



# Open Document Format for Office Applications (OpenDocument) Version 1.4. Part 3: OpenDocument Schema

## Committee Specification Draft 01 1 July 2024

### This stage:

<https://docs.oasis-open.org/office/OpenDocument/v1.4/csd01/part3-schema/OpenDocument-v1.4-csd01-part3-schema.odt> (Authoritative)  
<https://docs.oasis-open.org/office/OpenDocument/v1.4/csd01/part3-schema/OpenDocument-v1.4-csd01-part3-schema.html>  
<https://docs.oasis-open.org/office/OpenDocument/v1.4/csd01/part3-schema/OpenDocument-v1.4-csd01-part3-schema.pdf>

### Previous stage:

<https://docs.oasis-open.org/office/OpenDocument/v1.4/csd01/part3-schema/OpenDocument-v1.4-csd01-part3-schema.odt> (Authoritative)  
<https://docs.oasis-open.org/office/OpenDocument/v1.4/csd01/part3-schema/OpenDocument-v1.4-csd01-part3-schema.html>  
<https://docs.oasis-open.org/office/OpenDocument/v1.4/csd01/part3-schema/OpenDocument-v1.4-csd01-part3-schema.pdf>

### Latest stage:

<https://docs.oasis-open.org/office/OpenDocument/v1.4/OpenDocument-v1.4-part3-schema.odt> (Authoritative)  
<https://docs.oasis-open.org/office/OpenDocument/v1.4/OpenDocument-v1.4-part3-schema.html>  
<https://docs.oasis-open.org/office/OpenDocument/v1.4/OpenDocument-v1.4-part3-schema.pdf>

### Technical Committee:

OASIS Open Document Format for Office Applications (OpenDocument) TC

### Chairs:

Patrick Durusau ([patrick@durusau.net](mailto:patrick@durusau.net)), Individual  
Svante Schubert ([svante.schubert@gmail.com](mailto:svante.schubert@gmail.com)), Individual

### Editors:

Francis Cave ([francis@franciscave.com](mailto:francis@franciscave.com)), Individual  
Patrick Durusau ([patrick@durusau.net](mailto:patrick@durusau.net)), Individual  
Svante Schubert ([svante.schubert@gmail.com](mailto:svante.schubert@gmail.com)), Individual  
Michael Stahl ([michael.stahl@allotropia.de](mailto:michael.stahl@allotropia.de)), allotropia software GmbH

### Additional artifacts:

This prose specification is one component of a Work Product which includes:

- *Open Document Format for Office Applications (OpenDocument) Version 1.4. Part 1: Introduction.* <https://docs.oasis-open.org/office/OpenDocument/v1.4/csd01/part1-introduction/OpenDocument-v1.4-csd01-part1-introduction.html>.
- *Open Document Format for Office Applications (OpenDocument) Version 1.4. Part 2: Packages.* <https://docs.oasis-open.org/office/OpenDocument/v1.4/csd01/part2-packages/OpenDocument-v1.4-csd01-part2-packages.html>.
- *Open Document Format for Office Applications (OpenDocument) Version 1.4. Part 3: OpenDocument Schema (this part).* <https://docs.oasis-open.org/office/OpenDocument/v1.4/csd01/part3-schema/OpenDocument-v1.4-csd01-part3-schema.html>.
- *Open Document Format for Office Applications (OpenDocument) Version 1.4. Part 4: Recalculated Formula (OpenFormula) Format.* <https://docs.oasis-open.org/office/OpenDocument/v1.4/csd01/part4-formula/OpenDocument-v1.4-csd01-part4-formula.html>.
- XML/RNG schemas and OWL ontologies. <https://docs.oasis-open.org/office/OpenDocument/v1.4/csd01/schemas/>.

#### **Related work:**

This specification replaces or supersedes:

- *OASIS Open Document Format for Office Applications (OpenDocument) Version 1.3.* 27 April 2021. OASIS Standard. Latest stage: <https://docs.oasis-open.org/office/OpenDocument/v1.3/OpenDocument-v1.3-part3-schema.html>.

#### **Declared XML namespaces:**

urn:oasis:names:tc:opendocument:xmlns:animation:1.0  
 urn:oasis:names:tc:opendocument:xmlns:chart:1.0  
 urn:oasis:names:tc:opendocument:xmlns:config:1.0  
 urn:oasis:names:tc:opendocument:xmlns:database:1.0  
 urn:oasis:names:tc:opendocument:xmlns:dr3d:1.0  
 urn:oasis:names:tc:opendocument:xmlns:drawing:1.0  
 urn:oasis:names:tc:opendocument:xmlns:form:1.0  
 urn:oasis:names:tc:opendocument:xmlns:manifest:1.0  
 urn:oasis:names:tc:opendocument:xmlns:meta:1.0  
 urn:oasis:names:tc:opendocument:xmlns:datastyle:1.0  
 urn:oasis:names:tc:opendocument:xmlns:office:1.0  
 urn:oasis:names:tc:opendocument:xmlns:presentation:1.0  
 urn:oasis:names:tc:opendocument:xmlns:script:1.0  
 urn:oasis:names:tc:opendocument:xmlns:table:1.0  
 urn:oasis:names:tc:opendocument:xmlns:text:1.0  
 urn:oasis:names:tc:opendocument:xmlns:style:1.0  
 urn:oasis:names:tc:opendocument:xmlns:xsl-fo-compatible:1.0  
 urn:oasis:names:tc:opendocument:xmlns:svg-compatible:1.0  
 urn:oasis:names:tc:opendocument:xmlns:smil-compatible:1.0  
 urn:oasis:names:tc:opendocument:xmlns:of:1.2  
<http://docs.oasis-open.org/hs/office/1.2/meta/odf#>  
<http://docs.oasis-open.org/hs/office/1.2/meta/pkg#>

#### **Abstract:**

This document is Part 3 of the Open Document Format for Office Applications (OpenDocument) Version 1.4 specification.

#### **Status:**

This document was last revised or approved by the OASIS Open Document Format for Office Applications (OpenDocument) TC on the above date. The level of approval is also listed above. Check the "Latest stage" location noted above for possible later revisions of this document. Any other numbered Versions and other technical work produced by the Technical Committee (TC) are listed at <https://groups.oasis-open.org/communities/tc-community-home2?CommunityKey=4bf06d41-79ad-4c58-9e8e-018dc7d05da8#technical>.

TC members should send comments on this specification to the TC's email list. Others should send comments to the TC's public comment list at [Technical-Committee-Comments@oasis-open.org](mailto:Technical-Committee-Comments@oasis-open.org).

This specification is provided under the [RF on Limited Terms Mode](#) of the [OASIS IPR Policy](#), the mode chosen when the Technical Committee was established. For information on whether any patents have been disclosed that may be essential to implementing this specification, and any offers of patent licensing terms, please refer to the Intellectual Property Rights section of the TC's web page (<https://www.oasis-open.org/committees/office/ipr.php>).

Note that any machine-readable content ([Computer Language Definitions](#)) declared Normative for this Work Product is provided in separate plain text files. In the event of a discrepancy between any such plain text file and display content in the Work Product's prose narrative document(s), the content in the separate plain text file prevails.

**Citation format:**

When referencing this specification the following citation format should be used:

**[OpenDocument-v1.4-part3]**

Open Document Format for Office Applications (OpenDocument) Version 1.4. Part 3: OpenDocument Schema. Edited by Francis Cave, Patrick Durusau, Svante Schubert and Michael Stahl. 1 July 2024. OASIS Committee Specification Draft 01. <https://docs.oasis-open.org/office/OpenDocument/v1.4/csd01/part3-schema/OpenDocument-v1.4-csd01-part3-schema.html>. Latest stage: <https://docs.oasis-open.org/office/OpenDocument/v1.4/OpenDocument-v1.4-part3-schema.html>.

---

## Notices

Copyright © OASIS Open 2024. All Rights Reserved.

All capitalized terms in the following text have the meanings assigned to them in the OASIS Intellectual Property Rights Policy (the "OASIS IPR Policy"). The full [Policy](#) may be found at the OASIS website.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published, and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this section are included on all such copies and derivative works. However, this document itself may not be modified in any way, including by removing the copyright notice or references to OASIS, except as needed for the purpose of developing any document or deliverable produced by an OASIS Technical Committee (in which case the rules applicable to copyrights, as set forth in the OASIS IPR Policy, must be followed) or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by OASIS or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and OASIS DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY OWNERSHIP RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

As stated in the OASIS IPR Policy, the following three paragraphs in brackets apply to OASIS Standards Final Deliverable documents (Committee Specification, OASIS Standard, or Approved Errata).

[OASIS requests that any OASIS Party or any other party that believes it has patent claims that would necessarily be infringed by implementations of this OASIS Standards Final Deliverable, to notify OASIS TC Administrator and provide an indication of its willingness to grant patent licenses to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this deliverable.]

[OASIS invites any party to contact the OASIS TC Administrator if it is aware of a claim of ownership of any patent claims that would necessarily be infringed by implementations of this OASIS Standards Final Deliverable by a patent holder that is not willing to provide a license to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this OASIS Standards Final Deliverable. OASIS may include such claims on its website, but disclaims any obligation to do so.]

[OASIS takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this OASIS Standards Final Deliverable or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on OASIS' procedures with respect to rights in any document or deliverable produced by an OASIS Technical Committee can be found on the OASIS website. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this OASIS Standards Final Deliverable, can be obtained from the OASIS TC Administrator. OASIS makes no representation that any information or list of intellectual property rights will at any time be complete, or that any claims in such list are, in fact, Essential Claims.]

The name "OASIS" is a trademark of [OASIS](#), the owner and developer of this specification, and should be used only to refer to the organization and its official outputs. OASIS welcomes reference to, and implementation and use of, specifications, while reserving the right to enforce its marks against misleading uses. Please see <https://www.oasis-open.org/policies-guidelines/trademark/> for above guidance.

---

# Table of Contents

1	Introduction.....	77
1.1	Introduction.....	77
1.2	Terminology.....	77
1.3	Normative References.....	77
1.4	Non Normative References.....	81
1.5	Namespaces.....	81
2	OpenDocument Documents, Consumers and Producers.....	84
2.1	Introduction.....	84
2.2	Document Conformance.....	84
2.2.1	OpenDocument Document.....	84
2.2.2	OpenDocument Extended Document.....	85
2.2.3	OpenDocument Text Document.....	85
2.2.4	OpenDocument Spreadsheet Document.....	86
2.2.5	OpenDocument Drawing Document.....	86
2.2.6	OpenDocument Presentation Document.....	86
2.2.7	OpenDocument Chart Document.....	86
2.2.8	OpenDocument Image Document.....	87
2.2.9	OpenDocument Formula Document.....	87
2.2.10	OpenDocument Database Front End Document.....	87
2.3	Producers.....	87
2.3.1	OpenDocument Producer.....	87
2.3.2	OpenDocument Extended Producer.....	87
2.4	Consumer.....	88
3	Document Structure.....	89
3.1	Document Representation.....	89
3.1.1	General.....	89
3.1.2	<office:document>(Single OpenDocument XML Files).....	89
3.1.3	Package OpenDocument Files.....	89
3.1.3.1	General.....	89
3.1.3.2	<office:document-content>.....	89
3.1.3.3	<office:document-styles>.....	90
3.1.3.4	<office:document-meta>.....	90
3.1.3.5	<office:document-settings>.....	90
3.1.4	Summary of Document Representation.....	90
3.2	<office:meta>.....	91
3.3	<office:body>.....	91
3.4	<office:text>.....	91
3.5	<office:drawing>.....	92
3.6	<office:presentation>.....	92

3.7 <office:spreadsheet>.....	92
3.8 <office:chart>.....	93
3.9 <office:image>.....	93
3.10 <office:settings>.....	93
3.10.1 General.....	93
3.10.2 <config:config-item-set>.....	94
3.10.3 <config:config-item>.....	94
3.10.4 <config:config-item-map-indexed>.....	94
3.10.5 <config:config-item-map-entry>.....	94
3.10.6 <config:config-item-map-named>.....	95
3.11 Cursor Position Setting.....	95
3.12 <office:scripts>.....	95
3.13 <office:script>.....	95
3.14 <office:font-face-decls>.....	96
3.15 Styles.....	96
3.15.1 General.....	96
3.15.2 <office:styles>.....	96
3.15.3 <office:automatic-styles>.....	96
3.15.4 <office:master-styles>.....	97
3.16 Document Signatures.....	97
3.17 Foreign Elements and Attributes.....	97
3.18 White Space Processing and EOL Handling.....	98
3.19 MIME Types and File Name Extensions.....	98
4 Metadata.....	99
4.1 General.....	99
4.2 RDF Metadata.....	99
4.2.1 In-Content Metadata (RDFa).....	99
4.2.2 manifest.rdf.....	99
4.2.2.1 General.....	99
4.2.2.2 odf:ContentFile.....	99
4.2.2.3 odf:StylesFile.....	99
4.2.2.4 odf:Element.....	100
4.2.2.5 odf:prefix.....	100
4.2.2.6 odf:suffix.....	100
4.3 Non-RDF Metadata.....	100
4.3.1 General.....	100
4.3.2 Pre-Defined Metadata Elements.....	101
4.3.2.1 <meta:generator>.....	101
4.3.2.2 <dc:title>.....	101
4.3.2.3 <dc:description>.....	101
4.3.2.4 <dc:subject>.....	101
4.3.2.5 <meta:keyword>.....	102
4.3.2.6 <meta:initial-creator>.....	102

4.3.2.7 <dc:creator>.....	102
4.3.2.8 <meta:printed-by>.....	102
4.3.2.9 <meta:creation-date>.....	102
4.3.2.10 <dc:date>.....	103
4.3.2.11 <meta:print-date>.....	103
4.3.2.12 <meta:template>.....	103
4.3.2.13 <meta:auto-reload>.....	103
4.3.2.14 <meta:hyperlink-behaviour>.....	103
4.3.2.15 <dc:language>.....	104
4.3.2.16 <meta:editing-cycles>.....	104
4.3.2.17 <meta:editing-duration>.....	104
4.3.2.18 <meta:document-statistic>.....	104
4.3.3 <meta:user-defined>.....	105
5 Text Content.....	106
5.1 Headings, Paragraphs and Basic Text Structure.....	106
5.1.1 General.....	106
5.1.2 <text:h>.....	106
5.1.3 <text:p>.....	107
5.2 <text:page-sequence>.....	109
5.2.1 General.....	109
5.2.2 <text:page>.....	109
5.3 Lists.....	109
5.3.1 <text:list>.....	109
5.3.2 Default List Style.....	110
5.3.3 <text:list-header>.....	110
5.3.4 <text:list-item>.....	110
5.3.5 List Item Style Rules.....	110
5.3.6 <text:numbered-paragraph> (deprecated).....	111
5.3.7 Numbered Paragraph Style Rules (deprecated).....	111
5.4 <text:section>.....	111
5.4.1 General.....	111
5.4.2 <text:section-source>.....	112
5.5 Change Tracking.....	113
5.5.1 General.....	113
5.5.2 <text:tracked-changes>.....	113
5.5.3 <text:changed-region>.....	113
5.5.4 <text:insertion>.....	113
5.5.5 <text:deletion>.....	114
5.5.6 <text:format-change>.....	115
5.5.7 <office:change-info>.....	115
5.5.8 Change Marks.....	115
5.5.8.1 General.....	115
5.5.8.2 <text:change-start>.....	115
5.5.8.3 <text:change-end>.....	116

5.5.8.4 <text:change>.....	116
5.6 <text:soft-page-break>.....	116
5.7 Document Declarations.....	116
6 Paragraph Elements Content.....	118
6.1 Basic Text Content.....	118
6.1.1 General.....	118
6.1.2 White Space Characters.....	118
6.1.3 <text:s>.....	119
6.1.4 <text:tab>.....	119
6.1.5 <text:line-break>.....	120
6.1.6 Soft Hyphens, Hyphens, and Non-breaking Spaces.....	120
6.1.7 <text:span>.....	120
6.1.8 <text:a>.....	121
6.1.9 <text:meta>.....	122
6.1.10 <text:number>.....	124
6.2 Bookmarks and References.....	124
6.2.1 Bookmarks.....	124
6.2.1.1 General.....	124
6.2.1.2 <text:bookmark>.....	124
6.2.1.3 <text:bookmark-start>.....	124
6.2.1.4 <text:bookmark-end>.....	125
6.2.2 References.....	125
6.2.2.1 General.....	125
6.2.2.2 <text:reference-mark>.....	125
6.2.2.3 <text:reference-mark-start>.....	125
6.2.2.4 <text:reference-mark-end>.....	126
6.3 Notes.....	126
6.3.1 General.....	126
6.3.2 <text:note>.....	126
6.3.3 <text:note-citation>.....	126
6.3.4 <text:note-body>.....	126
6.4 <text:ruby>.....	127
6.4.1 General.....	127
6.4.2 <text:ruby-base>.....	127
6.4.3 <text:ruby-text>.....	128
7 Text Fields.....	129
7.1 General.....	129
7.2 Common Characteristics of Field Elements.....	129
7.3 Document Fields.....	129
7.3.1 General.....	129
7.3.2 <text:date>.....	129
7.3.3 <text:time>.....	129

7.3.4 <text:page-number>.....	130
7.3.5 <text:page-continuation>.....	130
7.3.6 Subsequent Author Fields.....	130
7.3.6.1 General.....	130
7.3.6.2 <text:sender-firstname>.....	130
7.3.6.3 <text:sender-lastname>.....	131
7.3.6.4 <text:sender-initials>.....	131
7.3.6.5 <text:sender-title>.....	131
7.3.6.6 <text:sender-position>.....	131
7.3.6.7 <text:sender-email>.....	132
7.3.6.8 <text:sender-phone-private>.....	132
7.3.6.9 <text:sender-fax>.....	132
7.3.6.10 <text:sender-company>.....	132
7.3.6.11 <text:sender-phone-work>.....	132
7.3.6.12 <text:sender-street>.....	133
7.3.6.13 <text:sender-city>.....	133
7.3.6.14 <text:sender-postal-code>.....	133
7.3.6.15 <text:sender-country>.....	133
7.3.6.16 <text:sender-state-or-province>.....	134
7.3.7 Author Fields.....	134
7.3.7.1 <text:author-name>.....	134
7.3.7.2 <text:author-initials>.....	134
7.3.8 <text:chapter>.....	134
7.3.9 <text:file-name>.....	135
7.3.10 <text:template-name>.....	135
7.3.11 <text:sheet-name>.....	135
7.4 Variable Fields.....	135
7.4.1 General.....	135
7.4.2 <text:variable-decls>.....	136
7.4.3 <text:variable-decl>.....	136
7.4.4 <text:variable-set>.....	137
7.4.5 <text:variable-get>.....	137
7.4.6 <text:variable-input>.....	137
7.4.7 <text:user-field-decls>.....	137
7.4.8 <text:user-field-decl>.....	138
7.4.9 <text:user-field-get>.....	138
7.4.10 <text:user-field-input>.....	138
7.4.11 <text:sequence-decls>.....	138
7.4.12 <text:sequence-decl>.....	139
7.4.13 <text:sequence>.....	139
7.4.14 <text:expression>.....	139
7.4.15 <text:text-input>.....	140
7.4.16 <text:drop-down>.....	140
7.4.17 <text:label>.....	140
7.5 Metadata Fields.....	140

7.5.1	General.....	140
7.5.2	<text:initial-creator>.....	141
7.5.3	<text:creation-date>.....	141
7.5.4	<text:creation-time>.....	141
7.5.5	<text:description>.....	141
7.5.6	<text:user-defined>.....	142
7.5.7	<text:print-time>.....	142
7.5.8	<text:print-date>.....	142
7.5.9	<text:printed-by>.....	142
7.5.10	<text:title>.....	143
7.5.11	<text:subject>.....	143
7.5.12	<text:keywords>.....	143
7.5.13	<text:editing-cycles>.....	143
7.5.14	<text:editing-duration>.....	144
7.5.15	<text:modification-time>.....	144
7.5.16	<text:modification-date>.....	144
7.5.17	<text:creator>.....	144
7.5.18	Document Statistics Fields.....	145
7.5.18.1	General.....	145
7.5.18.2	<text:page-count>.....	145
7.5.18.3	<text:paragraph-count>.....	145
7.5.18.4	<text:word-count>.....	145
7.5.18.5	<text:character-count>.....	145
7.5.18.6	<text:table-count>.....	146
7.5.18.7	<text:image-count>.....	146
7.5.18.8	<text:object-count>.....	146
7.5.19	<text:meta-field>.....	146
7.6	Database Fields.....	148
7.6.1	General.....	148
7.6.2	<form:connection-resource>.....	148
7.6.3	<text:database-display>.....	148
7.6.4	<text:database-next>.....	149
7.6.5	<text:database-row-select>.....	149
7.6.6	<text:database-row-number>.....	149
7.6.7	<text:database-name>.....	149
7.7	More Fields.....	150
7.7.1	Page Variable Fields.....	150
7.7.1.1	General.....	150
7.7.1.2	<text:page-variable-set>.....	150
7.7.1.3	<text:page-variable-get>.....	150
7.7.2	<text:placeholder>.....	150
7.7.3	<text:conditional-text>.....	151
7.7.4	<text:hidden-text>.....	151

7.7.5 <text:reference-ref>.....	151
7.7.6 <text:bookmark-ref>.....	152
7.7.7 <text:note-ref>.....	152
7.7.8 <text:sequence-ref>.....	152
7.7.9 <text:script>.....	152
7.7.10 <text:execute-macro>.....	153
7.7.11 <text:hidden-paragraph>.....	153
7.7.12 <text:dde-connection>.....	153
7.7.13 <text:measure>.....	154
7.7.14 <text:table-formula> (deprecated).....	154
<b>8 Text Indexes.....</b>	<b>155</b>
8.1 Index Marks.....	155
8.1.1 General.....	155
8.1.2 <text:toc-mark-start>.....	155
8.1.3 <text:toc-mark-end>.....	155
8.1.4 <text:toc-mark>.....	156
8.1.5 <text:user-index-mark-start>.....	156
8.1.6 <text:user-index-mark-end>.....	156
8.1.7 <text:user-index-mark>.....	156
8.1.8 <text:alphabetical-index-mark-start>.....	156
8.1.9 <text:alphabetical-index-mark-end>.....	157
8.1.10 <text:alphabetical-index-mark>.....	157
8.1.11 <text:bibliography-mark>.....	157
8.2 Index Structure.....	158
8.2.1 General.....	158
8.2.2 <text:index-body>.....	158
8.2.3 <text:index-title>.....	158
8.3 <text:table-of-content>.....	159
8.3.1 General.....	159
8.3.2 <text:table-of-content-source>.....	159
8.3.3 <text:table-of-content-entry-template>.....	160
8.4 <text:illustration-index>.....	160
8.4.1 General.....	160
8.4.2 <text:illustration-index-source>.....	160
8.4.3 <text:illustration-index-entry-template>.....	161
8.5 <text:table-index>.....	161
8.5.1 General.....	161
8.5.2 <text:table-index-source>.....	161
8.5.3 <text:table-index-entry-template>.....	161

8.6 <text:object-index>.....	162
8.6.1 General.....	162
8.6.2 <text:object-index-source>.....	162
8.6.3 <text:object-index-entry-template>.....	162
8.7 <text:user-index>.....	163
8.7.1 General.....	163
8.7.2 <text:user-index-source>.....	163
8.7.3 <text:user-index-entry-template>.....	163
8.8 <text:alphabetical-index>.....	164
8.8.1 General.....	164
8.8.2 <text:alphabetical-index-source>.....	164
8.8.3 <text:alphabetical-index-auto-mark-file>.....	164
8.8.4 <text:alphabetical-index-entry-template>.....	165
8.9 <text:bibliography>.....	165
8.9.1 General.....	165
8.9.2 <text:bibliography-source>.....	165
8.9.3 <text:bibliography-entry-template>.....	166
8.10 <text:index-source-styles>.....	166
8.11 <text:index-source-style>.....	166
8.12 <text:index-title-template>.....	166
8.13 Index Template Entries.....	167
8.13.1 <text:index-entry-chapter>.....	167
8.13.2 <text:index-entry-text>.....	167
8.13.3 <text:index-entry-page-number>.....	167
8.13.4 <text:index-entry-span>.....	167
8.13.5 <text:index-entry-bibliography>.....	168
8.13.6 <text:index-entry-tab-stop>.....	168
8.13.7 <text:index-entry-link-start>.....	168
8.13.8 <text:index-entry-link-end>.....	168
9 Tables.....	170
9.1 Basic Tables.....	170
9.1.1 General.....	170
9.1.2 <table:table>.....	170
9.1.3 <table:table-row>.....	171
9.1.4 <table:table-cell>.....	171
9.1.5 <table:covered-table-cell>.....	171
9.1.6 <table:table-column>.....	172
9.1.7 <table:table-header-rows>.....	172
9.1.8 <table:table-rows>.....	173

9.1.9 <table:table-row-group>.....	173
9.1.10 <table:table-column-group>.....	173
9.1.11 <table:table-header-columns>.....	174
9.1.12 <table:table-columns>.....	174
9.1.13 <table:title>.....	174
9.1.14 <table:desc>.....	174
9.2 Advanced Tables.....	175
9.2.1 Referencing Table Cells.....	175
9.2.2 Absolute and relative cell addressing.....	175
9.2.3 Cell Range Address.....	175
9.2.4 Column and Row Range Addresses.....	176
9.2.5 Cell Range Address List.....	176
9.2.6 <table:table-source>.....	176
9.2.7 <table:scenario>.....	176
9.2.8 <table:shapes>.....	177
9.3 Advanced Table Cells.....	177
9.3.1 <table:cell-range-source>.....	177
9.3.2 <table:detective>.....	177
9.3.3 <table:operation>.....	178
9.3.4 <table:highlighted-range>.....	178
9.4 Spreadsheet Document Content.....	178
9.4.1 <table:calculation-settings>.....	178
9.4.2 <table:null-date>.....	178
9.4.3 <table:iteration>.....	179
9.4.4 <table:content-validations>.....	179
9.4.5 <table:content-validation>.....	179
9.4.6 <table:help-message>.....	180
9.4.7 <table:error-message>.....	180
9.4.8 <table:error-macro>.....	180
9.4.9 <table:label-range>.....	180
9.4.10 <table:label-ranges>.....	181
9.4.11 <table:named-expressions>.....	181
9.4.12 <table:named-range>.....	181
9.4.13 <table:named-expression>.....	181
9.4.14 <table:database-ranges>.....	182
9.4.15 <table:database-range>.....	182
9.4.16 <table:database-source-sql>.....	182
9.4.17 <table:database-source-table>.....	182
9.4.18 <table:database-source-query>.....	183

9.4.19 <table:sort>.....	183
9.4.20 <table:sort-by>.....	183
9.4.21 <table:subtotal-rules>.....	183
9.4.22 <table:sort-groups>.....	184
9.4.23 <table:subtotal-rule>.....	184
9.4.24 <table:subtotal-field>.....	184
9.5 Filters.....	184
9.5.1 General.....	184
9.5.2 <table:filter>.....	184
9.5.3 <table:filter-and>.....	185
9.5.4 <table:filter-or>.....	185
9.5.5 <table:filter-condition>.....	185
9.5.6 <table:filter-set-item>.....	185
9.6 Data Pilot Tables.....	186
9.6.1 General.....	186
9.6.2 <table:data-pilot-tables>.....	186
9.6.3 <table:data-pilot-table>.....	186
9.6.4 Data Pilot Table Source.....	187
9.6.5 <table:source-cell-range>.....	187
9.6.6 <table:source-service>.....	187
9.6.7 <table:data-pilot-field>.....	187
9.6.8 <table:data-pilot-level>.....	187
9.6.9 <table:data-pilot-subtotals>.....	188
9.6.10 <table:data-pilot-subtotal>.....	188
9.6.11 <table:data-pilot-members>.....	188
9.6.12 <table:data-pilot-member>.....	188
9.6.13 <table:data-pilot-display-info>.....	189
9.6.14 <table:data-pilot-sort-info>.....	189
9.6.15 <table:data-pilot-layout-info>.....	189
9.6.16 <table:data-pilot-field-reference>.....	189
9.6.17 <table:data-pilot-groups>.....	189
9.6.18 <table:data-pilot-group>.....	190
9.6.19 <table:data-pilot-group-member>.....	190
9.7 <table:consolidation>.....	190
9.8 <table:dde-links>.....	190
9.9 Change Tracking in Spreadsheets.....	191
9.9.1 General.....	191
9.9.2 <table:tracked-changes>.....	191
9.9.3 <table:insertion>.....	191

9.9.4 <table:dependencies>.....	191
9.9.5 <table:dependency>.....	192
9.9.6 <table:deletions>.....	192
9.9.7 <table:cell-content-deletion>.....	192
9.9.8 <table:change-deletion>.....	192
9.9.9 <table:deletion>.....	192
9.9.10 <table:cut-offs>.....	193
9.9.11 <table:insertion-cut-off>.....	193
9.9.12 <table:movement-cut-off>.....	193
9.9.13 <table:movement>.....	193
9.9.14 <table:source-range-address>.....	194
9.9.15 <table:target-range-address>.....	194
9.9.16 <table:change-track-table-cell>.....	194
9.9.17 <table:cell-content-change>.....	194
9.9.18 <table:cell-address>.....	195
9.9.19 <table:previous>.....	195
<b>10 Graphic Content.....</b>	<b>196</b>
10.1 General.....	196
<b>10.2 Enhanced Page Features for Graphical Applications.....</b>	<b>196</b>
10.2.1 <style:handout-master>.....	196
10.2.2 <draw:layer-set>.....	196
10.2.3 <draw:layer>.....	196
10.2.4 <draw:page>.....	197
10.2.5 <draw:equation>.....	197
<b>10.3 Drawing Shapes.....</b>	<b>198</b>
10.3.1 General.....	198
10.3.2 <draw:rect>.....	198
10.3.3 <draw:line>.....	198
10.3.4 <draw:polyline>.....	199
10.3.5 <draw:polygon>.....	199
10.3.6 <draw:regular-polygon>.....	200
10.3.7 <draw:path>.....	200
10.3.8 <draw:circle>.....	201
10.3.9 <draw:ellipse>.....	201
10.3.10 <draw:connector>.....	202
10.3.11 <draw:caption>.....	202
10.3.12 <draw:measure>.....	203
10.3.13 <draw:control>.....	205
10.3.14 <draw:page-thumbnail>.....	205

10.3.15 <draw:g>.....	206
10.3.16 <draw:glue-point>.....	206
10.3.17 <svg:title>.....	207
10.3.18 <svg:desc>.....	207
10.3.19 <office:event-listeners>.....	207
10.4 Frames.....	208
10.4.1 General.....	208
10.4.2 <draw:frame>.....	208
10.4.3 <draw:text-box>.....	209
10.4.4 <draw:image>.....	209
10.4.5 <office:binary-data>.....	210
10.4.6 Embedded Objects.....	210
10.4.6.1 General.....	210
10.4.6.2 <draw:object>.....	210
10.4.6.3 <draw:object-ole>.....	211
10.4.7 <draw:applet> (Deprecated).....	211
10.4.8 <draw:plugin>.....	211
10.4.9 <draw:param>.....	212
10.4.10 <draw:floating-frame>.....	212
10.4.11 Contours.....	212
10.4.11.1 General.....	212
10.4.11.2 <draw:contour-polygon>.....	212
10.4.11.3 <draw:contour-path>.....	212
10.4.12 <draw:a>.....	213
10.4.13 Client Side Image Maps.....	213
10.4.13.1 General.....	213
10.4.13.2 <draw:image-map>.....	213
10.4.13.3 <draw:area-rectangle>.....	213
10.4.13.4 <draw:area-circle>.....	214
10.4.13.5 <draw:area-polygon>.....	214
10.5 3D Shapes.....	215
10.5.1 General.....	215
10.5.2 <dr3d:scene>.....	215
10.5.3 <dr3d:light>.....	215
10.5.4 <dr3d:cube>.....	216
10.5.5 <dr3d:sphere>.....	216
10.5.6 <dr3d:extrude>.....	216
10.5.7 <dr3d:rotate>.....	217
10.6 Custom Shape.....	217
10.6.1 <draw:custom-shape>.....	217
10.6.2 <draw:enhanced-geometry>.....	218
10.6.3 <draw:handle>.....	218
10.7 Presentation Shapes.....	220

10.8	Presentation Animations (deprecated).....	220
10.8.1	<presentation:animations> (deprecated).....	220
10.8.2	<presentation:sound> (moved to 10.9.3).....	221
10.8.3	<presentation:show-shape> (deprecated).....	221
10.8.4	<presentation:show-text> (deprecated).....	221
10.8.5	<presentation:hide-shape> (deprecated).....	221
10.8.6	<presentation:hide-text> (deprecated).....	221
10.8.7	<presentation:dim> (deprecated).....	222
10.8.8	<presentation:play> (deprecated).....	222
10.8.9	<presentation:animation-group> (deprecated).....	222
10.9	SMIL Presentation Animations.....	222
10.9.1	General.....	222
10.9.2	<presentation:event-listener>.....	223
10.9.3	<presentation:sound>.....	223
10.9.4	Presentation Document Content.....	223
10.9.4.1	<presentation:header>.....	223
10.9.4.2	<presentation:header-decl>.....	224
10.9.4.3	<presentation:footer>.....	224
10.9.4.4	<presentation:footer-decl>.....	224
10.9.4.5	<presentation:date-time>.....	224
10.9.4.6	<presentation:date-time-decl>.....	225
10.9.4.7	<presentation:settings>.....	225
10.9.4.8	<presentation:show>.....	225
11	Chart Content.....	227
11.1	<chart:chart>.....	227
11.2	Title, Subtitle and Footer.....	227
11.2.1	<chart:title>.....	227
11.2.2	<chart:subtitle>.....	227
11.2.3	<chart:footer>.....	228
11.3	<chart:coordinate-region>.....	228
11.4	<chart:legend>.....	228
11.5	<chart:plot-area>.....	229
11.6	3D Plot Area.....	230
11.7	<chart:wall>.....	230
11.8	<chart:floor>.....	230
11.9	<chart:axis>.....	230
11.10	<chart:categories>.....	231
11.11	<chart:grid>.....	231
11.12	<chart:series>.....	231
11.13	<chart:domain>.....	231
11.14	<chart:data-point>.....	232
11.15	<chart:data-label>.....	232

11.16 <chart:mean-value>.....	232
11.17 <chart:error-indicator>.....	232
11.18 <chart:regression-curve>.....	233
11.19 <chart:equation>.....	233
11.20 <chart:stock-gain-marker>.....	233
11.21 <chart:stock-loss-marker>.....	233
11.22 <chart:stock-range-line>.....	233
12 Database Front-end Document Content.....	235
12.1 <office:database>.....	235
12.2 <db:data-source>.....	235
12.3 <db:connection-data>.....	235
12.4 <db:database-description>.....	235
12.5 <db:file-based-database>.....	236
12.6 <db:server-database>.....	236
12.7 <db:connection-resource>.....	236
12.8 <db:login>.....	236
12.9 <db:driver-settings>.....	236
12.10 <db:auto-increment>.....	237
12.11 <db:delimiter>.....	237
12.12 <db:character-set>.....	237
12.13 <db:table-settings>.....	237
12.14 <db:table-setting>.....	237
12.15 <db:application-connection-settings>.....	238
12.16 <db:table-filter>.....	238
12.17 <db:table-include-filter>.....	238
12.18 <db:table-exclude-filter>.....	238
12.19 <db:table-filter-pattern>.....	239
12.20 <db:table-type-filter>.....	239
12.21 <db:table-type>.....	239
12.22 <db:data-source-settings>.....	239
12.23 <db:data-source-setting>.....	240
12.24 <db:data-source-setting-value>.....	240
12.25 Forms and Reports.....	240
12.25.1 General.....	240
12.25.2 <db:forms>.....	240
12.25.3 <db:reports>.....	240
12.25.4 <db:component-collection>.....	241
12.25.5 <db:component>.....	241
12.26 <db:queries>.....	241

12.27 <db:query-collection>.....	241
12.28 <db:query>.....	241
12.29 <db:order-statement>.....	242
12.30 <db:filter-statement>.....	242
12.31 <db:update-table>.....	242
12.32 <db:table-representations>.....	242
12.33 <db:table-representation>.....	243
12.34 <db:columns>.....	243
12.35 <db:column>.....	243
12.36 <db:schema-definition>.....	243
12.37 <db:table-definitions>.....	243
12.38 <db:table-definition>.....	244
12.39 <db:column-definitions>.....	244
12.40 <db:column-definition>.....	244
12.41 <db:keys>.....	244
12.42 <db:key>.....	244
12.43 <db:key-columns>.....	245
12.44 <db:key-column>.....	245
12.45 <db:indices>.....	245
12.46 <db:index>.....	245
12.47 <db:index-columns>.....	245
12.48 <db:index-column>.....	246
13 Form Content.....	247
13.1 General.....	247
13.2 <office:forms>.....	247
13.3 <form:form>.....	247
13.4 <xforms:model>.....	248
13.5 Controls.....	248
13.5.1 General.....	248
13.5.2 <form:text>.....	248
13.5.3 <form:textarea>.....	249
13.5.4 <form:password>.....	249
13.5.5 <form:file>.....	249
13.5.6 <form:formatted-text>.....	249
13.5.7 <form:number>.....	250
13.5.8 <form:date>.....	250
13.5.9 <form:time>.....	250
13.5.10 <form:fixed-text>.....	251
13.5.11 <form:combobox>.....	251

13.5.12 <form:item>.....	251
13.5.13 <form:listbox>.....	252
13.5.14 <form:option>.....	252
13.5.15 <form:button>.....	252
13.5.16 <form:image>.....	252
13.5.17 <form:checkbox>.....	253
13.5.18 <form:radio>.....	253
13.5.19 <form:frame>.....	254
13.5.20 <form:image-frame>.....	254
13.5.21 <form:hidden>.....	254
13.5.22 <form:grid>.....	254
13.5.23 <form:column>.....	255
13.5.24 <form:value-range>.....	255
13.5.25 <form:generic-control>.....	255
13.6 Event Listeners.....	255
13.7 <form:properties>.....	257
13.8 <form:property>.....	257
13.9 <form:list-property>.....	257
13.10 <form:list-value>.....	258
14 Common Content.....	259
14.1 <office:annotation>.....	259
14.2 <office:annotation-end>.....	259
14.3 <meta:date-string> (deprecated).....	259
14.4 <meta:creator initials>.....	260
14.5 Event Listener Tables.....	260
14.5.1 General.....	260
14.5.2 <script:event-listener>.....	260
14.5.3 Event Types.....	260
14.5.4 Macro Name and Location.....	261
14.6 <math:math>.....	261
14.7 DDE Connections.....	261
14.7.1 General.....	261
14.7.2 <text:dde-connection-decls>.....	261
14.7.3 <text:dde-connection-decl>.....	261
14.7.4 <table:dde-link>.....	262
14.7.5 <office:dde-source>.....	262
15 SMIL Animations.....	263
15.1 General.....	263
15.2 Basic Animation Elements.....	263

15.2.1	General.....	263
15.2.2	<anim:animate>.....	263
15.2.3	<anim:animateTransform>.....	263
15.2.4	<anim:set>.....	264
15.2.5	<anim:animateMotion>.....	264
15.2.6	<anim:animateColor>.....	264
15.2.7	<anim:transitionFilter>.....	265
15.3	Animation Model Attributes.....	265
15.3.1	General.....	265
15.3.2	Simple Animation Functions.....	265
15.4	Animation Timing.....	265
15.4.1	General.....	265
15.4.2	<anim:par>.....	265
15.4.3	<anim:seq>.....	266
15.4.4	<anim:iterate>.....	266
15.5	<anim:audio>.....	267
15.6	Animation Command Elements.....	267
15.6.1	<anim:command>.....	267
15.6.2	<anim:param>.....	267
16	Styles.....	268
16.1	General.....	268
16.2	<style:style>.....	268
16.3	<style:map>.....	269
16.4	<style:default-style>.....	270
16.5	<style:page-layout>.....	270
16.6	<style:header-style>.....	270
16.7	<style:footer-style>.....	270
16.8	<style:default-page-layout>.....	271
16.9	<style:master-page>.....	271
16.10	<style:header>.....	272
16.11	<style:header-first>.....	272
16.12	<style:footer>.....	273
16.13	<style:footer-first>.....	273
16.14	<style:header-left>.....	273
16.15	<style:footer-left>.....	274
16.16	<style:region-left>.....	274
16.17	<style:region-center>.....	274
16.18	<style:region-right>.....	275
16.19	<presentation:notes>.....	275

16.20 <table:table-template>.....	275
16.21 Cell Styles.....	276
16.21.1 General.....	276
16.21.2 <table:first-row>.....	276
16.21.3 <table:last-row>.....	276
16.21.4 <table:first-column>.....	276
16.21.5 <table:last-column>.....	277
16.21.6 <table:body>.....	277
16.21.7 <table:even-rows>.....	277
16.21.8 <table:odd-rows>.....	277
16.21.9 <table:even-columns>.....	277
16.21.10 <table:odd-columns>.....	278
16.22 <table:background>.....	278
16.23 <style:font-face>.....	278
16.24 <svg:font-face-src>.....	279
16.25 <svg:font-face-name>.....	279
16.26 <svg:font-face-uri>.....	279
16.27 <svg:definition-src>.....	279
16.28 <svg:font-face-format>.....	279
16.29 Data Styles.....	280
16.29.1 General.....	280
16.29.2 <number:number-style>.....	280
16.29.3 <number:number>.....	280
16.29.4 <number:embedded-text>.....	280
16.29.5 <number:fill-character>.....	281
16.29.6 <number:scientific-number>.....	281
16.29.7 <number:fraction>.....	281
16.29.8 <number:currency-style>.....	282
16.29.9 <number:currency-symbol>.....	282
16.29.10 <number:percentage-style>.....	282
16.29.11 <number:date-style>.....	283
16.29.12 <number:day>.....	283
16.29.13 <number:month>.....	283
16.29.14 <number:year>.....	283
16.29.15 <number:era>.....	284
16.29.16 <number:day-of-week>.....	284
16.29.17 <number:week-of-year>.....	284
16.29.18 <number:quarter>.....	284
16.29.19 <number:time-style>.....	284

16.29.20 <number:hours>.....	285
16.29.21 <number:minutes>.....	285
16.29.22 <number:seconds>.....	285
16.29.23 <number:am-pm>.....	285
16.29.24 <number:boolean-style>.....	286
16.29.25 <number:boolean>.....	286
16.29.26 <number:text-style>.....	286
16.29.27 <number:text>.....	286
16.29.28 <number:text-content>.....	287
16.29.29 <style:text-properties>.....	287
16.29.30 <number:num-list-format>.....	288
16.29.31 <number:num-list-label>.....	288
16.30 Text Content Style Families.....	288
16.30.1 Text Styles.....	288
16.30.2 Paragraph Styles.....	288
16.30.3 Section Styles.....	288
16.30.4 Ruby Styles.....	289
16.31 Enhanced Text Styles.....	289
16.31.1 <text:linenumbering-configuration>.....	289
16.31.2 <text:linenumbering-separator>.....	289
16.31.3 <text:notes-configuration>.....	289
16.31.4 <text:note-continuation-notice-forward>.....	290
16.31.5 <text:note-continuation-notice-backward>.....	290
16.31.6 <text:bibliography-configuration>.....	290
16.31.7 <text:sort-key>.....	290
16.32 <text:list-style>.....	290
16.33 <text:list-level-style-bullet>.....	291
16.34 <text:list-level-style-number>.....	291
16.35 <text:list-level-style-image>.....	292
16.36 <text:outline-style>.....	292
16.37 <text:outline-level-style>.....	292
16.38 Table Styles.....	292
16.38.1 Table Styles.....	292
16.38.2 Table Column Styles.....	292
16.38.3 Table Row Styles.....	292
16.38.4 Table Cell Styles.....	293
16.39 Graphic Styles.....	293
16.40 Presentation Styles.....	293
16.41 Drawing Page Style.....	293

16.42 Enhanced Graphic Style Elements.....	293
16.42.1 <draw:gradient>.....	293
16.42.2 <svg:linearGradient>.....	293
16.42.3 <svg:radialGradient>.....	294
16.42.4 <svg:stop>.....	294
16.42.5 <draw:hatch>.....	294
16.42.6 <draw:fill-image>.....	295
16.42.7 <draw:opacity>.....	295
16.42.8 <draw:marker>.....	295
16.42.9 <draw:stroke-dash>.....	296
16.43 <style:presentation-page-layout>.....	296
16.44 <presentation:placeholder>.....	297
16.45 Chart Styles.....	297
17 Formatting Properties Elements.....	298
17.1 General.....	298
17.2 <style:page-layout-properties>.....	298
17.3 <style:background-image>.....	298
17.4 <style:footnote-sep>.....	299
17.5 <style:header-footer-properties>.....	299
17.6 <style:paragraph-properties>.....	299
17.7 <style:tab-stops>.....	300
17.8 <style:tab-stop>.....	300
17.9 <style:drop-cap>.....	301
17.10 <style:ruby-properties>.....	301
17.11 <style:section-properties>.....	301
17.12 <style:columns>.....	301
17.13 <style:column>.....	301
17.14 <style:column-sep>.....	302
17.15 <style:table-properties>.....	302
17.16 <style:table-column-properties>.....	302
17.17 <style:table-row-properties>.....	302
17.18 <style:table-cell-properties>.....	303
17.19 <style:list-level-properties>.....	303
17.20 <style:list-level-label-alignment>.....	303
17.21 <style:graphic-properties>.....	304
17.22 <style:chart-properties>.....	305
17.23 <chart:symbol-image>.....	306
17.24 <chart:label-separator>.....	306
17.25 <style:drawing-page-properties>.....	307

17.26 <style:background-fill>.....	307
18 Datatypes.....	309
18.1 Introduction.....	309
18.2 W3C Schema Datatypes.....	309
18.3 Other Datatypes.....	309
18.3.1 angle.....	309
18.3.2 anyIRI.....	310
18.3.3 boolean.....	310
18.3.4 cellAddress.....	310
18.3.5 cellRangeAddress.....	310
18.3.6 cellRangeAddressList.....	310
18.3.7 character.....	310
18.3.8 clipShape.....	310
18.3.9 color.....	310
18.3.10 coordinate.....	310
18.3.11 countryCode.....	310
18.3.12 CURIE.....	310
18.3.13 CURIEs.....	310
18.3.14 dateOrDateTime.....	310
18.3.15 distance.....	311
18.3.16 .....	311
18.3.17 language.....	311
18.3.18 languageCode.....	311
18.3.19 length.....	311
18.3.20 namespacedToken.....	311
18.3.21 nonNegativeLength.....	311
18.3.22 nonNegativePixelLength.....	311
18.3.23 pathData.....	311
18.3.24 percent.....	311
18.3.25 point3D.....	311
18.3.26 points.....	311
18.3.27 positiveLength.....	311
18.3.28 relativeLength.....	312
18.3.29 safeCURIE.....	312
18.3.30 scriptCode.....	312
18.3.31 signedZeroToHundredPercent.....	312
18.3.32 styleName.....	312
18.3.33 styleNameRef.....	312
18.3.34 styleNameRefs.....	312

18.3.35	targetFrameName.....	312
18.3.36	textEncoding.....	312
18.3.37	timeOrDateTime.....	312
18.3.38	URIorSafeCURIE.....	312
18.3.39	valueType.....	312
18.3.40	variableName.....	312
18.3.41	vector3D.....	313
18.3.42	zeroToHundredPercent.....	313
18.3.43	error.....	313
18.3.44	nonNegativePercent.....	313
19	General Attributes.....	314
19.1	General.....	314
19.2	anim:color-interpolation.....	314
19.3	anim:color-interpolation-direction.....	314
19.4	anim:audio-level.....	314
19.5	anim:command.....	315
19.6	anim:formula.....	315
19.7	anim:id (deprecated).....	316
19.8	anim:iterate-interval.....	316
19.9	anim:iterate-type.....	317
19.10	anim:name.....	317
19.11	anim:sub-item.....	317
19.12	anim:value.....	317
19.13	chart:attached-axis.....	318
19.14	chart:automatic-content.....	318
19.15	chart:class.....	318
19.15.1	<chart:chart>, <chart:series>.....	318
19.15.2	<chart:grid>.....	325
19.16	chart:column-mapping (deprecated).....	325
19.17	chart:data-source-has-labels (deprecated).....	325
19.18	chart:dimension.....	326
19.19	chart:display-equation.....	326
19.20	chart:display-r-square.....	326
19.21	chart:label-cell-address.....	327
19.22	chart:legend-align.....	327
19.23	chart:legend-position.....	327
19.24	chart:name.....	328
19.25	chart:repeated.....	328
19.26	chart:row-mapping (deprecated).....	328

19.27	chart:style-name.....	328
19.28	chart:values-cell-range-address.....	329
19.29	config:name.....	329
19.30	config:type.....	329
19.31	db:additional-column-statement.....	329
19.32	db:append-table-alias-name.....	330
19.33	db:apply-command.....	330
19.34	db:as-template.....	330
19.35	db:base-dn.....	330
19.36	db:boolean-comparison-mode.....	331
19.37	db:catalog-name.....	331
19.38	db:command.....	331
19.39	db:data-source-setting-is-list.....	331
19.40	db:data-source-setting-name.....	332
19.41	db:data-source-setting-type.....	332
19.42	db:data-type.....	332
19.43	db:database-name.....	332
19.44	db:decimal.....	333
19.45	db:default-cell-style-name.....	333
19.46	db:default-row-style-name.....	333
19.47	db:delete-rule.....	333
19.48	db:description.....	333
19.49	db:enable-sql92-check.....	334
19.50	db:encoding.....	334
19.51	db:escape-processing.....	334
19.52	db:extension.....	334
19.53	db:field.....	335
19.54	db:hostname.....	335
19.55	db:ignore-driver-privileges.....	335
19.56	db:is-ascending.....	335
19.57	db:is-autoincrement.....	335
19.58	db:is-clustered.....	336
19.59	db:is-empty-allowed.....	336
19.60	db:is-first-row-header-line.....	336
19.61	db:login-timeout.....	336
19.62	db:is-nullable.....	336
19.63	db:is-password-required.....	337
19.64	db:is-unique.....	337
19.65	db:precision.....	337

19.66	db:is-table-name-length-limited.....	337
19.67	db:local-socket.....	338
19.68	db:max-row-count.....	338
19.69	db:media-type.....	338
19.70	db:name.....	338
19.70.1	General.....	338
19.70.2	<db:column>.....	338
19.70.3	<db:column-definition>.....	338
19.70.4	<db:component>.....	339
19.70.5	<db:component-collection>.....	339
19.70.6	<db:index>.....	339
19.70.7	<db:index-column>.....	339
19.70.8	<db:key>.....	339
19.70.9	<db:key-column>.....	339
19.70.10	<db:query>.....	339
19.70.11	<db:query-collection>.....	339
19.70.12	<db:table-definition>.....	340
19.70.13	<db:table-representation>.....	340
19.70.14	<db:update-table>.....	340
19.71	db:parameter-name-substitution.....	340
19.72	db:port.....	340
19.73	db:referenced-table-name.....	340
19.74	db:related-column-name.....	341
19.75	db:row-retrieving-statement.....	341
19.76	db:scale.....	341
19.77	db:schema-name.....	341
19.78	db:show-deleted.....	341
19.79	db:string.....	342
19.80	db:style-name.....	342
19.80.1	General.....	342
19.80.2	<db:column>.....	342
19.80.3	<db:query>.....	342
19.80.4	<db:table-representation>.....	342
19.81	db:suppress-version-columns.....	342
19.82	db:system-driver-settings.....	342
19.83	db:thousand.....	343
19.84	db:title.....	343
19.85	db:type.....	343
19.85.1	General.....	343

19.85.2 <db:key>.....	343
19.85.3 <db:server-database>.....	343
19.85.4 <db:table-definition>.....	343
19.86 db:type-name.....	344
19.87 db:update-rule.....	344
19.88 db:use-catalog.....	344
19.89 db:use-system-user.....	345
19.90 db:user-name.....	345
19.91 db:visible.....	345
19.92 dr3d:ambient-color.....	345
19.93 dr3d:center.....	346
19.94 dr3d:diffuse-color.....	346
19.95 dr3d:direction.....	346
19.96 dr3d:distance.....	346
19.97 dr3d:enabled.....	346
19.98 dr3d:focal-length.....	346
19.99 dr3d:lighting-mode.....	347
19.99.1 General.....	347
19.99.2 <chart:plot-area>.....	347
19.99.3 <dr3d:scene>.....	347
19.100 dr3d:max-edge.....	347
19.101 dr3d:min-edge.....	347
19.102 dr3d:projection.....	347
19.103 dr3d:shade-mode.....	348
19.104 dr3d:shadow-slant.....	348
19.105 dr3d:size.....	348
19.106 dr3d:specular.....	348
19.107 dr3d:transform.....	349
19.108 dr3d:vpn.....	349
19.109 dr3d:vrp.....	349
19.110 dr3d:vup.....	350
19.111 draw:align.....	350
19.112 draw:angle.....	350
19.113 draw:archive.....	350
19.114 draw:border.....	351
19.115 draw:caption-id.....	351
19.116 draw:caption-point-x.....	351
19.117 draw:caption-point-y.....	351
19.118 draw:chain-next-name.....	351

19.119	draw:class-id.....	352
19.120	draw:class-names.....	352
19.121	draw:code.....	352
19.122	draw:color.....	352
19.122.1	General.....	352
19.122.2	<draw:hatch>.....	352
19.122.3	<presentation:dim>.....	353
19.123	draw:concave.....	353
19.124	draw:concentric-gradient-fill-allowed.....	353
19.125	draw:control.....	353
19.126	draw:copy-of.....	354
19.127	draw:corner-radius.....	354
19.128	draw:corners.....	354
19.129	draw:cx.....	354
19.130	draw:cy.....	355
19.131	draw:data.....	355
19.132	draw:display.....	355
19.133	draw:display-name.....	355
19.133.1	General.....	355
19.133.2	<draw:fill-image>.....	355
19.133.3	<draw:gradient>.....	355
19.133.4	<draw:hatch>.....	356
19.133.5	<draw:marker>.....	356
19.133.6	<draw:opacity>.....	356
19.133.7	<draw:stroke-dash>.....	356
19.133.8	<svg:linearGradient>.....	356
19.133.9	<svg:radialGradient>.....	356
19.134	draw:distance.....	356
19.134.1	General.....	356
19.134.2	<draw:hatch>.....	357
19.134.3	<draw:stroke-dash>.....	357
19.135	draw:dots1.....	357
19.136	draw:dots1-length.....	357
19.137	draw:dots2.....	357
19.138	draw:dots2-length.....	357
19.139	draw:end.....	358
19.140	draw:end-angle.....	358
19.141	draw:end-color.....	358
19.142	draw:end-glue-point.....	358

19.143	draw:end-intensity.....	358
19.144	draw:end-shape.....	359
19.145	draw:enhanced-path.....	359
19.146	draw:engine.....	362
19.147	draw:escape-direction.....	362
19.148	draw:extrusion.....	362
19.149	draw:extrusion-allowed.....	363
19.150	draw:extrusion-brightness.....	363
19.151	draw:extrusion-color.....	363
19.152	draw:extrusion-depth.....	363
19.153	draw:extrusion-diffusion.....	364
19.154	draw:extrusion-first-light-direction.....	364
19.155	draw:extrusion-first-light-harsh.....	364
19.156	draw:extrusion-first-light-level.....	364
19.157	draw:extrusion-light-face.....	365
19.158	draw:extrusion-metal.....	365
19.159	draw:extrusion-number-of-line-segments.....	365
19.160	draw:extrusion-origin.....	365
19.161	draw:extrusion-rotation-angle.....	366
19.162	draw:extrusion-rotation-center.....	366
19.163	draw:extrusion-second-light-direction.....	366
19.164	draw:extrusion-second-light-harsh.....	367
19.165	draw:extrusion-second-light-level.....	367
19.166	draw:extrusion-shininess.....	367
19.167	draw:extrusion-skew.....	367
19.168	draw:extrusion-specularity.....	368
19.169	draw:extrusion-viewpoint.....	368
19.170	draw:filter-name.....	368
19.171	draw:formula.....	368
19.172	draw:frame-name.....	370
19.173	draw:glue-point-leaving-directions.....	370
19.174	draw:glue-point-type.....	370
19.175	draw:glue-points.....	371
19.176	draw:handle-mirror-horizontal.....	371
19.177	draw:handle-mirror-vertical.....	372
19.178	draw:handle-polar (deprecated).....	372
19.179	draw:handle-position (deprecated).....	372
19.180	draw:handle-radius-range-maximum.....	373
19.181	draw:handle-radius-range-minimum.....	374

19.182	draw:handle-range-x-maximum.....	374
19.183	draw:handle-range-x-minimum.....	374
19.184	draw:handle-range-y-maximum.....	374
19.185	draw:handle-range-y-minimum.....	374
19.186	draw:handle-switched.....	375
19.187	draw:id.....	375
19.187.1	General.....	375
19.187.2	<draw:glue-point>.....	375
19.187.3	<dr3d:cube>, <dr3d:extrude>, <dr3d:rotate>, <dr3d:scene>, <dr3d:sphere>, <draw:caption>, <draw:circle>, <draw:connector>, <draw:control>, <draw:custom-shape>, <draw:ellipse>, <draw:frame>, <draw:g>, <draw:line>, <draw:measure>, <draw:page>, <draw:page-thumbnail>, <draw:path>, <draw:polygon>, <draw:polyline>, <draw:rect>, <draw:regular-polygon> and <office:annotation> Deprecated.....	375
19.188	draw:kind.....	376
19.189	draw:layer.....	376
19.190	draw:line-skew.....	376
19.191	draw:mime-type.....	377
19.192	draw:mirror-horizontal.....	377
19.193	draw:mirror-vertical.....	377
19.194	draw:master-page-name.....	377
19.195	draw:may-script.....	377
19.196	draw:modifiers.....	378
19.197	draw:name.....	378
19.197.1	General.....	378
19.197.2	<draw:caption>.....	378
19.197.3	<draw:circle>.....	378
19.197.4	<draw:connector>.....	378
19.197.5	<draw:control>.....	379
19.197.6	<draw:custom-shape>.....	379
19.197.7	<draw:ellipse>.....	379
19.197.8	<draw:equation>.....	379
19.197.9	<draw:fill-image>.....	379
19.197.10	<draw:frame>.....	379
19.197.11	<draw:g>.....	379
19.197.12	<draw:gradient>.....	380
19.197.13	<draw:hatch>.....	380
19.197.14	<draw:layer>.....	380
19.197.15	<draw:line>.....	380
19.197.16	<draw:marker>.....	380
19.197.17	<draw:measure>.....	380

19.197.18 <draw:opacity>.....	380
19.197.19 <draw:page>.....	381
19.197.20 <draw:page-thumbnail>.....	381
19.197.21 <draw:param>.....	381
19.197.22 <draw:path>.....	381
19.197.23 <draw:polygon>.....	381
19.197.24 <draw:polyline>.....	381
19.197.25 <draw:rect>.....	381
19.197.26 <draw:regular-polygon>.....	382
19.197.27 <draw:stroke-dash>.....	382
19.197.28 <office:annotation> (deprecated).....	382
19.197.29 <svg:linearGradient>.....	382
19.197.30 <svg:radialGradient>.....	382
19.197.31 <dr3d:cube>.....	383
19.197.32 <dr3d:extrude>.....	383
19.197.33 <dr3d:rotate>.....	383
19.197.34 <dr3d:scene>.....	383
19.197.35 <dr3d:sphere>.....	383
19.198 draw:nav-order.....	383
19.199 draw:nohref.....	383
19.200 draw:notify-on-update-of-ranges.....	384
19.201 draw:object.....	384
19.202 draw:opacity.....	384
19.203 draw:page-number.....	384
19.204 draw:path-stretchpoint-x.....	384
19.205 draw:path-stretchpoint-y.....	385
19.206 draw:points.....	385
19.207 draw:protected.....	385
19.208 draw:recreate-on-edit.....	386
19.209 draw:rotation.....	386
19.210 draw:shape-id.....	386
19.211 draw:sharpness.....	386
19.212 draw:start.....	387
19.213 draw:start-angle.....	387
19.214 draw:start-color.....	387
19.215 draw:start-glue-point.....	388
19.216 draw:start-intensity.....	388
19.217 draw:start-shape.....	388
19.218 draw:style.....	388

19.218.1	General.....	388
19.218.2	<draw:gradient>.....	388
19.218.3	<draw:hatch>.....	389
19.218.4	<draw:opacity>.....	389
19.218.5	<draw:stroke-dash>.....	390
19.219	draw:style-name.....	390
19.219.1	General.....	390
19.219.2	<dr3d:cube>.....	390
19.219.3	<dr3d:extrude>.....	390
19.219.4	<dr3d:rotate>.....	390
19.219.5	<dr3d:scene>.....	390
19.219.6	<dr3d:sphere>.....	391
19.219.7	<draw:caption>.....	391
19.219.8	<draw:circle>.....	391
19.219.9	<draw:connector>.....	391
19.219.10	<draw:control>.....	391
19.219.11	<draw:custom-shape>.....	391
19.219.12	<draw:ellipse>.....	392
19.219.13	<draw:frame>.....	392
19.219.14	<draw:g>.....	392
19.219.15	<draw:line>.....	392
19.219.16	<draw:measure>.....	392
19.219.17	<draw:page>.....	392
19.219.18	<draw:page-thumbnail>.....	392
19.219.19	<draw:path>.....	393
19.219.20	<draw:polygon>.....	393
19.219.21	<draw:polyline>.....	393
19.219.22	<draw:rect>.....	393
19.219.23	<draw:regular-polygon>.....	393
19.219.24	<office:annotation>.....	393
19.219.25	<presentation:notes>.....	394
19.219.26	<style:handout-master>.....	394
19.219.27	<style:master-page>.....	394
19.220	draw:text-areas.....	394
19.221	draw:text-path.....	395
19.222	draw:text-path-allowed.....	395
19.223	draw:text-path-mode.....	395
19.224	draw:text-path-same-letter-heights.....	396
19.225	draw:text-path-scale.....	396

19.226	draw:text-rotate-angle.....	396
19.227	draw:text-style-name.....	396
19.228	draw:transform.....	397
19.229	draw:type.....	397
19.229.1	General.....	397
19.229.2	<draw:connector>.....	397
19.229.3	<draw:enhanced-geometry>.....	398
19.230	draw:value.....	398
19.231	draw:z-index.....	398
19.232	fo:column-count.....	399
19.233	fo:column-gap.....	399
19.234	fo:country.....	399
19.234.1	<text:alphabetical-index-source>.....	399
19.234.2	<text:bibliography-configuration>.....	399
19.235	fo:end-indent.....	400
19.236	fo:language.....	400
19.236.1	General.....	400
19.236.2	<text:alphabetical-index-source>.....	400
19.236.3	<text:bibliography-configuration>.....	400
19.237	fo:margin-left.....	400
19.238	fo:max-height.....	401
19.239	fo:max-width.....	401
19.240	fo:min-height.....	401
19.241	fo:min-width.....	401
19.242	fo:script.....	402
19.242.1	General.....	402
19.242.2	<text:alphabetical-index-source>.....	402
19.242.3	<text:bibliography-configuration>.....	402
19.243	fo:space-after.....	402
19.244	fo:space-before.....	402
19.245	fo:start-indent.....	403
19.246	fo:text-indent.....	403
19.247	form:allow-deletes.....	403
19.248	form:allow-inserts.....	403
19.249	form:allow-updates.....	404
19.250	form:apply-design-mode.....	404
19.251	form:apply-filter.....	404
19.252	form:auto-complete.....	404
19.253	form:automatic-focus.....	405

19.254	form:bound-column.....	405
19.255	form:button-type.....	405
19.256	form:command.....	406
19.257	form:command-type.....	406
19.258	form:control-implementation.....	406
19.259	form:convert-empty-to-null.....	407
19.260	form:current-selected.....	407
19.261	form:current-state.....	407
19.262	form:current-value.....	407
19.263	form:data-field.....	408
19.264	form:datasource.....	408
19.265	form:default-button.....	409
19.266	form:delay-for-repeat.....	409
19.267	form:detail-fields.....	409
19.268	form:disabled.....	409
19.269	form:dropdown.....	410
19.270	form:echo-char.....	410
19.271	form:enctype.....	410
19.272	form:escape-processing.....	410
19.273	form:filter.....	411
19.274	form:focus-on-click.....	411
19.275	form:for.....	411
19.276	form:id (deprecated).....	411
19.277	form:ignore-result.....	412
19.278	form:image-align.....	412
19.279	form:image-data.....	412
19.280	form:image-position.....	412
19.281	form:is-tristate.....	413
19.282	form:label.....	413
19.283	form:linked-cell.....	413
19.284	form:list-linkage-type.....	414
19.285	form:list-source.....	414
19.286	form:list-source-type.....	415
19.287	form:master-fields.....	415
19.288	form:max-length.....	416
19.289	form:max-value.....	416
19.289.1	General.....	416
19.289.2	<form:date>.....	416
19.289.3	<form:formatted-text>.....	416

19.289.4 <form:number> and <form:value-range>.....	416
19.289.5 <form:time>.....	416
19.290 form:min-value.....	416
19.290.1 General.....	416
19.290.2 <form:date>.....	416
19.290.3 <form:formatted-text>.....	417
19.290.4 <form:number> and <form:value-range>.....	417
19.290.5 <form:time>.....	417
19.291 form:method.....	417
19.292 form:multi-line.....	417
19.293 form:multiple.....	418
19.294 form:name.....	418
19.295 form:navigation-mode.....	418
19.296 form:order.....	418
19.297 form:orientation.....	419
19.298 form:page-step-size.....	419
19.299 form:printable.....	419
19.300 form:property-name.....	419
19.301 form:readonly.....	420
19.302 form:repeat.....	420
19.303 form:selected.....	420
19.304 form:size.....	420
19.305 form:source-cell-range.....	421
19.306 form:spin-button.....	421
19.307 form:state.....	421
19.308 form:step-size.....	421
19.309 form:tab-cycle.....	422
19.310 form:tab-index.....	422
19.311 form:tab-stop.....	423
19.312 form:text-style-name.....	423
19.313 form:title.....	423
19.314 form:toggle.....	423
19.315 form:validation.....	424
19.316 form:value.....	424
19.316.1 General.....	424
19.316.2 <form:button>.....	424
19.316.3 <form:date>.....	424
19.316.4 <form:number> and <form:value-range>.....	424

19.316.5 <form:checkbox>, <form:combobox>, <form:file>, <form:formatted-text>, <form:hidden>, <form:image>, <form:option>, <form:password>, <form:radio>, <form:text>, <form:textarea>.....	425
19.316.6 <form:time>.....	425
19.317 form:visual-effect.....	425
19.318 form:xforms-list-source.....	425
19.319 form:xforms-submission.....	425
19.320 grddl:transformation.....	426
19.321 meta:cell-count.....	426
19.322 meta:character-count.....	426
19.323 meta:date.....	426
19.324 meta:delay.....	426
19.325 meta:draw-count.....	427
19.326 meta:frame-count.....	427
19.327 meta:image-count.....	427
19.328 meta:name.....	427
19.329 meta:non-whitespace-character-count.....	427
19.330 meta:object-count.....	428
19.331 meta:ole-object-count.....	428
19.332 meta:page-count.....	428
19.333 meta:paragraph-count.....	428
19.334 meta:row-count.....	428
19.335 meta:sentence-count.....	428
19.336 meta:syllable-count.....	429
19.337 meta:table-count.....	429
19.338 meta:value-type.....	429
19.339 meta:word-count.....	429
19.340 number:automatic-order.....	429
19.341 number:calendar.....	430
19.342 number:country.....	430
19.343 number:decimal-places.....	431
19.343.1 General.....	431
19.343.2 <number:number>.....	431
19.343.3 <number:seconds>.....	431
19.343.4 <number:scientific-number>.....	431
19.344 number:decimal-replacement.....	431
19.345 number:denominator-value.....	432
19.346 number:display-factor.....	432
19.347 number:exponent-interval.....	432
19.348 number:format-source.....	432

19.349	number:forced-exponent-sign.....	433
19.350	number:grouping.....	433
19.351	number:language.....	433
19.352	number:max-denominator-value.....	433
19.353	number:min-denominator-digits.....	434
19.354	number:min-exponent-digits.....	434
19.355	number:min-integer-digits.....	434
19.356	number:min-decimal-places.....	434
19.356.1	General.....	434
19.356.2	<number:number>.....	435
19.356.3	<number:scientific-number>.....	435
19.357	number:min-numerator-digits.....	435
19.358	number:position.....	435
19.359	number:possessive-form.....	436
19.360	number:rfc-language-tag.....	436
19.361	number:script.....	436
19.362	number:style.....	436
19.362.1	General.....	436
19.362.2	<number:day>.....	437
19.362.3	<number:day-of-week>.....	437
19.362.4	<number:era>.....	437
19.362.5	<number:hours>.....	437
19.362.6	<number:minutes>.....	437
19.362.7	<number:month>.....	438
19.362.8	<number:quarter>.....	438
19.362.9	<number:seconds>.....	438
19.362.10	<number:year>.....	438
19.363	number:textual.....	439
19.364	number:title.....	439
19.365	number:transliteration-country.....	439
19.366	number:transliteration-format.....	439
19.367	number:transliteration-language.....	440
19.368	number:transliteration-style.....	440
19.369	number:truncate-on-overflow.....	440
19.370	office:automatic-update.....	441
19.371	office:boolean-value.....	441
19.372	office:conversion-mode.....	441
19.373	office:currency.....	442
19.374	office:date-value.....	442

19.375	office:dde-application.....	442
19.376	office:dde-item.....	443
19.377	office:dde-topic.....	443
19.378	office:display.....	443
19.379	office:mimetype.....	443
19.380	office:name.....	443
19.380.1	General.....	443
19.380.2	<draw:a>.....	443
19.380.3	<draw:area-circle>.....	443
19.380.4	<draw:area-rectangle>.....	444
19.380.5	<draw:area-polygon>.....	444
19.380.6	<office:annotation>.....	444
19.380.7	<office:annotation-end>.....	444
19.380.8	<office:dde-source>.....	444
19.380.9	<text:a>.....	445
19.380.10	<text:dde-connection-decl>.....	445
19.382	office:server-map.....	445
19.383	office:string-value.....	445
19.384	office:target-frame.....	446
19.385	office:target-frame-name.....	446
19.386	office:time-value.....	446
19.387	office:title.....	447
19.388	office:value.....	447
19.389	office:value-type.....	447
19.390	office:version.....	448
19.391	presentation:action.....	449
19.392	presentation:animations.....	449
19.393	presentation:class.....	450
19.394	presentation:class-names.....	451
19.395	presentation:delay (deprecated).....	451
19.396	presentation:direction.....	451
19.397	presentation:endless.....	452
19.398	presentation:effect.....	452
19.399	presentation:force-manual.....	453
19.400	presentation:full-screen.....	454
19.401	presentation:group-id.....	454
19.402	presentation:master-element.....	454
19.403	presentation:mouse-as-pen.....	454
19.404	presentation:mouse-visible.....	455

19.405	presentation:name.....	455
19.406	presentation:node-type.....	455
19.407	presentation:object.....	456
19.408	presentation:pages.....	456
19.409	presentation:path-id (deprecated).....	456
19.410	presentation:pause.....	456
19.411	presentation:placeholder.....	456
19.412	presentation:play-full.....	457
19.413	presentation:presentation-page-layout-name.....	457
19.414	presentation:preset-class.....	457
19.415	presentation:preset-id.....	458
19.416	presentation:preset-sub-type.....	458
19.417	presentation:show.....	458
19.418	presentation:show-end-of-presentation-slide.....	458
19.419	presentation:show-logo.....	458
19.420	presentation:source.....	459
19.421	presentation:speed.....	459
19.422	presentation:start-scale.....	459
19.423	presentation:start-page.....	460
19.424	presentation:start-with-navigator.....	460
19.425	presentation:stay-on-top.....	460
19.426	presentation:style-name.....	460
19.427	presentation:transition-on-click.....	461
19.428	presentation:use-date-time-name.....	461
19.429	presentation:use-footer-name.....	461
19.430	presentation:use-header-name.....	461
19.431	presentation:user-transformed.....	462
19.432	presentation:verb.....	462
19.433	script:event-name.....	462
19.434	script:language.....	463
19.435	script:macro-name.....	463
19.436	smil:accelerate.....	463
19.437	smil:accumulate.....	464
19.438	smil:additive.....	464
19.439	smil:attributeName.....	464
19.440	smil:autoReverse.....	465
19.441	smil:begin.....	465
19.442	smil:by.....	465
19.443	smil:calcMode.....	465

19.444	smil:decelerate.....	466
19.445	smil:direction.....	466
19.446	smil:dur.....	466
19.447	smil:end.....	466
19.448	smil:endsync.....	466
19.449	smil:fadeColor.....	467
19.450	smil:fill.....	467
19.451	smil:fillDefault.....	467
19.452	smil:from.....	467
19.453	smil:keySplines.....	467
19.454	smil:keyTimes.....	467
19.455	smil:mode.....	468
19.456	smil:repeatCount.....	468
19.457	smil:repeatDur.....	468
19.458	smil:restart.....	468
19.459	smil:restartDefault.....	468
19.460	smil:subtype.....	469
19.461	smil:targetElement.....	469
19.462	smil:to.....	469
19.463	smil:type.....	469
19.464	smil:values.....	470
19.465	style:adjustment.....	470
19.466	style:apply-style-name.....	470
19.467	style:auto-update.....	470
19.468	style:base-cell-address.....	471
19.469	style:char.....	471
19.470	style:class.....	471
19.471	style:color.....	471
19.472	style:condition.....	471
19.473	style:data-style-name.....	473
19.473.1	General.....	473
19.473.2	<presentation:date-time-decl>.....	473
19.473.3	<style:style>.....	473
19.473.4	<text:creation-date>.....	473
19.473.5	<text:creation-time>.....	473
19.473.6	<text:database-display>.....	473
19.473.7	<text:date>.....	473
19.473.8	<text:editing-duration>.....	474
19.473.9	<text:expression>.....	474

19.473.10 <text:meta-field>.....	474
19.473.11 <text:modification-date>.....	474
19.473.12 <text:modification-time>.....	474
19.473.13 <text:print-date>.....	474
19.473.14 <text:print-time>.....	474
19.473.15 <text:table-formula>.....	475
19.473.16 <text:time>.....	475
19.473.17 <text:user-defined>.....	475
19.473.18 <text:user-field-get>.....	475
19.473.19 <text:user-field-input>.....	475
19.473.20 <text:variable-get>.....	475
19.473.21 <text:variable-input>.....	476
19.473.22 <text:variable-set>.....	476
19.474 style:default-outline-level.....	476
19.475 style:display.....	476
19.476 style:display-name.....	477
19.477 style:distance.....	477
19.478 style:distance-before-sep.....	477
19.479 style:distance-after-sep.....	477
19.480 style:family.....	477
19.481 style:filter-name.....	478
19.482 style:font-adornments.....	478
19.483 style:font-charset.....	478
19.484 style:font-family-generic.....	479
19.485 style:font-pitch.....	479
19.486 style:height.....	479
19.487 style:leader-char.....	479
19.488 style:leader-color.....	480
19.489 style:leader-style.....	480
19.490 style:leader-text.....	480
19.491 style:leader-text-style.....	481
19.492 style:leader-type.....	481
19.493 style:leader-width.....	481
19.494 style:legend-expansion.....	481
19.495 style:legend-expansion-aspect-ratio.....	482
19.496 style:length.....	482
19.497 style:line-style.....	482
19.498 style:lines.....	482
19.499 style:list-level.....	483

19.500 style:list-style-name.....	483
19.501 style:master-page-name.....	483
19.502 style:name.....	483
19.502.1 General.....	483
19.502.2 <number:boolean-style>, <number:currency-style>, <number:date-style>, <number:num-list-format>, <number:number-style>, <number:percentage-style>, <number:text-style>, <number:time-style>, <style:master-page>, <style:page-layout>, <style:presentation-page-layout>, <style:style>, <text:list-style>, <text:outline-style>...	484
19.502.3 <style:font-face>.....	484
19.503 style:next-style-name.....	485
19.503.1 General.....	485
19.503.2 <style:master-page>.....	485
19.503.3 <style:style>.....	485
19.504 style:num-format.....	485
19.505 style:num-letter-sync.....	486
19.506 style:num-prefix.....	486
19.507 style:num-suffix.....	486
19.508 style:page-layout-name.....	487
19.509 style:page-usage.....	487
19.510 style:parent-style-name.....	488
19.511 style:percentage-data-style-name.....	488
19.512 style:position.....	489
19.512.1 General.....	489
19.512.2 <style:background-image>.....	489
19.512.3 <style:tab-stop>.....	489
19.512.4 <text:index-entry-tab-stop>.....	489
19.513 style:rel-height.....	489
19.514 style:rel-width.....	490
19.514.1 General.....	490
19.514.2 <draw:frame>.....	490
19.514.3 <style:column>.....	490
19.514.4 <style:footnote-sep>.....	491
19.515 style:repeat.....	491
19.516 style:rfc-language-tag.....	491
19.517 style:style.....	491
19.518 style:style-name.....	492
19.519 style:type.....	492
19.519.1 General.....	492
19.519.2 <text:index-entry-tab-stop>.....	492
19.519.3 <style:tab-stop>.....	492

19.520	style:vertical-align.....	493
19.521	style:volatile.....	493
19.522	style:width.....	493
19.522.1	General.....	493
19.522.2	<style:column-sep>.....	493
19.522.3	<style:footnote-sep>.....	493
19.523	svg:accent-height.....	494
19.524	svg:alphabetic.....	494
19.525	svg:ascent.....	494
19.526	svg:bbox.....	494
19.527	svg:cap-height.....	494
19.528	svg:cx.....	494
19.528.1	<draw:area-circle>.....	494
19.528.2	<draw:circle>.....	494
19.528.3	<draw:ellipse>.....	495
19.528.4	<svg:radialGradient>.....	495
19.529	svg:cy.....	495
19.529.1	<draw:area-circle>.....	495
19.529.2	<draw:circle>.....	495
19.529.3	<draw:ellipse>.....	495
19.529.4	<svg:radialGradient>.....	495
19.530	svg:d.....	496
19.531	svg:descent.....	496
19.532	svg:font-family.....	496
19.533	svg:font-size.....	496
19.534	svg:font-stretch.....	496
19.535	svg:font-style.....	496
19.536	svg:font-variant.....	496
19.537	svg:font-weight.....	497
19.538	svg:fx.....	497
19.539	svg:fy.....	497
19.540	svg:gradientTransform.....	497
19.541	svg:gradientUnits.....	497
19.542	svg:hanging.....	497
19.543	svg:height.....	498
19.543.1	<chart:chart>.....	498
19.543.2	<chart:legend>.....	498
19.543.3	<chart:coordinate-region>.....	498
19.543.4	<chart:plot-area>.....	498

19.543.5 <dr3d:scene>.....	498
19.543.6 <draw:area-polygon>.....	498
19.543.7 <draw:area-rectangle>.....	498
19.543.8 <draw:contour-polygon>, <draw:contour-path>.....	499
19.543.9 <draw:fill-image>.....	499
19.543.10 <draw:caption>, <draw:custom-shape>, <draw:circle>, <draw:control>, <draw:ellipse>, <draw:frame>, <draw:rect>, <draw:page-thumbnail>, <draw:path>, <draw:polygon>, <draw:polyline>, <draw:regular-polygon>.....	499
19.543.11 <office:annotation>.....	499
19.543.12 <presentation:placeholder>.....	499
19.544 svg:ideographic.....	500
19.545 svg:mathematical.....	500
19.546 svg:name.....	500
19.547 svg:offset.....	500
19.548 svg:origin.....	500
19.549 svg:overline-position.....	500
19.550 svg:overline-thickness.....	500
19.551 svg:panose-1.....	501
19.552 svg:path.....	501
19.553 svg:r.....	501
19.553.1 <draw:area-circle>.....	501
19.553.2 <draw:circle>.....	501
19.553.3 <svg:radialGradient>.....	501
19.554 svg:rx.....	501
19.555 svg:ry.....	502
19.556 svg:slope.....	502
19.557 svg:spreadMethod.....	502
19.558 svg:stemh.....	502
19.559 svg:stemv.....	502
19.560 svg:stop-color.....	503
19.561 svg:stop-opacity.....	503
19.562 svg:strikethrough-position.....	503
19.563 svg:strikethrough-thickness.....	503
19.564 svg:string.....	503
19.565 svg:type.....	503
19.566 svg:underline-position.....	503
19.567 svg:underline-thickness.....	504
19.568 svg:unicode-range.....	504
19.569 svg:units-per-em.....	504
19.570 svg:v-alphabetic.....	504

19.571	svg:v-hanging.....	504
19.572	svg:v-ideographic.....	504
19.573	svg:v-mathematical.....	504
19.574	svg:viewBox.....	505
19.575	svg:width.....	505
19.575.1	<chart:chart>.....	505
19.575.2	<chart:coordinate-region>.....	505
19.575.3	<chart:floor>.....	505
19.575.4	<chart:legend>.....	505
19.575.5	<chart:plot-area>.....	506
19.575.6	<chart:wall>.....	506
19.575.7	<dr3d:scene>.....	506
19.575.8	<draw:area-polygon>.....	506
19.575.9	<draw:area-rectangle>.....	506
19.575.10	<draw:contour-polygon>, <draw:contour-path>.....	506
19.575.11	<draw:fill-image>.....	506
19.575.12	<draw:caption>, <draw:custom-shape>, <draw:circle>, <draw:control>, <draw:ellipse>, <draw:frame>, <draw:rect>, <draw:page-thumbnail>, <draw:path>, <draw:polygon>, <draw:polyline>, <draw:regular-polygon>.....	507
19.575.13	<office:annotation>.....	507
19.575.14	<presentation:placeholder>.....	507
19.576	svg:widths.....	507
19.577	svg:x.....	507
19.577.1	<chart:coordinate-region>.....	507
19.577.2	<chart:data-label>, <chart:equation>, <chart:footer>, <chart:legend>, <chart:plot-area>, <chart:subtitle>, <chart:title>.....	508
19.577.3	<draw:area-polygon>.....	508
19.577.4	<draw:area-rectangle>.....	508
19.577.5	<dr3d:scene>.....	508
19.577.6	<draw:caption>, <draw:custom-shape>, <draw:circle>, <draw:control>, <draw:ellipse>, <draw:frame>, <draw:rect>, <draw:page-thumbnail>, <draw:path>, <draw:polygon>, <draw:polyline>, <draw:regular-polygon>.....	508
19.577.7	<draw:glue-point>.....	509
19.577.8	<office:annotation>.....	509
19.577.9	<presentation:placeholder>.....	509
19.578	svg:x1.....	509
19.578.1	<draw:connector>, <draw:line> and <draw:measure>.....	509
19.578.2	<svg:linearGradient>.....	509
19.579	svg:x2.....	509
19.579.1	<draw:connector>, <draw:line> and <draw:measure>.....	509
19.579.2	<svg:linearGradient>.....	510

19.580	svg:x-height.....	510
19.581	svg:y.....	510
19.581.1	<chart:coordinate-region>.....	510
19.581.2	<chart:data-label>, <chart:equation>, <chart:footer>, <chart:legend>, <chart:plot-area>, <chart:subtitle>, <chart:title>.....	510
19.581.3	<draw:area-polygon>.....	510
19.581.4	<draw:area-rectangle>.....	511
19.581.5	<dr3d:scene>.....	511
19.581.6	<draw:caption>, <draw:custom-shape>, <draw:circle>, <draw:control>, <draw:ellipse>, <draw:frame>, <draw:g>, <draw:rect>, <draw:page-thumbnail>, <draw:path>, <draw:polygon>, <draw:polyline>, <draw:regular-polygon>.....	511
19.581.7	<draw:glue-point>.....	511
19.581.8	<office:annotation>.....	511
19.581.9	<presentation:placeholder>.....	511
19.582	svg:y1.....	512
19.582.1	<draw:connector>, <draw:line> and <draw:measure>.....	512
19.582.2	<svg:linearGradient>.....	512
19.583	svg:y2.....	512
19.583.1	<draw:connector>, <draw:line> and <draw:measure>.....	512
19.583.2	<svg:linearGradient>.....	512
19.584	table:acceptance-state.....	513
19.585	table:algorithm.....	513
19.586	table:add-empty-lines.....	513
19.587	table:allow-empty-cell.....	513
19.588	table:application-data.....	514
19.589	table:automatic-find-labels.....	514
19.590	table:base-cell-address.....	514
19.591	table:bind-styles-to-content.....	514
19.592	table:border-color.....	515
19.593	table:buttons.....	515
19.594	table:case-sensitive.....	515
19.595	table:cell-address.....	515
19.596	table:cell-range.....	516
19.597	table:cell-range-address.....	516
19.597.1	General.....	516
19.597.2	<chart:categories>.....	516
19.597.3	<chart:domain> - bubble.....	516
19.597.4	<chart:domain>- scatter.....	516
19.597.5	<chart:domain> - surface.....	516
19.597.6	<chart:plot-area> (removed).....	517

19.597.7 <table:highlighted-range>.....	517
19.597.8 <table:named-range>.....	517
19.597.9 <table:source-cell-range>.....	517
19.598 table:column.....	517
19.599 table:comment.....	518
19.600 table:condition.....	518
19.601 table:condition-source.....	519
19.602 table:condition-source-range-address.....	520
19.603 table:contains-error.....	520
19.604 table:contains-header.....	520
19.605 table:content-validation-name.....	520
19.606 table:copy-back.....	521
19.607 table:copy-formulas.....	521
19.608 table:copy-styles.....	521
19.609 table:count.....	521
19.610 table:country.....	522
19.611 table:data-field.....	522
19.612 table:data-cell-range-address.....	522
19.613 table:database-name.....	522
19.614 table:database-table-name.....	522
19.615 table:data-type.....	522
19.615.1 General.....	522
19.615.2 <table:filter-condition>.....	522
19.615.3 <table:sort-by>.....	523
19.615.4 <table:sort-groups>.....	523
19.616 table:date-end.....	524
19.617 table:date-start.....	524
19.618 table:date-value.....	524
19.619 table:default-cell-style-name.....	524
19.620 table:direction.....	525
19.621 table:display.....	525
19.622 table:display-border.....	525
19.623 table:display-duplicates.....	526
19.624 table:display-filter-buttons.....	526
19.625 table:display-list.....	526
19.626 table:display-member-mode.....	527
19.627 table:drill-down-on-double-click.....	527
19.628 table:embedded-number-behavior.....	527
19.629 table:enabled.....	529

19.630	table:end.....	529
19.631	table:end-cell-address.....	529
19.632	table:end-column.....	529
19.633	table:end-position.....	529
19.634	table:end-row.....	530
19.635	table:end-table.....	530
19.636	table:end-x.....	530
19.637	table:end-y.....	530
19.638	table:execute.....	530
19.639	table:expression.....	531
19.640	table:field-name.....	531
19.641	table:field-number.....	531
19.642	table:filter-name.....	532
19.643	table:filter-options.....	532
19.644	table:first-row-end-column (deprecated).....	532
19.645	table:first-row-start-column (deprecated).....	532
19.646	table:formula.....	532
19.647	table:function.....	533
19.647.1	General.....	533
19.647.2	<table:consolidation>.....	534
19.647.3	<table:data-pilot-field>.....	534
19.647.4	<table:data-pilot-subtotal>.....	534
19.647.5	<table:subtotal-field>.....	535
19.648	table:grand-total.....	535
19.649	table:group-by-field-number.....	535
19.650	table:grouped-by.....	535
19.651	table:has-persistent-data.....	536
19.652	table:id.....	536
19.653	table:identify-categories.....	536
19.654	table:ignore-empty-rows.....	537
19.655	table:index.....	537
19.656	table:is-active.....	537
19.657	table:is-data-layout-field.....	537
19.658	table:is-selection.....	538
19.659	table:is-sub-table.....	538
19.660	table:label-cell-range-address.....	538
19.661	table:language.....	538
19.662	table:last-column-spanned.....	539
19.663	table:last-row-end-column (deprecated).....	539

19.664	table:last-row-spanned.....	539
19.665	table:last-row-start-column (deprecated).....	539
19.666	table:layout-mode.....	539
19.667	table:link-to-source-data.....	540
19.668	table:marked-invalid.....	540
19.669	table:matrix-covered.....	540
19.670	table:maximum-difference.....	541
19.671	table:member-count.....	541
19.672	table:member-name.....	541
19.673	table:member-type.....	541
19.674	table:message-type.....	541
19.675	table:mode.....	542
19.676	table:multi-deletion-spanned.....	542
19.677	table:name.....	542
19.677.1	General.....	542
19.677.2	<table:cell-range-source>.....	542
19.677.3	<table:content-validation>.....	542
19.677.4	<table:data-pilot-group>.....	543
19.677.5	<table:data-pilot-group-member>.....	543
19.677.6	<table:data-pilot-member>.....	543
19.677.7	<table:data-pilot-table>.....	543
19.677.8	<table:database-range>.....	543
19.677.9	<table:named-expression>.....	543
19.677.10	<table:named-range>.....	544
19.677.11	<table:operation>.....	544
19.677.12	<table:source-cell-range>.....	544
19.677.13	<table:source-service>.....	544
19.677.14	<table:table>.....	544
19.677.15	<table:table-template>.....	545
19.678	table:null-year.....	545
19.679	table:number-columns-repeated.....	545
19.679.1	General.....	545
19.679.2	<table:covered-table-cell>.....	545
19.679.3	<table:table-cell>.....	545
19.679.4	<table:table-column>.....	546
19.680	table:number-columns-spanned.....	546
19.681	table:number-rows-repeated.....	546
19.682	table:number-rows-spanned.....	546
19.683	table:number-matrix-columns-spanned.....	547

19.684	table:number-matrix-rows-spanned.....	547
19.685	table:object-name.....	547
19.686	table:on-update-keep-size.....	547
19.687	table:on-update-keep-styles.....	547
19.688	table:operator.....	548
19.689	table:order.....	549
19.690	table:orientation.....	549
19.690.1	General.....	549
19.690.2	<table:database-range>.....	549
19.690.3	<table:label-range>.....	550
19.690.4	<table:label-range>.....	550
19.691	table:page-breaks-on-group-change.....	550
19.692	table:paragraph-style-name.....	550
19.693	table:parse-sql-statement.....	551
19.694	table:password.....	551
19.695	table:position.....	551
19.696	table:precision-as-shown.....	551
19.697	table:print.....	552
19.698	table:print-ranges.....	552
19.699	table:protect (deprecated).....	552
19.700	table:protected.....	552
19.700.1	General.....	552
19.700.2	<table:covered-table-cell>.....	552
19.700.3	<table:scenario>.....	553
19.700.4	<table:table>.....	553
19.700.5	<table:table-cell>.....	553
19.701	table:protection-key.....	554
19.702	table:protection-key-digest-algorithm.....	554
19.703	table:query-name.....	554
19.704	table:range-usable-as.....	554
19.705	table:refresh-delay.....	555
19.706	table:rejecting-change-id.....	555
19.707	table:rfc-language-tag.....	555
19.708	table:row.....	555
19.709	table:scenario-ranges.....	556
19.710	table:script.....	556
19.711	table:search-criteria-must-apply-to-whole-cell.....	556
19.712	table:selected-page.....	556
19.713	table:show-details.....	557

19.714	table:show-empty.....	557
19.715	table:show-filter-button.....	557
19.716	table:sort-mode.....	557
19.717	table:source-cell-range-addresses.....	558
19.718	table:source-field-name.....	558
19.719	table:source-name.....	558
19.720	table:sql-statement.....	558
19.721	table:start.....	558
19.722	table:start-column.....	559
19.723	table:start-position.....	559
19.724	table:start-row.....	559
19.725	table:start-table.....	559
19.726	table:status.....	559
19.727	table:step.....	560
19.728	table:steps.....	560
19.729	table:structure-protected.....	560
19.730	table:style-name.....	560
19.730.1	General.....	560
19.730.2	<table:background>.....	560
19.730.3	<table:body>.....	561
19.730.4	<table:covered-table-cell>.....	561
19.730.5	<table:even-columns>.....	561
19.730.6	<table:even-rows>.....	561
19.730.7	<table:first-column>.....	561
19.730.8	<table:first-row>.....	561
19.730.9	<table:last-column>.....	562
19.730.10	<table:last-row>.....	562
19.730.11	<table:odd-columns>.....	562
19.730.12	<table:odd-rows>.....	562
19.730.13	<table:table-cell>.....	562
19.730.14	<table:table-column>.....	562
19.730.15	<table:table-row>.....	563
19.730.16	<table:table>.....	563
19.731	table:tab-color (moved to 20.442).....	563
19.732	table:table.....	563
19.733	table:table-background.....	563
19.734	table:table-name.....	564
19.735	table:target-cell-address.....	564
19.736	table:target-range-address.....	564

19.736.1	General.....	564
19.736.2	<table:database-range>.....	564
19.736.3	<table:filter>.....	564
19.736.4	<table:sort>.....	565
19.737	table:template-name.....	565
19.738	table:title.....	565
19.739	table:track-changes.....	565
19.740	table:type.....	565
19.740.1	General.....	565
19.740.2	<table:filter-condition>.....	565
19.740.3	<table:filter-set-item>.....	566
19.740.4	<table:insertion>.....	566
19.741	table:use-banding-columns-styles.....	567
19.742	table:use-banding-rows-styles.....	567
19.743	table:use-first-column-styles.....	567
19.744	table:use-first-row-styles.....	568
19.745	table:use-last-column-styles.....	568
19.746	table:use-last-row-styles.....	568
19.747	table:use-labels.....	568
19.748	table:use-regular-expressions.....	569
19.749	table:use-wildcards.....	569
19.750	table:used-hierarchy.....	570
19.751	table:user-name.....	570
19.752	table:value.....	570
19.752.1	<table:filter-condition>.....	570
19.752.2	<table:filter-set-item>.....	570
19.753	table:value-type.....	571
19.754	table:visibility.....	571
19.755	text:active.....	571
19.756	text:address.....	571
19.757	text:alphabetical-separators.....	571
19.758	text:anchor-page-number.....	572
19.759	text:anchor-type.....	572
19.760	text:annotate.....	573
19.761	text:author.....	573
19.762	text:bibliography-data-field.....	574
19.763	text:bibliography-type.....	575
19.764	text:booktitle.....	576

19.765	text:bullet-char.....	576
19.766	text:bullet-relative-size.....	576
19.767	text:capitalize-entries.....	576
19.768	text:c.....	577
19.769	text:caption-sequence-format.....	577
19.770	text:caption-sequence-name.....	577
19.771	text:change-id.....	578
19.772	text:chapter.....	578
19.773	text:citation-body-style-name.....	578
19.774	text:citation-style-name.....	578
19.775	text:class-names.....	578
19.775.1	General.....	578
19.775.2	<text:h>.....	578
19.775.3	<text:p>.....	579
19.775.4	<text:span>.....	579
19.776	text:column-name.....	579
19.777	text:combine-entries.....	579
19.778	text:combine-entries-with-dash.....	579
19.779	text:combine-entries-with-pp.....	580
19.780	text:comma-separated.....	580
19.781	text:cond-style-name.....	580
19.782	text:condition.....	580
19.782.1	General.....	580
19.782.2	<text:conditional-text>.....	581
19.782.3	<text:database-next>.....	581
19.782.4	<text:database-row-select>.....	581
19.782.5	<text:hidden-paragraph>.....	581
19.782.6	<text:hidden-text>.....	581
19.782.7	<text:section>.....	582
19.783	text:connection-name.....	582
19.784	text:consecutive-numbering.....	582
19.785	text:continue-list.....	582
19.786	text:continue-numbering.....	582
19.787	text:copy-outline-levels.....	583
19.788	text:count-empty-lines.....	583
19.789	text:count-in-text-boxes.....	583
19.790	text:current-selected.....	584
19.791	text:current-value.....	584
19.792	text:custom1.....	584

19.793	text:custom2.....	584
19.794	text:custom3.....	584
19.795	text:custom4.....	584
19.796	text:custom5.....	585
19.797	text:database-name.....	585
19.798	text:date-adjust.....	585
19.799	text:date-value.....	585
19.800	text:default-style-name.....	585
19.801	text:description.....	586
19.802	text:display.....	586
19.802.1	General.....	586
19.802.2	<text:chapter>.....	586
19.802.3	<text:expression>.....	586
19.802.4	<text:file-name>.....	586
19.802.5	<text:index-entry-chapter>.....	587
19.802.6	<text:section>.....	587
19.802.7	<text:table-formula>.....	588
19.802.8	<text:template-name>.....	588
19.802.9	<text:user-field-get>.....	588
19.802.10	<text:variable-get>.....	588
19.802.11	<text:variable-input>.....	589
19.802.12	<text:variable-set>.....	589
19.803	text:display-levels.....	589
19.804	text:display-outline-level.....	589
19.805	text:duration.....	590
19.806	text:edition.....	590
19.807	text:editor.....	590
19.808	text:filter-name.....	590
19.809	text:fixed.....	590
19.810	text:footnotes-position.....	591
19.811	text:formula.....	591
19.812	text:global (deprecated).....	591
19.813	text:howpublished.....	592
19.814	text:increment.....	592
19.815	text:id.....	592
19.815.1	General.....	592
19.815.2	<draw:text-box> (deprecated).....	592
19.815.3	<text:alphabetical-index-mark-end>.....	592
19.815.4	<text:alphabetical-index-mark-start>.....	592

19.815.5 <text:changed-region> (deprecated).....	593
19.815.6 <text:h> (deprecated).....	593
19.815.7 <text:note>.....	593
19.815.8 <text:p> (deprecated).....	593
19.815.9 <text:toc-mark-end>.....	594
19.815.10 <text:toc-mark-start>.....	594
19.815.11 <text:user-index-mark-end>.....	594
19.815.12 <text:user-index-mark-start>.....	594
19.816 text:identifier.....	594
19.817 text:index-name.....	595
19.818 text:index-scope.....	595
19.819 text:ignore-case.....	595
19.820 text:institution.....	596
19.821 text:is-hidden.....	596
19.822 text:is-list-header.....	596
19.823 text:isbn.....	596
19.824 text:issn.....	596
19.825 text:journal.....	597
19.826 text:key.....	597
19.827 text:key1.....	598
19.828 text:key2.....	598
19.829 text:key1-phonetic.....	598
19.830 text:key2-phonetic.....	598
19.831 text:kind.....	599
19.832 text:label.....	599
19.833 text:label-followed-by.....	599
19.834 text:level.....	599
19.835 text:list-id (deprecated).....	599
19.836 text:list-tab-stop-position.....	600
19.837 text:main-entry.....	600
19.838 text:main-entry-style-name.....	601
19.839 text:master-page-name.....	601
19.840 text:month.....	601
19.841 text:name.....	601
19.841.1 General.....	601
19.841.2 <text:alphabetical-index>.....	601
19.841.3 <text:bibliography>.....	601
19.841.4 <text:bookmark>.....	602
19.841.5 <text:bookmark-end>.....	602

19.841.6 <text:bookmark-start>.....	602
19.841.7 <text:drop-down>.....	602
19.841.8 <text:execute-macro>.....	602
19.841.9 <text:illustration-index>.....	602
19.841.10 <text:index-title>.....	602
19.841.11 <text:object-index>.....	603
19.841.12 <text:reference-mark>.....	603
19.841.13 <text:reference-mark-end>.....	603
19.841.14 <text:reference-mark-start>.....	603
19.841.15 <text:section>.....	603
19.841.16 <text:sequence>.....	603
19.841.17 <text:sequence-decl>.....	603
19.841.18 <text:table-of-content>.....	604
19.841.19 <text:table-index>.....	604
19.841.20 <text:user-defined>.....	604
19.841.21 <text:user-field-decl>.....	604
19.841.22 <text:user-field-get>.....	604
19.841.23 <text:user-field-input>.....	604
19.841.24 <text:user-index>.....	604
19.841.25 <text:variable-set>.....	605
19.841.26 <text:variable-decl>.....	605
19.841.27 <text:variable-get>.....	605
19.841.28 <text:variable-input>.....	605
19.842 text:note.....	605
19.843 text:note-class.....	605
19.844 text:number.....	606
19.845 text:number-lines.....	606
19.846 text:number-position.....	606
19.847 text:numbered-entries.....	606
19.848 text:offset.....	607
19.849 text:organizations.....	607
19.850 text:outline-level.....	607
19.850.1 General.....	607
19.850.2 <text:alphabetical-index-entry-template>.....	607
19.850.3 <text:chapter>.....	607
19.850.4 <text:h>.....	607
19.850.5 <text:index-entry-chapter>.....	608
19.850.6 <text:index-source-styles>.....	608
19.850.7 <text:user-index-mark>.....	608

19.850.8 <text:table-of-content-entry-template>.....	608
19.850.9 <text:table-of-content-source>.....	608
19.850.10 <text:toc-mark>.....	608
19.850.11 <text:toc-mark-start>.....	608
19.850.12 <text:user-index-entry-template>.....	609
19.850.13 <text:user-index-mark-start>.....	609
19.851 text:page-adjust.....	609
19.851.1 General.....	609
19.851.2 <text:page-number>.....	609
19.851.3 <text:page-variable-set>.....	609
19.852 text:pages.....	609
19.853 text:placeholder-type.....	610
19.854 text:prefix.....	610
19.855 text:protected.....	610
19.856 text:protection-key.....	610
19.857 text:protection-key-digest-algorithm.....	611
19.858 text:publisher.....	611
19.859 text:ref-name.....	611
19.859.1 General.....	611
19.859.2 <text:bookmark-ref>.....	611
19.859.3 <text:note-ref>.....	612
19.859.4 <text:reference-ref>.....	612
19.859.5 <text:sequence>.....	612
19.859.6 <text:sequence-ref>.....	612
19.860 text:reference-format.....	612
19.861 text:relative-tab-stop-position.....	613
19.862 text:report-type.....	614
19.863 text:restart-numbering.....	614
19.864 text:restart-on-page.....	614
19.865 text:row-number.....	614
19.866 text:school.....	615
19.867 text:section-name.....	615
19.868 text:series.....	615
19.869 text:select-page.....	615
19.869.1 General.....	615
19.869.2 <text:page-continuation>.....	615
19.869.3 <text:page-number>.....	615
19.870 text:separation-character.....	616
19.871 text:sort-algorithm.....	616

19.872	text:sort-ascending.....	616
19.873	text:sort-by-position.....	616
19.874	text:start-value.....	617
19.874.1	General.....	617
19.874.2	<text:h>.....	617
19.874.3	<text:list-item>.....	617
19.874.4	<text:list-level-style-number>.....	617
19.874.5	<text:notes-configuration>.....	617
19.874.6	<text:numbered-paragraph> (deprecated).....	618
19.874.7	<text:outline-level-style>.....	618
19.875	text:start-numbering-at.....	618
19.876	text:string-value-phonetic.....	618
19.877	text:string-value.....	618
19.877.1	General.....	618
19.877.2	<text:alphabetical-index-mark>.....	618
19.877.3	<text:hidden-text>.....	619
19.877.4	<text:page-continuation>.....	619
19.877.5	<text:toc-mark>.....	619
19.877.6	<text:user-index-mark>.....	619
19.878	text:string-value-if-false.....	619
19.879	text:string-value-if-true.....	619
19.880	text:style-name.....	620
19.880.1	General.....	620
19.880.2	<text:a>.....	620
19.880.3	<text:alphabetical-index>.....	620
19.880.4	<text:alphabetical-index-entry-template>.....	620
19.880.5	<text:bibliography>.....	620
19.880.6	<text:bibliography-entry-template>.....	620
19.880.7	<text:h>.....	620
19.880.8	<text:illustration-index>.....	621
19.880.9	<text:illustration-index-entry-template>.....	621
19.880.10	<text:index-entry-bibliography>.....	621
19.880.11	<text:index-entry-chapter>.....	621
19.880.12	<text:index-entry-link-end>.....	621
19.880.13	<text:index-entry-link-start>.....	621
19.880.14	<text:index-entry-page-number>.....	621
19.880.15	<text:index-entry-span>.....	622
19.880.16	<text:index-entry-tab-stop>.....	622
19.880.17	<text:index-entry-text>.....	622

19.880.18 <text:index-source-style>.....	622
19.880.19 <text:index-title>.....	622
19.880.20 <text:index-title-template>.....	622
19.880.21 <text:linenumbering-configuration>.....	622
19.880.22 <text:list>.....	623
19.880.23 <text:list-level-style-number>.....	623
19.880.24 <text:list-level-style-bullet>.....	623
19.880.25 <text:numbered-paragraph> (deprecated).....	623
19.880.26 <text:object-index>.....	624
19.880.27 <text:object-index-entry-template>.....	624
19.880.28 <text:outline-level-style>.....	624
19.880.29 <text:p>.....	624
19.880.30 <text:ruby>.....	624
19.880.31 <text:ruby-text>.....	624
19.880.32 <text:section>.....	624
19.880.33 <text:span>.....	625
19.880.34 <text:table-index-entry-template>.....	625
19.880.35 <text:table-of-content>.....	625
19.880.36 <text:table-of-content-entry-template>.....	625
19.880.37 <text:table-index>.....	625
19.880.38 <text:user-index>.....	625
19.880.39 <text:user-index-entry-template>.....	626
19.881 text:style-override.....	626
19.882 text:suffix.....	626
19.883 text:tab-ref.....	626
19.884 text:table-name.....	626
19.885 text:table-type.....	626
19.886 text:time-adjust.....	627
19.887 text:time-value.....	627
19.888 text:title.....	627
19.889 text:track-changes.....	627
19.890 text:url.....	627
19.891 text:use-caption.....	628
19.892 text:use-chart-objects.....	628
19.893 text:use-draw-objects.....	628
19.894 text:use-floating-frames.....	628
19.895 text:use-graphics.....	629
19.896 text:use-index-marks.....	629
19.897 text:use-index-source-styles.....	629

19.898	text:use-keys-as-entries.....	630
19.899	text:use-math-objects.....	630
19.900	text:use-objects.....	630
19.901	text:use-other-objects.....	631
19.902	text:use-outline-level.....	631
19.903	text:use-soft-page-breaks.....	631
19.904	text:use-spreadsheet-objects.....	632
19.905	text:use-tables.....	632
19.906	text:value.....	633
19.906.1	General.....	633
19.906.2	<text:label>.....	633
19.906.3	<text:database-row-number>.....	633
19.907	text:visited-style-name.....	633
19.908	text:volume.....	633
19.909	text:year.....	633
19.910	xforms:bind.....	633
19.911	xhtml:about.....	634
19.912	xhtml:content.....	634
19.913	xhtml:datatype.....	635
19.914	xhtml:property.....	635
19.915	xlink:actuate.....	635
19.916	xlink:href.....	636
19.916.1	General.....	636
19.916.2	<anim:audio>.....	636
19.916.3	<chart:chart>.....	636
19.916.4	<chart:symbol-image>.....	636
19.916.5	<db:component>.....	636
19.916.6	<db:connection-resource>.....	637
19.916.7	<db:file-based-database>.....	637
19.916.8	<draw:a>.....	637
19.916.9	<draw:applet>.....	637
19.916.10	<draw:area-circle>.....	637
19.916.11	<draw:area-polygon>.....	637
19.916.12	<draw:area-rectangle>.....	637
19.916.13	<draw:fill-image>.....	638
19.916.14	<draw:floating-frame>.....	638
19.916.15	<draw:image>.....	638
19.916.16	<draw:object>.....	638
19.916.17	<draw:object-ole>.....	638

19.916.18 <draw:plugin>.....	638
19.916.19 <form:button>.....	638
19.916.20 <form:connection-resource>.....	639
19.916.21 <form:image>.....	639
19.916.22 <form:form>.....	639
19.916.23 <meta:auto-reload>.....	639
19.916.24 <meta:template>.....	639
19.916.25 <presentation:event-listener>.....	639
19.916.26 <presentation:sound>.....	639
19.916.27 <script:event-listener>.....	640
19.916.28 <style:background-image>.....	640
19.916.29 <svg:definition-src>.....	640
19.916.30 <svg:font-face-uri>.....	640
19.916.31 <table:cell-range-source>.....	640
19.916.32 <table:table-source>.....	640
19.916.33 <text:a>.....	640
19.916.34 <text:alphabetical-index-auto-mark-file>.....	641
19.916.35 <text:list-level-style-image>.....	641
19.916.36 <text:script>.....	641
19.916.37 <text:section-source>.....	641
19.917 xlink:show.....	641
19.918 xlink:title.....	642
19.919 xlink:type.....	642
19.920 xml:id.....	642
19.921 form:input-required.....	643
19.922 style:num-list-format-name.....	644
19.923 draw:extrusion-metal-type.....	644
19.924 draw:handle-position-x.....	644
19.925 draw:handle-position-y.....	644
19.926 draw:handle-polar-pole-x.....	645
19.927 draw:handle-polar-pole-y.....	645
19.928 draw:handle-polar-radius.....	645
19.929 draw:handle-polar-angle.....	646
20 Formatting Attributes.....	647
20.1 General.....	647
20.2 chart:angle-offset.....	647
20.3 chart:auto-position.....	647
20.4 chart:auto-size.....	647
20.5 chart:axis-label-position.....	648

20.6	chart:axis-position.....	648
20.7	chart:connect-bars.....	649
20.8	chart:data-label-number.....	649
20.9	chart:data-label-series.....	649
20.10	chart:data-label-symbol.....	650
20.11	chart:data-label-text.....	650
20.12	chart:deep.....	650
20.13	chart:display-label.....	651
20.14	chart:error-category.....	651
20.15	chart:error-lower-indicator.....	652
20.16	chart:error-lower-limit.....	652
20.17	chart:error-lower-range.....	652
20.18	chart:error-margin.....	652
20.19	chart:error-percentage.....	652
20.20	chart:error-upper-indicator.....	653
20.21	chart:error-upper-limit.....	653
20.22	chart:error-upper-range.....	653
20.23	chart:gap-width.....	653
20.24	chart:group-bars-per-axis.....	654
20.25	chart:hole-size.....	654
20.26	chart:include-hidden-cells.....	654
20.27	chart:interpolation.....	654
20.28	chart:interval-major.....	657
20.29	chart:interval-minor-divisor.....	658
20.30	chart:japanese-candle-stick.....	658
20.31	chart:label-arrangement.....	658
20.32	chart:label-position.....	659
20.33	chart:label-position-negative.....	659
20.34	chart:lines (deprecated).....	660
20.35	chart:link-data-style-to-source.....	660
20.36	chart:logarithmic.....	661
20.37	chart:maximum.....	661
20.38	chart:mean-value.....	661
20.39	chart:minimum.....	662
20.40	chart:origin.....	662
20.41	chart:overlap.....	662
20.42	chart:pie-offset.....	662
20.43	chart:percentage.....	663
20.44	chart:regression-force-intercept.....	663

20.45	chart:regression-intercept-value.....	663
20.46	chart:regression-max-degree.....	663
20.47	chart:regression-moving-type.....	664
20.48	chart:regression-name.....	664
20.49	chart:regression-period.....	665
20.50	chart:regression-type.....	665
20.51	chart:reverse-direction.....	666
20.52	chart:right-angled-axes.....	666
20.53	chart:scale-text.....	666
20.54	chart:series-source (deprecated).....	667
20.55	chart:sort-by-x-values.....	667
20.56	chart:solid-type.....	667
20.57	chart:spline-order.....	668
20.58	chart:spline-resolution.....	668
20.59	chart:stacked.....	668
20.60	chart:symbol-height.....	668
20.61	chart:symbol-name.....	668
20.62	chart:symbol-type.....	669
20.63	chart:symbol-width.....	669
20.64	chart:text-overlap.....	669
20.65	chart:three-dimensional.....	670
20.66	chart:tick-marks-major-inner.....	670
20.67	chart:tick-marks-major-outer.....	670
20.68	chart:tick-marks-minor-inner.....	671
20.69	chart:tick-marks-minor-outer.....	671
20.70	chart:tick-mark-position.....	671
20.71	chart:treat-empty-cells.....	672
20.72	chart:vertical.....	672
20.73	chart:visible.....	673
20.74	dr3d:ambient-color.....	673
20.75	dr3d:back-scale.....	673
20.76	dr3d:backface-culling.....	673
20.77	dr3d:close-back.....	673
20.78	dr3d:close-front.....	674
20.79	dr3d:depth.....	674
20.80	dr3d:diffuse-color.....	674
20.81	dr3d:edge-rounding.....	674
20.82	dr3d:edge-rounding-mode.....	674
20.83	dr3d:emissive-color.....	675

20.84 dr3d:end-angle.....	675
20.85 dr3d:horizontal-segments.....	675
20.86 dr3d:lighting-mode.....	675
20.87 dr3d:normals-direction.....	676
20.88 dr3d:normals-kind.....	676
20.89 dr3d:shadow.....	676
20.90 dr3d:shininess.....	677
20.91 dr3d:specular-color.....	677
20.92 dr3d:texture-filter.....	677
20.93 dr3d:texture-kind.....	677
20.94 dr3d:texture-mode.....	677
20.95 dr3d:texture-generation-mode-x.....	678
20.96 dr3d:texture-generation-mode-y.....	678
20.97 dr3d:vertical-segments.....	678
20.98 draw:auto-grow-height.....	678
20.99 draw:auto-grow-width.....	679
20.100 draw:background-size.....	679
20.101 draw:blue.....	679
20.102 draw:caption-angle.....	680
20.103 draw:caption-angle-type.....	680
20.104 draw:caption-escape.....	680
20.105 draw:caption-escape-direction.....	680
20.106 draw:caption-fit-line-length.....	681
20.107 draw:caption-gap.....	681
20.108 draw:caption-line-length.....	681
20.109 draw:caption-type.....	681
20.110 draw:color-inversion.....	682
20.111 draw:color-mode.....	682
20.112 draw:contrast.....	682
20.113 draw:decimal-places.....	682
20.114 draw:draw-aspect.....	683
20.115 draw:end-guide.....	683
20.116 draw:end-line-spacing-horizontal.....	683
20.117 draw:end-line-spacing-vertical.....	683
20.118 draw:fill.....	683
20.119 draw:fill-color.....	684
20.120 draw:fill-gradient-name.....	684
20.121 draw:fill-hatch-name.....	684
20.122 draw:fill-hatch-solid.....	684

20.123	draw:fill-image-height.....	685
20.124	draw:fill-image-name.....	685
20.125	draw:fill-image-ref-point.....	685
20.126	draw:fill-image-ref-point-x.....	686
20.127	draw:fill-image-ref-point-y.....	686
20.128	draw:fill-image-width.....	686
20.129	draw:fit-to-contour.....	686
20.130	draw:fit-to-size.....	687
20.131	draw:frame-display-border.....	687
20.132	draw:frame-margin-horizontal.....	687
20.133	draw:frame-display-scrollbar.....	687
20.134	draw:frame-margin-vertical.....	688
20.135	draw:gamma.....	688
20.136	draw:green.....	688
20.137	draw:gradient-step-count.....	688
20.138	draw:guide-distance.....	688
20.139	draw:guide-overhang.....	689
20.140	draw:image-opacity.....	689
20.141	draw:line-distance.....	689
20.142	draw:luminance.....	689
20.143	draw:marker-end.....	689
20.144	draw:marker-end-center.....	690
20.145	draw:marker-end-width.....	690
20.146	draw:marker-start.....	690
20.147	draw:marker-start-center.....	690
20.148	draw:marker-start-width.....	690
20.149	draw:measure-align.....	691
20.150	draw:measure-vertical-align.....	691
20.151	draw:ole-draw-aspect (deprecated).....	691
20.152	draw:opacity.....	692
20.153	draw:opacity-name.....	692
20.154	draw:parallel.....	692
20.155	draw:placing.....	692
20.156	draw:red.....	693
20.157	draw:secondary-fill-color.....	693
20.158	draw:shadow.....	693
20.159	draw:shadow-color.....	693
20.160	draw:shadow-offset-x.....	693
20.161	draw:shadow-offset-y.....	694

20.162	draw:shadow-opacity.....	694
20.163	draw:show-unit.....	694
20.164	draw:start-guide.....	694
20.165	draw:start-line-spacing-horizontal.....	694
20.166	draw:start-line-spacing-vertical.....	694
20.167	draw:stroke.....	695
20.168	draw:stroke-dash.....	695
20.169	draw:stroke-dash-names.....	695
20.170	draw:stroke-linejoin.....	695
20.171	svg:stroke-linecap.....	696
20.172	draw:symbol-color.....	696
20.173	draw:textarea-horizontal-align.....	696
20.174	draw:textarea-vertical-align.....	697
20.175	draw:tile-repeat-offset.....	697
20.176	draw:visible-area-height.....	697
20.177	draw:visible-area-left.....	698
20.178	draw:visible-area-top.....	698
20.179	draw:visible-area-width.....	698
20.180	draw:unit.....	698
20.181	draw:wrap-influence-on-position.....	699
20.182	fo:background-color.....	700
20.183	fo Border Properties.....	701
20.183.1	General.....	701
20.183.2	fo:border.....	701
20.183.3	fo:border-bottom.....	701
20.183.4	fo:border-left.....	701
20.183.5	fo:border-right.....	701
20.183.6	fo:border-top.....	702
20.184	fo:break-after.....	702
20.185	fo:break-before.....	702
20.186	fo:clip.....	702
20.187	fo:color.....	702
20.188	fo:country.....	703
20.189	fo:font-family.....	703
20.190	fo:font-size.....	703
20.191	fo:font-style.....	703
20.192	fo:font-variant.....	704
20.193	fo:font-weight.....	704
20.194	fo:height.....	704

20.195	fo:hyphenate.....	704
20.196	fo:hyphenation-keep.....	704
20.197	fo:hyphenation-ladder-count.....	705
20.198	fo:hyphenation-push-char-count.....	705
20.199	fo:hyphenation-remain-char-count.....	705
20.200	fo:keep-together.....	705
20.201	fo:keep-with-next.....	705
20.202	fo:language.....	706
20.203	fo:letter-spacing.....	706
20.204	fo:line-height.....	706
20.205	fo:margin.....	707
20.206	fo:margin-bottom.....	707
20.207	fo:margin-left.....	707
20.208	fo:margin-right.....	708
20.209	fo:margin-top.....	708
20.210	fo:max-height.....	708
20.211	fo:max-width.....	708
20.212	fo:min-height.....	709
20.212.1	<style:graphic-properties>.....	709
20.212.2	<style:header-footer-properties>.....	709
20.213	fo:min-width.....	709
20.214	fo:orphans.....	709
20.215	fo:page-height.....	709
20.216	fo:page-width.....	709
20.217	fo:padding.....	710
20.218	fo:padding-bottom.....	710
20.219	fo:padding-left.....	710
20.220	fo:padding-right.....	710
20.221	fo:padding-top.....	710
20.222	fo:script.....	710
20.223	fo:text-align.....	711
20.223.1	<style:paragraph-properties>.....	711
20.223.2	<style:list-level-properties>.....	711
20.224	fo:text-align-last.....	712
20.225	fo:text-indent.....	712
20.226	fo:text-shadow.....	712
20.227	fo:text-transform.....	712
20.228	fo:widows.....	713
20.229	fo:width.....	713

20.230	fo:wrap-option.....	713
20.231	presentation:background-objects-visible.....	713
20.232	presentation:background-visible.....	714
20.233	presentation:display-date-time.....	714
20.234	presentation:display-footer.....	714
20.235	presentation:display-header.....	714
20.236	presentation:display-page-number.....	715
20.237	presentation:duration.....	715
20.238	presentation:transition-speed.....	715
20.239	presentation:transition-style.....	716
20.240	presentation:transition-type.....	719
20.241	presentation:visibility.....	719
20.242	smil:direction.....	719
20.243	smil:fadeColor.....	719
20.244	smil:subtype.....	719
20.245	smil:type.....	720
20.246	style:auto-text-indent.....	720
20.247	style:background-transparency.....	720
20.248	style:border-line-width.....	720
20.249	style:border-line-width-bottom.....	721
20.250	style:border-line-width-left.....	721
20.251	style:border-line-width-right.....	721
20.252	style:border-line-width-top.....	722
20.253	style:cell-protect.....	722
20.254	style:column-width.....	722
20.255	style:contextual-spacing.....	723
20.256	style:country-asian.....	723
20.257	style:country-complex.....	723
20.258	style:decimal-places.....	723
20.259	style:diagonal-bl-tr.....	724
20.260	style:diagonal-bl-tr-widths.....	724
20.261	style:diagonal-tl-br.....	724
20.262	style:diagonal-tl-br-widths.....	724
20.263	style:direction.....	724
20.264	style:dynamic-spacing.....	725
20.265	style:editable.....	725
20.266	style:first-page-number.....	725
20.267	style:flow-with-text.....	725
20.268	style:font-charset.....	726

20.269	style:font-charset-asian.....	727
20.270	style:font-charset-complex.....	727
20.271	style:font-family-asian.....	727
20.272	style:font-family-complex.....	728
20.273	style:font-family-generic.....	728
20.274	style:font-family-generic-asian.....	728
20.275	style:font-family-generic-complex.....	729
20.276	style:font-independent-line-spacing.....	729
20.277	style:font-name.....	730
20.278	style:font-name-asian.....	730
20.279	style:font-name-complex.....	730
20.280	style:font-pitch.....	730
20.281	style:font-pitch-asian.....	731
20.282	style:font-pitch-complex.....	731
20.283	style:font-relief.....	731
20.284	style:font-size-asian.....	732
20.285	style:font-size-complex.....	732
20.286	style:font-size-rel.....	732
20.287	style:font-size-rel-asian.....	732
20.288	style:font-size-rel-complex.....	733
20.289	style:font-style-asian.....	733
20.290	style:font-style-complex.....	733
20.291	style:font-style-name.....	733
20.292	style:font-style-name-asian.....	734
20.293	style:font-style-name-complex.....	734
20.294	style:font-weight-asian.....	734
20.295	style:font-weight-complex.....	734
20.296	style:footnote-max-height.....	735
20.297	style:glyph-orientation-vertical.....	735
20.298	style:horizontal-pos.....	735
20.299	style:horizontal-rel.....	737
20.300	style:join-border.....	737
20.301	style:justify-single-word.....	738
20.302	style:language-asian.....	738
20.303	style:language-complex.....	739
20.304	style:layout-grid-base-height.....	739
20.305	style:layout-grid-base-width.....	739
20.306	style:layout-grid-color.....	739
20.307	style:layout-grid-display.....	739

20.308	style:layout-grid-lines.....	740
20.309	style:layout-grid-mode.....	740
20.310	style:layout-grid-print.....	740
20.311	style:layout-grid-ruby-below.....	740
20.312	style:layout-grid-ruby-height.....	741
20.313	style:layout-grid-snap-to.....	741
20.314	style:layout-grid-standard-mode.....	741
20.315	style:line-break.....	742
20.316	style:letter-kerning.....	742
20.317	style:line-height-at-least.....	742
20.318	style:line-spacing.....	742
20.319	style:may-break-between-rows.....	743
20.320	style:min-row-height.....	743
20.321	style:mirror.....	743
20.322	style:num-format.....	743
20.323	style:num-letter-sync.....	744
20.324	style:num-prefix.....	744
20.325	style:num-suffix.....	744
20.326	style:number-wrapped-paragraphs.....	745
20.327	style:overflow-behavior.....	745
20.328	style:page-number.....	745
20.329	style:paper-tray-name.....	746
20.330	style:print.....	746
20.331	style:print-content.....	746
20.331.1	General.....	746
20.331.2	<style:graphic-properties>.....	746
20.331.3	<style:table-cell-properties>.....	747
20.332	style:print-page-order.....	747
20.333	style:print-orientation.....	747
20.334	style:protect.....	747
20.334.1	General.....	747
20.334.2	<style:graphic-properties>.....	747
20.334.3	<style:section-properties>.....	748
20.335	style:punctuation-wrap.....	748
20.336	style:register-true.....	748
20.337	style:register-truth-ref-style-name.....	749
20.338	style:rel-column-width.....	749
20.339	style:rel-height.....	749
20.340	style:rel-width.....	749

20.340.1 <style:graphic-properties>.....	749
20.340.2 <style:table-properties>.....	750
20.341 style:repeat.....	750
20.342 style:repeat-content.....	750
20.343 style:rfc-language-tag.....	751
20.344 style:rfc-language-tag-asian.....	751
20.345 style:rfc-language-tag-complex.....	751
20.346 style:rotation-align.....	752
20.347 style:rotation-angle.....	752
20.348 style:row-height.....	752
20.349 style:ruby-align.....	752
20.350 style:ruby-position.....	753
20.351 style:run-through.....	753
20.352 style:scale-to.....	753
20.353 style:scale-to-pages.....	754
20.354 style:scale-to-X.....	754
20.355 style:scale-to-Y.....	754
20.356 style:script-asian.....	754
20.357 style:script-complex.....	755
20.358 style:script-type.....	755
20.359 style:shadow.....	756
20.360 style:shrink-to-fit.....	756
20.361 style:snap-to-layout-grid.....	756
20.362 style:tab-stop-distance.....	757
20.363 style:table-centering.....	757
20.364 style:text-align-source.....	757
20.365 style:text-autospace.....	757
20.366 style:text-blinking.....	758
20.367 style:text-combine.....	758
20.368 style:text-combine-start-char.....	758
20.369 style:text-combine-end-char.....	758
20.370 style:text-emphasize.....	759
20.371 style:text-line-through-color.....	759
20.372 style:text-line-through-mode.....	759
20.373 style:text-line-through-style.....	760
20.374 style:text-line-through-text.....	760
20.375 style:text-line-through-text-style.....	760
20.376 style:text-line-through-type.....	761
20.377 style:text-line-through-width.....	761

20.378	style:text-outline.....	762
20.379	style:text-overline-color.....	762
20.380	style:text-overline-mode.....	762
20.381	style:text-overline-style.....	762
20.382	style:text-overline-type.....	763
20.383	style:text-overline-width.....	763
20.384	style:text-position.....	764
20.385	style:text-rotation-angle.....	764
20.386	style:text-rotation-scale.....	764
20.387	style:text-scale.....	764
20.388	style:text-underline-color.....	765
20.389	style:text-underline-mode.....	765
20.390	style:text-underline-style.....	765
20.391	style:text-underline-type.....	766
20.392	style:text-underline-width.....	766
20.393	style:use-optimal-column-width.....	767
20.394	style:use-optimal-row-height.....	767
20.395	style:use-window-font-color.....	767
20.396	style:vertical-align.....	767
20.396.1	<style:paragraph-properties>.....	767
20.396.2	<style:table-cell-properties>.....	768
20.397	style:vertical-pos.....	768
20.398	style:vertical-rel.....	768
20.399	style:width.....	770
20.400	style:wrap.....	770
20.401	style:wrap-contour.....	770
20.402	style:wrap-contour-mode.....	770
20.403	style:wrap-dynamic-threshold.....	771
20.404	style:writing-mode.....	771
20.404.1	General.....	771
20.404.2	<style:graphic-properties>.....	772
20.404.3	<style:page-layout-properties>.....	773
20.404.4	<style:paragraph-properties>.....	773
20.404.5	<style:section-properties>.....	774
20.404.6	<style:table-cell-properties>.....	774
20.404.7	<style:table-properties>.....	774
20.405	style:writing-mode-automatic.....	774
20.406	svg:fill-rule.....	775
20.407	svg:height.....	775

20.407.1 <style:graphic-properties>.....	775
20.407.2 <style:header-footer-properties>.....	775
20.408 svg:stroke-color.....	775
20.409 svg:stroke-opacity.....	775
20.410 svg:stroke-width.....	776
20.411 svg:x.....	776
20.412 svg:y.....	776
20.412.1 <style:graphic-properties>.....	776
20.412.2 <style:list-level-properties>.....	776
20.413 svg:width.....	776
20.414 table:align.....	776
20.415 table:border-model.....	777
20.416 table:display.....	777
20.417 text:anchor-page-number.....	777
20.418 text:anchor-type.....	778
20.419 text:animation.....	779
20.420 text:animation-delay.....	779
20.421 text:animation-direction.....	779
20.422 text:animation-repeat.....	779
20.423 text:animation-start-inside.....	779
20.424 text:animation-steps.....	780
20.425 text:animation-stop-inside.....	780
20.426 text:condition.....	780
20.427 text:display.....	780
20.428 text:dont-balance-text-columns.....	780
20.429 text:line-break.....	781
20.430 text:line-number.....	781
20.431 text:list-level-position-and-space-mode.....	781
20.432 text:min-label-distance.....	782
20.433 text:min-label-width.....	782
20.434 text:number-lines.....	782
20.435 text:space-before.....	783
20.436 chart:major-origin.....	783
20.437 chart:minor-logarithmic.....	783
20.438 draw:allow-overlap.....	783
20.439 style:margin-gutter.....	783
20.440 text:relative-tab-stop-position.....	784
20.441 draw:decorative.....	784
20.442 table:tab-color.....	784

Appendix A OpenDocument Relax NG Schema.....	786
Appendix B OpenDocument Metadata Manifest Ontology.....	787
Appendix C MIME Types and File Name Extensions (Non Normative).....	788
Appendix D Accessibility Guidelines (Non Normative).....	790
D.1 Title, Description and Caption of Graphical Elements.....	790
D.1.1 Guidance for authors.....	790
D.1.2 Authoring tool responsibility for presenting and prompting for the <svg:title> and <svg:desc> elements.....	790
D.2 Hyperlink Titles.....	790
D.3 Tables in Presentations.....	791
D.4 Further Guidelines.....	791
Appendix E Bidirectional (BiDi) Scripts, Numeric Digits Presentation and Calendars (Non Normative).....	792
E.1 Paragraph and Layout Direction.....	792
E.2 Numeric Digits Presentation and Calendars.....	792
Appendix F Recommended Usage of SMIL.....	793
F.1 Slide Animation.....	793
F.2 Main Sequence.....	793
F.3 Interactive Sequence.....	794
Appendix G Changes From ODF 1.3 (Non Normative).....	795

---

# 1 Introduction

## 1.1 Introduction

This document is part of the Open Document Format for Office Applications (OpenDocument) Version 1.4 specification. It defines an XML schema for office documents. Office documents include text documents, spreadsheets, charts and graphical documents like drawings or presentations, and other forms of documents.

## 1.2 Terminology

All text is normative unless otherwise labeled.

Text with a gray background color which is contained in boxes is informative. -

Within the normative text of this specification, the terms "shall", "shall not", "should", "should not", "may" and "need not" are to be interpreted as described in Annex H of [ISO/IEC Directives].

Attributes and elements may be marked as deprecated. Deprecated attributes or elements have "Deprecated" inserted into their section title. Attributes and elements marked as deprecated should not be used any longer, and may be removed from future versions of (OpenDocument).

XML Element, attribute names, attribute value types, and attribute values appear in monospace font.

This standard, for illustrative purposes, describes functionality using terminology common in desktop computing environments that contain a display terminal, keyboard, and mouse, attached to a computer hosting an operating system with a graphical user interface which includes user interface controls such as input controls, command buttons, selection boxes, etc.

This standard is not limited to such environments. This format can be used by alternative computing environments, with other form factors, non-GUI consumers and producers, assistive technologies, using analogous user interface operations.

Implementation-defined is used in this standard for values or processing that may differ between ODF implementations but is required to be specified by the implementor for each particular ODF-implementation.

Implementation-dependent is used in this standard for values or processing that may differ between ODF implementations but is not required to be specified by the implementor for each particular ODF-implementation.

## 1.3 Normative References

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

[CSS2] Bert Bos, Håkon Wium Lie, Chris Lilley, Ian Jacobs, Cascading Style Sheets, level 2, <http://www.w3.org/TR/1998/REC-CSS2-19980512>, W3C, 1998.

[CSS3Text] Michel Suignard, CSS3 Text Module, <http://www.w3.org/TR/2003/CR-css3-text-20030514>, W3C, 2003.

[CSS4WritingModes] Elika J. Etemad, Koji Ishii, CSS Writing Modes Level 4, <https://www.w3.org/TR/css-writing-modes-4>, W3C, 2019

[DAISY] ANSI/NISO Z39.86-2005 (R2012) Specifications for the Digital Talking Book, <http://www.niso.org/standards/resources/Z39-86-2005.html>, 2005.

[DCMI] Dublin Core Metadata Element Set, Version 1.1: Reference Description, <http://www.dublincore.org/documents/2008/01/14/dces/>, Dublin Core Metadata Initiative, 2003.

[DOMEvents2] Tom Pixley, Document Object Model (DOM) Level 2 Events Specification, <http://www.w3.org/TR/2000/REC-DOM-Level-2-Events-20001113>, W3C, 2000.

[GRDDL] Don Conolly, Gleaning Resource Descriptions from Dialects of Languages (GRDDL), <http://www.w3.org/TR/2007/REC-grddl-20070911/>, W3C, 2007.

[HTML4] Dave Raggett, Arnaud Le Hors, Ian Jacobs, HTML 4.01 Specification, <http://www.w3.org/TR/1999/REC-html401-19991224>, W3C, 1999.

[ISO/IEC Directives] ISO/IEC Directives, Part 2 (Fifth Edition) Rules for the structure and drafting of International Standards, International Organization for Standardization and International Electrotechnical Commission, 2004.

[ISO639-1] ISO 639-1:2002 Codes for the representation of names of languages -- Part 1: Alpha-2 code, International Organization for Standardization and International Electrotechnical Commission, 2002.

[ISO639-2] ISO 639-2:1998 Codes for the representation of names of languages -- Part 2: Alpha-3 code, International Organization for Standardization and International Electrotechnical Commission, 1998.

[ISO639-3] ISO 639-3:2007 Codes for the representation of names of languages -- Part 3: Alpha-3 code for comprehensive coverage of languages, International Organization for Standardization and International Electrotechnical Commission, 2007.

[ISO3166-1] ISO 3166-1:2006/Cor 1:2007 Codes for the representation of names of countries and their subdivisions -- Part 1: Country codes, International Organization for Standardization and International Electrotechnical Commission, 2007.

[ISO3166-2] ISO 3166-2:2007 Codes for the representation of names of countries and their subdivisions -- Part 2: Country subdivision code, International Organization for Standardization and International Electrotechnical Commission, 2007.

[ISO3166-3] ISO 3166-3:1999 Codes for the representation of names of countries and their subdivisions -- Part 3: Code for formerly used names of countries, International Organization for Standardization and International Electrotechnical Commission, 1999.

[ISO8601] ISO 8601:2019 Date and time — Representations for information interchange — Part 1: Basic rules, International Organization for Standardization, 2019.

[ISO15924] ISO 15924:2004 Information and documentation -- Codes for the representation of names of scripts, International Organization for Standardization and International Electrotechnical Commission, 2004.

[JIS X 0301] JIS X 0301:2002 Data elements and interchange formats -- Information interchange -- Representation of dates and times, Japanese Standards Association, 2002.

[MathML] David Carlisle, Patrick Ion, Robert Miner (deceased), Mathematical Markup Language (MathML), <http://www.w3.org/TR/MathML>, W3C, 2003.

[MIMETYPES] MIME Media Types, <http://www.iana.org/assignments/media-types/>, IANA.

[ODF10] OASIS Standard, Open Document Format for Office Applications (OpenDocument) v1.0, May 2005,  
<http://docs.oasis-open.org/office/v1.0/OpenDocument-v1.0-os.pdf>

[ODF11] OASIS Standard, Open Document Format for Office Applications (OpenDocument) v1.1, February 2007,  
<http://docs.oasis-open.org/office/v1.1/OS/OpenDocument-v1.1.pdf>

[ODF1.4-Part-2:Packages] Open Document Format for Office Applications (OpenDocument) v.1.4, Part 2: Packages. Linked in Additional Artifacts on front page.

[ODF1.4-Part-4:OpenFormula] Open Document Format for Office Applications (OpenDocument) Version 1.4 Part 4 Recalculated Formula (OpenFormula) Format. Linked in Additional Artifacts on front page.

[OLE] Kraig Brockschmidt, Inside OLE, Microsoft Press, 1995, ISBN: 1-55615-843-2

[OWL] Deborah L. McGuinness, Frank van Harmelen, OWL Web Ontology Language Overview, <http://www.w3.org/TR/2004/REC-owl-features-20040210/>, W3C, 2004.

[PNG] David Duce, Portable Network Graphics (PNG) Specification (Second Edition), <http://www.w3.org/TR/2003/REC-PNG-20031110>, W3C, 2003.

[RDF-CONCEPTS] Graham Klyne, Jeremy J. Carroll, Brian McBride, Resource Description Framework (RDF): Concepts and Abstract Syntax, <http://www.w3.org/TR/2004/REC-rdf-concepts-20040210/>, W3C, 2004.

[RDFa] Ben Adida, Mark Birbeck, Shane McCarron, Steven Pemberton, RDFa in XHTML: Syntax and Processing, <http://www.w3.org/TR/2008/REC-rdfa-syntax-20081014>, W3C, 2008.

[RFC2045] N. Freed and N. Borenstein, Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies, <http://www.ietf.org/rfc/rfc2045.txt>, IETF, 1996.

[RFC2616] R. Fielding, et al., Hypertext Transfer Protocol -- HTTP/1.1, <http://www.ietf.org/rfc/rfc2616.txt>, IETF, 1999.

[RFC3066] H. Alvestrand, Tags for the Identification of Languages, <http://www.ietf.org/rfc/rfc3066.txt>, IETF, 2001.

- [RFC3987] M. Duerst, M. Suignard, Internationalized Resource Identifiers (IRIs), <http://www.ietf.org/rfc/rfc3987.txt>, IETF, 2005
- [RFC4514] K. Zeilenga, Lightweight Directory Access Protocol (LDAP): String Representation of Distinguished Names, <http://tools.ietf.org/html/rfc4514>, IETF, 2006.
- [RFC5646] A. Phillips, M. Davis, Tags for Identifying Languages, <http://www.ietf.org/rfc/rfc5646.txt>, IETF, 2009.
- [RFC6838] N. Freed, J. Klensin, T. Hansen Media Type Specifications and Registration Procedures, <http://www.ietf.org/rfc/rfc6838.txt>, IETF, 2013.
- [RNG] ISO/IEC 19757-2 Document Schema Definition Language (DSDL) -- Part 2: Regular-grammar-based validation -- RELAX NG, International Organization for Standardization and International Electrotechnical Commission, 2003.
- [SMIL20] Jeff Ayars, etc al., Synchronized Multimedia Integration Language (SMIL 2.0) - [Second Edition], <http://www.w3.org/TR/2005/REC-SMIL2-20050107/>, W3C, 2001.
- [SQL] ISO/IEC 9075-1:2008 Information Technology -- Database Language -- SQL -- Part 2: Foundation (SQL/Foundation), International Organization for Standardization and International Electrotechnical Commission, 2008.
- [SVG] Jon Ferraiolo, 藤沢 淳 (FUJISAWA Jun), Dean Jackson, Scalable Vector Graphics (SVG) 1.1, <http://www.w3.org/TR/2003/REC-SVG11-20030114/>, W3C, 2003.
- [UAX9] Mark Davis, The Bidirectional Algorithm, Unicode Standard Annex #9, <http://www.unicode.org/reports/tr9/tr9-15.html>, 2005.
- [UNICODE] The Unicode Consortium. The Unicode Standard, <https://www.unicode.org/versions/latest>.
- [UTR18] Mark Davis, Andy Heninger, Unicode Regular Expressions, Unicode Technical Report #18, <http://www.unicode.org/reports/tr18/tr18-13.html>, 2008.
- [UTR20] Martin Dürst and Asmus Freytag, Unicode in XML and other Markup Languages, Unicode Technical Report #20, <http://www.unicode.org/reports/tr20/tr20-7.html>, 2003.
- [XForms] John M. Boyer, XForms 1.0 (Third Edition), <http://www.w3.org/TR/2007/REC-xforms-20071029/>, W3C, 2007.
- [XLink] Steve DeRose, Eve Maler, David Orchard, XML Linking Language (XLink) Version 1.0, <http://www.w3.org/TR/2001/REC-xlink-20010627/>, W3C, 2001.
- [XML-ID] Jonathan Marsh, Daniel Veillard, Norman Walsh, xml:id Version 1.0, <http://www.w3.org/TR/2005/REC-xml-id-20050909/>, W3C, 2005.
- [XML-Names] Tim Bray et al., Namespaces in XML 1.0 (Second Edition), <http://www.w3.org/TR/2006/REC-xml-names-20060816>, W3C, 2006.
- [XML1.0] Tim Bray, Jean Paoli, C. M. Sperberg-McQueen, Eve Maler, François Yergeau, Extensible Markup Language (XML) 1.0 (Fourth Edition), <http://www.w3.org/TR/2006/REC-xml-20060816/>, W3C, 2006.

[**XML-ENC-CORE**] Donald Eastlake, Joseph Reagle, XML Encryption Syntax and Processing, <http://www.w3.org/TR/2002/REC-xmlenc-core-20021210/>, W3C, 2002.

[**XPath**] James Clark, Steve DeRose, XML Path Language (XPath) Version 1.0, <http://www.w3.org/TR/1999/REC-xpath-19991116>, W3C, 1999.

[**XMLSchema-2**] Paul V. Biron, Ashok Malhotra, XML Schema Part 2: Datatypes Second Edition, <http://www.w3.org/TR/2004/REC-xmlschema-2-20041028/>, W3C, 2004.

[**XSL**] Sharon Adler, et al., Extensible Stylesheet Language (XSL), <http://www.w3.org/TR/2001/REC-xsl-20011015/>, W3C, 2001.

[**XSLT**] James Clark, XSL Transformations (XSLT) Version 1.0, <http://www.w3.org/TR/1999/REC-xslt-19991116>, W3C, 1999.

## 1.4 Non Normative References

[**BibTeXing**] Oren Patashnik, *BIBTEXing*, <http://mirrors.ctan.org/biblio/bibtex/base/btxdoc.pdf>

[**UAX9**] Mark Davis, *The Bidirectional Algorithm*, Unicode Standard Annex #9, <http://www.unicode.org/reports/tr9/tr9-15.html>, 2005.

[**UTR20**] Martin Dürst and Asmus Freytag, *Unicode in XML and other Markup Languages*, Unicode Technical Report #20, <http://www.unicode.org/reports/tr20/tr20-7.html>, 2003.

## 1.5 Namespaces

Tables 1-6 list namespaces used or defined by OpenDocument.

**Note:** The defined XML namespaces conform to the *Namespaces in XML* specification [XML-Names].

This specification uses the prefixes defined in Tables 1 and 2 when referring to elements and attributes in this specification. However, implementors may use any prefix, provided that there is a namespace declaration that binds the prefix to the IRI of the corresponding namespace.

*Table 1 - XML Namespaces defined by the OpenDocument schema*

<b>Prefix</b>	<b>Description</b>	<b>Namespace</b>
anim	Elements and attributes that describe animation content.	urn:oasis:names:tc:opendocument:xmlns:animation:1.0
chart	Elements and attributes that describe chart content.	urn:oasis:names:tc:opendocument:xmlns:chart:1.0
config	Elements and attributes that describe application specific settings.	urn:oasis:names:tc:opendocument:xmlns:config:1.0
db	For elements and attributes that describe database specific objects.	urn:oasis:names:tc:opendocument:xmlns:database:1.0
dr3d	Elements and attributes that describe 3D graphic content.	urn:oasis:names:tc:opendocument:xmlns:dr3d:1.0
draw	Elements and attributes that describe	urn:oasis:names:tc:opendocument:xmlns:

<b>Prefix</b>	<b>Description</b>	<b>Namespace</b>
	graphic content.	drawing:1.0
form	Elements and attributes that describe forms and controls.	urn:oasis:names:tc:opendocument:xmlns:form:1.0
manifest	Elements and attribute contained in the package manifest.	urn:oasis:names:tc:opendocument:xmlns:manifest:1.0
meta	Elements and attributes that describe meta information.	urn:oasis:names:tc:opendocument:xmlns:meta:1.0
number	Elements and attributes that describe data style information.	urn:oasis:names:tc:opendocument:xmlns:datastyle:1.0
office	All common pieces of information not contained in another, more specific namespace.	urn:oasis:names:tc:opendocument:xmlns:office:1.0
presentation	Elements and attributes that describe presentation content.	urn:oasis:names:tc:opendocument:xmlns:presentation:1.0
script	Elements and attributes that represent scripts or events.	urn:oasis:names:tc:opendocument:xmlns:script:1.0
table	Elements and attributes that may occur in spreadsheets or in table in a text document.	urn:oasis:names:tc:opendocument:xmlns:table:1.0
text	Elements and attributes that may occur within text documents and text parts of other document types.	urn:oasis:names:tc:opendocument:xmlns:text:1.0
style	Elements and attributes that describe the style and inheritance model used by the OpenDocument format.	urn:oasis:names:tc:opendocument:xmlns:style:1.0

*Table 2 - XML Namespaces defined by the OpenDocument metadata manifest ontology*

<b>Prefix</b>	<b>Description</b>	<b>Namespace</b>
odf	RDF node property and node elements for OpenDocument package entities	http://docs.oasis-open.org/ns/office/1.2/meta/odf#

*Table 3 - XML Namespaces defined by the OpenDocument schema that include elements and attributes that are compatible to elements and attributes of other standards.*

<b>Prefix</b>	<b>Description</b>	<b>Namespace</b>
fo	Attributes that are compatible to attributes defined in [XSL].	urn:oasis:names:tc:opendocument:xmlns: :xsl-fo-compatible:1.0
svg	Elements and attributes that are derived from elements or attributes defined in [SVG].	urn:oasis:names:tc:opendocument:xmlns: :svg-compatible:1.0
smil	Attributes that are derived from to	urn:oasis:names:tc:opendocument:xmlns

<b>Prefix</b>	<b>Description</b>	<b>Namespace</b>
	attributes defined in [SMIL20].	: smil-compatible:1.0

*Table 4 - XML Namespaces used by the OpenDocument schema*

<b>Prefix</b>	<b>Description</b>	<b>Namespace</b>
dc	The Dublin Core Namespace (see [DCMI]).	<a href="http://purl.org/dc/elements/1.1/">http://purl.org/dc/elements/1.1/</a>
math	MathML Namespace (see [MathML])	<a href="http://www.w3.org/1998/Math/MathML">http://www.w3.org/1998/Math/MathML</a>
xforms	The XForms namespace (see [XForms]).	<a href="http://www.w3.org/2002/xforms">http://www.w3.org/2002/xforms</a>
xlink	The XLink namespace (see [XLink]).	<a href="http://www.w3.org/1999/xlink">http://www.w3.org/1999/xlink</a>
xhtml	RDFa attributes (see [RDFa]).	<a href="http://www.w3.org/1999/xhtml">http://www.w3.org/1999/xhtml</a>
grddl	GRDDL attributes (see [GRDDL])	<a href="http://www.w3.org/2003/g/data-view#">http://www.w3.org/2003/g/data-view#</a>

*Table 5 - XML Namespaces defined by the OpenDocument metadata manifest ontology*

<b>Prefix</b>	<b>Description</b>	<b>Namespace</b>
pkg	OWL classes and properties contained in metadata manifest files.	<a href="http://docs.oasis-open.org/ns/office/1.2/meta/pkg#">http://docs.oasis-open.org/ns/office/1.2/meta/pkg#</a>

*Table 6 - Other XML Namespaces defined by this specification*

<b>Prefix</b>	<b>Description</b>	<b>Namespace</b>
of	Namespace for formulas defined by Part 4 of this specification that occur in attribute values.	<a href="urn:oasis:names:tc:opendocument:xmlns:of:1.2">urn:oasis:names:tc:opendocument:xmlns:of:1.2</a>

---

## 2 OpenDocument Documents, Consumers and Producers

### 2.1 Introduction

The OpenDocument specification defines conformance for documents, consumers, and producers, with two conformance classes called conforming and extended conforming. It further defines conforming text, spreadsheet, drawing, presentation, chart, image, formula and database front end documents. This chapter defines the basic requirements for the individual conformance targets.

### 2.2 Document Conformance

#### 2.2.1 OpenDocument Document

An *OpenDocument document* shall meet the following requirements:

A) If the document is an OpenDocument package, then

- A.1) it shall be a conforming OpenDocument package [ODF1.4-Part-2:Packages], 2.2 Packages, and
- A.2) the package shall contain at least one of the following files: content.xml and styles.xml. It may contain additional files.

B) If the document is an OpenDocument package, then the following requirements shall be met for its contained files named content.xml, styles.xml, settings.xml, and meta.xml if present:

B.1) The files shall be well-formed XML documents with respect to the XML 1.0 [XML1.0] specification.

B.2) The XML root elements of the files shall be

- B.2.1) `<office:document-content>` 3.1.3.2 or `<math:math>` 14.6 for files named content.xml,
- B.2.2) `<office:document-styles>` 3.1.3.3 for files named styles.xml,
- B.2.3) `<office:document-meta>` 3.1.3.4 for files named meta.xml,
- B.2.4) `<office:document-settings>` 3.1.3.5 for files named settings.xml.

B.3) If the XML root element of a file is `<office:document-content>`, `<office:document-styles>`, `<office:document-meta>` or `<office:document-settings>`, then the XML file shall be valid with respect to the schema defined in appendix A.

B.4) If the XML root element of a file is `<math:math>`, then the XML file shall be valid with respect to the MathML 2.0 [MathML] schema.

C) If the document is a single XML file, then

- C.1) the file shall be a well-formed XML document with respect to the XML 1.0 [XML1.0] specification
- C.2) the XML root element of the file shall be `<office:document>`.
- C.3) the XML file shall validate against the schema defined in appendix A.

D) The files contained in a package listed in B) or the single file listed in C) meet the following requirements:

- D.1) They shall conform to the XML Namespaces specification [XML-Names].
- D.2) They shall conform to the xml:id Version 1.0 specification [XML-ID].
- D.3) If a `style:condition`, `table:condition`, `table:expression`, `table:formula` or `text:formula` attribute value begins with a namespace prefix bound to namespace "urn:oasis:names:tc:opendocument:xmlns:of:1.2," the syntax and semantics of the attribute value portions that are expressions determined by the prefix shall conform to Part4 of this specification. If a `style:condition`, `table:condition`, `table:expression`, `table:formula` or `text:formula` attribute value has no namespace prefix, the attribute value portions that are expressions determined by a prefix shall conform as if there were a prefix bound to namespace "urn:oasis:names:tc:opendocument:xmlns:of:1.2".
- E) It shall conform to one of: OpenDocument Text Document (2.2.3), OpenDocument Spreadsheet Document (2.2.4), OpenDocument Drawing Document (2.2.5), OpenDocument Presentation Document (2.2.6), OpenDocument Chart Document (2.2.7), OpenDocument Image Document (2.2.8), OpenDocument Formula Document (2.2.9), OpenDocument Database Front End Document (2.2.10).

## 2.2.2 OpenDocument Extended Document

An *OpenDocument extended document* shall meet all requirements of a conforming document listed in 2.2.1 except A.1), B.3), C.3) and E) and shall meet the following requirements:

- A) If the document is an OpenDocument package, then it shall be a conforming ODF extended package [ODF1.4-Part-2:Packages], 2.2.2 OpenDocument Extended Package.
- B) If the document is an OpenDocument package, then the following requirement shall be met for its contained files named `content.xml`, `styles.xml`, `settings.xml`, and `meta.xml` if present:
  - B.1) If the XML root element of a file is `<office:document-content>` 3.1.3.2, `<office:document-styles>` 3.1.3.3, `<office:document-meta>` 3.1.3.4 or `<office:document-settings>` 3.1.3.5, then the XML file shall be valid with respect to the schema defined in appendix A, after the following processing of foreign elements has taken place:
    - B.1.1) Foreign elements and attributes, as defined in section 3.17 have been removed.
    - B.1.2) The content of foreign elements has been processed as defined in section 3.17.
- C) If the document is a single XML file, then the XML files shall be valid against the schema defined in appendix A, after processing of foreign elements as defined in B.1) has taken place.

## 2.2.3 OpenDocument Text Document

An *OpenDocument Text document* shall meet all requirements of a Conforming OpenDocument Document, as well as the following additional requirements:

- A) The `<office:document>` 3.1.2 element shall have an `office:mimetype` 3.1.2 attribute with one of these values: "application/vnd.oasis.opendocument.text", "application/vnd.oasis.opendocument.text-template", "application/vnd.oasis.opendocument.text-master" or "application/vnd.oasis.opendocument.text-master-template".
- B) If the document is an OpenDocument package then it shall contain a file named `mimetype` containing one of these strings: "application/vnd.oasis.opendocument.text", "application/vnd.oasis.opendocument.text-template", "application/vnd.oasis.opendocument.text-master" or "application/vnd.oasis.opendocument.text-master-template".
- C) The `<office:body>` 3.3 element shall have the child element `<office:text>` 3.4.

## 2.2.4 OpenDocument Spreadsheet Document

An *OpenDocument Spreadsheet document* shall meet all requirements of a Conforming OpenDocument Document, as well as the following additional requirements:

- A) The `<office:document>` 3.1.2 element shall have an `office:mimetype` 19.379 attribute with one of these values: "application/vnd.oasis.opendocument.spreadsheet" or "application/vnd.oasis.opendocument.spreadsheet-template".
- B) If the document is an OpenDocument package then it shall contain a file named mimetype containing one of these strings: "application/vnd.oasis.opendocument.spreadsheet" or "application/vnd.oasis.opendocument.spreadsheet-template".
- C) The `<office:body>` 3.3 element shall have the child element `<office:spreadsheet>` 3.7.
- D) All namespace prefixes used in the values of `table:formula` attributes values shall be bound to the "urn:oasis:names:tc:opendocument:xmlns:of:1.2" namespace.
- E) All `table:formula` attribute values shall be a conforming OpenDocument Formula expression [ODF1.4-Part-4:OpenFormula] 2.2 OpenDocument Formula Expression.

## 2.2.5 OpenDocument Drawing Document

An *OpenDocument Drawing document* shall meet all requirements of a Conforming OpenDocument Document, as well as the following additional requirements:

- A) The `<office:document>` 3.1.2 element shall have an `office:mimetype` 19.379 attribute with one of these values: "application/vnd.oasis.opendocument.graphics" or "application/vnd.oasis.opendocument.graphics-template".
- B) If the document is an OpenDocument package then it shall contain a file named mimetype containing one of these strings: "application/vnd.oasis.opendocument.graphics" or "application/vnd.oasis.opendocument.graphics-template".
- C) The `<office:body>` 3.3 element shall have the child element `<office:drawing>` 3.5.

## 2.2.6 OpenDocument Presentation Document

An *OpenDocument Presentation document* shall meet all requirements of a Conforming OpenDocument Document, as well as the following additional requirements:

- A) The `<office:document>` 3.1.2 element shall have an `office:mimetype` 19.379 attribute with one of these values: "application/vnd.oasis.opendocument.presentation" or "application/vnd.oasis.opendocument.presentation-template".
- B) If the document is an OpenDocument package then it shall contain a file named mimetype containing one of these strings: "application/vnd.oasis.opendocument.presentation" or "application/vnd.oasis.opendocument.presentation-template".
- C) The `<office:body>` 3.3 element shall have the child element 3.6  
`<office:presentation>`.

## 2.2.7 OpenDocument Chart Document

An *OpenDocument Chart document* shall meet all requirements of a Conforming OpenDocument Document, as well as the following additional requirements:

- A) The `<office:document>` 3.1.2 element shall have an `office:mimetype` 19.379 attribute with one of these values: "application/vnd.oasis.opendocument.chart" or "application/vnd.oasis.opendocument.chart-template".
- B) If the document is an OpenDocument package then it shall contain a file named mimetype containing one of these strings: "application/vnd.oasis.opendocument.chart" or "application/vnd.oasis.opendocument.chart-template".

C) The `<office:body>` 3.3 element shall have the child element `<office:chart>` 19.379.

## 2.2.8 OpenDocument Image Document

An *OpenDocument Image document* shall meet all requirements of a Conforming OpenDocument Document, as well as the following additional requirements:

- A) The `<office:document>` 3.1.2 element shall have an `office:mimetype` 19.379 attribute with one of these values: "application/vnd.oasis.opendocument.image" or "application/vnd.oasis.opendocument.image-template".
- B) If the document is an OpenDocument package then it shall contain a file named mimetype containing one of these strings: "application/vnd.oasis.opendocument.image" or "application/vnd.oasis.opendocument.image-template".
- C) The `<office:body>` 3.3 element shall have the child element `<office:image>` 3.9.

## 2.2.9 OpenDocument Formula Document

An *OpenDocument Formula document* shall meet all requirements of a Conforming OpenDocument Document, as well as the following additional requirements:

- A) If the document is an OpenDocument package, the package shall contain a file named mimetype with content being the string "application/vnd.oasis.opendocument.formula".
- B) The package "content.xml" file shall have a `<math:math>` 14.6 root element.

## 2.2.10 OpenDocument Database Front End Document

An *OpenDocument Database Front End document* shall meet all requirements of a Conforming OpenDocument Document, as well as the following additional requirements:

- A) The `<office:document>` 3.1.2 element shall have an `office:mimetype` 19.379 attribute with the value "application/vnd.oasis.opendocument.base".
- B) If the document is an OpenDocument package then it shall contain a file named mimetype containing the string "application/vnd.oasis.opendocument.base".
- C) The `<office:body>` 3.3 element shall have the child element `<office:database>` 12.1.

## 2.3 Producers

### 2.3.1 OpenDocument Producer

An *OpenDocument producer* is a program that creates at least one conforming OpenDocument document, and that meets the following additional requirements:

- A) It may produce conforming OpenDocument extended documents, but it shall have a mode of operation where it creates conforming OpenDocument documents.
- B) It shall be accompanied by a document that defines all implementation-defined values used by the OpenDocument producer.

### 2.3.2 OpenDocument Extended Producer

An *OpenDocument extended producer* is a program that creates at least one conforming OpenDocument extended document, and that meets the following additional requirements:

- A) It shall be accompanied by a document that defines all implementation-defined values used by the OpenDocument extended producer.
- B) It should be accompanied by a document that defines all foreign elements and attributes used by the OpenDocument extended producer.

## 2.4 Consumer

An *OpenDocument consumer* is a program that can parse and interpret OpenDocument documents according to the semantics defined by this specification, and that meets the following additional requirements:

- A) It shall be able to parse and interpret OpenDocument documents of one or more of the document types defined by this specification (see 3.3), but it need not interpret the semantics of all elements, attributes and attribute values.
- B) It may be able to parse and interpret OpenDocument documents stored as a single XML document, but it need not interpret the semantics of all elements, attributes and attribute values.
- C) It shall interpret those elements and attributes it does interpret consistent with the semantics defined for the element or attribute by this specification.
- D) It should be able to parse and interpret conforming OpenDocument extended documents, but it need not interpret the semantics of all elements, attributes and attribute values.
- E) The XML parser used to parse the files contained in a package listed in 2.2.1, item B) or the single document listed in 2.2.1, item C) meets the following requirements:
  - E.1) It shall be a nonvalidating XML processor with regard to the XML 1.0 specification [XML1.0]
  - E.2) It shall be and be a conforming processor with regard to the XML Namespaces specification [XML-Names].
  - E.3) It shall conform to the xml-id specification [XML-ID].

## 3 Document Structure

### 3.1 Document Representation

#### 3.1.1 General

OpenDocument defines two methods of document representation:

- A single XML document.
- A collection of files within a package ([ODF1.4-Part-2:Packages]), each of which stores a part of a complete document.

#### 3.1.2 <office:document>(Single OpenDocument XML Files)

The `<office:document>` element is the root element of a document in OpenDocument format which is represented as a single XML document. It contains the entire document.

The `<office:document>` element is a root element.

The `<office:document>` element is usable within the following elements:

`<db:component>` 12.25.5 and `<draw:object>` 10.4.6.2.

The `<office:document>` element has the following attributes: `grddl:transformation` 19.320, `office:mimetype` 19.379 and `office:version` 19.390.

The `<office:document>` element has the following child elements:

`<office:automatic-styles>` 3.15.3, `<office:body>` 3.3, `<office:font-face-decls>` 3.14, `<office:master-styles>` 3.15.4, `<office:meta>` 3.2, `<office:scripts>` 3.12, `<office:settings>` 3.10 and `<office:styles>` 3.15.2.

#### 3.1.3 Package OpenDocument Files

##### 3.1.3.1 General

The potential root elements for an OpenDocument document represented as a package are defined in OpenDocument Document, 2.2.1.

A package may also contain image files, embedded objects and implementation-dependent files.

##### 3.1.3.2 <office:document-content>

The `<office:document-content>` root element contains document content and automatic styles used in a document. The file within a package for the `<office:document-content>` element is `content.xml`.

The `<office:document-content>` element is a root element.

The `<office:document-content>` element has the following attributes: `grddl:transformation` 19.320 and `office:version` 19.390.

The `<office:document-content>` element has the following child elements:

`<office:automatic-styles>` 3.15.3, `<office:body>` 3.3, `<office:font-face-decls>` 3.14 and `<office:scripts>` 3.12.

### 3.1.3.3 <office:document-styles>

The `<office:document-styles>` root element contains styles used in document content and automatic styles used in styles. The file within a package for the `<office:document-styles>` element is `styles.xml`.

The `<office:document-styles>` element is a root element.

The `<office:document-styles>` element has the following attributes: `grddl:transformation 19.320` and `office:version 19.390`.

The `<office:document-styles>` element has the following child elements: `<office:automatic-styles> 3.15.3`, `<office:font-face-decls> 3.14`, `<office:master-styles> 3.15.4` and `<office:styles> 3.15.2`.

### 3.1.3.4 <office:document-meta>

The `<office:document-meta>` root element contains document metadata elements. The file within a package for the `<office:document-meta>` element is `meta.xml`.

The `<office:document-meta>` element is a root element.

The `<office:document-meta>` element has the following attributes: `grddl:transformation 19.320` and `office:version 19.390`.

The `<office:document-meta>` element has the following child element: `<office:meta> 3.2`.

### 3.1.3.5 <office:document-settings>

The `<office:document-settings>` root element contains implementation-dependent settings. The file within a package for the `<office:document-settings>` element is `settings.xml`.

The `<office:document-settings>` element is a root element.

The `<office:document-settings>` element has the following attributes: `grddl:transformation 19.320` and `office:version 19.390`.

The `<office:document-settings>` element has the following child element: `<office:settings> 3.10`.

## 3.1.4 Summary of Document Representation

OpenDocument file models are summarized in Table 7.

Table 7 - Root element content models

<b>Root Element</b>	<code>&lt;of-fice :met-a&gt;</code>	<code>&lt;of-fice :e:st-in-gs&gt;</code>	<code>&lt;of-fice :scr-ipts&gt;</code>	<code>&lt;of-fice :fon-t-face-de-cls&gt;</code>	<code>&lt;of-fice :sty-les&gt;</code>	<code>&lt;of-fice :au-to-mati-c-sty-les&gt;</code>	<code>&lt;of-fice :mas-ter-styl-es&gt;</code>	<code>&lt;of-fice :bod-y&gt;</code>
<code>&lt;office:document&gt;</code>	X	X	X	X	X	X	X	X
<code>&lt;office:document-content&gt;</code>			X	X		X		X
<code>&lt;office:document-styles&gt;</code>				X	X	X	X	

<b>Root Element</b>	<code>&lt;of-fice :met a&gt;</code>	<code>&lt;of-fic e:s et-tin gs&gt;</code>	<code>&lt;of-fice :scr ipts&gt;</code>	<code>&lt;of-fice :fon t-face -de-cls&gt;</code>	<code>&lt;of-fice :sty les&gt;</code>	<code>&lt;of-fice :au-to-mati c-st yles &gt;</code>	<code>&lt;of-fice :mas ter-styl es&gt;</code>	<code>&lt;of-fice :bod y&gt;</code>
<code>&lt;office:document-meta&gt;</code>	X							
<code>&lt;office:document-settings&gt;</code>		X						

### 3.2 `<office:meta>`

The `<office:meta>` element contains metadata elements for a document that are not specified by a manifest.rdf file. 4.2.2

All the child elements of an `<office:meta>` element contain metadata about a document as a whole.

The `<office:meta>` element is usable within the following elements:

`<office:document>` 3.1.2 and `<office:document-meta>` 3.1.3.4.

The `<office:meta>` element has no attributes.

The `<office:meta>` element has the following child elements: `<dc:creator>` 4.3.2.7, `<dc:date>` 4.3.2.10, `<dc:description>` 4.3.2.3, `<dc:language>` 4.3.2.15, `<dc:subject>` 4.3.2.4, `<dc:title>` 4.3.2.2, `<meta:auto-reload>` 4.3.2.13, `<meta:creation-date>` 4.3.2.9, `<meta:document-statistic>` 4.3.2.18, `<meta:editing-cycles>` 4.3.2.16, `<meta:editing-duration>` 4.3.2.17, `<meta:generator>` 4.3.2.1, `<meta:hyperlink-behaviour>` 4.3.2.14, `<meta:initial-creator>` 4.3.2.6, `<meta:keyword>` 4.3.2.5, `<meta:print-date>` 4.3.2.11, `<meta:printed-by>` 4.3.2.8, `<meta:template>` 4.3.2.12 and `<meta:user-defined>` 4.3.3.

### 3.3 `<office:body>`

The `<office:body>` element contains the elements that represent the content of a document.

The `<office:body>` element is usable within the following elements:

`<office:document>` 3.1.2 and `<office:document-content>` 3.1.3.2.

The `<office:body>` element has no attributes.

The `<office:body>` element has the following child elements: `<office:chart>` 3.8,

`<office:database>` 12.1, `<office:drawing>` 3.5, `<office:image>` 3.9,

`<office:presentation>` 3.6, `<office:spreadsheet>` 3.7 and `<office:text>` 3.4.

### 3.4 `<office:text>`

The `<office:text>` element represents the content of a text document.

The `<office:text>` element is usable within the following element: `<office:body>` 3.3.

The `<office:text>` element has the following attributes: `text:global` 19.812 and `text:use-soft-page-breaks` 19.903.

The `<office:text>` element has the following child elements: `<dr3d:scene>` 10.5.2, `<draw:a>` 10.4.12, `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8,

`<draw:connector>` 10.3.10, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1,

```
<draw:ellipse> 10.3.9, <draw:frame> 10.4.2, <draw:g> 10.3.15, <draw:line> 10.3.3, <draw:measure> 10.3.12, <draw:page-thumbnail> 10.3.14, <draw:path> 10.3.7, <draw:polygon> 10.3.5, <draw:polyline> 10.3.4, <draw:rect> 10.3.2, <draw:regular-polygon> 10.3.6, <office:forms> 13.2, <table:calculation-settings> 9.4.1, <table:consolidation> 9.7, <table:content-validations> 9.4.4, <table:data-pilot-tables> 9.6.2, <table:database-ranges> 9.4.14, <table:dde-links> 9.8, <table:label-ranges> 9.4.10, <table:named-expressions> 9.4.11, <table:table> 9.1.2, <text:alphabetical-index> 8.8, <text:alphabetical-index-auto-mark-file> 8.8.3, <text:bibliography> 8.9, <text:change> 5.5.8.4, <text:change-end> 5.5.8.3, <text:change-start> 5.5.8.2, <text:dde-connection-decls> 14.7.2, <text:h> 5.1.2, <text:illustration-index> 8.4, <text:list> 5.3.1, <text:numbered-paragraph> 5.3.6, <text:object-index> 8.6, <text:p> 5.1.3, <text:page-sequence> 5.2, <text:section> 5.4, <text:sequence-decls> 7.4.11, <text:soft-page-break> 5.6, <text:table-index> 8.5, <text:table-of-content> 8.3, <text:tracked-changes> 5.5.2, <text:user-field-decls> 7.4.7, <text:user-index> 8.7 and <text:variable-decls> 7.4.2.
```

### 3.5 <office:drawing>

The `<office:drawing>` element represents a drawing document.

The `<office:drawing>` element is usable within the following element: `<office:body>` 3.3.

The `<office:drawing>` element has no attributes.

The `<office:drawing>` element has the following child elements: `<draw:page>` 10.2.4, `<table:calculation-settings>` 9.4.1, `<table:consolidation>` 9.7, `<table:content-validations>` 9.4.4, `<table:data-pilot-tables>` 9.6.2, `<table:database-ranges>` 9.4.14, `<table:dde-links>` 9.8, `<table:label-ranges>` 9.4.10, `<table:named-expressions>` 9.4.11, `<text:alphabetical-index-auto-mark-file>` 8.8.3, `<text:dde-connection-decls>` 14.7.2, `<text:sequence-decls>` 7.4.11, `<text:user-field-decls>` 7.4.7 and `<text:variable-decls>` 7.4.2.

### 3.6 <office:presentation>

The `<office:presentation>` element represents a presentation document.

The `<office:presentation>` element is usable within the following element: `<office:body>` 3.3.

The `<office:presentation>` element has no attributes.

The `<office:presentation>` element has the following child elements: `<draw:page>` 10.2.4, `<presentation:date-time-decl>` 10.9.4.6, `<presentation:footer-decl>` 10.9.4.4, `<presentation:header-decl>` 10.9.4.2, `<presentation:settings>` 10.9.4.7, `<table:calculation-settings>` 9.4.1, `<table:consolidation>` 9.7, `<table:content-validations>` 9.4.4, `<table:data-pilot-tables>` 9.6.2, `<table:database-ranges>` 9.4.14, `<table:dde-links>` 9.8, `<table:label-ranges>` 9.4.10, `<table:named-expressions>` 9.4.11, `<text:alphabetical-index-auto-mark-file>` 8.8.3, `<text:dde-connection-decls>` 14.7.2, `<text:sequence-decls>` 7.4.11, `<text:user-field-decls>` 7.4.7 and `<text:variable-decls>` 7.4.2.

### 3.7 <office:spreadsheet>

The `<office:spreadsheet>` element represents a spreadsheet document.

The `<office:spreadsheet>` element is usable within the following element: `<office:body>` 3.3.

The `<office:spreadsheet>` element has the following attributes: `table:protection-key` 19.701, `table:protection-key-digest-algorithm` 19.702 and `table:structure-protected` 19.729.

The `<office:spreadsheet>` element has the following child elements: `<table:calculation-settings>` 9.4.1, `<table:consolidation>` 9.7, `<table:content-validations>` 9.4.4, `<table:data-pilot-tables>` 9.6.2, `<table:database-ranges>` 9.4.14, `<table:dde-links>` 9.8, `<table:label-ranges>` 9.4.10, `<table:named-expressions>` 9.4.11, `<table:table>` 9.1.2, `<table:tracked-changes>` 9.9.2, `<text:alphabetical-index-auto-mark-file>` 8.8.3, `<text:dde-connection-decls>` 14.7.2, `<text:sequence-decls>` 7.4.11, `<text:user-field-decls>` 7.4.7 and `<text:variable-decls>` 7.4.2.

## 3.8 `<office:chart>`

The `<office:chart>` element represents a chart document.

The `<office:chart>` element is usable within the following element: `<office:body>` 3.3.

The `<office:chart>` element has no attributes.

The `<office:chart>` element has the following child elements: `<chart:chart>` 11.1, `<table:calculation-settings>` 9.4.1, `<table:consolidation>` 9.7, `<table:content-validations>` 9.4.4, `<table:data-pilot-tables>` 9.6.2, `<table:database-ranges>` 9.4.14, `<table:dde-links>` 9.8, `<table:label-ranges>` 9.4.10, `<table:named-expressions>` 9.4.11, `<text:alphabetical-index-auto-mark-file>` 8.8.3, `<text:dde-connection-decls>` 14.7.2, `<text:sequence-decls>` 7.4.11, `<text:user-field-decls>` 7.4.7 and `<text:variable-decls>` 7.4.2.

## 3.9 `<office:image>`

The `<office:image>` element represents an image document.

The `<office:image>` element contains a `<draw:frame>` 10.4.2 element which shall contain a single `<draw:image>` 10.4.4 element.

The `<office:image>` element is usable within the following element: `<office:body>` 3.3.

The `<office:image>` element has no attributes.

The `<office:image>` element has the following child element: `<draw:frame>` 10.4.2.

## 3.10 `<office:settings>`

### 3.10.1 General

The `<office:settings>` element contains one or more `<config:config-item-set>` elements, each of which represents a set of application settings.

The `<office:settings>` element is usable within the following elements: `<office:document>` 3.1.2 and `<office:document-settings>` 3.1.3.5.

The `<office:settings>` element has no attributes.

The `<office:settings>` element has the following child element: `<config:config-item-set>` 3.10.2.

### 3.10.2 <config:config-item-set>

The <config:config-item-set> element is a container element for application setting elements. Setting child elements may occur in any order.

An OpenDocument Consumer shall use application settings to alter application behavior only within the dimensions of variability permitted by this standard.

The <config:config-item-set> element is usable within the following elements: <config:config-item-map-entry> 3.10.5, <config:config-item-set> 3.10.2 and <office:settings> 3.10.

The <config:config-item-set> element has the following attribute: config:name 19.29.

The <config:config-item-set> element has the following child elements:

<config:config-item> 3.10.3, <config:config-item-map-indexed> 3.10.4, <config:config-item-map-named> 3.10.6 and <config:config-item-set> 3.10.2.

### 3.10.3 <config:config-item>

The <config:config-item> element contains the value of an application setting whose name is specified by its config:name attribute.

The <config:config-item> element is usable within the following elements: <config:config-item-map-entry> 3.10.5 and <config:config-item-set> 3.10.2.

The <config:config-item> element has the following attributes: config:name 19.29 and config:type 19.30.

The <config:config-item> element has no child elements.

The <config:config-item> element has character data content.

### 3.10.4 <config:config-item-map-indexed>

The <config:config-item-map-indexed> element is a container element for ordered sequences of application settings.

The <config:config-item-map-indexed> element is usable within the following elements: <config:config-item-map-entry> 3.10.5 and <config:config-item-set> 3.10.2.

The <config:config-item-map-indexed> element has the following attribute: config:name 19.29.

The <config:config-item-map-indexed> element has the following child element: <config:config-item-map-entry> 3.10.5.

### 3.10.5 <config:config-item-map-entry>

The <config:config-item-map-entry> element represents a single setting in a sequence of settings. The setting itself is defined by the child element of <config:config-item-map-entry>, and may be a single value, a set of settings, or a sequence of settings.

The <config:config-item-map-entry> element is usable within the following elements: <config:config-item-map-indexed> 3.10.4 and <config:config-item-map-named> 3.10.6.

The <config:config-item-map-entry> element has the following attribute: config:name 19.29.

The `<config:config-item-map-entry>` element has the following child elements: `<config:config-item>` 3.10.3, `<config:config-item-map-indexed>` 3.10.4, `<config:config-item-map-named>` 3.10.6 and `<config:config-item-set>` 3.10.2.

### 3.10.6 `<config:config-item-map-named>`

The `<config:config-item-map-named>` element contains a sequence of application setting elements. Each sequence is identified by the value of its `config:name` attribute.

The `<config:config-item-map-named>` element is usable within the following elements: `<config:config-item-map-entry>` 3.10.5 and `<config:config-item-set>` 3.10.2.

The `<config:config-item-map-named>` element has the following attribute: `config:name` 19.29.

The `<config:config-item-map-named>` element has the following child element: `<config:config-item-map-entry>` 3.10.5.

## 3.11 Cursor Position Setting

A processing instruction with `PITarget` (see §2.6 of [XML1.0]) followed by the string `cursor-position` represents a text cursor position in a document. The string `cursor-position` may be followed by an arbitrary implementation-defined string.

**Note:** Where a text cursor position is not sufficient to recreate a document view, producers may use arbitrary document-specific settings in addition to a cursor position processing instruction.

### 3.12 `<office:scripts>`

The `<office:scripts>` element contains all the `<office:script>` elements for a document.

Each `<office:script>` element contains a script. A document may contain scripts in different scripting languages.

**Note:** Scripts need not use XML document models.

The `<office:scripts>` element may also contain `<office:event-listeners>` 10.4.4 elements which contain the events assigned to the document itself.

The `<office:scripts>` element is usable within the following elements: `<office:document>` 3.1.2 and `<office:document-content>` 3.1.3.2.

The `<office:scripts>` element has no attributes.

The `<office:scripts>` element has the following child elements: `<office:event-listeners>` 10.3.19 and `<office:script>` 3.13.

### 3.13 `<office:script>`

The `<office:script>` element contains a script.

**Note:** In most situations, the element contains the source code of the script, but it may also contain a compiled version of the script or a link to external script code.

The `<office:script>` element is usable within the following element: `<office:scripts>` 3.12.

The `<office:script>` element has the following attribute: `script:language` 19.434.

The `<office:script>` element has mixed content where arbitrary child elements are permitted.

## 3.14 `<office:font-face-decls>`

The `<office:font-face-decls>` element contains all the font face declarations (`<style:font-face>` 16.23 elements) for a document.

The `<office:font-face-decls>` element is usable within the following elements: `<office:document>` 3.1.2, `<office:document-content>` 3.1.3.2 and `<office:document-styles>` 3.1.3.3.

The `<office:font-face-decls>` element has no attributes.

The `<office:font-face-decls>` element has the following child element: `<style:font-face>` 16.23.

## 3.15 Styles

### 3.15.1 General

There are three types of styles for documents: common, automatic and master.

Page layouts and styles are defined by `<style:page-layout>` 16.5 and `<style:master-page>` 16.9 elements, respectively.

### 3.15.2 `<office:styles>`

The `<office:styles>` element contains common styles used in a document. A common style is a style chosen by a user for a document or portion thereof.

The `<office:styles>` element is usable within the following elements: `<office:document>` 3.1.2 and `<office:document-styles>` 3.1.3.3.

The `<office:styles>` element has no attributes.

The `<office:styles>` element has the following child elements: `<draw:fill-image>` 16.42.6, `<draw:gradient>` 16.42.1, `<draw:hatch>` 16.42.5, `<draw:marker>` 16.42.8, `<draw:opacity>` 16.42.7, `<draw:stroke-dash>` 16.42.9, `<number:boolean-style>` 16.29.24, `<number:currency-style>` 16.29.8, `<number:date-style>` 16.29.11, `<number:num-list-format>` 16.29.30, `<number:number-style>` 16.29.2, `<number:percentage-style>` 16.29.10, `<number:text-style>` 16.29.26, `<number:time-style>` 16.29.19, `<style:default-page-layout>` 16.8, `<style:default-style>` 16.4, `<style:presentation-page-layout>` 16.43, `<style:style>` 16.2, `<svg:linearGradient>` 16.42.2, `<svg:radialGradient>` 16.42.3, `<table:table-template>` 16.20, `<text:bibliography-configuration>` 16.31.6, `<text:linenumbering-configuration>` 16.31.1, `<text:list-style>` 16.32, `<text:notes-configuration>` 16.31.3 and `<text:outline-style>` 16.36.

### 3.15.3 `<office:automatic-styles>`

The `<office:automatic-styles>` element contains automatic styles used in a document.

An automatic style is a set of formatting properties treated as properties of the object to which the style is assigned.

**Note:** Common and automatic styles behave differently in OpenDocument editing consumers. Common styles present to a user as a named set of

formatting properties. The formatting properties of an automatic style present to a user as properties of the object to which the style is applied.

The `<office:automatic-styles>` element is usable within the following elements: `<office:document>` 3.1.2, `<office:document-content>` 3.1.3.2 and `<office:document-styles>` 3.1.3.3.

The `<office:automatic-styles>` element has no attributes.

The `<office:automatic-styles>` element has the following child elements:

`<number:boolean-style>` 16.29.24, `<number:currency-style>` 16.29.8, `<number:date-style>` 16.29.11, `<number:num-list-format>` 16.29.30, `<number:number-style>` 16.29.2, `<number:percentage-style>` 16.29.10, `<number:text-style>` 16.29.26, `<number:time-style>` 16.29.19, `<style:page-layout>` 16.5, `<style:style>` 16.2 and `<text:list-style>` 16.32.

### 3.15.4 `<office:master-styles>`

The `<office:master-styles>` element contains master styles that are used in a document. A master style contains formatting and other content that is displayed with document content when the style is used.

The `<office:master-styles>` element is usable within the following elements: `<office:document>` 3.1.2 and `<office:document-styles>` 3.1.3.3.

The `<office:master-styles>` element has no attributes.

The `<office:master-styles>` element has the following child elements: `<draw:layer-set>` 10.2.2, `<style:handout-master>` 10.2.1 and `<style:master-page>` 16.9.

## 3.16 Document Signatures

An OpenDocument document that is stored in a package may have one or more digital signatures applied to the package.

Document signatures are stored in a file called META-INF/documentsignatures.xml in the package as described in section 3.5 of the OpenDocument specification Part2. Document signatures shall contain a `<ds:Reference>` element for each file within the package, with the exception that `<ds:Reference>` elements for the META-INF/documentsignatures.xml file containing the signature, and any files contained in the package whose relative path starts with "external-data/" should be omitted.

Signatures other than document signatures are implementation-defined.

## 3.17 Foreign Elements and Attributes

OpenDocument extended documents may contain elements and attributes not defined by the OpenDocument schema. Elements and attributes not defined by the OpenDocument schema are called *foreign elements* and *attributes*. Foreign elements and attributes shall not use a namespace listed in tables 1, 2 or 3 of section 1.5.

If a foreign element has a `<text:h>` 5.1.2 or `<text:p>` 5.1.3 ancestor element, and is a child element of an element for which the OpenDocument schema permits the inclusion of character data, and if the OpenDocument schema permits the inclusion of character data for all its ancestors up to the `<text:p>` or `<text:h>` element ancestor element, or a `<text:ruby-base>` 6.4.2 ancestor element, then the element's content may be interpreted by conforming OpenDocument consumers, and the document itself shall be valid against the OpenDocument schema as if the foreign element's start- and end-tags or its empty-element-tag are removed.

For a foreign element that occurs at another location, conforming consumers should not interpret but may preserve the element's content.

Conforming extended producers should not use foreign elements and attributes for features defined in the OpenDocument specification.

When a conforming consumer encounters an OpenDocument defined attribute that has a value that is not defined by OpenDocument, it should:

- 1) If the attribute has a specified default value, use its default value, or
- 2) If the attribute does not have a specified default value, ignore the attribute.

## 3.18 White Space Processing and EOL Handling

ODF processing of white space characters is in conformance with the provisions of [XML1.0].

In addition, OpenDocument Consumers shall ignore all element children ([RNG] section 5, Data Model) of elements defined in this specification that are strings consisting entirely of white space characters and which do not satisfy a pattern of the OpenDocument schema definition for the element.

Additional treatment of occurrences of white space characters depends on the provisions for specific elements, attributes, and their datatypes in this specification.

**Note:** There are special OpenDocument rules for the reduction of white space sequences 3.18 to single space characters in the text obtained from `<text:p> 5.1.3, <text:h> 5.1.2` and their descendant elements.

## 3.19 MIME Types and File Name Extensions

Office documents contained in a package use the MIME types and file name extensions listed in Appendix C. See 3.1.3.

Office documents not contained in a package should use the MIME type `text/xml`.

Only MIME types and extensions that have been registered according to [RFC6838] should be used for office documents that conform to this specification. The MIME types and extensions listed in appendix C should be used where appropriate.

---

## 4 Metadata

### 4.1 General

Metadata is general information about a document or its content.

OpenDocument supports five types of metadata:

- 1) RDF metadata describing documents or the content of identifiable OpenDocument elements. 4.2.2
- 2) Text content being used as RDF metadata. 4.2.1
- 3) Pre-defined metadata (meta.xml). 4.3.2
- 4) User-defined metadata (using the `<meta:user-defined>` element). 4.3.3
- 5) Custom metadata (custom XML elements within meta.xml). 4.3.1

Pre-defined and user-defined metadata are both stored in the metadata `<office:meta>` 3.18 element. The elements representing this metadata may be omitted or occur multiple times. The updating of multiple instances of the same metadata elements is implementation-dependent.

### 4.2 RDF Metadata

#### 4.2.1 In-Content Metadata (RDFa)

Metadata in OpenDocument documents may be expressed using the model of the W3C Resource Description Framework [RDF-CONCEPTS].

RDF metadata is supported as in-content metadata and by manifest.rdf. 4.2.2. RDF metadata can be attached to elements in the "content.xml" and "styles.xml" file. Elements supporting this type of metadata have the following attributes: `xhtml:about` 19.911, `xhtml:property` 19.914, `xhtml:content` 19.912, and `xhtml:datatype` 19.913. When metadata is added using these attributes, it is referred to as in-content metadata. If "content.xml" and "styles.xml" files contain in-content metadata they shall be listed in the package's "manifest.rdf" file.

#### 4.2.2 manifest.rdf

##### 4.2.2.1 General

The OpenDocument document contains a metadata manifest, as defined in Part 2 of this specification. The [OWL] Metadata Manifest Description ontology defined in Part 2 is extended with additional properties and classes in appendix B. The following OWL classes and properties are defined.

##### 4.2.2.2 odf:ContentFile

An instance of the `odf:ContentFile` class represents a content.xml file.

The `odf:ContentFile` class is a subclass of `pkg:File`.

##### 4.2.2.3 odf:StylesFile

An instance of the `odf:StylesFile` class represents a styles.xml file.

The `odf:StylesFile` class is a subclass of `pkg:File`

#### 4.2.2.4 odf:Element

An instance of the `odf:Element` class represents an OpenDocument XML element in a `content.xml` or `styles.xml` file. It is a subclass of the `pkg:Element` class, which represents any XML element in the OpenDocument package.

**Note:** One or more `rdf:type` properties specify the metadata type of an `odf:Element`.

Every XML element that is defined by this specification and has an attribute of type ID may be described using an OWL class. The IRI of an OWL class for an element is the concatenation of the element's QName namespace IRI and its local name part. These classes are subclasses of `odf:Element`.

The following properties are defined for the subclass describing a `<text:meta-field>` element (`urn:oasis:names:tc:opendocument:xmlns:text:1.0meta-field`): `odf:prefix` 4.2.2.5 and `odf:suffix` 4.2.2.6.

#### 4.2.2.5 odf:prefix

The `odf:prefix` property defines the prefix content of a `<text:meta-field>` element 7.5.19.

This property can be used with the following class:

`urn:oasis:names:tc:opendocument:xmlns:text:1.0meta-field` 4.2.2.4.

#### 4.2.2.6 odf:suffix

The `odf:suffix` property defines the suffix content of a `<text:meta-field>` element 7.5.19.

This property can be used with the following class:

`urn:oasis:names:tc:opendocument:xmlns:text:1.0meta-field` 4.2.2.4.

### 4.3 Non-RDF Metadata

#### 4.3.1 General

Non-RDF metadata in OpenDocument is composed of pre-defined metadata elements, user-defined metadata elements, and custom metadata elements.

The pre-defined metadata elements have defined semantics. They should be processed by consumers and updated by producers. They can be referenced from within the document using text fields.

The pre-defined metadata elements borrow heavily upon the metadata standards developed by the Dublin Core Metadata Initiative (<http://www.dublincore.org>). Metadata elements drawn directly from the Dublin Core work are in the `http://purl.org/dc/elements/1.1` namespace.

User-defined metadata specifies a triplet of name, type, and value. Consumers can present these values to the user.

Custom metadata elements are arbitrary elements inside an `<office:meta>` element. The semantics of custom metadata elements is implementation-defined.

Custom metadata elements shall only occur in OpenDocument extended documents. The use of custom metadata is deprecated in favor of RDF/XML based metadata.

**Note:** Consumers need not process custom metadata other than to preserve it when editing a document.

## 4.3.2 Pre-Defined Metadata Elements

### 4.3.2.1 <meta:generator>

The `<meta:generator>` element contains a string that identifies the OpenDocument producer that was used to create or last modify the document. This string should match the definition for user-agents in the HTTP protocol as specified in section 14.43 of [RFC2616]. The generator string should allow OpenDocument consumers to distinguish between all released versions of a producer.

**Note:** Release versions of a producer could be distinguished based on build identifiers or patch-level information.

If an OpenDocument producer that creates a document cannot provide an identifier string, the producer shall not export this element. If a producer stores a modified document created by another producer cannot provide a unique identifier, it shall not export the original identifier belonging to the producer that created the document.

The `<meta:generator>` element is usable within the following element: `<office:meta> 3.2`.

The `<meta:generator>` element has no attributes.

The `<meta:generator>` element has no child elements.

The `<meta:generator>` element has content of data type `string 18.2`.

### 4.3.2.2 <dc:title>

The `<dc:title>` element specifies the title of a document.

The `<dc:title>` element is usable within the following element: `<office:meta> 3.2`.

The `<dc:title>` element has no attributes.

The `<dc:title>` element has no child elements.

The `<dc:title>` element has content of data type `string 18.2`.

### 4.3.2.3 <dc:description>

The `<dc:description>` element contains a description of a document.

The `<dc:description>` element is usable within the following element: `<office:meta> 3.2`.

The `<dc:description>` element has no attributes.

The `<dc:description>` element has no child elements.

The `<dc:description>` element has content of data type `string 18.2`.

### 4.3.2.4 <dc:subject>

The `<dc:subject>` element specifies the subject of a document.

The `<dc:subject>` element is usable within the following element: `<office:meta> 3.2`.

The `<dc:subject>` element has no attributes.

The `<dc:subject>` element has no child elements.

The `<dc:subject>` element has content of data type `string 18.2`.

#### 4.3.2.5 <meta:keyword>

The `<meta:keyword>` element contains a keyword pertaining to a document.

The `<meta:keyword>` element is usable within the following element: `<office:meta> 3.2`.

The `<meta:keyword>` element has no attributes.

The `<meta:keyword>` element has no child elements.

The `<meta:keyword>` element has content of data type `string 18.2`.

#### 4.3.2.6 <meta:initial-creator>

The `<meta:initial-creator>` element specifies the name of the initial creator of a document.

The `<meta:initial-creator>` element is usable within the following element: `<office:meta> 3.2`.

The `<meta:initial-creator>` element has no attributes.

The `<meta:initial-creator>` element has no child elements.

The `<meta:initial-creator>` element has content of data type `string 18.2`.

#### 4.3.2.7 <dc:creator>

The `<dc:creator>` element specifies the name of the person who last modified a document (`<office:meta>`), who created an annotation (`<office:annotation>`), who authored a change (`<office:change-info>`).

The `<dc:creator>` element is usable within the following elements: `<office:annotation> 14.1`, `<office:change-info> 5.5.7` and `<office:meta> 3.2`.

The `<dc:creator>` element has no attributes.

The `<dc:creator>` element has no child elements.

The `<dc:creator>` element has content of data type `string 18.2`.

#### 4.3.2.8 <meta:printed-by>

The `<meta:printed-by>` element specifies the name of the last person who printed a document.

The `<meta:printed-by>` element is usable within the following element: `<office:meta> 3.2`.

The `<meta:printed-by>` element has no attributes.

The `<meta:printed-by>` element has no child elements.

The `<meta:printed-by>` element has content of data type `string 18.2`.

#### 4.3.2.9 <meta:creation-date>

The `<meta:creation-date>` element specifies the date and time when a document was created.

The `<meta:creation-date>` element is usable within the following element: `<office:meta> 3.2`.

The `<meta:creation-date>` element has no attributes.

The `<meta:creation-date>` element has no child elements.

The `<meta:creation-date>` element has content of data type `dateTime` 18.2.

#### 4.3.2.10 `<dc:date>`

The `<dc:date>` element specifies the date and time when the document was last modified (`<office:meta>`), when an annotation was created (`<office:annotation>`), when a change was made (`<office:change-info>`).

The `<dc:date>` element is usable within the following elements: `<office:annotation>` 14.1, `<office:change-info>` 5.5.7 and `<office:meta>` 3.2.

The `<dc:date>` element has no attributes.

The `<dc:date>` element has no child elements.

The `<dc:date>` element has content of data type `dateTime` 18.2.

#### 4.3.2.11 `<meta:print-date>`

The `<meta:print-date>` element specifies the date and time when a document was last printed.

The `<meta:print-date>` element is usable within the following element: `<office:meta>` 3.2.

The `<meta:print-date>` element has no attributes.

The `<meta:print-date>` element has no child elements.

The `<meta:print-date>` element has content of data type `dateTime` 18.2.

#### 4.3.2.12 `<meta:template>`

The `<meta:template>` element specifies an IRI for the document template that was used to create a document. The IRI is specified using the `xlink:href` attribute. See [XLink].

The `<meta:template>` element is usable within the following element: `<office:meta>` 3.2.

The `<meta:template>` element has the following attributes: `meta:date` 19.323, `xlink:actuate` 19.915, `xlink:href` 19.916, `xlink:title` 19.918 and `xlink:type` 19.919.

The `<meta:template>` element has no child elements.

#### 4.3.2.13 `<meta:auto-reload>`

The `<meta:auto-reload>` element specifies whether a document is reloaded or replaced by another document after a specified period of time has elapsed.

The `<meta:auto-reload>` element is usable within the following element: `<office:meta>` 3.2.

The `<meta:auto-reload>` element has the following attributes: `meta:delay` 19.324, `xlink:actuate` 19.915, `xlink:href` 19.916, `xlink:show` 19.917 and `xlink:type` 19.919.

The `<meta:auto-reload>` element has no child elements.

#### 4.3.2.14 `<meta:hyperlink-behaviour>`

The `<meta:hyperlink-behaviour>` element specifies the default behavior for hyperlinks in a document.

The `<meta:hyperlink-behaviour>` element is usable within the following element: `<office:meta> 3.2.`

The `<meta:hyperlink-behaviour>` element has the following attributes: `office:target-frame-name 19.385` and `xlink:show 19.917`.

The `<meta:hyperlink-behaviour>` element has no child elements.

#### 4.3.2.15 `<dc:language>`

The `<dc:language>` element specifies the default language of a document.

The `<dc:language>` element is usable within the following element: `<office:meta> 3.2.`

The `<dc:language>` element has no attributes.

The `<dc:language>` element has no child elements.

The `<dc:language>` element has content of data type `language 18.3.17.`

#### 4.3.2.16 `<meta:editing-cycles>`

The `<meta:editing-cycles>` element specifies the number of times a document has been edited. When a document is created, this value is set to 1. Each time a document is saved, the editing-cycles number is incremented by 1.

The `<meta:editing-cycles>` element is usable within the following element: `<office:meta> 3.2.`

The `<meta:editing-cycles>` element has no attributes.

The `<meta:editing-cycles>` element has no child elements.

The `<meta:editing-cycles>` element has content of data type `nonNegativeInteger 18.2.`

#### 4.3.2.17 `<meta:editing-duration>`

The `<meta:editing-duration>` element specifies the total time spent editing a document.

The `<meta:editing-duration>` element is usable within the following element: `<office:meta> 3.2.`

The `<meta:editing-duration>` element has no attributes.

The `<meta:editing-duration>` element has no child elements.

The `<meta:editing-duration>` element has content of data type `duration 18.2.`

#### 4.3.2.18 `<meta:document-statistic>`

The `<meta:document-statistic>` element represents statistics about a document.

The `<meta:document-statistic>` element is usable within the following element: `<office:meta> 3.2.`

The `<meta:document-statistic>` element has the following attributes: `meta:cell-count 19.321`, `meta:character-count 19.322`, `meta:draw-count 19.325`, `meta:frame-count 19.326`, `meta:image-count 19.327`, `meta:non-whitespace-character-count 19.329`, `meta:object-count 19.330`, `meta:ole-object-count 19.331`, `meta:page-count 19.332`, `meta:paragraph-count 19.333`, `meta:row-count 19.334`, `meta:sentence-count 19.335`, `meta:syllable-count 19.336`, `meta:table-count 19.337` and `meta:word-count 19.339`.

The `<meta:document-statistic>` element has no child elements.

### 4.3.3 <meta:user-defined>

The `<meta:user-defined>` element specifies any additional user-defined metadata for a document.

The `<meta:user-defined>` element is usable within the following element:  
`<office:meta> 3.2.`

The `<meta:user-defined>` element has the following attributes: `meta:name 19.328`, `meta:value-type 19.338`.

The `<meta:user-defined>` element has no child elements.

The `<meta:user-defined>` element has character data content, or depending on the value of the `meta:value-type` attribute content of type `double 18.2`, `date 18.2`, `dateTime 18.2`, `duration 18.2`, `boolean 18.3.3` or `string 18.2`.

# 5 Text Content

## 5.1 Headings, Paragraphs and Basic Text Structure

### 5.1.1 General

The `<text:h>` and `<text:p>` elements represent headings and paragraphs, respectively.

Headings and paragraphs are collectively referred to as *paragraph elements*.

### 5.1.2 `<text:h>`

The `<text:h>` element represents a heading in a document. Headings define the division structure for a document. A chapter or section begins with a heading and extends to the next heading at the same or higher level.

The `<text:h>` element is usable within the following elements: `<draw:text-box>` 10.4.3, `<office:text>` 3.4, `<style:footer>` 16.12, `<style:footer-first>` 16.13, `<style:footer-left>` 16.15, `<style:header>` 16.10, `<style:header-first>` 16.11, `<style:header-left>` 16.14, `<table:covered-table-cell>` 9.1.5, `<table:table-cell>` 9.1.4, `<text:deletion>` 5.5.5, `<text:index-body>` 8.2.2, `<text:index-title>` 8.2.3, `<text:list-header>` 5.3.3, `<text:list-item>` 5.3.4, `<text:note-body>` 6.3.4, `<text:numbered-paragraph>` 5.3.6 and `<text:section>` 5.4.

The `<text:h>` element has the following attributes: `text:class-names` 19.775, `text:cond-style-name` 19.781, `text:id` 19.815, `text:is-list-header` 19.822, `text:outline-level` 19.850, `text:restart-numbering` 19.863, `text:start-value` 19.874, `text:style-name` 19.880, `xhtml:about` 19.911, `xhtml:content` 19.912, `xhtml:datatype` 19.913, `xhtml:property` 19.914 and `xml:id` 19.920.

The `<text:h>` element has the following child elements: `<dr3d:scene>` 10.5.2, `<draw:a>` 10.4.12, `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:connector>` 10.3.10, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:g>` 10.3.15, `<draw:line>` 10.3.3, `<draw:measure>` 10.3.12, `<draw:page-thumbnail>` 10.3.14, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2, `<draw:regular-polygon>` 10.3.6, `<office:annotation>` 14.1, `<office:annotation-end>` 14.2, `<presentation:date-time>` 10.9.4.5, `<presentation:footer>` 10.9.4.3, `<presentation:header>` 10.9.4.1, `<text:a>` 6.1.8, `<text:alphabetical-index-mark>` 8.1.10, `<text:alphabetical-index-mark-end>` 8.1.9, `<text:alphabetical-index-mark-start>` 8.1.8, `<text:author-initials>` 7.3.7.2, `<text:author-name>` 7.3.7.1, `<text:bibliography-mark>` 8.1.11, `<text:bookmark>` 6.2.1.2, `<text:bookmark-end>` 6.2.1.4, `<text:bookmark-ref>` 7.7.6, `<text:bookmark-start>` 6.2.1.3, `<text:change>` 5.5.8.4, `<text:change-end>` 5.5.8.3, `<text:change-start>` 5.5.8.2, `<text:chapter>` 7.3.8, `<text:character-count>` 7.5.18.5, `<text:conditional-text>` 7.7.3, `<text:creation-date>` 7.5.3, `<text:creation-time>` 7.5.4, `<text:creator>` 7.5.17, `<text:database-display>` 7.6.3, `<text:database-name>` 7.6.7, `<text:database-next>` 7.6.4, `<text:database-row-number>` 7.6.6, `<text:database-row-select>` 7.6.5, `<text:date>` 7.3.2, `<text:dde-connection>` 7.7.12, `<text:description>` 7.5.5, `<text:drop-down>` 7.4.16, `<text:editing-cycles>` 7.5.13, `<text:editing-duration>` 7.5.14, `<text:execute-macro>` 7.7.10, `<text:expression>` 7.4.14, `<text:file-name>` 7.3.9, `<text:hidden-paragraph>` 7.7.11, `<text:hidden-text>`

```

7.7.4, <text:image-count> 7.5.18.7, <text:initial-creator> 7.5.2,
<text:keywords> 7.5.12, <text:line-break> 6.1.5, <text:measure> 7.7.13,
<text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:modification-date> 7.5.16,
<text:modification-time> 7.5.15, <text:note> 6.3.2, <text:note-ref> 7.7.7,
<text:number> 6.1.10, <text:object-count> 7.5.18.8, <text:page-
continuation> 7.3.5, <text:page-count> 7.5.18.2, <text:page-number> 7.3.4,
<text:page-variable-get> 7.7.1.3, <text:page-variable-set> 7.7.1.2,
<text:paragraph-count> 7.5.18.3, <text:placeholder> 7.7.2, <text:print-
date> 7.5.8, <text:print-time> 7.5.7, <text:printed-by> 7.5.9,
<text:reference-mark> 6.2.2.2, <text:reference-mark-end> 6.2.2.4,
<text:reference-mark-start> 6.2.2.3, <text:reference-ref> 7.7.5,
<text:ruby> 6.4, <text:s> 6.1.3, <text:script> 7.7.9, <text:sender-city>
7.3.6.13, <text:sender-company> 7.3.6.10, <text:sender-country> 7.3.6.15,
<text:sender-email> 7.3.6.7, <text:sender-fax> 7.3.6.9, <text:sender-
firstname> 7.3.6.2, <text:sender-initials> 7.3.6.4, <text:sender-lastname>
7.3.6.3, <text:sender-phone-private> 7.3.6.8, <text:sender-phone-work>
7.3.6.11, <text:sender-position> 7.3.6.6, <text:sender-postal-code> 7.3.6.14,
<text:sender-state-or-province> 7.3.6.16, <text:sender-street> 7.3.6.12,
<text:sender-title> 7.3.6.5, <text:sequence> 7.4.13, <text:sequence-ref>
7.7.8, <text:sheet-name> 7.3.11, <text:soft-page-break> 5.6, <text:span>
6.1.7, <text:subject> 7.5.11, <text:tab> 6.1.4, <text:table-count> 7.5.18.6,
<text:table-formula> 7.7.14, <text:template-name> 7.3.10, <text:text-
input> 7.4.15, <text:time> 7.3.3, <text:title> 7.5.10, <text:toc-mark> 8.1.4,
<text:toc-mark-end> 8.1.3, <text:toc-mark-start> 8.1.2, <text:user-
defined> 7.5.6, <text:user-field-get> 7.4.9, <text:user-field-input> 7.4.10,
<text:user-index-mark> 8.1.7, <text:user-index-mark-end> 8.1.6,
<text:user-index-mark-start> 8.1.5, <text:variable-get> 7.4.5,
<text:variable-input> 7.4.6, <text:variable-set> 7.4.4 and <text:word-
count> 7.5.18.4.

```

### 5.1.3 <text:p>

The <text:p> element represents a paragraph, which is the basic unit of text in an OpenDocument file.

The <text:p> element is usable within the following elements: <chart: data-label> 11.15, <chart: equation> 11.19, <chart: footer> 11.2.3, <chart: label-
separator> 17.24, <chart: legend> 11.4, <chart: subtitle> 11.2.2,
<chart: title> 11.2.1, <draw: caption> 10.3.11, <draw: circle> 10.3.8,
<draw: connector> 10.3.10, <draw: custom-shape> 10.6.1, <draw: ellipse> 10.3.9,
<draw: image> 10.4.4, <draw: line> 10.3.3, <draw: measure> 10.3.12, <draw: path>
10.3.7, <draw: polygon> 10.3.5, <draw: polyline> 10.3.4, <draw: rect> 10.3.2,
<draw: regular-polygon> 10.3.6, <draw: text-box> 10.4.3, <form: textarea>
13.5.3, <office: annotation> 14.1, <office: change-info> 5.5.7, <office: text>
3.4, <style: footer> 16.12, <style: footer-first> 16.13, <style: footer-left>
16.15, <style: header> 16.10, <style: header-first> 16.11, <style: header-
left> 16.14, <style: region-center> 16.17, <style: region-left> 16.16,
<style: region-right> 16.18, <table: change-track-table-cell> 9.9.16,
<table: covered-table-cell> 9.1.5, <table: error-message> 9.4.7,
<table: help-message> 9.4.6, <table: table-cell> 9.1.4, <text: deletion> 5.5.5,
<text: index-body> 8.2.2, <text: index-title> 8.2.3, <text: list-header> 5.3.3,
<text: list-item> 5.3.4, <text: note-body> 6.3.4, <text: numbered-paragraph>
5.3.6 and <text: section> 5.4.

The <text:p> element has the following attributes: text: class-names 19.775, text: cond-style-name 19.781, text: id 19.815, text: style-name 19.880,

xhtml:about 19.911, xhtml:content 19.912, xhtml:datatype 19.913, xhtml:property 19.914 and xml:id 19.920.

The `<text:p>` element has the following child elements: `<dr3d:scene>` 10.5.2, `<draw:a>` 10.4.12, `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:connector>` 10.3.10, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:g>` 10.3.15, `<draw:line>` 10.3.3, `<draw:measure>` 10.3.12, `<draw:page-thumbnail>` 10.3.14, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2, `<draw:regular-polygon>` 10.3.6, `<office:annotation>` 14.1, `<office:annotation-end>` 14.2, `<presentation:date-time>` 10.9.4.5, `<presentation:footer>` 10.9.4.3, `<presentation:header>` 10.9.4.1, `<text:a>` 6.1.8, `<text:alphabetical-index-mark>` 8.1.10, `<text:alphabetical-index-mark-end>` 8.1.9, `<text:alphabetical-index-mark-start>` 8.1.8, `<text:author-initials>` 7.3.7.2, `<text:author-name>` 7.3.7.1, `<text:bibliography-mark>` 8.1.11, `<text:bookmark>` 6.2.1.2, `<text:bookmark-end>` 6.2.1.4, `<text:bookmark-ref>` 7.7.6, `<text:bookmark-start>` 6.2.1.3, `<text:change>` 5.5.8.4, `<text:change-end>` 5.5.8.3, `<text:change-start>` 5.5.8.2, `<text:chapter>` 7.3.8, `<text:character-count>` 7.5.18.5, `<text:conditional-text>` 7.7.3, `<text:creation-date>` 7.5.3, `<text:creation-time>` 7.5.4, `<text:creator>` 7.5.17, `<text:database-display>` 7.6.3, `<text:database-name>` 7.6.7, `<text:database-next>` 7.6.4, `<text:database-row-number>` 7.6.6, `<text:database-row-select>` 7.6.5, `<text:date>` 7.3.2, `<text:dde-connection>` 7.7.12, `<text:description>` 7.5.5, `<text:drop-down>` 7.4.16, `<text:editing-cycles>` 7.5.13, `<text:editing-duration>` 7.5.14, `<text:execute-macro>` 7.7.10, `<text:expression>` 7.4.14, `<text:file-name>` 7.3.9, `<text:hidden-paragraph>` 7.7.11, `<text:hidden-text>` 7.7.4, `<text:image-count>` 7.5.18.7, `<text:initial-creator>` 7.5.2, `<text:keywords>` 7.5.12, `<text:line-break>` 6.1.5, `<text:measure>` 7.7.13, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:modification-date>` 7.5.16, `<text:modification-time>` 7.5.15, `<text:note>` 6.3.2, `<text:note-ref>` 7.7.7, `<text:object-count>` 7.5.18.8, `<text:page-continuation>` 7.3.5, `<text:page-count>` 7.5.18.2, `<text:page-number>` 7.3.4, `<text:page-variable-get>` 7.7.1.3, `<text:page-variable-set>` 7.7.1.2, `<text:paragraph-count>` 7.5.18.3, `<text:placeholder>` 7.7.2, `<text:print-date>` 7.5.8, `<text:print-time>` 7.5.7, `<text:printed-by>` 7.5.9, `<text:reference-mark>` 6.2.2.2, `<text:reference-mark-end>` 6.2.2.4, `<text:reference-mark-start>` 6.2.2.3, `<text:reference-ref>` 7.7.5, `<text:ruby>` 6.4, `<text:s>` 6.1.3, `<text:script>` 7.7.9, `<text:sender-city>` 7.3.6.13, `<text:sender-company>` 7.3.6.10, `<text:sender-country>` 7.3.6.15, `<text:sender-email>` 7.3.6.7, `<text:sender-fax>` 7.3.6.9, `<text:sender-firstname>` 7.3.6.2, `<text:sender-initials>` 7.3.6.4, `<text:sender-lastname>` 7.3.6.3, `<text:sender-phone-private>` 7.3.6.8, `<text:sender-phone-work>` 7.3.6.11, `<text:sender-position>` 7.3.6.6, `<text:sender-postal-code>` 7.3.6.14, `<text:sender-state-or-province>` 7.3.6.16, `<text:sender-street>` 7.3.6.12, `<text:sender-title>` 7.3.6.5, `<text:sequence>` 7.4.13, `<text:sequence-ref>` 7.7.8, `<text:sheet-name>` 7.3.11, `<text:soft-page-break>` 5.6, `<text:span>` 6.1.7, `<text:subject>` 7.5.11, `<text:tab>` 6.1.4, `<text:table-count>` 7.5.18.6, `<text:table-formula>` 7.7.14, `<text:template-name>` 7.3.10, `<text:text-input>` 7.4.15, `<text:time>` 7.3.3, `<text:title>` 7.5.10, `<text:toc-mark>` 8.1.4, `<text:toc-mark-end>` 8.1.3, `<text:toc-mark-start>` 8.1.2, `<text:user-defined>` 7.5.6, `<text:user-field-get>` 7.4.9, `<text:user-field-input>` 7.4.10, `<text:user-index-mark>` 8.1.7, `<text:user-index-mark-end>` 8.1.6, `<text:user-index-mark-start>` 8.1.5, `<text:variable-get>` 7.4.5, `<text:variable-input>` 7.4.6, `<text:variable-set>` 7.4.4 and `<text:word-count>` 7.5.18.4.

## 5.2 <text:page-sequence>

### 5.2.1 General

The <text:page-sequence> element contains at least one <text:page> element. If there is more than one <text:page> element, the text:master-page-name attribute of the first <text:page> element specifies the default master style for a document.

The <text:page-sequence> element is usable within the following element:

<office:text> 3.4.

The <text:page-sequence> element has no attributes.

The <text:page-sequence> element has the following child element: <text:page> 5.2.2.

### 5.2.2 <text:page>

The <text:page> element represents a page in a <text:page-sequence> element.

The <text:page> element is usable within the following element: <text:page-sequence> 5.2.

The <text:page> element has the following attribute: text:master-page-name 19.839.

The <text:page> element has no child elements.

## 5.3 Lists

### 5.3.1 <text:list>

The <text:list> element represents a list. It may contain a <text:list-header> element, followed by any number of <text:list-item> elements.

Lists may be numbered. The numbering may be restarted with a specific numbering at each list item. Lists may also continue numbering from other lists in order to merge lists into a single, discontinuous list. Whether list numbering is displayed or not depends on the list style being used.

Every list has a list level, which is determined by the nesting of the <text:list> element that represents that list. If a list is not contained in another list, its list level is 1. If a list is contained within another list, the list level of the contained list is the list level of the list in which it is contained incremented by one. If a list is contained in a table cell or text box, its list level returns to 1, even if the table or text box is nested in another list.

Every list with a list level of 1 defines a list and the counter domain for its list items and any sub-list of that list. Each sub-list starts a counter for its list items and any sub-list it may contain.

Every list, including sub-lists, may have a list style which is applied to its list items and sub-lists. A list style specified for a sub-list overrides the list style specified for the list in which the sub-list is contained.

The <text:list> element is usable within the following elements: <draw:caption> 10.3.11, <draw:circle> 10.3.8, <draw:connector> 10.3.10, <draw:custom-shape> 10.6.1, <draw:ellipse> 10.3.9, <draw:image> 10.4.4, <draw:line> 10.3.3, <draw:measure> 10.3.12, <draw:path> 10.3.7, <draw:polygon> 10.3.5, <draw:polyline> 10.3.4, <draw:rect> 10.3.2, <draw:regular-polygon> 10.3.6, <draw:text-box> 10.4.3, <office:annotation> 14.1, <office:text> 3.4, <style:footer> 16.12, <style:footer-first> 16.13, <style:footer-left> 16.15, <style:header> 16.10, <style:header-first> 16.11, <style:header-left> 16.14, <table:covered-table-cell> 9.1.5, <table:table-cell> 9.1.4,

<text:deletion> 5.5.5, <text:index-body> 8.2.2, <text:index-title> 8.2.3, <text:list-header> 5.3.3, <text:list-item> 5.3.4, <text:note-body> 6.3.4 and <text:section> 5.4.

The <text:list> element has the following attributes: text:continue-list 19.785, text:continue-numbering 19.786, text:style-name 19.880 and xml:id 19.920.

The <text:list> element has the following child elements: <text:list-header> 5.3.3 and <text:list-item> 5.3.4.

### 5.3.2 Default List Style

If a list does not have a style:name attribute and therefore no list style is specified, one of the following actions is taken:

- If the list is contained in another list, the list style defaults to the style of the surrounding list.
- If there is no list style specified for the surrounding list, but the list contains paragraphs that have paragraph styles attached that specify a list style, that list style is used.
- An implementation-dependent default is applied to the list.

To determine which formatting properties are applied to a list, the list level and its style name are taken into account. 16.32.

### 5.3.3 <text:list-header>

The <text:list-header> element represents a list header and is a type of list item. It contains one or more paragraphs that are displayed before a list. The paragraphs are formatted as list items but they do not have a preceding number or bullet.

The <text:list-header> element is usable within the following element: <text:list> 5.3.1.

The <text:list-header> element has the following attribute: xml:id 19.920.

The <text:list-header> element has the following child elements: <text:h> 5.1.2, <text:list> 5.3.1, <text:number> 6.1.10, <text:p> 5.1.3 and <text:soft-page-break> 5.6.

### 5.3.4 <text:list-item>

The <text:list-item> element represents an item in a list.

The <text:list-item> element is usable within the following element: <text:list> 5.3.1.

The <text:list-item> element has the following attributes: text:start-value 19.874, text:style-override 19.881 and xml:id 19.920.

The <text:list-item> element has the following child elements: <text:h> 5.1.2, <text:list> 5.3.1, <text:number> 6.1.10, <text:p> 5.1.3 and <text:soft-page-break> 5.6.

### 5.3.5 List Item Style Rules

The first line in a list item is may be preceded by a bullet or number, depending on the list style assigned to the list. If a list item starts another list and does not contain any text, no bullet or number is displayed.

The list style which is applied to a list need not contain a list level style for the list level the list item specifies. The following rules are applied to determine the list level style in this case

- If the list is contained in another list, the list level style defined in the list style which is applied to the surrounding list for the level of the item is used.
- If a list is not contained in (surrounded by) another list or does not have an assigned list level style, the list level style defined by the default list style is used. See 5.3.2.

### 5.3.6 <text:numbered-paragraph> (deprecated)

The `<text:numbered-paragraph>` element numbers an individual paragraph, as if it was part of a list at a specified level.

**Note:** Numbered paragraphs form an equivalent, alternative way of specifying lists. A list in `<text:list>` representation can be converted into a list in `<text:numbered-paragraph>` representation and vice versa.

The `<text:numbered-paragraph>` element is deprecated in favor of the `<text:list>` 5.3.1 element.

A numbered paragraph can be assigned a list level. A numbered paragraph is equivalent to a list nested to the given level, containing one list item with one paragraph. If no level is given, the numbered paragraph is interpreted as being on level 1.

If a list style is not specified for a numbered paragraph, the list style of the previous numbered paragraph in the same sublist is used. If this paragraph is the first in the sublist, the list style of the previous paragraph in the higher list level is used. If there is no specified list style, the default list style is applied. 5.3.2

If a list level style is missing in the applied list style of a numbered paragraph, the same rules are applied to determine this list level style as for list items.

The text of a formatted number can be included in a `<text:number>` element.

**Note:** This text can be used by consumers that do not support numbering.

The `<text:numbered-paragraph>` element is usable within the following elements: `<draw:text-box>` 10.4.3, `<office:text>` 3.4, `<table:covered-table-cell>` 9.1.5, `<table:table-cell>` 9.1.4, `<text:deletion>` 5.5.5, `<text:index-body>` 8.2.2, `<text:index-title>` 8.2.3, `<text:note-body>` 6.3.4 and `<text:section>` 5.4.

The `<text:numbered-paragraph>` element has the following attributes: `text:continue-numbering` 19.786, `text:level` 19.834, `text:list-id` 19.835, `text:start-value` 19.874, `text:style-name` 19.880 and `xml:id` 19.920.

The `<text:numbered-paragraph>` element has the following child elements: `<text:h>` 5.1.2, `<text:number>` 6.1.10 and `<text:p>` 5.1.3.

### 5.3.7 Numbered Paragraph Style Rules (deprecated)

If a list style is not specified for a numbered paragraph, the list style of the previous numbered paragraph in the same list is used. If this paragraph is the first in the list, the list style of the previous numbered paragraph in a higher list level is used. If there is no specified style available, the default style is applied. 5.3.2

If a list level style is missing in the applied list style of a numbered paragraph, the rules for styles on list items are applied. 5.3.5

## 5.4 <text:section>

### 5.4.1 General

The `<text:section>` element represents a named portion of content in a document.

Sections specify formatting properties for a region of text or text that is automatically acquired from an external data source or document, or another text section.

Sections support two ways of linking to external content. If a section is linked to another document, the link can be through one of the following:

- A resource identified by an XLink, represented by a `<text:section-source>` element
- Dynamic Data Exchange (DDE), represented by a `<office:dde-source>` element

The `<text:section-source>` or `<office:dde-source>` elements are mutually exclusive alternatives. Either element may occur only once as the first child element of a `<text:section>` element.

A section that links to external content should contain a full representation of the external content that appears in the section.

Linked external content may have an arbitrary file format. Consumers may process links to external content and the external content itself to update the content that appears in a section.

The `<text:section>` element is usable within the following elements: `<draw:text-box>` 10.4.3, `<office:text>` 3.4, `<style:footer>` 16.12, `<style:footer-first>` 16.13, `<style:footer-left>` 16.15, `<style:header>` 16.10, `<style:header-first>` 16.11, `<style:header-left>` 16.14, `<table:covered-table-cell>` 9.1.5, `<table:table-cell>` 9.1.4, `<text:deletion>` 5.5.5, `<text:index-body>` 8.2.2, `<text:index-title>` 8.2.3, `<text:note-body>` 6.3.4 and `<text:section>` 5.4.

The `<text:section>` element has the following attributes: `text:condition` 19.782, `text:display` 19.802, `text:name` 19.841, `text:protected` 19.855, `text:protection-key` 19.856, `text:protection-key-digest-algorithm` 19.857, `text:style-name` 19.880 and `xml:id` 19.920.

The `<text:section>` element has the following child elements: `<dr3d:scene>` 10.5.2, `<draw:a>` 10.4.12, `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:connector>` 10.3.10, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:g>` 10.3.15, `<draw:line>` 10.3.3, `<draw:measure>` 10.3.12, `<draw:page-thumbnail>` 10.3.14, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2, `<draw:regular-polygon>` 10.3.6, `<office:dde-source>` 14.7.5, `<table:table>` 9.1.2, `<text:alphabetical-index>` 8.8, `<text:bibliography>` 8.9, `<text:change>` 5.5.8.4, `<text:change-end>` 5.5.8.3, `<text:change-start>` 5.5.8.2, `<text:h>` 5.1.2, `<text:illustration-index>` 8.4, `<text:list>` 5.3.1, `<text:numbered-paragraph>` 5.3.6, `<text:object-index>` 8.6, `<text:p>` 5.1.3, `<text:section>` 5.4, `<text:section-source>` 5.4.2, `<text:soft-page-break>` 5.6, `<text:table-index>` 8.5, `<text:table-of-content>` 8.3 and `<text:user-index>` 8.7.

## 5.4.2 `<text:section-source>`

The `<text:section-source>` element indicates that a section is a linked section.

The `<text:section-source>` element is usable within the following element:  
`<text:section>` 5.4.

The `<text:section-source>` element has the following attributes: `text:filter-name` 19.808, `text:section-name` 19.867, `xlink:href` 19.916, `xlink:show` 19.917 and `xlink:type` 19.919.

The `<text:section-source>` element has no child elements.

## 5.5 Change Tracking

### 5.5.1 General

The under-specification of change tracking in ODF 1.2 resulted in varying implementations of this feature. Where interoperability between implementations is required, this feature should be checked for interoperability.

### 5.5.2 <text:tracked-changes>

The `<text:tracked-changes>` element acts as a container for `<text:changed-region>` elements that represent changes in a certain scope of an OpenDocument document. This scope is the element in which the `<text:tracked-changes>` element occurs. Changes in this scope shall be tracked by `<text:changed-region>` elements contained in the `<text:tracked-changes>` element in this scope.

If a `<text:tracked-changes>` element is absent, there are no tracked changes in the corresponding scope. In this case, all change mark elements (5.5.8) in this scope shall be ignored.

The `<text:tracked-changes>` element is usable within the following elements:  
`<office:text> 3.4, <style:footer> 16.12, <style:footer-first> 16.13, <style:footer-left> 16.15, <style:header> 16.10, <style:header-first> 16.11 and <style:header-left> 16.14.`

The `<text:tracked-changes>` element has the following attribute: `text:track-changes 19.889`.

The `<text:tracked-changes>` element has the following child element:  
`<text:changed-region> 5.5.3.`

### 5.5.3 <text:changed-region>

Each `<text:changed-region>` element contains a single element, one of `<text:insertion>`, `<text:deletion>`, or `<text:format-change>` that corresponds to a change being tracked within the scope of the `<text:tracked-changes>` element that contains the `<text:changed-region>` instance.

The `xml:id` attribute of the `<text:changed-region>` is referenced from the `<text:change>`, `<text:change-start>` and `<text:change-end>` elements that identify where the change applies to markup in the scope of the `<text:tracked-changes>` element.

A `<text:changed-region>` can be referenced by more than one change, but the corresponding referencing change mark elements shall be of the same change type - insertion, format change or deletion.

The `<text:changed-region>` element is usable within the following element:  
`<text:tracked-changes> 5.5.2.`

The `<text:changed-region>` element has the following attributes: `text:id 19.815` and `xml:id 19.920`.

The `<text:changed-region>` element has the following child elements:  
`<text:deletion> 5.5.5, <text:format-change> 5.5.6 and <text:insertion> 5.5.4.`

### 5.5.4 <text:insertion>

The `<text:insertion>` element contains the information that identifies the person responsible for a change and the date of that change. This information may also contain one or more `<text:p>` elements which contains a comment on the insertion.

The `<text:insertion>` element's parent `<text:changed-region>` element has an `xml:id` or `text:id` attribute, the value of which binds that parent element to the `text:change-id` attribute on the `<text:change-start>` and `<text:change-end>` elements.

The `<text:insertion>` element is usable within the following element: `<text:changed-region>` 5.5.3.

The `<text:insertion>` element has no attributes.

The `<text:insertion>` element has the following child element: `<office:change-info>` 5.5.7.

## 5.5.5 `<text:deletion>`

The `<text:deletion>` element contains information that identifies the person responsible for a deletion and the date of that deletion. This information may also contain one or more `<text:p>` elements which contains a comment on the deletion. The `<text:deletion>` element may also contain content that was deleted while change tracking was enabled. The position where the text was deleted is marked by a `<text:change>` element.

Deleted text is contained in a paragraph element. To reconstruct the original text, the paragraph containing the deleted text is merged with its surrounding paragraph or heading element.

To reconstruct the text before a deletion took place:

- If the change mark is inside a paragraph, insert the content that was deleted, but remove all leading start tags up to and including the first `<text:p>` element and all trailing end tags up to and including the last `</text:p>` or `</text:h>` element. If the last trailing element is a `</text:h>`, change the end tag `</text:p>` following this insertion to a `</text:h>` element.
- If the change mark is inside a heading, insert the content that was deleted, but remove all leading start tags up to and including the first `<text:h>` element and all trailing end tags up to and including the last `</text:h>` or `</text:p>` element. If the last trailing element is a `</text:p>`, change the end tag `</text:h>` following this insertion to a `</text:p>` element.
- Otherwise, copy the text content of the `<text:deletion>` element in place of the change mark.

The `<text:deletion>` element is usable within the following element: `<text:changed-region>` 5.5.3.

The `<text:deletion>` element has no attributes.

The `<text:deletion>` element has the following child elements: `<dr3d:scene>` 10.5.2, `<draw:a>` 10.4.12, `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:connector>` 10.3.10, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:g>` 10.3.15, `<draw:line>` 10.3.3, `<draw:measure>` 10.3.12, `<draw:page-thumbnail>` 10.3.14, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2, `<draw:regular-polygon>` 10.3.6, `<office:change-info>` 5.5.7, `<table:table>` 9.1.2, `<text:alphabetical-index>` 8.8, `<text:bibliography>` 8.9, `<text:change>` 5.5.8.4, `<text:change-end>` 5.5.8.3, `<text:change-start>` 5.5.8.2, `<text:h>` 5.1.2, `<text:illustration-index>` 8.4, `<text:list>` 5.3.1, `<text:numbered-paragraph>` 5.3.6, `<text:object-index>` 8.6, `<text:p>` 5.1.3, `<text:section>` 5.4, `<text:soft-page-break>` 5.6, `<text:table-index>` 8.5, `<text:table-of-content>` 8.3 and `<text:user-index>` 8.7.

## 5.5.6 <text:format-change>

The `<text:format-change>` element represents any change in formatting attributes. The region where the change took place is marked by `<text:change-start>`, `<text:change-end>` or `<text:change>` elements.

**Note:** This element does not contain formatting changes that have taken place.

The `<text:format-change>` element is usable within the following element: `<text:changed-region>` 5.5.3.

The `<text:format-change>` element has no attributes.

The `<text:format-change>` element has the following child element: `<office:change-info>` 5.5.7.

## 5.5.7 <office:change-info>

The `<office:change-info>` element represents who made a change and when. It may also contain a comment (one or more `<text:p>` elements) on the change.

The `<office:change-info>` element is usable within the following elements:

`<table:cell-content-change>` 9.9.17, `<table:deletion>` 9.9.9, `<table:insertion>` 9.9.3, `<table:movement>` 9.9.13, `<text:deletion>` 5.5.5, `<text:format-change>` 5.5.6 and `<text:insertion>` 5.5.4.

The `<office:change-info>` element has no attributes.

The `<office:change-info>` element has the following child elements: `<dc:creator>` 4.3.2.7, `<dc:date>` 4.3.2.10 and `<text:p>` 5.1.3.

## 5.5.8 Change Marks

### 5.5.8.1 General

The change mark elements mark start, end and positions in a text where a change has occurred.

These elements use the `text:change-id` attribute to link to `<text:changed-region>` elements that contain the change information.

### 5.5.8.2 <text:change-start>

The `<text:change-start>` element marks the start of a region with content where text has been inserted or the format has been changed.

The `<text:change-start>` element is usable within the following elements: `<draw:text-box>` 10.4.3, `<office:text>` 3.4, `<style:footer>` 16.12, `<style:footer-first>` 16.13, `<style:footer-left>` 16.15, `<style:header>` 16.10, `<style:header-first>` 16.11, `<style:header-left>` 16.14, `<table:covered-table-cell>` 9.1.5, `<table:table-cell>` 9.1.4, `<text:a>` 6.1.8, `<text:deletion>` 5.5.5, `<text:h>` 5.1.2, `<text:index-body>` 8.2.2, `<text:index-title>` 8.2.3, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:note-body>` 6.3.4, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2, `<text:section>` 5.4 and `<text:span>` 6.1.7.

The `<text:change-start>` element has the following attribute: `text:change-id` 19.771.

The `<text:change-start>` element has no child elements.

### 5.5.8.3 <text:change-end>

The `<text:change-end>` element marks the end of a region with content where text has been inserted or the format has been changed.

The `<text:change-end>` element is usable within the following elements: `<draw:text-box> 10.4.3, <office:text> 3.4, <style:footer> 16.12, <style:footer-first> 16.13, <style:footer-left> 16.15, <style:header> 16.10, <style:header-first> 16.11, <style:header-left> 16.14, <table:covered-table-cell> 9.1.5, <table:table-cell> 9.1.4, <text:a> 6.1.8, <text:deletion> 5.5.5, <text:h> 5.1.2, <text:index-body> 8.2.2, <text:index-title> 8.2.3, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:note-body> 6.3.4, <text:p> 5.1.3, <text:ruby-base> 6.4.2, <text:section> 5.4 and <text:span> 6.1.7.`

The `<text:change-end>` element has the following attribute: `text:change-id 19.771`.

The `<text:change-end>` element has no child elements.

### 5.5.8.4 <text:change>

The `<text:change>` element marks a position in an empty region where text has been deleted.

The `<text:change>` element is usable within the following elements: `<draw:text-box> 10.4.3, <office:text> 3.4, <style:footer> 16.12, <style:footer-first> 16.13, <style:footer-left> 16.15, <style:header> 16.10, <style:header-first> 16.11, <style:header-left> 16.14, <table:covered-table-cell> 9.1.5, <table:table-cell> 9.1.4, <text:a> 6.1.8, <text:deletion> 5.5.5, <text:h> 5.1.2, <text:index-body> 8.2.2, <text:index-title> 8.2.3, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:note-body> 6.3.4, <text:p> 5.1.3, <text:ruby-base> 6.4.2, <text:section> 5.4 and <text:span> 6.1.7.`

The `<text:change>` element has the following attribute: `text:change-id 19.771`.

The `<text:change>` element has no child elements.

## 5.6 <text:soft-page-break>

The `<text:soft-page-break>` element represents a soft page break within or between paragraph elements.

As a child element of a `<table:table>` element it represents a soft page break between two table rows. It may appear in front of a `<table:table-row>` element.

The `<text:soft-page-break>` element is usable within the following elements: `<draw:text-box> 10.4.3, <office:text> 3.4, <table:covered-table-cell> 9.1.5, <table:table> 9.1.2, <table:table-cell> 9.1.4, <table:table-header-rows> 9.1.7, <table:table-row-group> 9.1.9, <table:table-rows> 9.1.8, <text:a> 6.1.8, <text:deletion> 5.5.5, <text:h> 5.1.2, <text:index-body> 8.2.2, <text:index-title> 8.2.3, <text:list-header> 5.3.3, <text:list-item> 5.3.4, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:note-body> 6.3.4, <text:p> 5.1.3, <text:ruby-base> 6.4.2, <text:section> 5.4 and <text:span> 6.1.7.`

The `<text:soft-page-break>` element has no attributes.

The `<text:soft-page-break>` element has no child elements.

## 5.7 Document Declarations

The following declaration elements appear before the main content element of a document:

- `<text:variable-decl>` 7.4.3 – declarations for variable fields.

- <text:user-field-decl> 7.4.8 – declarations for user-defined fields.
- <text:sequence-decl> 7.4.12 – declarations for sequence fields.
- <text:dde-connection> 7.7.12 – declarations for DDE fields and DDE sections.
- <text:alphabetical-index-auto-mark-file> 8.8.3 – declaration for generation of alphabetical indexes.

---

# 6 Paragraph Elements Content

## 6.1 Basic Text Content

### 6.1.1 General

The paragraph elements `<text:p>` and `<text:h>` and their descendant elements contain the text content of any document. The *character content* of a paragraph consists of the character data of the paragraph element and the character data of its descendant elements concatenated in document order, with the following exceptions:

Character data contained in the following elements or their descendant elements are not included in the character content of a paragraph:

- `<text:note-body>`. 6.3.4.
- `<text:ruby-text>`. 7.5.19
- `<office:annotation>`. 14.1
- Drawing shape and frame elements defined in sections 10.3, 10.4, 10.5 and 10.6.

**Note:** This should significantly ease transformations into other formats, since transformations need not interpret most of the descendant elements of a paragraph element to obtain the character content of a paragraph.

The phrase "document order" when used with reference to an OpenDocument document instance is defined by §5.0 Data Model [XPath].

### 6.1.2 White Space Characters

Consumers shall collapse white space characters that occur in

- a `<text:p>` or `<text:h>` element (so-called paragraph elements), and
- in their descendant elements, if the OpenDocument schema permits `<text:s>` 6.1.3, `<text:tab>` 6.1.4 or `<text:line-break>` 6.1.4 as element content..

Collapsing white space characters is defined by the following algorithm:

- 1) Descendant `<text:ruby>` elements are replaced with their `<text:ruby-base>` child elements.
- 2) Descendant elements of the paragraph element which are not `<text:s>`, `<text:tab>` or `<text:line-break>` elements and for which the OpenDocument schema does not permit `<text:s>`, `<text:tab>` and `<text:line-break>` as child elements are removed from the paragraph element.
- 3) Descendant elements of the paragraph element for which the OpenDocument schema permits `<text:s>`, `<text:tab>` and `<text:line-break>` as child elements are replaced by their character data and `<text:s>`, `<text:tab>` and `<text:line-break>` element children.
- 4) The following [UNICODE] characters are replaced by a “ “ (U+0020, SPACE) character:
  - HORIZONTAL TABULATION (U+0009)
  - CARRIAGE RETURN (U+000D)
  - LINE FEED (U+000A)
- 5) Leading “ “ (U+0020, SPACE) characters at the start of the resulting text and trailing SPACE characters at the end of the resulting text are removed.

- 6) Sequences of “ “ (U+0020, SPACE) characters are replaced by a single “ “ (U+0020, SPACE) character.
  - 7) The remaining `<text:s>`, `<text:tab>` and `<text:line-break>` elements are interpreted as the [UNICODE] white space characters they represent.
- OpenDocument producers shall produce paragraph elements that, when consumed according to this algorithm, result in the expected amount of white space.
- OpenDocument consumers shall either process white space such that the result is equivalent to the result of the given algorithm, or implement a variation that increases interoperability with popular OpenDocument 1.2 producers. The variation replaces step 2 of the algorithm with steps 2a and 2b:

- 2a) Descendant elements of the paragraph element that are mark elements (`<text:change>` 5.5.8.4, `<text:change-end>` 5.5.8.3, `<text:change-start>` 5.5.8.2, `<text:bookmark>` 6.2.1.2, `<text:bookmark-end>` 6.2.1.4, `<text:bookmark-start>` 6.2.1.3, `<text:reference-mark>` 6.2.2.2, `<text:reference-mark-end>` 6.2.2.4, `<text:reference-mark-start>` 6.2.2.3, `<text:toc-mark>` 8.1.4, `<text:toc-mark-end>` 8.1.3, `<text:toc-mark-start>` 8.1.2, `<text:user-index-mark>` 8.1.7, `<text:user-index-mark-end>` 8.1.6, `<text:user-index-mark-start>` 8.1.5, `<text:alphabetical-index-mark>` 8.1.10, `<text:alphabetical-index-mark-end>` 8.1.9, `<text:alphabetical-index-mark-start>` 8.1.8) are removed from the paragraph element.
- 2b) Descendant elements of the paragraph element which are not `<text:s>`, `<text:tab>` or `<text:line-break>` elements and for which the OpenDocument schema does not permit `<text:s>`, `<text:tab>` and `<text:line-break>` as child elements are replaced with a hypothetical `<text:s text:c="0"/>` element.

**Note:** XML formatting software that does not implement the ODF white space rules may introduce or remove spaces.

### 6.1.3 `<text:s>`

The `<text:s>` element is used to represent the [UNICODE] character “ “ (U+0020, SPACE). This element shall be used to represent the second and all following “ “ (U+0020, SPACE) characters in a sequence of “ “ (U+0020, SPACE) characters.

**Note:** It is not an error if the character preceding the element is not a white space character, but it is good practice to use this element only for the second and all following “ “ (U+0020, SPACE) characters in a sequence.

The `<text:s>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:s>` element has the following attribute: `text:c` 19.768.

The `<text:s>` element has no child elements.

### 6.1.4 `<text:tab>`

The `<text:tab>` element represents the [UNICODE] tab character (HORIZONTAL TABULATION, U+0009). A `<text:tab>` element specifies that content immediately following it should align at the next tab stop.

The `<text:tab>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:tab>` element has the following attribute: `text:tab-ref` 19.883.

The `<text:tab>` element has no child elements.

### 6.1.5 `<text:line-break>`

The `<text:line-break>` element represents a line break.

The `<text:line-break>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:line-break>` element has no attributes.

The `<text:line-break>` element has no child elements.

### 6.1.6 Soft Hyphens, Hyphens, and Non-breaking Spaces

Soft hyphens, hyphens, and non-breaking blanks are represented by [UNICODE] characters.

*Table 8 - Unicode characters*

<i>The [UNICODE] character...</i>	<i>Represents...</i>
U-00AD SOFT HYPHEN	soft hyphen
U+2011 NON-BREAKING HY-PHEN	non-breaking hyphen
U+OOA0 NO BREAK SPACE	non-breaking space

### 6.1.7 `<text:span>`

The `<text:span>` element represents the application of a style to the character data of a portion of text. The content of this element is the text which uses that text style.

The `<text:span>` element can be nested.

White space characters contained in this element are collapsed.

The `<text:span>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:span>` element has the following attributes: `text:class-names` 19.775 and `text:style-name` 19.880.

The `<text:span>` element has the following child elements: `<dr3d:scene>` 10.5.2, `<draw:a>` 10.4.12, `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:connector>` 10.3.10, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:g>` 10.3.15, `<draw:line>` 10.3.3, `<draw:measure>` 10.3.12, `<draw:page-thumbnail>` 10.3.14, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2, `<draw:regular-polygon>` 10.3.6, `<office:annotation>` 14.1, `<office:annotation-end>` 14.2, `<presentation:date-time>` 10.9.4.5, `<presentation:footer>` 10.9.4.3, `<presentation:header>` 10.9.4.1, `<text:a>` 6.1.8, `<text:alphabetical-index-mark>` 8.1.10, `<text:alphabetical-index-mark-end>` 8.1.9, `<text:alphabetical-index-mark-start>` 8.1.8, `<text:author-initials>` 7.3.7.2, `<text:author-name>` 7.3.7.1, `<text:bibliography-mark>` 8.1.11, `<text:bookmark>` 6.2.1.2, `<text:bookmark-end>` 6.2.1.4, `<text:bookmark-ref>` 7.7.6, `<text:bookmark-start>` 6.2.1.3, `<text:change>` 5.5.8.4, `<text:change-end>` 5.5.8.3, `<text:change-start>` 5.5.8.2, `<text:chapter>` 7.3.8, `<text:character-count>` 7.5.18.5, `<text:conditional-text>` 7.7.3,

```

<text:creation-date> 7.5.3, <text:creation-time> 7.5.4, <text:creator>
7.5.17, <text:database-display> 7.6.3, <text:database-name> 7.6.7,
<text:database-next> 7.6.4, <text:database-row-number> 7.6.6,
<text:database-row-select> 7.6.5, <text:date> 7.3.2, <text:dde-connection>
7.7.12, <text:description> 7.5.5, <text:drop-down> 7.4.16, <text:editing-
cycles> 7.5.13, <text:editing-duration> 7.5.14, <text:execute-macro> 7.7.10,
<text:expression> 7.4.14, <text:file-name> 7.3.9, <text:hidden-paragraph>
7.7.11, <text:hidden-text> 7.7.4, <text:image-count> 7.5.18.7, <text:initial-
creator> 7.5.2, <text:keywords> 7.5.12, <text:line-break> 6.1.5,
<text:measure> 7.7.13, <text:meta> 6.1.9, <text:meta-field> 7.5.19,
<text:modification-date> 7.5.16, <text:modification-time> 7.5.15,
<text:note> 6.3.2, <text:note-ref> 7.7.7, <text:object-count> 7.5.18.8,
<text:page-continuation> 7.3.5, <text:page-count> 7.5.18.2, <text:page-
number> 7.3.4, <text:page-variable-get> 7.7.1.3, <text:page-variable-set>
7.7.1.2, <text:paragraph-count> 7.5.18.3, <text:placeholder> 7.7.2,
<text:print-date> 7.5.8, <text:print-time> 7.5.7, <text:printed-by> 7.5.9,
<text:reference-mark> 6.2.2.2, <text:reference-mark-end> 6.2.2.4,
<text:reference-mark-start> 6.2.2.3, <text:reference-ref> 7.7.5,
<text:ruby> 6.4, <text:s> 6.1.3, <text:script> 7.7.9, <text:sender-city>
7.3.6.13, <text:sender-company> 7.3.6.10, <text:sender-country> 7.3.6.15,
<text:sender-email> 7.3.6.7, <text:sender-fax> 7.3.6.9, <text:sender-
firstname> 7.3.6.2, <text:sender-initials> 7.3.6.4, <text:sender-lastname>
7.3.6.3, <text:sender-phone-private> 7.3.6.8, <text:sender-phone-work>
7.3.6.11, <text:sender-position> 7.3.6.6, <text:sender-postal-code> 7.3.6.14,
<text:sender-state-or-province> 7.3.6.16, <text:sender-street> 7.3.6.12,
<text:sender-title> 7.3.6.5, <text:sequence> 7.4.13, <text:sequence-ref>
7.7.8, <text:sheet-name> 7.3.11, <text:soft-page-break> 5.6, <text:span>
6.1.7, <text:subject> 7.5.11, <text:tab> 6.1.4, <text:table-count> 7.5.18.6,
<text:table-formula> 7.7.14, <text:template-name> 7.3.10, <text:text-
input> 7.4.15, <text:time> 7.3.3, <text:title> 7.5.10, <text:toc-mark> 8.1.4,
<text:toc-mark-end> 8.1.3, <text:toc-mark-start> 8.1.2, <text:user-
defined> 7.5.6, <text:user-field-get> 7.4.9, <text:user-field-input> 7.4.10,
<text:user-index-mark> 8.1.7, <text:user-index-mark-end> 8.1.6,
<text:user-index-mark-start> 8.1.5, <text:variable-get> 7.4.5,
<text:variable-input> 7.4.6, <text:variable-set> 7.4.4 and <text:word-
count> 7.5.18.4.

```

## 6.1.8 <text:a>

The <text:a> element represents a hyperlink.

The anchor of a hyperlink is composed of the character data contained in the <text:a> element and any of its descendant elements which constitute the character data of the paragraph element which contains the <text:a> element. 6.1.1

The <text:a> element is usable within the following elements: <text:h> 5.1.2, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:p> 5.1.3, <text:ruby-base> 6.4.2 and <text:span> 6.1.7.

The <text:a> element has the following attributes: office:name 19.380, office:target-frame-name 19.385, office:title 19.387, text:style-name 19.880, text:visited-style-name 19.907, xlink:actuate 19.915, xlink:href 19.916, xlink:show 19.917 and xlink:type 19.919.

The <text:a> element has the following child elements: <dr3d:scene> 10.5.2, <draw:a> 10.4.12, <draw:caption> 10.3.11, <draw:circle> 10.3.8, <draw:connector> 10.3.10, <draw:control> 10.3.13, <draw:custom-shape> 10.6.1, <draw:ellipse> 10.3.9,

```

<draw:frame> 10.4.2, <draw:g> 10.3.15, <draw:line> 10.3.3, <draw:measure>
10.3.12, <draw:page-thumbnail> 10.3.14, <draw:path> 10.3.7, <draw:polygon>
10.3.5, <draw:polyline> 10.3.4, <draw:rect> 10.3.2, <draw:regular-polygon>
10.3.6, <office:annotation> 14.1, <office:annotation-end> 14.2,
<office:event-listeners> 10.3.19, <presentation:date-time> 10.9.4.5,
<presentation:footer> 10.9.4.3, <presentation:header> 10.9.4.1,
<text:alphabetical-index-mark> 8.1.10, <text:alphabetical-index-mark-end> 8.1.9, <text:alphabetical-index-mark-start> 8.1.8, <text:author-initials> 7.3.7.2, <text:author-name> 7.3.7.1, <text:bibliography-mark>
8.1.11, <text:bookmark> 6.2.1.2, <text:bookmark-end> 6.2.1.4, <text:bookmark-ref> 7.7.6, <text:bookmark-start> 6.2.1.3, <text:change> 5.5.8.4,
<text:change-end> 5.5.8.3, <text:change-start> 5.5.8.2, <text:chapter> 7.3.8,
<text:character-count> 7.5.18.5, <text:conditional-text> 7.7.3,
<text:creation-date> 7.5.3, <text:creation-time> 7.5.4, <text:creator>
7.5.17, <text:database-display> 7.6.3, <text:database-name> 7.6.7,
<text:database-next> 7.6.4, <text:database-row-number> 7.6.6,
<text:database-row-select> 7.6.5, <text:date> 7.3.2, <text:dde-connection>
7.7.12, <text:description> 7.5.5, <text:drop-down> 7.4.16, <text:editing-cycles> 7.5.13, <text:editing-duration> 7.5.14, <text:execute-macro> 7.7.10,
<text:expression> 7.4.14, <text:file-name> 7.3.9, <text:hidden-paragraph>
7.7.11, <text:hidden-text> 7.7.4, <text:image-count> 7.5.18.7, <text:initial-creator> 7.5.2, <text:keywords> 7.5.12, <text:line-break> 6.1.5,
<text:measure> 7.7.13, <text:meta> 6.1.9, <text:meta-field> 7.5.19,
<text:modification-date> 7.5.16, <text:modification-time> 7.5.15,
<text:note> 6.3.2, <text:note-ref> 7.7.7, <text:object-count> 7.5.18.8,
<text:page-continuation> 7.3.5, <text:page-count> 7.5.18.2, <text:page-number> 7.3.4, <text:page-variable-get> 7.7.1.3, <text:page-variable-set> 7.7.1.2, <text:paragraph-count> 7.5.18.3, <text:placeholder> 7.7.2,
<text:print-date> 7.5.8, <text:print-time> 7.5.7, <text:printed-by> 7.5.9,
<text:reference-mark> 6.2.2.2, <text:reference-mark-end> 6.2.2.4,
<text:reference-mark-start> 6.2.2.3, <text:reference-ref> 7.7.5,
<text:ruby> 6.4, <text:s> 6.1.3, <text:script> 7.7.9, <text:sender-city>
7.3.6.13, <text:sender-company> 7.3.6.10, <text:sender-country> 7.3.6.15,
<text:sender-email> 7.3.6.7, <text:sender-fax> 7.3.6.9, <text:sender-firstname> 7.3.6.2, <text:sender-initials> 7.3.6.4, <text:sender-lastname>
7.3.6.3, <text:sender-phone-private> 7.3.6.8, <text:sender-phone-work>
7.3.6.11, <text:sender-position> 7.3.6.6, <text:sender-postal-code> 7.3.6.14,
<text:sender-state-or-province> 7.3.6.16, <text:sender-street> 7.3.6.12,
<text:sender-title> 7.3.6.5, <text:sequence> 7.4.13, <text:sequence-ref>
7.7.8, <text:sheet-name> 7.3.11, <text:soft-page-break> 5.6, <text:span>
6.1.7, <text:subject> 7.5.11, <text:tab> 6.1.4, <text:table-count> 7.5.18.6,
<text:table-formula> 7.7.14, <text:template-name> 7.3.10, <text:text-input> 7.4.15, <text:time> 7.3.3, <text:title> 7.5.10, <text:toc-mark> 8.1.4,
<text:toc-mark-end> 8.1.3, <text:toc-mark-start> 8.1.2, <text:user-defined> 7.5.6, <text:user-field-get> 7.4.9, <text:user-field-input> 7.4.10,
<text:user-index-mark> 8.1.7, <text:user-index-mark-end> 8.1.6,
<text:user-index-mark-start> 8.1.5, <text:variable-get> 7.4.5,
<text:variable-input> 7.4.6, <text:variable-set> 7.4.4 and <text:word-count> 7.5.18.4.

```

## 6.1.9 <text:meta>

The <text:meta> element represents portions of text that have in-content metadata attached. See 19.911.

The `<text:meta>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:meta>` element has the following attributes: `xhtml:about` 19.911, `xhtml:content` 19.912, `xhtml:datatype` 19.913, `xhtml:property` 19.914 and `xml:id` 19.920.

The `<text:meta>` element has the following child elements: `<dr3d:scene>` 10.5.2, `<draw:a>` 10.4.12, `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:connector>` 10.3.10, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:g>` 10.3.15, `<draw:line>` 10.3.3, `<draw:measure>` 10.3.12, `<draw:page-thumbnail>` 10.3.14, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2, `<draw:regular-polygon>` 10.3.6, `<office:annotation>` 14.1, `<office:annotation-end>` 14.2, `<presentation:date-time>` 10.9.4.5, `<presentation:footer>` 10.9.4.3, `<presentation:header>` 10.9.4.1, `<text:a>` 6.1.8, `<text:alphabetical-index-mark>` 8.1.10, `<text:alphabetical-index-mark-end>` 8.1.9, `<text:alphabetical-index-mark-start>` 8.1.8, `<text:author-initials>` 7.3.7.2, `<text:author-name>` 7.3.7.1, `<text:bibliography-mark>` 8.1.11, `<text:bookmark>` 6.2.1.2, `<text:bookmark-end>` 6.2.1.4, `<text:bookmark-ref>` 7.7.6, `<text:bookmark-start>` 6.2.1.3, `<text:change>` 5.5.8.4, `<text:change-end>` 5.5.8.3, `<text:change-start>` 5.5.8.2, `<text:chapter>` 7.3.8, `<text:character-count>` 7.5.18.5, `<text:conditional-text>` 7.7.3, `<text:creation-date>` 7.5.3, `<text:creation-time>` 7.5.4, `<text:creator>` 7.5.17, `<text:database-display>` 7.6.3, `<text:database-name>` 7.6.7, `<text:database-next>` 7.6.4, `<text:database-row-number>` 7.6.6, `<text:database-row-select>` 7.6.5, `<text:date>` 7.3.2, `<text:dde-connection>` 7.7.12, `<text:description>` 7.5.5, `<text:drop-down>` 7.4.16, `<text:editing-cycles>` 7.5.13, `<text:editing-duration>` 7.5.14, `<text:execute-macro>` 7.7.10, `<text:expression>` 7.4.14, `<text:file-name>` 7.3.9, `<text:hidden-paragraph>` 7.7.11, `<text:hidden-text>` 7.7.4, `<text:image-count>` 7.5.18.7, `<text:initial-creator>` 7.5.2, `<text:keywords>` 7.5.12, `<text:line-break>` 6.1.5, `<text:measure>` 7.7.13, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:modification-date>` 7.5.16, `<text:modification-time>` 7.5.15, `<text:note>` 6.3.2, `<text:note-ref>` 7.7.7, `<text:object-count>` 7.5.18.8, `<text:page-continuation>` 7.3.5, `<text:page-count>` 7.5.18.2, `<text:page-number>` 7.3.4, `<text:page-variable-get>` 7.7.1.3, `<text:page-variable-set>` 7.7.1.2, `<text:paragraph-count>` 7.5.18.3, `<text:placeholder>` 7.7.2, `<text:print-date>` 7.5.8, `<text:print-time>` 7.5.7, `<text:printed-by>` 7.5.9, `<text:reference-mark>` 6.2.2.2, `<text:reference-mark-end>` 6.2.2.4, `<text:reference-mark-start>` 6.2.2.3, `<text:reference-ref>` 7.7.5, `<text:ruby>` 6.4, `<text:s>` 6.1.3, `<text:script>` 7.7.9, `<text:sender-city>` 7.3.6.13, `<text:sender-company>` 7.3.6.10, `<text:sender-country>` 7.3.6.15, `<text:sender-email>` 7.3.6.7, `<text:sender-fax>` 7.3.6.9, `<text:sender-firstname>` 7.3.6.2, `<text:sender-initials>` 7.3.6.4, `<text:sender-lastname>` 7.3.6.3, `<text:sender-phone-private>` 7.3.6.8, `<text:sender-phone-work>` 7.3.6.11, `<text:sender-position>` 7.3.6.6, `<text:sender-postal-code>` 7.3.6.14, `<text:sender-state-or-province>` 7.3.6.16, `<text:sender-street>` 7.3.6.12, `<text:sender-title>` 7.3.6.5, `<text:sequence>` 7.4.13, `<text:sequence-ref>` 7.7.8, `<text:sheet-name>` 7.3.11, `<text:soft-page-break>` 5.6, `<text:span>` 6.1.7, `<text:subject>` 7.5.11, `<text:tab>` 6.1.4, `<text:table-count>` 7.5.18.6, `<text:table-formula>` 7.7.14, `<text:template-name>` 7.3.10, `<text:text-input>` 7.4.15, `<text:time>` 7.3.3, `<text:title>` 7.5.10, `<text:toc-mark>` 8.1.4, `<text:toc-mark-end>` 8.1.3, `<text:toc-mark-start>` 8.1.2, `<text:user-defined>` 7.5.6, `<text:user-field-get>` 7.4.9, `<text:user-field-input>` 7.4.10,

```
<text:user-index-mark> 8.1.7, <text:user-index-mark-end> 8.1.6,  
<text:user-index-mark-start> 8.1.5, <text:variable-get> 7.4.5,  
<text:variable-input> 7.4.6, <text:variable-set> 7.4.4 and <text:word-  
count> 7.5.18.4.
```

## 6.1.10 <text:number>

The `<text:number>` element contains the text of a formatted number which is present when a list style is applied to an element whose corresponding list level style specifies that the list label is a number. This text may be used by consumers that do not support the automatic generation of numbering but should be ignored by consumers that do support it.

This element has no attributes.

The `<text:number>` element is usable within the following elements: `<text:h>` 5.1.2, `<text:list-header>` 5.3.3, `<text:list-item>` 5.3.4 and `<text:numbered-paragraph>` 5.3.6.

The `<text:number>` element has no attributes.

The `<text:number>` element has no child elements.

The `<text:number>` element has content of data type `string` 18.2.

## 6.2 Bookmarks and References

### 6.2.1 Bookmarks

#### 6.2.1.1 General

There are two types of bookmarks:

- A point reference
  - A point reference marks a position in text and is represented by a single `<text:bookmark>` element.
- A bookmark reference
  - A bookmark reference marks a range of characters in text and is represented by two elements; a `<text:bookmark-start>` element to mark the start of the range and a following `<text:bookmark-end>` element to mark the end of the range.

Every reference is identified by the value of its `text:name` attribute, which shall be unique. In a bookmark reference, the start and end elements shall use the same attribute value.

Bookmark references support overlapping references.

#### 6.2.1.2 <text:bookmark>

The `<text:bookmark>` element marks a text position.

The `<text:bookmark>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:bookmark>` element has the following attributes: `text:name` 19.841 and `xml:id` 19.920.

The `<text:bookmark>` element has no child elements.

### 6.2.1.3 <text:bookmark-start>

The `<text:bookmark-start>` element marks the start position of a bookmark in a text range.

The `<text:bookmark-start>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:bookmark-start>` element has the following attributes: `text:name` 19.841, `xhtml:about` 19.911, `xhtml:content` 19.912, `xhtml:datatype` 19.913, `xhtml:property` 19.914 and `xml:id` 19.920.

The `<text:bookmark-start>` element has no child elements.

### 6.2.1.4 <text:bookmark-end>

The `<text:bookmark-end>` element marks the end position of a bookmark in a text range.

The `<text:bookmark-end>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:bookmark-end>` element has the following attribute: `text:name` 19.841.

The `<text:bookmark-end>` element has no child elements.

## 6.2.2 References

### 6.2.2.1 General

There are two types of reference marks:

- A point reference

A point reference marks a position in text and is represented by a single `<text:reference-mark>` element.

- A range reference

A range reference marks a range of characters in text and is represented by two elements; a `<text:reference-mark-start>` element to mark the start of the range and a following `<text:reference-mark-end>` element to mark the end of the range.

Every reference is identified by the value of its `text:name` attribute, which shall be unique. In a range reference, the start and end elements shall use the same attribute value.

Range references support overlapping references.

### 6.2.2.2 <text:reference-mark>

The `<text:reference-mark>` element represents a point reference.

The `<text:reference-mark>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:reference-mark>` element has the following attribute: `text:name` 19.841.

The `<text:reference-mark>` element has no child elements.

### 6.2.2.3 <text:reference-mark-start>

The `<text:reference-mark-start>` element represents the start of a range reference.

The `<text:reference-mark-start>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:reference-mark-start>` element has the following attribute: `text:name` 19.841.

The `<text:reference-mark-start>` element has no child elements.

### 6.2.2.4 <text:reference-mark-end>

The `<text:reference-mark-end>` element represents the end of a range reference.

The `<text:reference-mark-end>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:reference-mark-end>` element has the following attribute: `text:name` 19.841.

The `<text:reference-mark-end>` element has no child elements.

## 6.3 Notes

### 6.3.1 General

Notes consist of a `<text:note>` element which occurs in the text stream at the position to which the note is anchored. The numbering and rendering of notes is specified by a `<text:notes-configuration>` element, which occurs inside an `<office:styles>` element.

### 6.3.2 <text:note>

The `<text:note>` element represents text notes which are attached to a text position.

The `<text:note>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:note>` element has the following attributes: `text:id` 19.815 and `text:note-class` 19.843.

The `<text:note>` element has the following child elements: `<text:note-body>` 6.3.4 and `<text:note-citation>` 6.3.3.

### 6.3.3 <text:note-citation>

The `<text:note-citation>` element contains a citation, either as a formatted number or a string.

The `<text:note-citation>` element is usable within the following element: `<text:note>` 6.3.2.

The `<text:note-citation>` element has the following attribute: `text:label` 19.832.

The `<text:note-citation>` element has no child elements.

The `<text:note-citation>` element has character data content.

### 6.3.4 <text:note-body>

The <text:note-body> element contains the content of a note. It does not have any attributes.

**Note:** The schema allows for the inclusion of <text:note> elements as a descendant of a child of the <text:note-body> element. While this may be reasonable for note types, it is not reasonable for footnotes and endnotes. Conforming consumers need not support notes inside notes.

The <text:note-body> element is usable within the following element: <text:note> 6.3.2.

The <text:note-body> element has no attributes.

The <text:note-body> element has the following child elements: <dr3d:scene> 10.5.2, <draw:a> 10.4.12, <draw:caption> 10.3.11, <draw:circle> 10.3.8, <draw:connector> 10.3.10, <draw:control> 10.3.13, <draw:custom-shape> 10.6.1, <draw:ellipse> 10.3.9, <draw:frame> 10.4.2, <draw:g> 10.3.15, <draw:line> 10.3.3, <draw:measure> 10.3.12, <draw:page-thumbnail> 10.3.14, <draw:path> 10.3.7, <draw:polygon> 10.3.5, <draw:polyline> 10.3.4, <draw:rect> 10.3.2, <draw:regular-polygon> 10.3.6, <table:table> 9.1.2, <text:alphabetical-index> 8.8, <text:bibliography> 8.9, <text:change> 5.5.8.4, <text:change-end> 5.5.8.3, <text:change-start> 5.5.8.2, <text:h> 5.1.2, <text:illustration-index> 8.4, <text:list> 5.3.1, <text:numbered-paragraph> 5.3.6, <text:object-index> 8.6, <text:p> 5.1.3, <text:section> 5.4, <text:soft-page-break> 5.6, <text:table-index> 8.5, <text:table-of-content> 8.3 and <text:user-index> 8.7.

## 6.4 <text:ruby>

### 6.4.1 General

A ruby is additional text that is displayed above or below a base text. The purpose of a ruby is to annotate a base text or to provide information about its pronunciation.

The element's `text:style-name` attribute references a ruby style that specifies formatting attributes of the ruby.

The <text:ruby> element is usable within the following elements: <text:a> 6.1.8, <text:h> 5.1.2, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:p> 5.1.3, <text:ruby-base> 6.4.2 and <text:span> 6.1.7.

The <text:ruby> element has the following attribute: `text:style-name` 19.880.

The <text:ruby> element has the following child elements: <text:ruby-base> 6.4.2 and <text:ruby-text> 6.4.3.

### 6.4.2 <text:ruby-base>

The <text:ruby-base> element contains the text that is to be annotated. It contains any paragraph element content, such as text spans.

The <text:ruby-base> element is usable within the following element: <text:ruby> 6.4.

The <text:ruby-base> element has no attributes.

The <text:ruby-base> element has the following child elements: <dr3d:scene> 10.5.2, <draw:a> 10.4.12, <draw:caption> 10.3.11, <draw:circle> 10.3.8, <draw:connector> 10.3.10, <draw:control> 10.3.13, <draw:custom-shape> 10.6.1, <draw:ellipse> 10.3.9, <draw:frame> 10.4.2, <draw:g> 10.3.15, <draw:line> 10.3.3, <draw:measure> 10.3.12, <draw:page-thumbnail> 10.3.14, <draw:path>

```
10.3.7, <draw:polygon> 10.3.5, <draw:polyline> 10.3.4, <draw:rect> 10.3.2,
<draw:regular-polygon> 10.3.6, <office:annotation> 14.1,
<office:annotation-end> 14.2, <presentation:date-time> 10.9.4.5,
<presentation:footer> 10.9.4.3, <presentation:header> 10.9.4.1, <text:a>
6.1.8, <text:alphabetical-index-mark> 8.1.10, <text:alphabetical-index-
mark-end> 8.1.9, <text:alphabetical-index-mark-start> 8.1.8, <text:author-
initials> 7.3.7.2, <text:author-name> 7.3.7.1, <text:bibliography-mark>
8.1.11, <text:bookmark> 6.2.1.2, <text:bookmark-end> 6.2.1.4, <text:bookmark-
ref> 7.7.6, <text:bookmark-start> 6.2.1.3, <text:change> 5.5.8.4,
<text:change-end> 5.5.8.3, <text:change-start> 5.5.8.2, <text:chapter> 7.3.8,
<text:character-count> 7.5.18.5, <text:conditional-text> 7.7.3,
<text:creation-date> 7.5.3, <text:creation-time> 7.5.4, <text:creator>
7.5.17, <text:database-display> 7.6.3, <text:database-name> 7.6.7,
<text:database-next> 7.6.4, <text:database-row-number> 7.6.6,
<text:database-row-select> 7.6.5, <text:date> 7.3.2, <text:dde-connection>
7.7.12, <text:description> 7.5.5, <text:drop-down> 7.4.16, <text:editing-
cycles> 7.5.13, <text:editing-duration> 7.5.14, <text:execute-macro> 7.7.10,
<text:expression> 7.4.14, <text:file-name> 7.3.9, <text:hidden-paragraph>
7.7.11, <text:hidden-text> 7.7.4, <text:image-count> 7.5.18.7, <text:initial-
creator> 7.5.2, <text:keywords> 7.5.12, <text:line-break> 6.1.5,
<text:measure> 7.7.13, <text:meta> 6.1.9, <text:meta-field> 7.5.19,
<text:modification-date> 7.5.16, <text:modification-time> 7.5.15,
<text:note> 6.3.2, <text:note-ref> 7.7.7, <text:object-count> 7.5.18.8,
<text:page-continuation> 7.3.5, <text:page-count> 7.5.18.2, <text:page-
number> 7.3.4, <text:page-variable-get> 7.7.1.3, <text:page-variable-set>
7.7.1.2, <text:paragraph-count> 7.5.18.3, <text:placeholder> 7.7.2,
<text:print-date> 7.5.8, <text:print-time> 7.5.7, <text:printed-by> 7.5.9,
<text:reference-mark> 6.2.2.2, <text:reference-mark-end> 6.2.2.4,
<text:reference-mark-start> 6.2.2.3, <text:reference-ref> 7.7.5,
<text:ruby> 6.4, <text:s> 6.1.3, <text:script> 7.7.9, <text:sender-city>
7.3.6.13, <text:sender-company> 7.3.6.10, <text:sender-country> 7.3.6.15,
<text:sender-email> 7.3.6.7, <text:sender-fax> 7.3.6.9, <text:sender-
firstname> 7.3.6.2, <text:sender-initials> 7.3.6.4, <text:sender-lastname>
7.3.6.3, <text:sender-phone-private> 7.3.6.8, <text:sender-phone-work>
7.3.6.11, <text:sender-position> 7.3.6.6, <text:sender-postal-code> 7.3.6.14,
<text:sender-state-or-province> 7.3.6.16, <text:sender-street> 7.3.6.12,
<text:sender-title> 7.3.6.5, <text:sequence> 7.4.13, <text:sequence-ref>
7.7.8, <text:sheet-name> 7.3.11, <text:soft-page-break> 5.6, <text:span>
6.1.7, <text:subject> 7.5.11, <text:tab> 6.1.4, <text:table-count> 7.5.18.6,
<text:table-formula> 7.7.14, <text:template-name> 7.3.10, <text:text-
input> 7.4.15, <text:time> 7.3.3, <text:title> 7.5.10, <text:toc-mark> 8.1.4,
<text:toc-mark-end> 8.1.3, <text:toc-mark-start> 8.1.2, <text:user-
defined> 7.5.6, <text:user-field-get> 7.4.9, <text:user-field-input> 7.4.10,
<text:user-index-mark> 8.1.7, <text:user-index-mark-end> 8.1.6,
<text:user-index-mark-start> 8.1.5, <text:variable-get> 7.4.5,
<text:variable-input> 7.4.6, <text:variable-set> 7.4.4 and <text:word-
count> 7.5.18.4.
```

### 6.4.3 <text:ruby-text>

The `<text:ruby-text>` element contains an annotation text. It contains only plain text. The element's `text:style-name` attribute references a text style that specifies further formatting attributes used for the annotation text.

The `<text:ruby-text>` element is usable within the following element: `<text:ruby>` 6.4.

The `<text:ruby-text>` element has the following attribute: `text:style-name 19.880`.

The `<text:ruby-text>` element has no child elements.

The `<text:ruby-text>` element has character data content.

# 7 Text Fields

## 7.1 General

OpenDocument text documents or OpenDocument text content embedded in other types of documents may contain variable text elements called fields. Fields are commonly used for:

- Page numbers  
A page number field displays the number of the page on which it appears.
- Creation dates  
A creation date field displays the date on which a document was created.

## 7.2 Common Characteristics of Field Elements

Each field type is represented by a corresponding element type. The content of an element is a textual representation of the current field value as it would be displayed or printed by an implementation that supports that field.

The value of a field is stored in an attribute. Storage of the value enables recalculation of the presentation of a field.

**Note:** For string values, if a value is identical to a presentation, the value attribute can be omitted to avoid duplicate storage of information.

The presentation of information in a field is determined by the style applied to the field.

## 7.3 Document Fields

### 7.3.1 General

Document fields display information about the current document or about a specific part of the current document.

### 7.3.2 <text:date>

The `<text:date>` element displays a date. By default this is the current date. The date can be adjusted to display a date other than the current date.

The `<text:date>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:date>` element has the following attributes: `style:data-style-name` 19.473, `text:date-adjust` 19.798, `text:date-value` 19.799 and `text:fixed` 19.809.

The `<text:date>` element has no child elements.

The `<text:date>` element has character data content.

### 7.3.3 <text:time>

The `<text:time>` element displays a time. By default this is the current time.

The `<text:time>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:time>` element has the following attributes: `style:data-style-name 19.473`, `text:fixed 19.809`, `text:time-adjust 19.886` and `text:time-value 19.887`.

The `<text:time>` element has no child elements.

The `<text:time>` element has character data content.

### 7.3.4 `<text:page-number>`

The `<text:page-number>` element displays the current page number.

If a number style is not specified for page numbers, the default is the default number style of the current page style.

**Note:** The `<text:page-count>` element displays the total number of pages in a document. 7.5.18.2

The `<text:page-number>` element is usable within the following elements: `<text:a> 6.1.8`, `<text:h> 5.1.2`, `<text:meta> 6.1.9`, `<text:meta-field> 7.5.19`, `<text:p> 5.1.3`, `<text:ruby-base> 6.4.2` and `<text:span> 6.1.7`.

The `<text:page-number>` element has the following attributes: `style:num-format 19.504`, `style:num-letter-sync 19.505`, `text:fixed 19.809`, `text:page-adjust 19.851` and `text:select-page 19.869`.

The `<text:page-number>` element has no child elements.

The `<text:page-number>` element has character data content.

### 7.3.5 `<text:page-continuation>`

The `<text:page-continuation>` element contains a text notice that is displayed or printed at the bottom of a page to indicate that the preceding text continues on the next page.

The `<text:page-continuation>` element is usable within the following elements: `<text:a> 6.1.8`, `<text:h> 5.1.2`, `<text:meta> 6.1.9`, `<text:meta-field> 7.5.19`, `<text:p> 5.1.3`, `<text:ruby-base> 6.4.2` and `<text:span> 6.1.7`.

The `<text:page-continuation>` element has the following attributes: `text:select-page 19.869` and `text:string-value 19.877`.

The `<text:page-continuation>` element has no child elements.

The `<text:page-continuation>` element has character data content.

## 7.3.6 Subsequent Author Fields

### 7.3.6.1 General

Subsequent author fields capture information about second and following authors for a document.

**Note:** In OpenDocument 1.0/1.1, these fields were referenced as “sender” fields and that terminology still appears in the names of these elements.

If a subsequent author field is marked fixed using the `text:fixed 19.809` attribute, the original information in that field is preserved. Otherwise, the information is updated each time the file is edited.

### 7.3.6.2 `<text:sender-firstname>`

The `<text:sender-firstname>` element represents the first name of a subsequent author.

The `<text:sender-ffirstname>` element is usable within the following elements:  
`<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19,  
`<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:sender-ffirstname>` element has the following attribute: `text:fixed 19.809`.

The `<text:sender-ffirstname>` element has no child elements.

The `<text:sender-ffirstname>` element has character data content.

### 7.3.6.3 `<text:sender-lastname>`

The `<text:sender-lastname>` element represents the last name of a subsequent author.

The `<text:sender-lastname>` element is usable within the following elements:  
`<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19,  
`<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:sender-lastname>` element has the following attribute: `text:fixed 19.809`.

The `<text:sender-lastname>` element has no child elements.

The `<text:sender-lastname>` element has character data content.

### 7.3.6.4 `<text:sender-initials>`

The `<text:sender-initials>` element represents the initials of a subsequent author.

The `<text:sender-initials>` element is usable within the following elements:  
`<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19,  
`<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:sender-initials>` element has the following attribute: `text:fixed 19.809`.

The `<text:sender-initials>` element has no child elements.

The `<text:sender-initials>` element has character data content.

### 7.3.6.5 `<text:sender-title>`

The `<text:sender-title>` element represents the title of a subsequent author.

The `<text:sender-title>` element is usable within the following elements: `<text:a>`  
6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>`  
5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:sender-title>` element has the following attribute: `text:fixed 19.809`.

The `<text:sender-title>` element has no child elements.

The `<text:sender-title>` element has character data content.

### 7.3.6.6 `<text:sender-position>`

The `<text:sender-position>` element represents the position of a subsequent author.

The `<text:sender-position>` element is usable within the following elements:  
`<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19,  
`<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:sender-position>` element has the following attribute: `text:fixed 19.809`.

The `<text:sender-position>` element has no child elements.

The `<text:sender-position>` element has character data content.

### 7.3.6.7 <text:sender-email>

The <text:sender-email> element represents the email address of a subsequent author.

The <text:sender-email> element is usable within the following elements: <text:a> 6.1.8, <text:h> 5.1.2, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:p> 5.1.3, <text:ruby-base> 6.4.2 and <text:span> 6.1.7.

The <text:sender-email> element has the following attribute: text:fixed 19.809.

The <text:sender-email> element has no child elements.

The <text:sender-email> element has character data content.

### 7.3.6.8 <text:sender-phone-private>

The <text:sender-phone-private> element represents the private telephone number of a subsequent author.

The <text:sender-phone-private> element is usable within the following elements: <text:a> 6.1.8, <text:h> 5.1.2, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:p> 5.1.3, <text:ruby-base> 6.4.2 and <text:span> 6.1.7.

The <text:sender-phone-private> element has the following attribute: text:fixed 19.809.

The <text:sender-phone-private> element has no child elements.

The <text:sender-phone-private> element has character data content.

### 7.3.6.9 <text:sender-fax>

The <text:sender-fax> element represents the facsimile number of a subsequent author.

The <text:sender-fax> element is usable within the following elements: <text:a> 6.1.8, <text:h> 5.1.2, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:p> 5.1.3, <text:ruby-base> 6.4.2 and <text:span> 6.1.7.

The <text:sender-fax> element has the following attribute: text:fixed 19.809.

The <text:sender-fax> element has no child elements.

The <text:sender-fax> element has character data content.

### 7.3.6.10 <text:sender-company>

The <text:sender-company> element represents the name of the company that employs a subsequent author.

The <text:sender-company> element is usable within the following elements: <text:a> 6.1.8, <text:h> 5.1.2, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:p> 5.1.3, <text:ruby-base> 6.4.2 and <text:span> 6.1.7.

The <text:sender-company> element has the following attribute: text:fixed 19.809.

The <text:sender-company> element has no child elements.

The <text:sender-company> element has character data content.

### 7.3.6.11 <text:sender-phone-work>

The <text:sender-phone-work> element represents the office telephone number of a subsequent author.

The `<text:sender-phone-work>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:sender-phone-work>` element has the following attribute: `text:fixed 19.809`.

The `<text:sender-phone-work>` element has no child elements.

The `<text:sender-phone-work>` element has character data content.

### 7.3.6.12 `<text:sender-street>`

The `<text:sender-street>` element represents the street address of a subsequent author.

The `<text:sender-street>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:sender-street>` element has the following attribute: `text:fixed 19.809`.

The `<text:sender-street>` element has no child elements.

The `<text:sender-street>` element has character data content.

### 7.3.6.13 `<text:sender-city>`

The `<text:sender-city>` element represents the city name of the address of a subsequent author.

The `<text:sender-city>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:sender-city>` element has the following attribute: `text:fixed 19.809`.

The `<text:sender-city>` element has no child elements.

The `<text:sender-city>` element has character data content.

### 7.3.6.14 `<text:sender-postal-code>`

The `<text:sender-postal-code>` element represents the postal code of the address of a subsequent author.

The `<text:sender-postal-code>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:sender-postal-code>` element has the following attribute: `text:fixed 19.809`.

The `<text:sender-postal-code>` element has no child elements.

The `<text:sender-postal-code>` element has character data content.

### 7.3.6.15 `<text:sender-country>`

The `<text:sender-country>` element represents the country name of the address of a subsequent author.

The `<text:sender-country>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:sender-country>` element has the following attribute: `text:fixed 19.809`.  
The `<text:sender-country>` element has no child elements.  
The `<text:sender-country>` element has character data content.

### 7.3.6.16 `<text:sender-state-or-province>`

The `<text:sender-state-or-province>` element represents the state or province name of the address of a subsequent author, if applicable.

The `<text:sender-state-or-province>` element is usable within the following elements: `<text:a> 6.1.8, <text:h> 5.1.2, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:p> 5.1.3, <text:ruby-base> 6.4.2 and <text:span> 6.1.7`.  
The `<text:sender-state-or-province>` element has the following attribute: `text:fixed 19.809`.  
The `<text:sender-state-or-province>` element has no child elements.  
The `<text:sender-state-or-province>` element has character data content.

## 7.3.7 Author Fields

### 7.3.7.1 `<text:author-name>`

The `<text:author-name>` element represents the full name of the author of a document.

The `<text:author-name>` element is usable within the following elements: `<text:a> 6.1.8, <text:h> 5.1.2, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:p> 5.1.3, <text:ruby-base> 6.4.2 and <text:span> 6.1.7`.  
The `<text:author-name>` element has the following attribute: `text:fixed 19.809`.  
The `<text:author-name>` element has no child elements.  
The `<text:author-name>` element has character data content.

### 7.3.7.2 `<text:author-initials>`

The `<text:author-initials>` element represents the initials of the author of a document.

The `<text:author-initials>` element is usable within the following elements: `<text:a> 6.1.8, <text:h> 5.1.2, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:p> 5.1.3, <text:ruby-base> 6.4.2 and <text:span> 6.1.7`.  
The `<text:author-initials>` element has the following attribute: `text:fixed 19.809`.  
The `<text:author-initials>` element has no child elements.  
The `<text:author-initials>` element has character data content.

## 7.3.8 `<text:chapter>`

The `<text:chapter>` element represents a field that displays one of the following:

- The name of the current chapter
- The number of the current chapter
- Both the name and number of the current chapter

If this element is placed inside a header or footer, it displays the current chapter name or number on every page.

The `<text:chapter>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:chapter>` element has the following attributes: `text:display` 19.802 and `text:outline-level` 19.850.

The `<text:chapter>` element has no child elements.

The `<text:chapter>` element has character data content.

### 7.3.9 `<text:file-name>`

The `<text:file-name>` element represents a field that displays the name of a file that is being edited.

The `<text:file-name>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:file-name>` element has the following attributes: `text:display` 19.802 and `text:fixed` 19.809.

The `<text:file-name>` element has no child elements.

The `<text:file-name>` element has character data content.

### 7.3.10 `<text:template-name>`

The `<text:template-name>` element represents a field that displays information about the document template in use.

The `<text:template-name>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:template-name>` element has the following attribute: `text:display` 19.802.

The `<text:template-name>` element has no child elements.

The `<text:template-name>` element has character data content.

### 7.3.11 `<text:sheet-name>`

The `<text:sheet-name>` element represents the name of the current sheet..

The `<text:sheet-name>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:sheet-name>` element has no attributes.

The `<text:sheet-name>` element has no child elements.

The `<text:sheet-name>` element has character data content.

## 7.4 Variable Fields

### 7.4.1 General

OpenDocument documents can contain variables, which are processed or displayed using variable fields. In an OpenDocument file, variable declarations shall precede in document order any use of those variable declarations.

There are three types of variables:

- **Simple variables**

Simple variables, or variables, can take different values at different positions in a document. Simple variables can be set using either setter or input fields. Setter fields contain an expression, which is used to compute the new value of the variable. Input fields prompt the user for the new value.

**Note:** Simple variables can be used to display different text in recurring elements, such as headers or footers.

- **User variables**

If a user variable is set anywhere within the document, all fields in a document that display that user variable have the same value. The value of the user variable can only be set after the variable is declared.

**Note:** In a consumer's user interface, a user variable can be set at any occurrence of a user field, or by using user variable input fields.

- **Sequence variables**

Sequence variables are used to number items in an OpenDocument text document

**Note:** Sequence variables can be used to number images or tables in a text.

The value of a sequence variable is initialized to 0 (zero) by its declaration.

Variables shall be declared before they can be used. The OpenDocument elements for declaring variables are described in sections 7.4.3, 7.4.8 and 7.4.12.

The display of variables is determined by styles which are specified by the element that defines the variable.

## 7.4.2 <text:variable-decls>

The `<text:variable-decls>` element is a container for `<text:variable-decl>` elements.

The `<text:variable-decls>` element is usable within the following elements: `<office:chart> 3.8`, `<office:drawing> 3.5`, `<office:presentation> 3.6`, `<office:spreadsheet> 3.7`, `<office:text> 3.4`, `<style:footer> 16.12`, `<style:footer-first> 16.13`, `<style:footer-left> 16.15`, `<style:header> 16.10`, `<style:header-first> 16.11` and `<style:header-left> 16.14`.

The `<text:variable-decls>` element has no attributes.

The `<text:variable-decls>` element has the following child element: `<text:variable-decl> 7.4.3`.

## 7.4.3 <text:variable-decl>

The `<text:variable-decl>` element is used to declare simple variables. This element specifies the name and the value type of a variable.

The `<text:variable-decl>` element is usable within the following element: `<text:variable-decls> 7.4.2`.

The `<text:variable-decl>` element has the following attributes: `office:value-type 19.389` and `text:name 19.841`.

The `<text:variable-decl>` element has no child elements.

#### 7.4.4 <text:variable-set>

The `<text:variable-set>` element represents a simple variable.

The `<text:variable-set>` element is usable within the following elements: `<text:a> 6.1.8, <text:h> 5.1.2, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:p> 5.1.3, <text:ruby-base> 6.4.2 and <text:span> 6.1.7.`

The `<text:variable-set>` element has the following attributes: `office:boolean-value 19.371, office:currency 19.373, office:date-value 19.374, office:string-value 19.383, office:time-value 19.386, office:value 19.388, office:value-type 19.389, style:data-style-name 19.473, text:display 19.802, text:formula 19.811 and text:name 19.841.`

The `<text:variable-set>` element has no child elements.

The `<text:variable-set>` element has character data content.

#### 7.4.5 <text:variable-get>

The `<text:variable-get>` element sets the display of the value of a simple variable. The value of this element is the value of the last preceding `<text:variable-set>` element with an identical value for its `text:name` attribute.

If there is no preceding `<text:variable-set>` element with an identical value for its `text:name` attribute the value of a `<text:variable-get>` field is undefined.

The `<text:variable-get>` element is usable within the following elements: `<text:a> 6.1.8, <text:h> 5.1.2, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:p> 5.1.3, <text:ruby-base> 6.4.2 and <text:span> 6.1.7.`

The `<text:variable-get>` element has the following attributes: `style:data-style-name 19.473, text:display 19.802 and text:name 19.841.`

The `<text:variable-get>` element has no child elements.

The `<text:variable-get>` element has character data content.

#### 7.4.6 <text:variable-input>

The `<text:variable-input>` element represents user supplied variable values. This element can be empty if its `text:display` attribute is set to `none`.

The `<text:variable-input>` element is usable within the following elements: `<text:a> 6.1.8, <text:h> 5.1.2, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:p> 5.1.3, <text:ruby-base> 6.4.2 and <text:span> 6.1.7.`

The `<text:variable-input>` element has the following attributes: `office:value-type 19.389, style:data-style-name 19.473, text:description 19.801, text:display 19.802 and text:name 19.841.`

The `<text:variable-input>` element has no child elements.

The `<text:variable-input>` element has character data content.

#### 7.4.7 <text:user-field-decls>

The `<text:user-field-decls>` element is a container for `<text:user-field-decl>` elements.

The `<text:user-field-decls>` element is usable within the following elements: `<office:chart> 3.8, <office:drawing> 3.5, <office:presentation> 3.6, <office:spreadsheet> 3.7, <office:text> 3.4, <style:footer> 16.12,`

<style:footer-first> 16.13, <style:footer-left> 16.15, <style:header> 16.10, <style:header-first> 16.11 and <style:header-left> 16.14.

The <text:user-field-decls> element has no attributes.

The <text:user-field-decls> element has the following child element: <text:user-field-decl> 7.4.8.

#### 7.4.8 <text:user-field-decl>

The <text:user-field-decl> element is used to specify user variable values that are displayed using the <text:user-field-get> element.

The <text:user-field-decl> element is usable within the following element: <text:user-field-decls> 7.4.7.

The <text:user-field-decl> element has the following attributes: office:boolean-value 19.371, office:currency 19.373, office:date-value 19.374, office:string-value 19.383, office:time-value 19.386, office:value 19.388, office:value-type 19.389, text:formula 19.811 and text:name 19.841.

The <text:user-field-decl> element has no child elements.

#### 7.4.9 <text:user-field-get>

The <text:user-field-get> element displays the content of a user variable. The value of this element is the value of a <text:user-field-decl> element with a text:name attribute value that matches the text:name attribute of this element.

The <text:user-field-get> element is usable within the following elements: <text:a> 6.1.8, <text:h> 5.1.2, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:p> 5.1.3, <text:ruby-base> 6.4.2 and <text:span> 6.1.7.

The <text:user-field-get> element has the following attributes: style:data-style-name 19.473, text:display 19.802 and text:name 19.841.

The <text:user-field-get> element has no child elements.

The <text:user-field-get> element has character data content.

#### 7.4.10 <text:user-field-input>

The <text:user-field-input> element is an alternative method of displaying user variables. The value of this element is the value of a <text:user-field-decl> element with a text:name attribute value that matches the text:name attribute of this element.

This element can be empty if the text:display attribute is set to none.

The <text:user-field-input> element is usable within the following elements: <text:a> 6.1.8, <text:h> 5.1.2, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:p> 5.1.3, <text:ruby-base> 6.4.2 and <text:span> 6.1.7.

The <text:user-field-input> element has the following attributes: style:data-style-name 19.473, text:description 19.801 and text:name 19.841.

The <text:user-field-input> element has no child elements.

The <text:user-field-input> element has character data content.

#### 7.4.11 <text:sequence-decls>

The <text:sequence-decls> element is a container for <text:sequence-decl> elements.

The `<text:sequence-decls>` element is usable within the following elements: `<office:chart>` 3.8, `<office:drawing>` 3.5, `<office:presentation>` 3.6, `<office:spreadsheet>` 3.7, `<office:text>` 3.4, `<style:footer>` 16.12, `<style:footer-first>` 16.13, `<style:footer-left>` 16.15, `<style:header>` 16.10, `<style:header-first>` 16.11 and `<style:header-left>` 16.14.

The `<text:sequence-decls>` element has no attributes.

The `<text:sequence-decls>` element has the following child element: `<text:sequence-decl>` 7.4.12.

#### 7.4.12 `<text:sequence-decl>`

The `<text:sequence-decl>` element is used to declare sequence variables used to number items within an OpenDocument text document.

**Note:** Sequence variables are commonly used for sequential numbering. However, expression formulas can be included in sequence fields to support more advanced sequences.

The `<text:sequence-decl>` element is usable within the following element: `<text:sequence-decls>` 7.4.11.

The `<text:sequence-decl>` element has the following attributes: `text:display-outline-level` 19.804, `text:name` 19.841 and `text:separation-character` 19.870.

The `<text:sequence-decl>` element has no child elements.

#### 7.4.13 `<text:sequence>`

A `<text:sequence>` element specifies a sequence field.

A sequence field changes the value of a sequence variable by a specified formula and displays the changed value of the sequence variable in a specified format.

**Note:** Sequence variables and fields are used to number repeating document elements such as pages, tables or images. In such cases, the formula could be: "sequenceVariableName + 1." A sequence variable can be set to a specific value or not changed at all.

The `<text:sequence>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:sequence>` element has the following attributes: `style:num-format` 19.504, `style:num-letter-sync` 19.505, `text:formula` 19.811, `text:name` 19.841 and `text:ref-name` 19.859.

The `<text:sequence>` element has no child elements.

The `<text:sequence>` element has character data content.

#### 7.4.14 `<text:expression>`

The `<text:expression>` element represents expressions that are evaluated and the resulting value displayed.

The `<text:expression>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:expression>` element has the following attributes: `office:boolean-value` 19.371, `office:currency` 19.373, `office:date-value` 19.374, `office:string-value` 19.383, `office:time-value` 19.386, `office:value` 19.388, `office:value-type` 19.389, `style:data-style-name` 19.473, `text:display` 19.802 and `text:formula` 19.811.

The `<text:expression>` element has no child elements.

The `<text:expression>` element has character data content.

#### 7.4.15 `<text:text-input>`

The `<text:text-input>` element is a text input field. A text input field is used in a user interface to prompt a user to input text. This text is stored and displayed as the content of this element until it is replaced by a user.

The `<text:text-input>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:text-input>` element has the following attribute: `text:description` 19.801.

The `<text:text-input>` element has no child elements.

The `<text:text-input>` element has character data content.

#### 7.4.16 `<text:drop-down>`

The `<text:drop-down>` element is a drop-down list input field. A drop-down list input field is used in a user interface to prompt a user to select a value from a list of options. The selected value is stored and displayed as the content of this element until it is changed by a user.

The `<text:drop-down>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:drop-down>` element has the following attribute: `text:name` 19.841.

The `<text:drop-down>` element has the following child element: `<text:label>` 7.4.17.

The `<text:drop-down>` element has character data content.

#### 7.4.17 `<text:label>`

The `<text:label>` element defines an entry in a drop-down list.

The `<text:label>` element is usable within the following element: `<text:drop-down>` 7.4.16.

The `<text:label>` element has the following attributes: `text:current-selected` 19.790 and `text:value` 19.906.

The `<text:label>` element has no child elements.

### 7.5 Metadata Fields

#### 7.5.1 General

Metadata field elements display meta information about a document.

All metadata field elements can be marked as fixed using the `text:fixed` attribute. 19.809

## 7.5.2 <text:initial-creator>

The `<text:initial-creator>` element displays the value contained in the `<meta:initial-creator>` element of a document. 4.3.2.6

The `<text:initial-creator>` element is usable within the following elements:

`<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:initial-creator>` element has the following attribute: `text:fixed` 19.809.

The `<text:initial-creator>` element has no child elements.

The `<text:initial-creator>` element has character data content.

## 7.5.3 <text:creation-date>

The `<text:creation-date>` element displays the value contained in the `<meta:creation-date>` element of a document. 4.3.2.9

The `<text:creation-date>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:creation-date>` element has the following attributes: `style:data-style-name` 19.473, `text:date-value` 19.799 and `text:fixed` 19.809.

The `<text:creation-date>` element has no child elements.

The `<text:creation-date>` element has character data content.

## 7.5.4 <text:creation-time>

The `<text:creation-time>` element represents the time at which a document was created.

The `<text:creation-time>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:creation-time>` element has the following attributes: `style:data-style-name` 19.473, `text:fixed` 19.809 and `text:time-value` 19.887.

The `<text:creation-time>` element has no child elements.

The `<text:creation-time>` element has character data content.

## 7.5.5 <text:description>

The `<text:description>` element displays the content contained in the `<dc:description>` element of a document. 4.3.2.3

The `<text:description>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:description>` element has the following attribute: `text:fixed` 19.809.

The `<text:description>` element has no child elements.

The `<text:description>` element has character data content.

## 7.5.6 <text:user-defined>

The `<text:user-defined>` element displays the value contained in a `<meta:user-defined>` element of a document. 4.3.3

**Note:** The link between a `<text:user-defined>` element and a `<meta:user-defined>` element is established by the `text:name` and `meta:name` attributes, respectively, containing the same value.

The `<text:user-defined>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:user-defined>` element has the following attributes: `office:boolean-value` 19.371, `office:date-value` 19.374, `office:string-value` 19.383, `office:time-value` 19.386, `office:value` 19.388, `style:data-style-name` 19.473, `text:fixed` 19.809 and `text:name` 19.841.

The `<text:user-defined>` element has no child elements.

The `<text:user-defined>` element has character data content.

## 7.5.7 <text:print-time>

The `<text:print-time>` element represents the time at which a document was last printed.

The `<text:print-time>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:print-time>` element has the following attributes: `style:data-style-name` 19.473, `text:fixed` 19.809 and `text:time-value` 19.887.

The `<text:print-time>` element has no child elements.

The `<text:print-time>` element has character data content.

## 7.5.8 <text:print-date>

The `<text:print-date>` element displays the value contained in the `<meta:print-date>` element of a document. 4.3.2.11

The `<text:print-date>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:print-date>` element has the following attributes: `style:data-style-name` 19.473, `text:date-value` 19.799 and `text:fixed` 19.809.

The `<text:print-date>` element has no child elements.

The `<text:print-date>` element has character data content.

## 7.5.9 <text:printed-by>

The `<text:printed-by>` element displays the value contained in the `<meta:printed-by>` element of a document. 4.3.2.8

The `<text:printed-by>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:printed-by>` element has the following attribute: `text:fixed` 19.809.

The `<text:printed-by>` element has no child elements.

The `<text:printed-by>` element has character data content.

### 7.5.10 `<text:title>`

The `<text:title>` element displays the value contained in the `<dc:title>` element of a document. 4.3.2.2

The `<text:title>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:title>` element has the following attribute: `text:fixed 19.809`.

The `<text:title>` element has no child elements.

The `<text:title>` element has character data content.

### 7.5.11 `<text:subject>`

The `<text:subject>` element displays the values contained in the `<dc:subject>` element of a document. 4.3.2.4

The `<text:subject>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:subject>` element has the following attribute: `text:fixed 19.809`.

The `<text:subject>` element has no child elements.

The `<text:subject>` element has character data content.

### 7.5.12 `<text:keywords>`

The `<text:keywords>` element displays a list of keywords contained in the `<meta:keyword>` elements of a document.

The format of a list of keywords is implementation-dependent.

The `<text:keywords>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:keywords>` element has the following attribute: `text:fixed 19.809`.

The `<text:keywords>` element has no child elements.

The `<text:keywords>` element has character data content.

### 7.5.13 `<text:editing-cycles>`

The `<text:editing-cycles>` element displays the value contained in the `<meta:editing-cycles>` element of a document. 4.3.2.16

The `<text:editing-cycles>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:editing-cycles>` element has the following attribute: `text:fixed 19.809`.

The `<text:editing-cycles>` element has no child elements.

The `<text:editing-cycles>` element has character data content.

### 7.5.14 <text:editing-duration>

The `<text:editing-duration>` element displays the value contained in the `<meta:editing-duration>` element of a document.

The `<text:editing-duration>` element is usable within the following elements:

`<text:a> 6.1.8, <text:h> 5.1.2, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:p> 5.1.3, <text:ruby-base> 6.4.2 and <text:span> 6.1.7.`

The `<text:editing-duration>` element has the following attributes: `style:data-style-name 19.473, text:duration 19.805 and text:fixed 19.809.`

The `<text:editing-duration>` element has no child elements.

The `<text:editing-duration>` element has character data content.

### 7.5.15 <text:modification-time>

The `<text:modification-time>` element represents the time at which a document was last modified.

This element displays the time information portion of the value of the `<dc:date>` element.

The `<text:modification-time>` element is usable within the following elements:

`<text:a> 6.1.8, <text:h> 5.1.2, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:p> 5.1.3, <text:ruby-base> 6.4.2 and <text:span> 6.1.7.`

The `<text:modification-time>` element has the following attributes: `style:data-style-name 19.473, text:fixed 19.809 and text:time-value 19.887.`

The `<text:modification-time>` element has no child elements.

The `<text:modification-time>` element has character data content.

### 7.5.16 <text:modification-date>

The `<text:modification-date>` element displays the value contained in the `<dc:date>` element of a document. 4.3.2.10

The `<text:modification-date>` element is usable within the following elements:

`<text:a> 6.1.8, <text:h> 5.1.2, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:p> 5.1.3, <text:ruby-base> 6.4.2 and <text:span> 6.1.7.`

The `<text:modification-date>` element has the following attributes: `style:data-style-name 19.473, text:date-value 19.799 and text:fixed 19.809.`

The `<text:modification-date>` element has no child elements.

The `<text:modification-date>` element has character data content.

### 7.5.17 <text:creator>

The `<text:creator>` element displays the value contained in the `<dc:creator>` element of a document. 4.3.2.7

The `<text:creator>` element is usable within the following elements: `<text:a> 6.1.8, <text:h> 5.1.2, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:p> 5.1.3, <text:ruby-base> 6.4.2 and <text:span> 6.1.7.`

The `<text:creator>` element has the following attribute: `text:fixed 19.809.`

The `<text:creator>` element has no child elements.

The `<text:creator>` element has character data content.

## 7.5.18 Document Statistics Fields

### 7.5.18.1 General

The contents of `<text:word-count>` and `<text:character-count>` elements are locale-specific and are not defined by OpenDocument.

### 7.5.18.2 `<text:page-count>`

The `<text:page-count>` element displays the value of the `meta:page-count` attribute of the `<meta:document-statistic>` element of a document. 19.332

The `<text:page-count>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:page-count>` element has the following attributes: `style:num-format` 19.504 and `style:num-letter-sync` 19.505.

The `<text:page-count>` element has no child elements.

The `<text:page-count>` element has character data content.

### 7.5.18.3 `<text:paragraph-count>`

The `<text:paragraph-count>` element displays the value of the `meta:paragraph-count` attribute of the `<meta:document-statistic>` element of a document. 19.333

The `<text:paragraph-count>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:paragraph-count>` element has the following attributes: `style:num-format` 19.504 and `style:num-letter-sync` 19.505.

The `<text:paragraph-count>` element has no child elements.

The `<text:paragraph-count>` element has character data content.

### 7.5.18.4 `<text:word-count>`

The `<text:word-count>` element displays the value of the `meta:word-count` attribute of the `<meta:document-statistic>` element of a document. 19.339

The `<text:word-count>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:word-count>` element has the following attributes: `style:num-format` 19.504 and `style:num-letter-sync` 19.505.

The `<text:word-count>` element has no child elements.

The `<text:word-count>` element has character data content.

### 7.5.18.5 `<text:character-count>`

The `<text:character-count>` element displays the value of the `meta:character-count` attribute of the `<meta:document-statistic>` element of a document. 19.322

The `<text:character-count>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:character-count>` element has the following attributes: `style:num-format 19.504` and `style:num-letter-sync 19.505`.

The `<text:character-count>` element has no child elements.

The `<text:character-count>` element has character data content.

### 7.5.18.6 `<text:table-count>`

The `<text:table-count>` element displays the value of the `meta:table-count` attribute of the `<meta:document-statistic>` element of a document. 19.337

The `<text:table-count>` element is usable within the following elements: `<text:a> 6.1.8, <text:h> 5.1.2, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:p> 5.1.3, <text:ruby-base> 6.4.2 and <text:span> 6.1.7`.

The `<text:table-count>` element has the following attributes: `style:num-format 19.504` and `style:num-letter-sync 19.505`.

The `<text:table-count>` element has no child elements.

The `<text:table-count>` element has character data content.

### 7.5.18.7 `<text:image-count>`

The `<text:image-count>` element displays the value of the `meta:image-count` attribute of the `<meta:document-statistic>` element of a document. 19.327

The `<text:image-count>` element is usable within the following elements: `<text:a> 6.1.8, <text:h> 5.1.2, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:p> 5.1.3, <text:ruby-base> 6.4.2 and <text:span> 6.1.7`.

The `<text:image-count>` element has the following attributes: `style:num-format 19.504` and `style:num-letter-sync 19.505`.

The `<text:image-count>` element has no child elements.

The `<text:image-count>` element has character data content.

### 7.5.18.8 `<text:object-count>`

The `<text:object-count>` element displays the value of the `meta:object-count` attribute of the `<meta:document-statistic>` element of a document. 19.330

The `<text:object-count>` element is usable within the following elements: `<text:a> 6.1.8, <text:h> 5.1.2, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:p> 5.1.3, <text:ruby-base> 6.4.2 and <text:span> 6.1.7`.

The `<text:object-count>` element has the following attributes: `style:num-format 19.504` and `style:num-letter-sync 19.505`.

The `<text:object-count>` element has no child elements.

The `<text:object-count>` element has character data content.

### 7.5.19 `<text:meta-field>`

The `<text:meta-field>` element represents content from a metadata source. The mixed content of this element should be generated from the metadata source. The source of the metadata and the means of generation of the mixed content is implementation-dependent. The `<text:meta-field>` element may contain any paragraph content.

**Note:** Alterations of the element's mixed content may be lost if the mixed content is subsequently refreshed from a metadata source.

The `<text:meta-field>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:meta-field>` element has the following attributes: `style:data-style-name` 19.473 and `xml:id` 19.920.

The `<text:meta-field>` element has the following child elements: `<dr3d:scene>` 10.5.2, `<draw:a>` 10.4.12, `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:connector>` 10.3.10, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:g>` 10.3.15, `<draw:line>` 10.3.3, `<draw:measure>` 10.3.12, `<draw:page-thumbnail>` 10.3.14, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2, `<draw:regular-polygon>` 10.3.6, `<office:annotation>` 14.1, `<office:annotation-end>` 14.2, `<presentation:date-time>` 10.9.4.5, `<presentation:footer>` 10.9.4.3, `<presentation:header>` 10.9.4.1, `<text:a>` 6.1.8, `<text:alphabetical-index-mark>` 8.1.10, `<text:alphabetical-index-mark-end>` 8.1.9, `<text:alphabetical-index-mark-start>` 8.1.8, `<text:author-initials>` 7.3.7.2, `<text:author-name>` 7.3.7.1, `<text:bibliography-mark>` 8.1.11, `<text:bookmark>` 6.2.1.2, `<text:bookmark-end>` 6.2.1.4, `<text:bookmark-ref>` 7.7.6, `<text:bookmark-start>` 6.2.1.3, `<text:change>` 5.5.8.4, `<text:change-end>` 5.5.8.3, `<text:change-start>` 5.5.8.2, `<text:chapter>` 7.3.8, `<text:character-count>` 7.5.18.5, `<text:conditional-text>` 7.7.3, `<text:creation-date>` 7.5.3, `<text:creation-time>` 7.5.4, `<text:creator>` 7.5.17, `<text:database-display>` 7.6.3, `<text:database-name>` 7.6.7, `<text:database-next>` 7.6.4, `<text:database-row-number>` 7.6.6, `<text:database-row-select>` 7.6.5, `<text:date>` 7.3.2, `<text:dde-connection>` 7.7.12, `<text:description>` 7.5.5, `<text:drop-down>` 7.4.16, `<text:editing-cycles>` 7.5.13, `<text:editing-duration>` 7.5.14, `<text:execute-macro>` 7.7.10, `<text:expression>` 7.4.14, `<text:file-name>` 7.3.9, `<text:hidden-paragraph>` 7.7.11, `<text:hidden-text>` 7.7.4, `<text:image-count>` 7.5.18.7, `<text:initial-creator>` 7.5.2, `<text:keywords>` 7.5.12, `<text:line-break>` 6.1.5, `<text:measure>` 7.7.13, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:modification-date>` 7.5.16, `<text:modification-time>` 7.5.15, `<text:note>` 6.3.2, `<text:note-ref>` 7.7.7, `<text:object-count>` 7.5.18.8, `<text:page-continuation>` 7.3.5, `<text:page-count>` 7.5.18.2, `<text:page-number>` 7.3.4, `<text:page-variable-get>` 7.7.1.3, `<text:page-variable-set>` 7.7.1.2, `<text:paragraph-count>` 7.5.18.3, `<text:placeholder>` 7.7.2, `<text:print-date>` 7.5.8, `<text:print-time>` 7.5.7, `<text:printed-by>` 7.5.9, `<text:reference-mark>` 6.2.2.2, `<text:reference-mark-end>` 6.2.2.4, `<text:reference-mark-start>` 6.2.2.3, `<text:reference-ref>` 7.7.5, `<text:ruby>` 6.4, `<text:s>` 6.1.3, `<text:script>` 7.7.9, `<text:sender-city>` 7.3.6.13, `<text:sender-company>` 7.3.6.10, `<text:sender-country>` 7.3.6.15, `<text:sender-email>` 7.3.6.7, `<text:sender-fax>` 7.3.6.9, `<text:sender-firstname>` 7.3.6.2, `<text:sender-initials>` 7.3.6.4, `<text:sender-lastname>` 7.3.6.3, `<text:sender-phone-private>` 7.3.6.8, `<text:sender-phone-work>` 7.3.6.11, `<text:sender-position>` 7.3.6.6, `<text:sender-postal-code>` 7.3.6.14, `<text:sender-state-or-province>` 7.3.6.16, `<text:sender-street>` 7.3.6.12, `<text:sender-title>` 7.3.6.5, `<text:sequence>` 7.4.13, `<text:sequence-ref>` 7.7.8, `<text:sheet-name>` 7.3.11, `<text:soft-page-break>` 5.6, `<text:span>` 6.1.7, `<text:subject>` 7.5.11, `<text:tab>` 6.1.4, `<text:table-count>` 7.5.18.6, `<text:table-formula>` 7.7.14, `<text:template-name>` 7.3.10, `<text:text-input>` 7.4.15, `<text:time>` 7.3.3, `<text:title>` 7.5.10, `<text:toc-mark>` 8.1.4, `<text:toc-mark-end>` 8.1.3, `<text:toc-mark-start>` 8.1.2, `<text:user>`

```
defined> 7.5.6, <text:user-field-get> 7.4.9, <text:user-field-input> 7.4.10,  
<text:user-index-mark> 8.1.7, <text:user-index-mark-end> 8.1.6,  
<text:user-index-mark-start> 8.1.5, <text:variable-get> 7.4.5,  
<text:variable-input> 7.4.6, <text:variable-set> 7.4.4 and <text:word-  
count> 7.5.18.4.
```

## 7.6 Database Fields

### 7.6.1 General

Documents can reference databases and display database information as text content. To display database information, the OpenDocument schema uses a group of text fields, collectively called database fields. Consumers may use database tables from SQL servers, therefore database fields can be used to access any SQL database, provided that the appropriate drivers are available. [SQL]

A database may contain the following components:

- Tables, which store data.
- Queries, which extract a subset of data from one or more tables.
- Forms, which present data. 12.25.2
- Reports, which summarize database content. 12.25.3

Database fields operate on a set of database rows, the so called *current selection*, which is either a database table, or the result of a database query. The current selection is determined by the data source information that is contained in database fields. The current selection is retrieved once per document and data source, regardless of the number of database fields that are associated with a data source.

Database fields display data contained in a particular database row within the current selection. The row within the current selection is controlled by database fields and is known as the current row.

### 7.6.2 <form:connection-resource>

The `<form:connection-resource>` element specifies a source database.

The `<form:connection-resource>` element is usable within the following elements:  
`<form:form>` 13.3, `<text:database-display>` 7.6.3, `<text:database-name>` 7.6.7,  
`<text:database-next>` 7.6.4, `<text:database-row-number>` 7.6.6 and  
`<text:database-row-select>` 7.6.5.

The `<form:connection-resource>` element has the following attribute: `xlink:href` 19.916.

The `<form:connection-resource>` element has no child elements.

### 7.6.3 <text:database-display>

The `<text:database-display>` element represents data from the current database row.

The `<text:database-display>` element is usable within the following elements:  
`<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19,  
`<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:database-display>` element has the following attributes: `style:style-name` 19.473, `text:column-name` 19.776, `text:database-name` 19.797, `text:table-name` 19.884 and `text:table-type` 19.885.

The `<text:database-display>` element has the following child element:

`<form:connection-resource>` 7.6.2.

The `<text:database-display>` element has character data content.

#### 7.6.4 `<text:database-next>`

The `<text:database-next>` element selects the database row following the one currently displayed in a `<text:database-display>` field.

The `<text:database-next>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:database-next>` element has the following attributes: `text:condition` 19.782, `text:database-name` 19.797, `text:table-name` 19.884 and `text:table-type` 19.885.

The `<text:database-next>` element has the following child element:

`<form:connection-resource>` 7.6.2.

#### 7.6.5 `<text:database-row-select>`

The `<text:database-row-select>` element selects a specific row from a current selection.

The `<text:database-row-select>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:database-row-select>` element has the following attributes: `text:condition` 19.782, `text:database-name` 19.797, `text:row-number` 19.865, `text:table-name` 19.884 and `text:table-type` 19.885.

The `<text:database-row-select>` element has the following child element:

`<form:connection-resource>` 7.6.2.

#### 7.6.6 `<text:database-row-number>`

The `<text:database-row-number>` element displays the current row number for a given table. The element displays the row number from a database and not the row number of a current selection that is used as an attribute value in the `<text:database-row-select>` element.

The `<text:database-row-number>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:database-row-number>` element has the following attributes: `style:num-format` 19.504, `style:num-letter-sync` 19.505, `text:database-name` 19.797, `text:table-name` 19.884, `text:table-type` 19.885 and `text:value` 19.906.

The `<text:database-row-number>` element has the following child element:

`<form:connection-resource>` 7.6.2.

The `<text:database-row-number>` element has character data content.

#### 7.6.7 `<text:database-name>`

The `<text:database-name>` element displays the database and table name of the last used table.

The `<text:database-name>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:database-name>` element has the following attributes: `text:database-name` 19.797, `text:table-name` 19.884 and `text:table-type` 19.885.

The `<text:database-name>` element has the following child element: `<form:connection-resource>` 7.6.2.

The `<text:database-name>` element has character data content.

## 7.7 More Fields

### 7.7.1 Page Variable Fields

#### 7.7.1.1 General

Page variables enable an alternative page numbering to be defined for documents. Each page has only one active set page variable field. The calculation of a page variable is controlled by the `text:active` attribute

#### 7.7.1.2 `<text:page-variable-set>`

The `<text:page-variable-set>` element is used to set a page variable.

The `<text:page-variable-set>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:page-variable-set>` element has the following attributes: `text:active` 19.755 and `text:page-adjust` 19.851.

The `<text:page-variable-set>` element has no child elements.

The `<text:page-variable-set>` element has character data content.

#### 7.7.1.3 `<text:page-variable-get>`

The `<text:page-variable-get>` element displays the value of a page variable.

If the `text:active` attribute 19.755, has the value of `false`, the `<text:page-variable-get>` element will have no display for that page.

The `<text:page-variable-get>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:page-variable-get>` element has the following attributes: `style:num-format` 19.504 and `style:num-letter-sync` 19.505.

The `<text:page-variable-get>` element has no child elements.

The `<text:page-variable-get>` element has character data content.

### 7.7.2 `<text:placeholder>`

The `<text:placeholder>` element is a placeholder for replacement by content of a particular type as specified in the mandatory `text:placeholder-type` attribute.

OpenDocument consumers that support the `<text:placeholder>` element shall use the `text:description` attribute value as the placeholder text, and shall ensure that this text clearly indicates that it is a placeholder for content of the specified type.

The `<text:placeholder>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:placeholder>` element has the following attributes: `text:description` 19.801 and `text:placeholder-type` 19.853.

The `<text:placeholder>` element has no child elements.

The `<text:placeholder>` element has character data content.

### 7.7.3 `<text:conditional-text>`

The `<text:conditional-text>` element specifies a condition for display of one text string or another. If the condition is true, the `text:string-value-if-true` 19.879 attribute value is displayed. If the condition is false, the `text:string-value-if-false` 19.878 attribute value is displayed.

The `<text:conditional-text>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:conditional-text>` element has the following attributes: `text:condition` 19.782, `text:current-value` 19.791, `text:string-value-if-false` 19.878 and `text:string-value-if-true` 19.879.

The `<text:conditional-text>` element has no child elements.

The `<text:conditional-text>` element has character data content.

### 7.7.4 `<text:hidden-text>`

The `<text:hidden-text>` element hides the text it contains when its `text:condition` attribute evaluates to true.

(more edits pending here)

If its `text:condition` attribute is missing or evaluates to false, the text content is displayed.

The `<text:hidden-text>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:hidden-text>` element has the following attributes: `text:condition` 19.782, `text:is-hidden` 19.821 and `text:string-value` 19.877.

The `<text:hidden-text>` element has no child elements.

The `<text:hidden-text>` element has character data content.

### 7.7.5 `<text:reference-ref>`

The `<text:reference-ref>` element represents a field that references a `<text:reference-mark-start>` 6.2.2.3 or `<text:reference-mark>` 6.2.2.2 element. Its `text:reference-format` 19.860 attribute specifies what is displayed according to the context of the referenced element.

The `<text:reference-ref>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:reference-ref>` element has the following attributes: `text:ref-name` 19.859 and `text:reference-format` 19.860.

The `<text:reference-ref>` element has no child elements.

### 7.7.6 `<text:bookmark-ref>`

The `<text:bookmark-ref>` element represents a field that references a `<text:bookmark-start>` 6.2.1.3 or `<text:bookmark>` 6.2.1.2 element. Its `text:reference-format` 19.860 attribute specifies what is displayed according to the context of the referenced element.

The `<text:bookmark-ref>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:bookmark-ref>` element has the following attributes: `text:ref-name` 19.859 and `text:reference-format` 19.860.

The `<text:bookmark-ref>` element has no child elements.

### 7.7.7 `<text:note-ref>`

The `<text:note-ref>` element represents a field that references a `<text:note>` 6.3.2 element. Its `text:reference-format` 19.860 attribute specifies what is displayed from the referenced element.

The `<text:note-ref>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:note-ref>` element has the following attributes: `text:note-class` 19.843, `text:ref-name` 19.859 and `text:reference-format` 19.860.

The `<text:note-ref>` element has no child elements.

### 7.7.8 `<text:sequence-ref>`

The `<text:sequence-ref>` element represents a field that references a `<text:sequence>` 7.4.13 element. Its `text:reference-format` 19.860 attribute specifies what is displayed according to the context of the referenced element.

The `<text:sequence-ref>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:sequence-ref>` element has the following attributes: `text:ref-name` 19.859 and `text:reference-format` 19.860.

The `<text:sequence-ref>` element has no child elements.

### 7.7.9 `<text:script>`

The `<text:script>` element represents a field that stores scripts or sections of scripts.

**Note:** The primary purpose of this field is to provide an equivalent to the `<script>` element in [HTML4], so that the content of a `<script>` element in HTML can be imported, edited, and exported using office application software.

The source code for the script can be stored in one of the following ways:

- The `<text:script>` element contains the source code.
- The source code is stored in an external file. The `xlink:href` attribute specifies the location of the source file.

The element shall have either an `xlink:href` attribute or content, but not both.

The `<text:script>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:script>` element has the following attributes: `script:language` 19.434, `xlink:href` 19.916 and `xlink:type` 19.919.

The `<text:script>` element has no child elements.

The `<text:script>` element has character data content. Character data content is only permitted if a `xlink:href` attribute is not present.

### 7.7.10 `<text:execute-macro>`

The `<text:execute-macro>` element represents a field that contains the name of a macro that is executed when the field is activated. The element may contain a text description of the macro. The name of the macro to be executed is specified by the `text:name` attribute.

The `<text:execute-macro>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:execute-macro>` element has the following attribute: `text:name` 19.841.

The `<text:execute-macro>` element has the following child element: `<office:event-listeners>` 10.3.19.

The `<text:execute-macro>` element has character data content.

### 7.7.11 `<text:hidden-paragraph>`

The `<text:hidden-paragraph>` element is a field that represents a condition. If the condition evaluates to `false`, the paragraph where this element appears is displayed unless the paragraph is enclosed in any other hiding element. If the condition evaluates to `true`, the paragraph is hidden unless the paragraph contains another `<text:hidden-paragraph>` element whose condition evaluates to `false`.

The `<text:hidden-paragraph>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:hidden-paragraph>` element has the following attributes: `text:condition` 19.782 and `text:is-hidden` 19.821.

The `<text:hidden-paragraph>` element has no child elements.

The `<text:hidden-paragraph>` element has character data content.

### 7.7.12 `<text:dde-connection>`

The `<text:dde-connection>` element represents a field that is information from a DDE connection. The content of this field is the last data that was received from a DDE connection. This is used as the content of this field if the DDE connection cannot be accessed.

See 14.7 for the use of DDE connections.

The `<text:dde-connection>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:dde-connection>` element has the following attribute: `text:connection-name 19.783`.

The `<text:dde-connection>` element has no child elements.

The `<text:dde-connection>` element has character data content.

### 7.7.13 `<text:measure>`

The `<text:measure>` element represents a field that displays measure text. It can only be used within text that is contained in measure drawing objects.

The `<text:measure>` element is usable within the following elements: `<text:a> 6.1.8, <text:h> 5.1.2, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:p> 5.1.3, <text:ruby-base> 6.4.2 and <text:span> 6.1.7`.

The `<text:measure>` element has the following attribute: `text:kind 19.831`.

The `<text:measure>` element has no child elements.

The `<text:measure>` element has character data content.

### 7.7.14 `<text:table-formula>` (deprecated)

The `<text:table-formula>` element field is deprecated. It stores a formula to be used in tables.

The `<text:table-formula>` element is usable within the following elements: `<text:a> 6.1.8, <text:h> 5.1.2, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:p> 5.1.3, <text:ruby-base> 6.4.2 and <text:span> 6.1.7`.

The `<text:table-formula>` element has the following attributes: `style:data-style-name 19.473, text:display 19.802 and text:formula 19.811`.

The `<text:table-formula>` element has no child elements.

The `<text:table-formula>` element has character data content.

# 8 Text Indexes

## 8.1 Index Marks

### 8.1.1 General

OpenDocument text documents may contain automatically generated indexes. An index contains a sorted list of items of a specified type, where the sorting and the type of items are determined by the type of index.

There are three types of index marks that correspond to the three types of index that make use of index marks. The three types of index marks are:

- Table of contents index marks
- User-defined index marks
- Alphabetical index marks

The rules for index marks are:

- Index marks are represented by start and end elements or by non-enclosing index point elements.
- Start and end index elements use the `text:id` attribute to match start and end elements for an index mark.
- Start and end index elements shall be contained in the same paragraph, with the start element occurring first in document order.
- The attributes associated with an index mark are attached to the start element.
- The text between the start and end index elements is the text of the index entry.
- Index marks can overlap.

### 8.1.2 <text:toc-mark-start>

The `<text:toc-mark-start>` element marks the start of a table of contents index entry.

The `<text:toc-mark-start>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:toc-mark-start>` element has the following attributes: `text:id` 19.815 and `text:outline-level` 19.850.

The `<text:toc-mark-start>` element has no child elements.

### 8.1.3 <text:toc-mark-end>

The `<text:toc-mark-end>` element marks the end of a table of contents index entry.

The `<text:toc-mark-end>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<text:toc-mark-end>` element has the following attribute: `text:id` 19.815.

The `<text:toc-mark-end>` element has no child elements.

#### 8.1.4 <text:toc-mark>

The <text:toc-mark> element represents a table of content index mark that does not enclose content.

The <text:toc-mark> element is usable within the following elements: <text:a> 6.1.8, <text:h> 5.1.2, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:p> 5.1.3, <text:ruby-base> 6.4.2 and <text:span> 6.1.7.

The <text:toc-mark> element has the following attributes: text:outline-level 19.850 and text:string-value 19.877.

The <text:toc-mark> element has no child elements.

#### 8.1.5 <text:user-index-mark-start>

The <text:user-index-mark-start> element marks the start of a user-defined index entry.

The <text:user-index-mark-start> element is usable within the following elements: <text:a> 6.1.8, <text:h> 5.1.2, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:p> 5.1.3, <text:ruby-base> 6.4.2 and <text:span> 6.1.7.

The <text:user-index-mark-start> element has the following attributes: text:id 19.815, text:index-name 19.817 and text:outline-level 19.850.

The <text:user-index-mark-start> element has no child elements.

#### 8.1.6 <text:user-index-mark-end>

The <text:user-index-mark-end> element marks the end of the user-defined index entry.

The <text:user-index-mark-end> element is usable within the following elements: <text:a> 6.1.8, <text:h> 5.1.2, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:p> 5.1.3, <text:ruby-base> 6.4.2 and <text:span> 6.1.7.

The <text:user-index-mark-end> element has the following attribute: text:id 19.815.

The <text:user-index-mark-end> element has no child elements.

#### 8.1.7 <text:user-index-mark>

The <text:user-index-mark> element represents a user-defined index mark that does not enclose content.

The <text:user-index-mark> element is usable within the following elements: <text:a> 6.1.8, <text:h> 5.1.2, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:p> 5.1.3, <text:ruby-base> 6.4.2 and <text:span> 6.1.7.

The <text:user-index-mark> element has the following attributes: text:index-name 19.817, text:outline-level 19.850 and text:string-value 19.877.

The <text:user-index-mark> element has no child elements.

#### 8.1.8 <text:alphabetical-index-mark-start>

The <text:alphabetical-index-mark-start> element marks the start of an alphabetical index entry.

The <text:alphabetical-index-mark-start> element is usable within the following elements: <text:a> 6.1.8, <text:h> 5.1.2, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:p> 5.1.3, <text:ruby-base> 6.4.2 and <text:span> 6.1.7.

The `<text:alphabetical-index-mark-start>` element has the following attributes: `text:id 19.815`, `text:key1 19.827`, `text:key1-phonetic 19.829`, `text:key2 19.828`, `text:key2-phonetic 19.830`, `text:main-entry 19.837` and `text:string-value-phonetic 19.876`.

The `<text:alphabetical-index-mark-start>` element has no child elements.

### 8.1.9 `<text:alphabetical-index-mark-end>`

The `<text:alphabetical-index-mark-end>` element marks the end of an alphabetical index entry.

The `<text:alphabetical-index-mark-end>` element is usable within the following elements: `<text:a> 6.1.8`, `<text:h> 5.1.2`, `<text:meta> 6.1.9`, `<text:meta-field> 7.5.19`, `<text:p> 5.1.3`, `<text:ruby-base> 6.4.2` and `<text:span> 6.1.7`.

The `<text:alphabetical-index-mark-end>` element has the following attribute: `text:id 19.815`.

The `<text:alphabetical-index-mark-end>` element has no child elements.

### 8.1.10 `<text:alphabetical-index-mark>`

The `<text:user-index-mark>` element represents a user-defined index mark that does not enclose content.

The `<text:alphabetical-index-mark>` element is usable within the following elements: `<text:a> 6.1.8`, `<text:h> 5.1.2`, `<text:meta> 6.1.9`, `<text:meta-field> 7.5.19`, `<text:p> 5.1.3`, `<text:ruby-base> 6.4.2` and `<text:span> 6.1.7`.

The `<text:alphabetical-index-mark>` element has the following attributes: `text:key1 19.827`, `text:key1-phonetic 19.829`, `text:key2 19.828`, `text:key2-phonetic 19.830`, `text:main-entry 19.837`, `text:string-value 19.877` and `text:string-value-phonetic 19.876`.

The `<text:alphabetical-index-mark>` element has no child elements.

### 8.1.11 `<text:bibliography-mark>`

The `<text:bibliography-mark>` element contains the text and information for a bibliography index entry.

The `<text:bibliography-mark>` element is usable within the following elements: `<text:a> 6.1.8`, `<text:h> 5.1.2`, `<text:meta> 6.1.9`, `<text:meta-field> 7.5.19`, `<text:p> 5.1.3`, `<text:ruby-base> 6.4.2` and `<text:span> 6.1.7`.

The `<text:bibliography-mark>` element has the following attributes: `text:address 19.756`, `text:annotate 19.760`, `text:author 19.761`, `text:bibliography-type 19.763`, `text:booktitle 19.764`, `text:chapter 19.772`, `text:custom1 19.792`, `text:custom2 19.793`, `text:custom3 19.794`, `text:custom4 19.795`, `text:custom5 19.796`, `text:edition 19.806`, `text:editor 19.807`, `text:howpublished 19.813`, `text:identifier 19.816`, `text:institution 19.820`, `text:isbn 19.823`, `text:issn 19.824`, `text:journal 19.825`, `text:month 19.840`, `text:note 19.842`, `text:number 19.844`, `text:organizations 19.849`, `text:pages 19.852`, `text:publisher 19.858`, `text:report-type 19.862`, `text:school 19.866`, `text:series 19.868`, `text:title 19.888`, `text:url 19.890`, `text:volume 19.908` and `text:year 19.909`.

The `<text:bibliography-mark>` element has no child elements.

The `<text:bibliography-mark>` element has character data content.

## 8.2 Index Structure

### 8.2.1 General

An index consists of two parts: the index source, and the index body.

The index source is specific to the type of index it is being used for. It contains the information necessary to generate the index content.

### 8.2.2 <text:index-body>

The `<text:index-body>` element contains an index.

The `<text:index-body>` element is used for all types of indexes. It contains the index content generated to form the index.

The `<text:index-body>` element is usable within the following elements:

`<text:alphabetical-index> 8.8, <text:bibliography> 8.9,`  
`<text:illustration-index> 8.4, <text:object-index> 8.6, <text:table-`  
`index> 8.5, <text:table-of-content> 8.3 and <text:user-index> 8.7.`

The `<text:index-body>` element has no attributes.

The `<text:index-body>` element has the following child elements: `<dr3d:scene>` 10.5.2, `<draw:a>` 10.4.12, `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:connector>` 10.3.10, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:g>` 10.3.15, `<draw:line>` 10.3.3, `<draw:measure>` 10.3.12, `<draw:page-thumbnail>` 10.3.14, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2, `<draw:regular-polygon>` 10.3.6, `<table:table>` 9.1.2, `<text:alphabetical-`  
`index>` 8.8, `<text:bibliography>` 8.9, `<text:change>` 5.5.8.4, `<text:change-end>` 5.5.8.3, `<text:change-start>` 5.5.8.2, `<text:h>` 5.1.2, `<text:illustration-`  
`index>` 8.4, `<text:index-title>` 8.2.3, `<text:list>` 5.3.1, `<text:numbered-`  
`paragraph>` 5.3.6, `<text:object-index>` 8.6, `<text:p>` 5.1.3, `<text:section>` 5.4, `<text:soft-page-break>` 5.6, `<text:table-index>` 8.5, `<text:table-of-`  
`content>` 8.3 and `<text:user-index>` 8.7.

### 8.2.3 <text:index-title>

The `<text:index-title>` element contains the title of an index.

The `<text:index-title>` element is usable within the following elements:

`<style:footer>` 16.12, `<style:footer-first>` 16.13, `<style:footer-left>` 16.15, `<style:header>` 16.10, `<style:header-first>` 16.11, `<style:header-left>` 16.14, `<text:index-body>` 8.2.2 and `<text:index-title>` 8.2.3.

The `<text:index-title>` element has the following attributes: `text:name` 19.841, `text:protected` 19.855, `text:protection-key` 19.856, `text:protection-key-digest-algorithm` 19.857, `text:style-name` 19.880 and `xml:id` 19.920.

The `<text:index-title>` element has the following child elements: `<dr3d:scene>` 10.5.2, `<draw:a>` 10.4.12, `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:connector>` 10.3.10, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:g>` 10.3.15, `<draw:line>` 10.3.3, `<draw:measure>` 10.3.12, `<draw:page-thumbnail>` 10.3.14, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2, `<draw:regular-polygon>` 10.3.6, `<table:table>` 9.1.2, `<text:alphabetical-`  
`index>` 8.8, `<text:bibliography>` 8.9, `<text:change>` 5.5.8.4, `<text:change-end>` 5.5.8.3, `<text:change-start>` 5.5.8.2, `<text:h>` 5.1.2, `<text:illustration-`  
`index>` 8.4, `<text:index-title>` 8.2.3, `<text:list>` 5.3.1, `<text:numbered-`

```
paragraph> 5.3.6, <text:object-index> 8.6, <text:p> 5.1.3, <text:section> 5.4, <text:soft-page-break> 5.6, <text:table-index> 8.5, <text:table-of-content> 8.3 and <text:user-index> 8.7.
```

## 8.3 <text:table-of-content>

### 8.3.1 General

The <text:table-of-content> element represents a table of contents for a document.

The items that can be listed in a table of contents are:

- Headings (as defined by the outline structure of the document), up to a selected level.
- Table of contents index marks.
- Paragraphs formatted with specified paragraph styles.

**Note:** Paragraphs may be selected for inclusion in a table of contents based on their styles. Selection is based on a paragraph's style matching a style specified by the <text:index-source-styles> child element of a <text:table-of-content-source> child element of a <text:table-of-content> element, when the text:use-index-source-styles attribute of the child <text:index-source-styles> element has the value true.

The <text:table-of-content> element is usable within the following elements:

```
<draw:text-box> 10.4.3, <office:text> 3.4, <style:footer> 16.12, <style:footer-first> 16.13, <style:footer-left> 16.15, <style:header> 16.10, <style:header-first> 16.11, <style:header-left> 16.14, <table:covered-table-cell> 9.1.5, <table:table-cell> 9.1.4, <text:deletion> 5.5.5, <text:index-body> 8.2.2, <text:index-title> 8.2.3, <text:note-body> 6.3.4 and <text:section> 5.4.
```

The <text:table-of-content> element has the following attributes: text:name 19.841, text:protected 19.855, text:protection-key 19.856, text:protection-key-digest-algorithm 19.857, text:style-name 19.880 and xml:id 19.920.

The <text:table-of-content> element has the following child elements:

```
<text:index-body> 8.2.2 and <text:table-of-content-source> 8.3.2.
```

### 8.3.2 <text:table-of-content-source>

The <text:table-of-content-source> element specifies how a table of contents is generated. It also specifies the styles to be applied to index entries.

The <text:table-of-content-source> element is usable within the following element: <text:table-of-content> 8.3.

The <text:table-of-content-source> element has the following attributes: text:index-scope 19.818, text:outline-level 19.850, text:relative-tab-stop-position 19.861, text:use-index-marks 19.896, text:use-index-source-styles 19.897 and text:use-outline-level 19.902.

The <text:table-of-content-source> element has the following child elements: <text:index-source-styles> 8.10, <text:index-title-template> 8.12 and <text:table-of-content-entry-template> 8.3.3.

### 8.3.3 <text:table-of-content-entry-template>

The `<text:table-of-content-entry-template>` element specifies the format of an index entry for a specified outline level.

For each `<text:table-of-content-source>` element, there shall not be more than one `<text:table-of-content-entry-template>` for a specified outline level.

The `<text:table-of-content-entry-template>` element is usable within the following element: `<text:table-of-content-source>` 8.3.2.

The `<text:table-of-content-entry-template>` element has the following attributes: `text:outline-level 19.850` and `text:style-name 19.880`.

The `<text:table-of-content-entry-template>` element has the following child elements: `<text:index-entry-chapter>` 8.13.1, `<text:index-entry-link-end>` 8.13.8, `<text:index-entry-link-start>` 8.13.7, `<text:index-entry-page-number>` 8.13.3, `<text:index-entry-span>` 8.13.4, `<text:index-entry-tab-stop>` 8.13.6 and `<text:index-entry-text>` 8.13.2.

## 8.4 <text:illustration-index>

### 8.4.1 General

The `<text:illustration-index>` element represents an index of the images and graphics in a current document or chapter.

The `<text:illustration-index>` element is usable within the following elements: `<draw:text-box>` 10.4.3, `<office:text>` 3.4, `<style:footer>` 16.12, `<style:footer-first>` 16.13, `<style:footer-left>` 16.15, `<style:header>` 16.10, `<style:header-first>` 16.11, `<style:header-left>` 16.14, `<table:covered-table-cell>` 9.1.5, `<table:table-cell>` 9.1.4, `<text:deletion>` 5.5.5, `<text:index-body>` 8.2.2, `<text:index-title>` 8.2.3, `<text:note-body>` 6.3.4 and `<text:section>` 5.4.

The `<text:illustration-index>` element has the following attributes: `text:name 19.841`, `text:protected 19.855`, `text:protection-key 19.856`, `text:protection-key-digest-algorithm 19.857`, `text:style-name 19.880` and `xml:id 19.920`.

The `<text:illustration-index>` element has the following child elements: `<text:illustration-index-source>` 8.4.2 and `<text:index-body>` 8.2.2.

### 8.4.2 <text:illustration-index-source>

The `<text:illustration-index-source>` element specifies how an index of illustrations is generated.

The `<text:illustration-index-source>` element is usable within the following element: `<text:illustration-index>` 8.4.

The `<text:illustration-index-source>` element has the following attributes: `text:caption-sequence-format 19.769`, `text:caption-sequence-name 19.770`, `text:index-scope 19.818`, `text:relative-tab-stop-position 19.861` and `text:use-caption 19.891`.

The `<text:illustration-index-source>` element has the following child elements: `<text:illustration-index-entry-template>` 8.4.3 and `<text:index-title-template>` 8.12.

### 8.4.3 <text:illustration-index-entry-template>

The `<text:illustration-index-entry-template>` element specifies the format of an index entry.

The `<text:illustration-index-entry-template>` element is usable within the following element: `<text:illustration-index-source>` 8.4.2.

The `<text:illustration-index-entry-template>` element has the following attribute: `text:style-name` 19.880.

The `<text:illustration-index-entry-template>` element has the following child elements: `<text:index-entry-chapter>` 8.13.1, `<text:index-entry-link-end>` 8.13.8, `<text:index-entry-link-start>` 8.13.7, `<text:index-entry-page-number>` 8.13.3, `<text:index-entry-span>` 8.13.4, `<text:index-entry-tab-stop>` 8.13.6 and `<text:index-entry-text>` 8.13.2.

## 8.5 <text:table-index>

### 8.5.1 General

The `<text:table-index>` element represents an index of the tables in a document or chapter.

The `<text:table-index>` element is usable within the following elements: `<draw:text-box>` 10.4.3, `<office:text>` 3.4, `<style:footer>` 16.12, `<style:footer-first>` 16.13, `<style:footer-left>` 16.15, `<style:header>` 16.10, `<style:header-first>` 16.11, `<style:header-left>` 16.14, `<table:covered-table-cell>` 9.1.5, `<table:table-cell>` 9.1.4, `<text:deletion>` 5.5.5, `<text:index-body>` 8.2.2, `<text:index-title>` 8.2.3, `<text:note-body>` 6.3.4 and `<text:section>` 5.4.

The `<text:table-index>` element has the following attributes: `text:name` 19.841, `text:protected` 19.855, `text:protection-key` 19.856, `text:protection-key-digest-algorithm` 19.857, `text:style-name` 19.880 and `xml:id` 19.920.

The `<text:table-index>` element has the following child elements: `<text:index-body>` 8.2.2 and `<text:table-index-source>` 8.5.2.

### 8.5.2 <text:table-index-source>

The `<text:table-index-source>` element specifies how an index of tables is generated.

The `<text:table-index-source>` element is usable within the following element: `<text:table-index>` 8.5.

The `<text:table-index-source>` element has the following attributes: `text:caption-sequence-format` 19.769, `text:caption-sequence-name` 19.770, `text:index-scope` 19.818, `text:relative-tab-stop-position` 19.861 and `text:use-caption` 19.891.

The `<text:table-index-source>` element has the following child elements: `<text:index-title-template>` 8.12 and `<text:table-index-entry-template>` 8.5.3.

### 8.5.3 <text:table-index-entry-template>

The `<text:table-index-entry-template>` element specifies the format of an index entry.

The `<text:table-index-entry-template>` element is usable within the following element: `<text:table-index-source>` 8.5.2.

The `<text:table-index-entry-template>` element has the following attribute: `text:style-name 19.880`.

The `<text:table-index-entry-template>` element has the following child elements: `<text:index-entry-chapter> 8.13.1`, `<text:index-entry-link-end> 8.13.8`, `<text:index-entry-link-start> 8.13.7`, `<text:index-entry-page-number> 8.13.3`, `<text:index-entry-span> 8.13.4`, `<text:index-entry-tab-stop> 8.13.6` and `<text:index-entry-text> 8.13.2`.

## 8.6 `<text:object-index>`

### 8.6.1 General

The `<text:object-index>` element represents an index of the objects in a document or chapter. The types of objects that should be included in the index are defined by the `<text:object-index-source>` element. 8.6.2

The `<text:object-index>` element is usable within the following elements:

`<draw:text-box> 10.4.3`, `<office:text> 3.4`, `<style:footer> 16.12`, `<style:footer-first> 16.13`, `<style:footer-left> 16.15`, `<style:header> 16.10`, `<style:header-first> 16.11`, `<style:header-left> 16.14`, `<table:covered-table-cell> 9.1.5`, `<table:table-cell> 9.1.4`, `<text:deletion> 5.5.5`, `<text:index-body> 8.2.2`, `<text:index-title> 8.2.3`, `<text:note-body> 6.3.4` and `<text:section> 5.4`.

The `<text:object-index>` element has the following attributes: `text:name 19.841`, `text:protected 19.855`, `text:protection-key 19.856`, `text:protection-key-digest-algorithm 19.857`, `text:style-name 19.880` and `xml:id 19.920`.

The `<text:object-index>` element has the following child elements: `<text:index-body> 8.2.2` and `<text:object-index-source> 8.6.2`.

### 8.6.2 `<text:object-index-source>`

The `<text:object-index-source>` element specifies which object types to include in the index of objects.

The `<text:object-index-source>` element is usable within the following element: `<text:object-index> 8.6`.

The `<text:object-index-source>` element has the following attributes: `text:index-scope 19.818`, `text:relative-tab-stop-position 19.861`, `text:use-chart-objects 19.892`, `text:use-draw-objects 19.893`, `text:use-math-objects 19.899`, `text:use-other-objects 19.901` and `text:use-spreadsheet-objects 19.904`.

The `<text:object-index-source>` element has the following child elements: `<text:index-title-template> 8.12` and `<text:object-index-entry-template> 8.6.3`.

### 8.6.3 `<text:object-index-entry-template>`

The `<text:object-index-entry-template>` element specifies the format of an index entry.

The `<text:object-index-entry-template>` element is usable within the following element: `<text:object-index-source> 8.6.2`.

The `<text:object-index-entry-template>` element has the following attribute: `text:style-name 19.880`.

The `<text:object-index-entry-template>` element has the following child elements: `<text:index-entry-chapter>` 8.13.1, `<text:index-entry-link-end>` 8.13.8, `<text:index-entry-link-start>` 8.13.7, `<text:index-entry-page-number>` 8.13.3, `<text:index-entry-span>` 8.13.4, `<text:index-entry-tab-stop>` 8.13.6 and `<text:index-entry-text>` 8.13.2.

## 8.7 `<text:user-index>`

### 8.7.1 General

The `<text:user-index>` element represents a user-defined index. A user-defined index may include entries from the following sources:

- Index marks
- Paragraphs formatted using specified paragraph styles
- Tables, images, or objects
- Text frames

The `<text:user-index>` element is usable within the following elements: `<draw:text-box>` 10.4.3, `<office:text>` 3.4, `<style:footer>` 16.12, `<style:footer-first>` 16.13, `<style:footer-left>` 16.15, `<style:header>` 16.10, `<style:header-first>` 16.11, `<style:header-left>` 16.14, `<table:covered-table-cell>` 9.1.5, `<table:table-cell>` 9.1.4, `<text:deletion>` 5.5.5, `<text:index-body>` 8.2.2, `<text:index-title>` 8.2.3, `<text:note-body>` 6.3.4 and `<text:section>` 5.4.

The `<text:user-index>` element has the following attributes: `text:name` 19.841, `text:protected` 19.855, `text:protection-key` 19.856, `text:protection-key-digest-algorithm` 19.857, `text:style-name` 19.880 and `xml:id` 19.920.

The `<text:user-index>` element has the following child elements: `<text:index-body>` 8.2.2 and `<text:user-index-source>` 8.7.2.

### 8.7.2 `<text:user-index-source>`

The `<text:user-index-source>` element defines the generation of user-defined indexes.

The `<text:user-index-source>` element is usable within the following element: `<text:user-index>` 8.7.

The `<text:user-index-source>` element has the following attributes: `text:copy-outline-levels` 19.787, `text:index-name` 19.817, `text:index-scope` 19.818, `text:relative-tab-stop-position` 19.861, `text:use-floating-frames` 19.894, `text:use-graphics` 19.895, `text:use-index-marks` 19.896, `text:use-index-source-styles` 19.897, `text:use-objects` 19.900 and `text:use-tables` 19.905.

The `<text:user-index-source>` element has the following child elements: `<text:index-source-styles>` 8.10, `<text:index-title-template>` 8.12 and `<text:user-index-entry-template>` 8.7.3.

### 8.7.3 `<text:user-index-entry-template>`

The `<text:user-index-entry-template>` element contains entry elements for chapter number, page number, entry text, text spans, and tab stops.

The `<text:user-index-entry-template>` element is usable within the following element: `<text:user-index-source>` 8.7.2.

The `<text:user-index-entry-template>` element has the following attributes: `text:outline-level` 19.850 and `text:style-name` 19.880.

The `<text:user-index-entry-template>` element has the following child elements:  
`<text:index-entry-chapter>` 8.13.1, `<text:index-entry-link-end>` 8.13.8,  
`<text:index-entry-link-start>` 8.13.7, `<text:index-entry-page-number>`  
8.13.3, `<text:index-entry-span>` 8.13.4, `<text:index-entry-tab-stop>` 8.13.6  
and `<text:index-entry-text>` 8.13.2.

## 8.8 `<text:alphabetical-index>`

### 8.8.1 General

The `<text:alphabetical-index>` element represents an index which is based on alphabetical index marks. The entries in an alphabetical index are sorted.

**Note:** Sort orders are locale-specific.

The `<text:alphabetical-index>` element is usable within the following elements:  
`<draw:text-box>` 10.4.3, `<office:text>` 3.4, `<style:footer>` 16.12,  
`<style:footer-first>` 16.13, `<style:footer-left>` 16.15, `<style:header>`  
16.10, `<style:header-first>` 16.11, `<style:header-left>` 16.14,  
`<table:covered-table-cell>` 9.1.5, `<table:table-cell>` 9.1.4,  
`<text:deletion>` 5.5.5, `<text:index-body>` 8.2.2, `<text:index-title>` 8.2.3,  
`<text:note-body>` 6.3.4 and `<text:section>` 5.4.

The `<text:alphabetical-index>` element has the following attributes: `text:name` 19.841, `text:protected` 19.855, `text:protection-key` 19.856, `text:protection-key-digest-algorithm` 19.857, `text:style-name` 19.880 and `xml:id` 19.920.

The `<text:alphabetical-index>` element has the following child elements:

`<text:alphabetical-index-source>` 8.8.2 and `<text:index-body>` 8.2.2.

### 8.8.2 `<text:alphabetical-index-source>`

The `<text:alphabetical-index-source>` element specifies how an alphabetical index is generated.

The `<text:alphabetical-index-source>` element is usable within the following element: `<text:alphabetical-index>` 8.8.

The `<text:alphabetical-index-source>` element has the following attributes:  
`fo:country` 19.234, `fo:language` 19.236, `fo:script` 19.242, `style:rfc-language-tag` 19.516, `text:alphabetical-separators` 19.757, `text:capitalize-entries` 19.767, `text:combine-entries` 19.777, `text:combine-entries-with-dash` 19.778, `text:combine-entries-with-pp` 19.779, `text:comma-separated` 19.780, `text:ignore-case` 19.819, `text:index-scope` 19.818, `text:main-entry-style-name` 19.838, `text:relative-tab-stop-position` 19.861, `text:sort-algorithm` 19.871 and `text:use-keys-as-entries` 19.898.

The `<text:alphabetical-index-source>` element has the following child elements:  
`<text:alphabetical-index-entry-template>` 8.8.4 and `<text:index-title-template>` 8.12.

### 8.8.3 `<text:alphabetical-index-auto-mark-file>`

The `<text:alphabetical-index-auto-mark-file>` element specifies an auto mark file. An auto mark file is a file that defines a list of terms that may be used by consumers to automatically insert alphabetical index marks (see 8.1.8, 8.1.10).

The format of an index mark file is implementation-defined.

The `<text:alphabetical-index-auto-mark-file>` element is usable within the following elements: `<office:chart>` 3.8, `<office:drawing>` 3.5, `<office:presentation>` 3.6, `<office:spreadsheet>` 3.7, `<office:text>` 3.4, `<style:footer>` 16.12, `<style:footer-first>` 16.13, `<style:footer-left>` 16.15, `<style:header>` 16.10, `<style:header-first>` 16.11 and `<style:header-left>` 16.14.

The `<text:alphabetical-index-auto-mark-file>` element has the following attributes: `xlink:href` 19.916 and `xlink:type` 19.919.

The `<text:alphabetical-index-auto-mark-file>` element has no child elements.

#### 8.8.4 `<text:alphabetical-index-entry-template>`

The `<text:alphabetical-index-entry-template>` element specifies the format of an alphabetical index entry.

The `<text:alphabetical-index-entry-template>` element is usable within the following element: `<text:alphabetical-index-source>` 8.8.2.

The `<text:alphabetical-index-entry-template>` element has the following attributes: `text:outline-level` 19.850 and `text:style-name` 19.880.

The `<text:alphabetical-index-entry-template>` element has the following child elements: `<text:index-entry-chapter>` 8.13.1, `<text:index-entry-page-number>` 8.13.3, `<text:index-entry-span>` 8.13.4, `<text:index-entry-tab-stop>` 8.13.6 and `<text:index-entry-text>` 8.13.2.

### 8.9 `<text:bibliography>`

#### 8.9.1 General

The `<text:bibliography>` element represents a bibliography constructed from bibliography index marks.

The `<text:bibliography>` element is usable within the following elements: `<draw:text-box>` 10.4.3, `<office:text>` 3.4, `<style:footer>` 16.12, `<style:footer-first>` 16.13, `<style:footer-left>` 16.15, `<style:header>` 16.10, `<style:header-first>` 16.11, `<style:header-left>` 16.14, `<table:covered-table-cell>` 9.1.5, `<table:table-cell>` 9.1.4, `<text:deletion>` 5.5.5, `<text:index-body>` 8.2.2, `<text:index-title>` 8.2.3, `<text:note-body>` 6.3.4 and `<text:section>` 5.4.

The `<text:bibliography>` element has the following attributes: `text:name` 19.841, `text:protected` 19.855, `text:protection-key` 19.856, `text:protection-key-digest-algorithm` 19.857, `text:style-name` 19.880 and `xml:id` 19.920.

The `<text:bibliography>` element has the following child elements: `<text:bibliography-source>` 8.9.2 and `<text:index-body>` 8.2.2.

#### 8.9.2 `<text:bibliography-source>`

The `<text:bibliography-source>` element specifies how a bibliography is generated.

The `<text:bibliography-source>` element is usable within the following element: `<text:bibliography>` 8.9.

The `<text:bibliography-source>` element has no attributes.

The `<text:bibliography-source>` element has the following child elements: `<text:bibliography-entry-template>` 8.9.3 and `<text:index-title-template>` 8.12.

### 8.9.3 `<text:bibliography-entry-template>`

The `<text:bibliography-entry-template>` element specifies the formatting for a bibliographic entry.

There shall be no more than one `<text:bibliography-entry-template>` element for each type of entry within any `<text:bibliography-source>` element. 8.9.2

The `<text:bibliography-entry-template>` element is usable within the following element: `<text:bibliography-source>` 8.9.2.

The `<text:bibliography-entry-template>` element has the following attributes: `text:bibliography-type` 19.763 and `text:style-name` 19.880.

The `<text:bibliography-entry-template>` element has the following child elements: `<text:index-entry-bibliography>` 8.13.5, `<text:index-entry-span>` 8.13.4 and `<text:index-entry-tab-stop>` 8.13.6.

### 8.10 `<text:index-source-styles>`

The `<text:index-source-styles>` element contains all of the `<text:index-source-style>` elements for a specified outline level. There can only be one `<text:index-source-styles>` element for each outline level.

The `<text:index-source-styles>` element is usable within the following elements: `<text:table-of-content-source>` 8.3.2 and `<text:user-index-source>` 8.7.2.

The `<text:index-source-styles>` element has the following attribute: `text:outline-level` 19.850.

The `<text:index-source-styles>` element has the following child element: `<text:index-source-style>` 8.11.

### 8.11 `<text:index-source-style>`

The `<text:index-source-style>` element specifies the name of a style or a class. Paragraphs formatted using that style or class are included in the index.

The `<text:index-source-style>` element is usable within the following element: `<text:index-source-styles>` 8.10.

The `<text:index-source-style>` element has the following attribute: `text:style-name` 19.880.

The `<text:index-source-style>` element has no child elements.

### 8.12 `<text:index-title-template>`

The `<text:index-title-template>` element specifies the style and content of the index title. There can only be one `<text:index-title-template>` element contained in a `<text:table-of-content-source>` element.

The `<text:index-title-template>` element is usable within the following elements: `<text:alphabetical-index-source>` 8.8.2, `<text:bibliography-source>` 8.9.2, `<text:illustration-index-source>` 8.4.2, `<text:object-index-source>` 8.6.2, `<text:table-index-source>` 8.5.2, `<text:table-of-content-source>` 8.3.2 and `<text:user-index-source>` 8.7.2.

The `<text:index-title-template>` element has the following attribute: `text:style-name 19.880`.

The `<text:index-title-template>` element has no child elements.

The `<text:index-title-template>` element has character data content.

## 8.13 Index Template Entries

### 8.13.1 <text:index-entry-chapter>

The `<text:index-entry-chapter>` element represents the chapter number where an index entry is located.

The `<text:index-entry-chapter>` element is usable within the following elements: `<text:alphabetical-index-entry-template> 8.8.4, <text:illustration-index-entry-template> 8.4.3, <text:object-index-entry-template> 8.6.3, <text:table-index-entry-template> 8.5.3, <text:table-of-content-entry-template> 8.3.3 and <text:user-index-entry-template> 8.7.3.`

The `<text:index-entry-chapter>` element has the following attributes: `text:display 19.802, text:outline-level 19.850` and `text:style-name 19.880`.

The `<text:index-entry-chapter>` element has no child elements.

### 8.13.2 <text:index-entry-text>

The `<text:index-entry-text>` element contains the text of an index entry.

The `<text:index-entry-text>` element is usable within the following elements: `<text:alphabetical-index-entry-template> 8.8.4, <text:illustration-index-entry-template> 8.4.3, <text:object-index-entry-template> 8.6.3, <text:table-index-entry-template> 8.5.3, <text:table-of-content-entry-template> 8.3.3 and <text:user-index-entry-template> 8.7.3.`

The `<text:index-entry-text>` element has the following attribute: `text:style-name 19.880`.

The `<text:index-entry-text>` element has no child elements.

### 8.13.3 <text:index-entry-page-number>

The `<text:index-entry-page-number>` element represents the number of the chapter where an index entry is located.

The `<text:index-entry-page-number>` element is usable within the following elements: `<text:alphabetical-index-entry-template> 8.8.4, <text:illustration-index-entry-template> 8.4.3, <text:object-index-entry-template> 8.6.3, <text:table-index-entry-template> 8.5.3, <text:table-of-content-entry-template> 8.3.3 and <text:user-index-entry-template> 8.7.3.`

The `<text:index-entry-page-number>` element has the following attribute: `text:style-name 19.880`.

The `<text:index-entry-page-number>` element has no child elements.

### 8.13.4 <text:index-entry-span>

The `<text:index-entry-span>` element represents a fixed string within an index entry.

The `<text:index-entry-span>` element is usable within the following elements: `<text:alphabetical-index-entry-template> 8.8.4, <text:bibliography-`

`<text:index-entry-template> 8.9.3, <text:illustration-index-entry-template> 8.4.3, <text:object-index-entry-template> 8.6.3, <text:table-index-entry-template> 8.5.3, <text:table-of-content-entry-template> 8.3.3 and <text:user-index-entry-template> 8.7.3.`

The `<text:index-entry-span>` element has the following attribute: `text:style-name 19.880`.

The `<text:index-entry-span>` element has no child elements.

The `<text:index-entry-span>` element has character data content.

### 8.13.5 `<text:index-entry-bibliography>`

The `<text:index-entry-bibliography>` element specifies the field of a bibliographic entry that will be displayed in an index.

The `<text:index-entry-bibliography>` element is usable within the following element: `<text:bibliography-entry-template> 8.9.3`.

The `<text:index-entry-bibliography>` element has the following attributes: `text:bibliography-data-field 19.762` and `text:style-name 19.880`.

The `<text:index-entry-bibliography>` element has no child elements.

### 8.13.6 `<text:index-entry-tab-stop>`

The `<text:index-entry-tab-stop>` element represents a tab stop within an index entry.

The `<text:index-entry-tab-stop>` element is usable within the following elements: `<text:alphabetical-index-entry-template> 8.8.4, <text:bibliography-entry-template> 8.9.3, <text:illustration-index-entry-template> 8.4.3, <text:object-index-entry-template> 8.6.3, <text:table-index-entry-template> 8.5.3, <text:table-of-content-entry-template> 8.3.3 and <text:user-index-entry-template> 8.7.3.`

The `<text:index-entry-tab-stop>` element has the following attributes: `style:leader-char 19.487`, `style:position 19.512`, `style:type 19.519` and `text:style-name 19.880`.

The `<text:index-entry-tab-stop>` element has no child elements.

### 8.13.7 `<text:index-entry-link-start>`

The `<text:index-entry-link-start>` element represents the start of a hyperlink index entry.

The `<text:index-entry-link-start>` element is usable within the following elements: `<text:illustration-index-entry-template> 8.4.3, <text:object-index-entry-template> 8.6.3, <text:table-index-entry-template> 8.5.3, <text:table-of-content-entry-template> 8.3.3 and <text:user-index-entry-template> 8.7.3.`

The `<text:index-entry-link-start>` element has the following attribute: `text:style-name 19.880`.

The `<text:index-entry-link-start>` element has no child elements.

### 8.13.8 `<text:index-entry-link-end>`

The `<text:index-entry-link-end>` element represents the end of a hyperlink index entry.

The `<text:index-entry-link-end>` element is usable within the following elements: `<text:illustration-index-entry-template>` 8.4.3, `<text:object-index-entry-template>` 8.6.3, `<text:table-index-entry-template>` 8.5.3, `<text:table-of-content-entry-template>` 8.3.3 and `<text:user-index-entry-template>` 8.7.3.

The `<text:index-entry-link-end>` element has the following attribute: `text:style-name` 19.880.

The `<text:index-entry-link-end>` element has no child elements.

# 9 Tables

## 9.1 Basic Tables

### 9.1.1 General

The representation of tables is based on the concept of grids of rows and columns. In markup, rows are partitioned into table cells. Columns are implied by taking together all cells with the same position within the rows.

Rows and columns appear in *row groups* and *column groups*. These groups specify whether to repeat a row or column on the next page.

**Note:** There is only one model for tables, although tables, such as for spreadsheets, may have additional capabilities when compared to tables in text documents.

Tables may be nested; that is, tables may occur within tables.

### 9.1.2 <table:table>

The `<table:table>` element is the root element for a table.

The `<table:table>` element is usable within the following elements: `<chart:chart> 11.1, <draw:caption> 10.3.11, <draw:circle> 10.3.8, <draw:connector> 10.3.10, <draw:custom-shape> 10.6.1, <draw:ellipse> 10.3.9, <draw:frame> 10.4.2, <draw:image> 10.4.4, <draw:line> 10.3.3, <draw:measure> 10.3.12, <draw:path> 10.3.7, <draw:polygon> 10.3.5, <draw:polyline> 10.3.4, <draw:rect> 10.3.2, <draw:regular-polygon> 10.3.6, <draw:text-box> 10.4.3, <office:spreadsheet> 3.7, <office:text> 3.4, <style:footer> 16.12, <style:footer-first> 16.13, <style:footer-left> 16.15, <style:header> 16.10, <style:header-first> 16.11, <style:header-left> 16.14, <table:covered-table-cell> 9.1.5, <table:dde-link> 14.7.4, <table:table-cell> 9.1.4, <text:deletion> 5.5.5, <text:index-body> 8.2.2, <text:index-title> 8.2.3, <text:note-body> 6.3.4 and <text:section> 5.4.`

The `<table:table>` element has the following attributes: `table:is-sub-table 19.659, table:name 19.677, table:print 19.697, table:print-ranges 19.698, table:protected 19.700, table:protection-key 19.701, table:protection-key-digest-algorithm 19.702, table:style-name 19.730, table:template-name 19.737, table:use-banding-columns-styles 19.741, table:use-banding-rows-styles 19.742, table:use-first-column-styles 19.743, table:use-first-row-styles 19.744, table:use-last-column-styles 19.745, table:use-last-row-styles 19.746 and xml:id 19.920.`

The `<table:table>` element has the following child elements: `<office:dde-source> 14.7.5, <office:forms> 13.2, <table:desc> 9.1.14, <table:named-expressions> 9.4.11, <table:scenario> 9.2.7, <table:shapes> 9.2.8, <table:table-column> 9.1.6, <table:table-column-group> 9.1.10, <table:table-columns> 9.1.12, <table:table-header-columns> 9.1.11, <table:table-header-rows> 9.1.7, <table:table-row> 9.1.3, <table:table-row-group> 9.1.9, <table:table-rows> 9.1.8, <table:table-source> 9.2.6, <table:title> 9.1.13, <text:soft-page-break> 5.6.`

### 9.1.3 <table:table-row>

The `<table:table-row>` element represents a row in a table. It contains elements that specify the cells of the table row.

The `<table:table-row>` element is usable within the following elements:

`<table:table>` 9.1.2, `<table:table-header-rows>` 9.1.7, `<table:table-row-group>` 9.1.9 and `<table:table-rows>` 9.1.8.

The `<table:table-row>` element has the following attributes: `table:default-cell-style-name` 19.619, `table:number-rows-repeated` 19.681, `table:style-name` 19.730, `table:visibility` 19.754 and `xml:id` 19.920.

The `<table:table-row>` element has the following child elements: `<table:covered-table-cell>` 9.1.5 and `<table:table-cell>` 9.1.4.

### 9.1.4 <table:table-cell>

The `<table:table-cell>` element represents a table cell. It is contained in a table row element. A table cell can contain paragraphs and other text content as well as nested tables. Table cells may span multiple columns and rows. Table cells may be empty.

The `<table:table-cell>` element is usable within the following element:

`<table:table-row>` 9.1.3.

The `<table:table-cell>` element has the following attributes: `office:boolean-value` 19.371, `office:currency` 19.373, `office:date-value` 19.374, `office:string-value` 19.383, `office:time-value` 19.386, `office:value` 19.388, `office:value-type` 19.389, `table:content-validation-name` 19.605, `table:formula` 19.646, `table:number-columns-repeated` 19.679, `table:number-columns-spanned` 19.680, `table:number-matrix-columns-spanned` 19.683, `table:number-matrix-rows-spanned` 19.684, `table:number-rows-spanned` 19.682, `table:protect` 19.699, `table:protected` 19.700, `table:style-name` 19.730, `xhtml:about` 19.911, `xhtml:content` 19.912, `xhtml:datatype` 19.913, `xhtml:property` 19.914 and `xml:id` 19.920.

The `<table:table-cell>` element has the following child elements: `<dr3d:scene>` 10.5.2, `<draw:a>` 10.4.12, `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:connector>` 10.3.10, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:g>` 10.3.15, `<draw:line>` 10.3.3, `<draw:measure>` 10.3.12, `<draw:page-thumbnail>` 10.3.14, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2, `<draw:regular-polygon>` 10.3.6, `<office:annotation>` 14.1, `<table:cell-range-source>` 9.3.1, `<table:detective>` 9.3.2, `<table:table>` 9.1.2, `<text:alphabetical-index>` 8.8, `<text:bibliography>` 8.9, `<text:change>` 5.5.8.4, `<text:change-end>` 5.5.8.3, `<text:change-start>` 5.5.8.2, `<text:h>` 5.1.2, `<text:illustration-index>` 8.4, `<text:list>` 5.3.1, `<text:numbered-paragraph>` 5.3.6, `<text:object-index>` 8.6, `<text:p>` 5.1.3, `<text:section>` 5.4, `<text:soft-page-break>` 5.6, `<text:table-index>` 8.5, `<text:table-of-content>` 8.3 and `<text:user-index>` 8.7.

### 9.1.5 <table:covered-table-cell>

The `<table:covered-table-cell>` element represents cells that are covered by a `<table:table-cell>` 9.1.4 element that spans multiple columns or rows. The `<table:covered-table-cell>` elements occupy the same position in document order as the cells spanned. There is one `<table:covered-table-cell>` element for each cell that is spanned.

The `<table:covered-table-cell>` is often used by spreadsheet applications, where covered cells contain content.

The `<table:covered-table-cell>` element is usable within the following element:  
`<table:table-row>` 9.1.3.

The `<table:covered-table-cell>` element has the following attributes:  
`office:boolean-value` 19.371, `office:currency` 19.373, `office:date-value` 19.374, `office:string-value` 19.383, `office:time-value` 19.386, `office:value` 19.388, `office:value-type` 19.389, `table:content-validation-name` 19.605, `table:formula` 19.646, `table:number-columns-repeated` 19.679, `table:protect` 19.699, `table:protected` 19.700, `table:style-name` 19.730, `xhtml:about` 19.911, `xhtml:content` 19.912, `xhtml:datatype` 19.913, `xhtml:property` 19.914 and `xml:id` 19.920.

The `<table:covered-table-cell>` element has the following child elements:  
`<dr3d:scene>` 10.5.2, `<draw:a>` 10.4.12, `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:connector>` 10.3.10, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:g>` 10.3.15, `<draw:line>` 10.3.3, `<draw:measure>` 10.3.12, `<draw:page-thumbnail>` 10.3.14, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2, `<draw:regular-polygon>` 10.3.6, `<office:annotation>` 14.1, `<table:cell-range-source>` 9.3.1, `<table:detective>` 9.3.2, `<table:table>` 9.1.2, `<text:alphabetical-index>` 8.8, `<text:bibliography>` 8.9, `<text:change>` 5.5.8.4, `<text:change-end>` 5.5.8.3, `<text:change-start>` 5.5.8.2, `<text:h>` 5.1.2, `<text:illustration-index>` 8.4, `<text:list>` 5.3.1, `<text:numbered-paragraph>` 5.3.6, `<text:object-index>` 8.6, `<text:p>` 5.1.3, `<text:section>` 5.4, `<text:soft-page-break>` 5.6, `<text:table-index>` 8.5, `<text:table-of-content>` 8.3 and `<text:user-index>` 8.7.

## 9.1.6 `<table:table-column>`

The `<table:table-column>` element specifies properties for one or more adjacent columns in a table.

The `<table:table-column>` element is usable within the following elements:  
`<table:table>` 9.1.2, `<table:table-column-group>` 9.1.10, `<table:table-columns>` 9.1.12 and `<table:table-header-columns>` 9.1.11.

The `<table:table-column>` element has the following attributes: `table:default-cell-style-name` 19.619, `table:number-columns-repeated` 19.679, `table:style-name` 19.730, `table:visibility` 19.754 and `xml:id` 19.920.

The `<table:table-column>` element has no child elements.

## 9.1.7 `<table:table-header-rows>`

The `<table:table-header-rows>` element represents header rows in a table. It is composed of adjacent `<table:table-row>` 9.1.3 elements.

**Note:** Use of the `<table:table-header-rows>` element increases accessibility. The use of styles to designate row headers diminishes accessibility.

If a table does not fit on a single page, table rows that are included in a `<table:table-header-rows>` element are automatically repeated on every page.

A table shall not contain more than one `<table:table-header-rows>` element, except that the table contains grouped rows defined by one or more `<table:table-row-group>` 9.1.9 elements. In this case, the table and each distinct group may contain one `<table:table-header-rows>` element, if and only if the table rows contained in the `<table:table-header-rows>` elements are adjacent.

Consumers that do not support header rows shall process header rows as non-header rows.

The `<table:table-header-rows>` element is usable within the following elements:  
`<table:table>` 9.1.2 and `<table:table-row-group>` 9.1.9.

The `<table:table-header-rows>` element has no attributes.

The `<table:table-header-rows>` element has the following child elements:  
`<table:table-row>` 9.1.3 and `<text:soft-page-break>` 5.6.

## 9.1.8 `<table:table-rows>`

The `<table:table-rows>` element specifies a sequence of one or more `<table:table-row>` 9.1.3 elements with optional `<text:soft-page-break>` 9.1.3 elements. The interpretation of the sequence is the same as if the `<table:table-rows>` start and end tags were not present.

The `<table:table-rows>` element is usable within the following elements:  
`<table:table>` 9.1.2 and `<table:table-row-group>` 9.1.9.

The `<table:table-rows>` element has no attributes.

The `<table:table-rows>` element has the following child elements: `<table:table-row>` 9.1.3 and `<text:soft-page-break>` 5.6.

## 9.1.9 `<table:table-row-group>`

The `<table:table-row-group>` element groups adjacent table rows. Every row group can contain header rows, and nested row groups. A row group can be visible or hidden.

The `<table:table-row-group>` element is usable within the following elements:  
`<table:table>` 9.1.2 and `<table:table-row-group>` 9.1.9.

The `<table:table-row-group>` element has the following attribute: `table:display` 19.621.

The `<table:table-row-group>` element has the following child elements:  
`<table:table-header-rows>` 9.1.7, `<table:table-row>` 9.1.3, `<table:table-row-group>` 9.1.9, `<table:table-rows>` 9.1.8, `<text:soft-page-break>` 5.6.

## 9.1.10 `<table:table-column-group>`

The `<table:table-column-group>` element groups adjacent table columns. Every column group can contain columns, header columns and nested column groups. A column group can be visible or hidden.

If a set of header columns and a column group overlap, the header column group breaks the column header set. That is, the `<table:table-column-group>` may contain `<table:table-header-columns>` elements, but not vice versa.

The `<table:table-column-group>` element is usable within the following elements:  
`<table:table>` 9.1.2 and `<table:table-column-group>` 9.1.10.

The `<table:table-column-group>` element has the following attribute: `table:display` 19.621.

The `<table:table-column-group>` element has the following child elements:  
`<table:table-column>` 9.1.6, `<table:table-column-group>` 9.1.10, `<table:table-columns>` 9.1.12, `<table:table-header-columns>` 9.1.11.

### 9.1.11 <table:table-header-columns>

The `<table:table-header-columns>` element represents column headers in a table. It is composed of adjacent `<table:table-column>` 9.1.6 elements.

**Note:** Use of the `<table:table-header-columns>` element increases accessibility. The use of styles to designate header columns diminishes accessibility.

If a table does not fit on a single page, table columns that are included in a `<table:table-header-columns>` element are automatically repeated on every page.

A table shall not contain more than one `<table:table-header-columns>` element, except that the table contains grouped rows defined by one or more `<table:table-column-group>` 9.1.10 elements. In this case, the table and each distinct group may contain one `<table:table-header-columns>` element, if and only if the table columns contained in the `<table:table-header-columns>` elements are adjacent.

Consumers that do not support header columns shall process header column as non-header columns.

The `<table:table-header-columns>` element is usable within the following elements: `<table:table>` 9.1.2 and `<table:table-column-group>` 9.1.10.

The `<table:table-header-columns>` element has no attributes.

The `<table:table-header-columns>` element has the following child element: `<table:table-column>` 9.1.6.

### 9.1.12 <table:table-columns>

The `<table:table-columns>` element specifies a sequence of one or more `<table:table-column>` 9.1.6 elements. The interpretation of the sequence is the same as if the `<table:table-columns>` start and end tags were not present.

The `<table:table-columns>` element is usable within the following elements: `<table:table>` 9.1.2 and `<table:table-column-group>` 9.1.10.

The `<table:table-columns>` element has no attributes.

The `<table:table-columns>` element has the following child element: `<table:table-column>` 9.1.6.

### 9.1.13 <table:title>

The `<table:title>` element specifies a title for a table. The element shall not be used if the parent `<table:table>` 9.1.2 element is contained within a `<draw:frame>` 10.4.2 element. In that case, the table's title is specified by `<svg:title>` 10.3.17 child element of `<draw:frame>`.

The `<table:title>` element is usable within the following element: `<table:table>` 9.1.2.

The `<table:title>` element has no attributes.

The `<table:title>` element has no child elements.

The `<table:title>` element has character data content.

### 9.1.14 <table:desc>

The `<table:desc>` element specifies a prose description of a table that may be used to support accessibility. See appendix. The element shall not be used if the parent `<table:table>` element is contained within a `<draw:frame>` 10.4.2 element. In that case,

the table's description is specified by `<svg:desc>` 10.3.18 child element of `<draw:frame>`.

The `<table:desc>` element is usable within the following element: `<table:table>` 9.1.2.

The `<table:desc>` element has no attributes.

The `<table:desc>` element has no child elements.

The `<table:desc>` element has character data content.

## 9.2 Advanced Tables

### 9.2.1 Referencing Table Cells

To reference cells in a table, cell addresses are used. Cell addresses are constructed as follows:

- 1) The name of the table.
- 2) A dot “.” (U+002E, FULL STOP).
- 3) An alphabetic value representing the column. The letter A represents column 1, B represents column 2, and so on. AA represents column 27, AB represents column 28, and so on.
- 4) A numeric value representing the row. The number 1 represents the first row, the number 2 represents the second row, and so on.

This means that A1 represents the cell in column 1 and row 1. B1 represents the cell in column 2 and row 1. A2 represents the cell in column 1 and row 2.

The structure of the address of a cell in a subtable is as follows:

- 1) The address of the cell that contains the subtable.
- 2) A dot “.” (U+002E, FULL STOP).
- 3) The address of the cell in the subtable.

If the name of the table contains blanks, dots “.” (U+002E, FULL STOP) or apostrophes “” (U+0027, APOSTROPHE), the name shall be quoted with apostrophes “” (U+0027, APOSTROPHE). Any apostrophes in the name shall be escaped by doubling the “” (U+0027, APOSTROPHE) character.

The behavior of a consumer when a cell is referenced but not declared is implementation-dependent.

### 9.2.2 Absolute and relative cell addressing

Cells can be referenced by using either absolute addresses or relative addresses. When an operation is performed on a table cell, absolute cell references do not change. Relative cell references are adapted to the address of the target cell of a copy operation.

To create an absolute address, a dollar sign “\$” (U+0024, DOLLAR SIGN) shall be placed before each table name, column reference, and row reference. Absolute and relative references may be mixed within a single cell address. Absolute addresses shall contain a table name.

When differentiation between absolute and relative addresses is not necessary, the dollar signs in absolute addresses may be omitted.

### 9.2.3 Cell Range Address

A cell range is a number of adjacent cells forming a rectangular shape. The rectangle stretches from the cell on the logical top left to the cell on the logical bottom right.

A cell range address references a cell range. A cell range address consists of:

- 1) The address of the cell at the top left of the range.
- 2) A colon ":" (U+003A, COLON).
- 3) The address of the cell at the bottom right of the range.

The smallest range is a single cell. The range address of a single cell is the same as the cell address.

#### 9.2.4 Column and Row Range Addresses

Column and row addresses are cell range addresses that reference entire rows or columns.

The syntax of a row address is the same as a cell address, except the alphabetic values that indicate the column are omitted.

The syntax of a column address is the same as a cell address, except the numeric values that indicate the row are omitted.

#### 9.2.5 Cell Range Address List

A cell range address list is a list of cell range addresses and cell addresses. Each item in the list is separated by white space. If any table name used in the list contains a " " (U+0020, SPACE) character, the table name is quoted within apostrophes "" ( U+0027, APOSTROPHE).

#### 9.2.6 <table:table-source>

The <table:table-source> element represents a link to a table that provides a source for a table.

The <table:table-source> element is usable within the following element:  
<table:table> 9.1.2.

The <table:table-source> element has the following attributes: table:filter-name 19.642, table:filter-options 19.643, table:mode 19.675, table:refresh-delay 19.705, table:table-name 19.734, xlink:actuate 19.915, xlink:href 19.916 and xlink:type 19.919.

The <table:table-source> element has no child elements.

#### 9.2.7 <table:scenario>

The <table:scenario> element defines a scenario for display on a table. A table represented by a <table:table> 9.1.2 element which contains a <table:scenario> child element is called a scenario table.

A scenario is one or more cell ranges of a table whose contents replaces the contents of the cell ranges in the nearest non-scenario table preceding the scenario table in document order. If multiple scenarios are defined for a cell range, a user may choose between the scenarios. The names of the scenarios are the same as the names of tables which contain the <table:scenario> elements.

Display of the border of a scenario table is controlled by its table:display-border attribute. 19.622

**Note:** Display of the border of a scenario is the only visible indication that the displayed data is the result of the application of a scenario to the data in the table.

Only one scenario table can be applied to a range at one time.

**Note:** A table can have multiple scenarios applied to it so long as the ranges of application do not overlap.

The `<table:scenario>` element is usable within the following element: `<table:table>` 9.1.2.

The `<table:scenario>` element has the following attributes: `table:border-color` 19.592, `table:comment` 19.599, `table:copy-back` 19.606, `table:copy-formulas` 19.607, `table:copy-styles` 19.608, `table:display-border` 19.622, `table:is-active` 19.656, `table:protected` 19.700 and `table:scenario-ranges` 19.709.

The `<table:scenario>` element has no child elements.

## 9.2.8 `<table:shapes>`

The `<table:shapes>` element contains all the elements that represent graphic shapes that are anchored on a table where this element occurs.

The `<table:shapes>` element is usable within the following element: `<table:table>` 9.1.2.

The `<table:shapes>` element has no attributes.

The `<table:shapes>` element has the following child elements: `<dr3d:scene>` 10.5.2, `<draw:a>` 10.4.12, `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:connector>` 10.3.10, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:g>` 10.3.15, `<draw:line>` 10.3.3, `<draw:measure>` 10.3.12, `<draw:page-thumbnail>` 10.3.14, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2 and `<draw:regular-polygon>` 10.3.6.

## 9.3 Advanced Table Cells

### 9.3.1 `<table:cell-range-source>`

The `<table:cell-range-source>` element represents a database range or named range of another file in a cell range. The first cell in the cell range contains this element. A cell range can be linked to a database range or named range of another file.

The `<table:cell-range-source>` element is usable within the following elements: `<table:covered-table-cell>` 9.1.5 and `<table:table-cell>` 9.1.4.

The `<table:cell-range-source>` element has the following attributes: `table:filter-name` 19.642, `table:filter-options` 19.643, `table:last-column-spanned` 19.662, `table:last-row-spanned` 19.664, `table:name` 19.677, `table:refresh-delay` 19.705, `xlink:actuate` 19.915, `xlink:href` 19.916 and `xlink:type` 19.919.

The `<table:cell-range-source>` element has no child elements.

### 9.3.2 `<table:detective>`

The `<table:detective>` element is a container for `<table:operation>` 9.3.3 and `<table:highlighted-range>` 9.3.4 elements, providing information about what relationships between the current cell and other cells are revealed in the presentation of the table.

The `<table:detective>` element is usable within the following elements: `<table:covered-table-cell>` 9.1.5 and `<table:table-cell>` 9.1.4.

The `<table:detective>` element has no attributes.

The `<table:detective>` element has the following child elements: `<table:highlighted-range>` 9.3.4 and `<table:operation>` 9.3.3.

### 9.3.3 <table:operation>

The `<table:operation>` element specifies the name of a detective operation that leads to the discovery of relationships between cells (`table:name` 19.677) and the order in which those operations are applied (`table:index` 19.655). Once relationships between cells have been discovered, those cells are highlighted to show those relationships.

**Note:** A single cell may be the subject of one or more detective operations.

The `<table:operation>` element is usable within the following element:  
`<table:detective>` 9.3.2.

The `<table:operation>` element has the following attributes: `table:index` 19.655 and `table:name` 19.677.

The `<table:operation>` element has no child elements.

### 9.3.4 <table:highlighted-range>

The `<table:highlighted-range>` element specifies a cell range that is highlighted in the UI either because of detective operations defined by a `<table:operation>` 9.3.3 element or because it contains an error or invalid data.

The nature of the highlighting is implementation-dependent.

This element represents the state of the document at the time the detective operations or error conditions were calculated. It need not accurately represent the current state of the document.

The `<table:highlighted-range>` element is usable within the following element:  
`<table:detective>` 9.3.2.

The `<table:highlighted-range>` element has the following attributes: `table:cell-range-address` 19.597, `table:contains-error` 19.603, `table:direction` 19.620 and `table:marked-invalid` 19.668.

The `<table:highlighted-range>` element has no child elements.

## 9.4 Spreadsheet Document Content

### 9.4.1 <table:calculation-settings>

The `<table:calculation-settings>` element is a container for settings that affect the calculation of formulae.

The `<table:calculation-settings>` element is usable within the following elements: `<office:chart>` 3.8, `<office:drawing>` 3.5, `<office:presentation>` 3.6, `<office:spreadsheet>` 3.7 and `<office:text>` 3.4.

The `<table:calculation-settings>` element has the following attributes: `table:automatic-find-labels` 19.589, `table:case-sensitive` 19.594, `table:null-year` 19.678, `table:precision-as-shown` 19.696, `table:search-criteria-must-apply-to-whole-cell` 19.711, `table:use-regular-expressions` 19.748 and `table:use-wildcards` 19.749.

The `<table:calculation-settings>` element has the following child elements: `<table:iteration>` 9.4.3 and `<table:null-date>` 9.4.2.

### 9.4.2 <table:null-date>

The `<table:null-date>` element specifies the null date for a spreadsheet document. The null date is the date that results in the value "0" if a date value is converted into a numeric value. The null date is specified in the element's `table:date-value` attribute.

**Note:** Commonly used values are 1899-12-30, 1900-01-01, and 1904-01-01.

The `<table:null-date>` element is usable within the following element:  
`<table:calculation-settings>` 9.4.1.

The `<table:null-date>` element has the following attributes: `table:date-value 19.618` and `table:value-type 19.753`.

The `<table:null-date>` element has no child elements.

### 9.4.3 `<table:iteration>`

The `<table:iteration>` element enables formulas with iterative (or cyclic) references to be calculated after a specific number of iterations. If iterative calculations are not enabled, a formula expression containing a cyclic cell reference returns an error.

If the `table:status` attribute has the value `enable` and iterating through the cyclic dependencies does not deliver a result that satisfies the condition specified by the `table:maximum-difference` attribute within a number of iterations specified by the `table:steps` attribute for all cells in the cyclic dependencies, an error is returned.

The `<table:iteration>` element is usable within the following element:  
`<table:calculation-settings>` 9.4.1.

The `<table:iteration>` element has the following attributes: `table:maximum-difference 19.670`, `table:status 19.726` and `table:steps 19.728`.

The `<table:iteration>` element has no child elements.

### 9.4.4 `<table:content-validations>`

The `<table:content-validations>` element contains all the validation rules for the content of table cells.

The `<table:content-validations>` element is usable within the following elements: `<office:chart> 3.8`, `<office:drawing> 3.5`, `<office:presentation> 3.6`, `<office:spreadsheet> 3.7` and `<office:text> 3.4`.

The `<table:content-validations>` element has no attributes.

The `<table:content-validations>` element has the following child element:  
`<table:content-validation>` 9.4.5.

### 9.4.5 `<table:content-validation>`

The `<table:content-validation>` element specifies a validation rule for table cell content.

The name of a validation rule is stored in the `table:name` attribute of a `<table:content-validation>` element.

The `<table:content-validation>` element is usable within the following element:  
`<table:content-validations>` 9.4.4.

The `<table:content-validation>` element has the following attributes: `table:allow-empty-cell 19.587`, `table:base-cell-address 19.590`, `table:condition 19.600`, `table:display-list 19.625` and `table:name 19.677`.

The `<table:content-validation>` element has the following child elements: `<office:event-listeners>` 10.3.19, `<table:error-macro>` 9.4.8, `<table:error-message>` 9.4.7 and `<table:help-message>` 9.4.6.

#### 9.4.6 `<table:help-message>`

The `<table:help-message>` element specifies a message to display if a user selects a cell.

The `<table:help-message>` element is usable within the following element: `<table:content-validation>` 9.4.5.

The `<table:help-message>` element has the following attributes: `table:display` 19.621 and `table:title` 19.738.

The `<table:help-message>` element has the following child element: `<text:p>` 5.1.3.

#### 9.4.7 `<table:error-message>`

The `<table:error-message>` element specifies a message to display when the `table:condition` 19.600 attribute value on the parent `<table:content-validation>` element evaluates to `false`."

The `<table:error-message>` element is usable within the following element: `<table:content-validation>` 9.4.5.

The `<table:error-message>` element has the following attributes: `table:display` 19.621, `table:message-type` 19.674 and `table:title` 19.738.

The `<table:error-message>` element has the following child element: `<text:p>` 5.1.3.

#### 9.4.8 `<table:error-macro>`

The `<table:error-macro>` element specifies if a macro should be executed upon an attempt to enter invalid content.

A macro is specified by an `<office:event-listeners>` element. 10.3.19 The event name shall be one that specifies an event that is called on invalid user input.

The `<table:error-macro>` element is usable within the following element: `<table:content-validation>` 9.4.5.

The `<table:error-macro>` element has the following attribute: `table:execute` 19.638.

The `<table:error-macro>` element has no child elements.

#### 9.4.9 `<table:label-range>`

The `<table:label-range>` element specifies a label cell range which contains labels, and a data cell range which specifies the rows or columns whose content is referenced by the labels.

There are two types of label ranges:

- Label ranges for columns.
- Label ranges for rows.

Data and label ranges should match in both height and vertical position for row orientation, or in both width and horizontal position for column orientation.

The `<table:label-range>` element is usable within the following element: `<table:label-ranges>` 9.4.10.

The `<table:label-range>` element has the following attributes: `table:data-cell-range-address 19.612`, `table:label-cell-range-address 19.660` and `table:orientation 19.690`.

The `<table:label-range>` element has no child elements.

#### 9.4.10 `<table:label-ranges>`

The `<table:label-ranges>` element contains all the `<table:label-range>` elements in a document.

The `<table:label-ranges>` element is usable within the following elements: `<office:chart> 3.8`, `<office:drawing> 3.5`, `<office:presentation> 3.6`, `<office:spreadsheet> 3.7` and `<office:text> 3.4`.

The `<table:label-ranges>` element has no attributes.

The `<table:label-ranges>` element has the following child element: `<table:label-range> 9.4.9`.

#### 9.4.11 `<table:named-expressions>`

The `<table:named-expressions>` element contains assignments of names to expressions.

The following expressions may have names:

- Cell ranges.
- Other expressions.

If the `<table:named-expressions>` element is used with a `<table:table>` element, the scope of the named expressions are local to that table element.

The `<table:named-expressions>` element is usable within the following elements: `<office:chart> 3.8`, `<office:drawing> 3.5`, `<office:presentation> 3.6`, `<office:spreadsheet> 3.7`, `<office:text> 3.4` and `<table:table> 9.1.2`.

The `<table:named-expressions>` element has no attributes.

The `<table:named-expressions>` element has the following child elements: `<table:named-expression> 9.4.13` and `<table:named-range> 9.4.12`.

#### 9.4.12 `<table:named-range>`

The `<table:named-range>` element specifies a cell range that has a name assigned.

The `<table:named-range>` element is usable within the following element: `<table:named-expressions> 9.4.11`.

The `<table:named-range>` element has the following attributes: `table:base-cell-address 19.590`, `table:cell-range-address 19.597`, `table:name 19.677` and `table:range-usable-as 19.704`.

The `<table:named-range>` element has no child elements.

#### 9.4.13 `<table:named-expression>`

The `<table:named-expression>` element represents an expression with a name.

The `<table:named-expression>` element is usable within the following element: `<table:named-expressions> 9.4.11`.

The `<table:database-ranges>` element has the following attributes: `table:base-cell-address` 19.590, `table:expression` 19.639 and `table:name` 19.677.

The `<table:database-ranges>` element has no child elements.

#### 9.4.14 `<table:database-ranges>`

The `<table:database-ranges>` element contains all the `<table:database-range>` elements in a document.

The `<table:database-ranges>` element is usable within the following elements: `<office:chart>` 3.8, `<office:drawing>` 3.5, `<office:presentation>` 3.6, `<office:spreadsheet>` 3.7 and `<office:text>` 3.4.

The `<table:database-ranges>` element has no attributes.

The `<table:database-ranges>` element has the following child element: `<table:database-range>` 9.4.15.

#### 9.4.15 `<table:database-range>`

The `<table:database-range>` element defines a single database range. A database range is a named area in a table upon which database operations are performed.

The `<table:database-range>` element is usable within the following element: `<table:database-ranges>` 9.4.14.

The `<table:database-range>` element has the following attributes: `table:contains-header` 19.604, `table:display-filter-buttons` 19.624, `table:has-persistent-data` 19.651, `table:is-selection` 19.658, `table:name` 19.677, `table:on-update-keep-size` 19.686, `table:on-update-keep-styles` 19.687, `table:orientation` 19.690, `table:refresh-delay` 19.705 and `table:target-range-address` 19.736.

The `<table:database-range>` element has the following child elements:

`<table:database-source-query>` 9.4.18, `<table:database-source-sql>` 9.4.16, `<table:database-source-table>` 9.4.17, `<table:filter>` 9.5.2, `<table:sort>` 9.4.19 and `<table:subtotal-rules>` 9.4.21.

#### 9.4.16 `<table:database-source-sql>`

The `<table:database-source-sql>` element specifies an [SQL] database that contains the source data for a database range.

The `<table:database-source-sql>` element is usable within the following elements: `<table:database-range>` 9.4.15 and `<table:database-source-table>` 9.4.17.

The `<table:database-source-sql>` element has the following attributes: `table:database-name` 19.613, `table:parse-sql-statement` 19.693 and `table:sql-statement` 19.720.

The `<table:database-source-sql>` element has no child elements.

#### 9.4.17 `<table:database-source-table>`

The `<table:database-source-table>` element specifies the database and table that acts as the source data for a database range.

The `<table:database-source-table>` element is usable within the following elements: `<table:database-range>` 9.4.15 and `<table:database-source-sql>` 9.4.16.

The `<table:database-source-table>` element has the following attributes: `table:database-name` 19.613 and `table:database-table-name` 19.614.

The `<table:database-source-table>` element has no child elements.

#### 9.4.18 `<table:database-source-query>`

The `<table:database-source-query>` element specifies the query that produces the source data for a database range.

The `<table:database-source-query>` element is usable within the following elements: `<table:data-pilot-table>` 9.6.3 and `<table:database-range>` 9.4.15.

The `<table:database-source-query>` element has the following attributes: `table:database-name` 19.613 and `table:query-name` 19.703.

The `<table:database-source-query>` element has no child elements.

#### 9.4.19 `<table:sort>`

The `<table:sort>` element `<table:sort>` specifies the sort keys to be applied to a database range.

The `<table:sort>` element is usable within the following element: `<table:database-range>` 9.4.15.

The `<table:sort>` element has the following attributes: `table:algorithm` 19.585, `table:bind-styles-to-content` 19.591, `table:case-sensitive` 19.594, `table:country` 19.610, `table:embedded-number-behavior` 19.628, `table:language` 19.661, `table:rfc-language-tag` 19.707, `table:script` 19.710 and `table:target-range-address` 19.736.

The `<table:sort>` element has the following child element: `<table:sort-by>` 9.4.20.

#### 9.4.20 `<table:sort-by>`

The `<table:sort-by>` element specifies a key or field to sort, the data type of this field, and how to sort it for a database range.

The `<table:sort-by>` element is usable within the following element: `<table:sort>` 9.4.19.

The `<table:sort-by>` element has the following attributes: `table:data-type` 19.615, `table:field-number` 19.641 and `table:order` 19.689.

The `<table:sort-by>` element has no child elements.

#### 9.4.21 `<table:subtotal-rules>`

The `<table:subtotal-rules>` element is a container for `<table:subtotal-rule>` elements which specify the calculation of provisional results (called subtotals) for a database range. Cells of the same field form a group. A provisional result is calculated and display at the end of each group.

The `<table:subtotal-rules>` element is usable within the following element: `<table:database-range>` 9.4.15.

The `<table:subtotal-rules>` element has the following attributes: `table:bind-styles-to-content` 19.591, `table:case-sensitive` 19.594 and `table:page-breaks-on-group-change` 19.691.

The `<table:subtotal-rules>` element has the following child elements: `<table:sort-groups>` 9.4.22 and `<table:subtotal-rule>` 9.4.23.

### 9.4.22 <table:sort-groups>

The `<table:sort-groups>` element specifies a sort order for columns or rows of a table.

The `<table:sort-groups>` element is usable within the following element:  
`<table:subtotal-rules>` 9.4.21.

The `<table:sort-groups>` element has the following attributes: `table:data-type` 19.615 and `table:order` 19.689.

The `<table:sort-groups>` element has no child elements.

### 9.4.23 <table:subtotal-rule>

The `<table:subtotal-rule>` element specifies the calculation of subtotals for columns or rows of a table. As the container element for `<table:subtotal-field>`, the calculation specified by this element may use provisional results in its calculation.

The `<table:subtotal-rule>` element is usable within the following element:  
`<table:subtotal-rules>` 9.4.21.

The `<table:subtotal-rule>` element has the following attribute: `table:group-by-field-number` 19.649.

The `<table:subtotal-rule>` element has the following child element:  
`<table:subtotal-field>` 9.4.24.

### 9.4.24 <table:subtotal-field>

The `<table:subtotal-field>` element specifies the field number and the function that is applied to the content of that field to calculate a provisional result.

The `<table:subtotal-field>` element is usable within the following element:  
`<table:subtotal-rule>` 9.4.23.

The `<table:subtotal-field>` element has the following attributes: `table:field-number` 19.641 and `table:function` 19.647.

The `<table:subtotal-field>` element has no child elements.

## 9.5 Filters

### 9.5.1 General

Filter elements specify conditions that, if not met, render portions of a table invisible.

### 9.5.2 <table:filter>

The `<table:filter>` element is a container for elements that specify filters for data contained in database ranges or data pilot tables. The filters are applied to all rows in the database range or the data pilot table. Rows where one or more filter conditions do not evaluate to true are made invisible.

The `<table:filter>` element is usable within the following elements:  
`<table:database-range>` 9.4.15 and `<table:source-cell-range>` 9.6.5.

The `<table:filter>` element has the following attributes: `table:condition-source` 19.601, `table:condition-source-range-address` 19.602, `table:display-duplicates` 19.623 and `table:target-range-address` 19.736.

The `<table:filter>` element has the following child elements: `<table:filter-and>` 9.5.3, `<table:filter-condition>` 9.5.5 and `<table:filter-or>` 9.5.4.

### 9.5.3 <table:filter-and>

The `<table:filter-and>` element specifies that the logical operator AND is applied to conditions specified by the child elements of this element.

The `<table:filter-and>` element is usable within the following elements:

`<table:filter>` 9.5.2 and `<table:filter-or>` 9.5.4.

The `<table:filter-and>` element has no attributes.

The `<table:filter-and>` element has the following child elements: `<table:filter-condition>` 9.5.5 and `<table:filter-or>` 9.5.4.

### 9.5.4 <table:filter-or>

The `<table:filter-or>` element specifies that the logical operator OR is applied to conditions specified by the child elements of this element.

The `<table:filter-or>` element is usable within the following elements:

`<table:filter>` 9.5.2 and `<table:filter-and>` 9.5.3.

The `<table:filter-or>` element has no attributes.

The `<table:filter-or>` element has the following child elements: `<table:filter-and>` 9.5.3 and `<table:filter-condition>` 9.5.5.

### 9.5.5 <table:filter-condition>

The `<table:filter-condition>` element specifies a single condition to apply in a filter operation.

The filter condition is either specified

- by the `table:operator` 19.688, `table:value` 19.752 and `table:data-type` 19.615 attributes or
- a set of `<table:filter-set-item>` 9.5.6 child elements as a set of values.

If one or more of `<table:filter-set-item>` elements are present as children of the `<table:filter-condition>` element, the `table:operator`, `table:value` and `table:data-type` attributes are ignored.

**Note:** To improve backward compatibility with OpenDocument 1.1 documents, if `<table:filter-set-item>` elements are included, set:

- `table:operator: "=";`
- `table:value: value of the first <table:filter-set-item> child element;`
- `table:data-type: "text".`

The `<table:filter-condition>` element is usable within the following elements:

`<table:filter>` 9.5.2, `<table:filter-and>` 9.5.3 and `<table:filter-or>` 9.5.4.

The `<table:filter-condition>` element has the following attributes: `table:case-sensitive` 19.594, `table:data-type` 19.615, `table:field-number` 19.641, `table:operator` 19.688, `table:value` 19.752.

The `<table:filter-condition>` element has the following child element:

`<table:filter-set-item>` 9.5.6.

### 9.5.6 <table:filter-set-item>

The `<table:filter-set-item>` element specifies a single value used in a filter condition where the `table:operator` attribute is set to `"="`, and the `table:data-type` attribute to

text. The filter sub-condition in which the element is included evaluates to true for a specified cell, if, and only if, the value of the cell is equal to one of the values in the set specified by the `<table:filter>` element.

The `<table:filter-set-item>` element is usable within the following element:  
`<table:filter-condition>` 9.5.5.

The `<table:filter-set-item>` element has the following attribute: `table:value` 19.752.

The `<table:filter-set-item>` element has no child elements.

## 9.6 Data Pilot Tables

### 9.6.1 General

Data pilot tables enable users to rearrange data from data sources so as to create new relationships between data elements or to view data from a variety of perspectives. This is in contrast to scenario tables 9.2.7, where the data elements remain fixed but the values and operations on them are varied to illustrate different outcomes.

The behavior of a data pilot table is specified by fields, where each field has a name and an orientation. The category columns are specified by fields with the orientation “row” or “column” and data columns are specified by fields that have the orientation “data”.

A third type of fields are data layout fields. Data layout fields are not connected to a column in the source table, but are used to change the layout of the data pilot table.

The order in which fields are specified determines the order in which the data of category columns are grouped and results are displayed.

### 9.6.2 `<table:data-pilot-tables>`

The `<table:data-pilot-tables>` element contains all the data pilot tables within a document.

The `<table:data-pilot-tables>` element is usable within the following elements:  
`<office:chart>` 3.8, `<office:drawing>` 3.5, `<office:presentation>` 3.6, `<office:spreadsheet>` 3.7 and `<office:text>` 3.4.

The `<table:data-pilot-tables>` element has no attributes.

The `<table:data-pilot-tables>` element has the following child element:  
`<table:data-pilot-table>` 9.6.3.

### 9.6.3 `<table:data-pilot-table>`

The `<table:data-pilot-table>` element contains the elements that define the sources of a data pilot table.

The `<table:data-pilot-table>` element is usable within the following element:  
`<table:data-pilot-tables>` 9.6.2.

The `<table:data-pilot-table>` element has the following attributes:  
`table:application-data` 19.588, `table:buttons` 19.593, `table:drill-down-on-double-click` 19.627, `table:grand-total` 19.648, `table:identify-categories` 19.653, `table:ignore-empty-rows` 19.654, `table:name` 19.677, `table:show-filter-button` 19.715 and `table:target-range-address` 19.736.

The `<table:data-pilot-table>` element has the following child elements:  
`<table:data-pilot-field>` 9.6.7, `<table:database-source-query>` 9.4.18, `<table:database-source-sql>` 9.4.16, `<table:database-source-table>` 9.4.17, `<table:source-cell-range>` 9.6.5 and `<table:source-service>` 9.6.6.

## 9.6.4 Data Pilot Table Source

The source of a data pilot table is specified by one of the following elements:

- <table:database-source-query> 9.4.18
- <table:database-source-sql> 9.4.16
- <table:database-source-table> 9.4.17
- <table:source-cell-range> 9.6.5
- <table:source-service> 9.6.6

## 9.6.5 <table:source-cell-range>

The <table:source-cell-range> element specifies a cell range as a source of data for a data pilot table. A filter may be specified by a <table:filter> child element that is applied prior to return of data to a data pilot table.

The <table:source-cell-range> element is usable within the following element:  
<table:data-pilot-table> 9.6.3.

The <table:source-cell-range> element has the following attributes: table:cell-range-address 19.597 and table:name 19.677.

The <table:source-cell-range> element has the following child element:  
<table:filter> 9.5.2.

## 9.6.6 <table:source-service>

The <table:source-service> element specifies a service that returns data to a data pilot table.

The <table:source-service> element is usable within the following element:  
<table:data-pilot-table> 9.6.3.

The <table:source-service> element has the following attributes: table:name 19.677, table:object-name 19.685, table:password 19.694, table:source-name 19.719 and table:user-name 19.751.

The <table:source-service> element has no child elements.

## 9.6.7 <table:data-pilot-field>

The <table:data-pilot-field> element specifies a field for a data pilot table.

The <table:data-pilot-field> element is usable within the following element:  
<table:data-pilot-table> 9.6.3.

The <table:data-pilot-field> element has the following attributes: table:function 19.647, table:is-data-layout-field 19.657, table:orientation 19.690, table:selected-page 19.712, table:source-field-name 19.718 and table:used-hierarchy 19.750.

The <table:data-pilot-field> element has the following child elements:  
<table:data-pilot-field-reference> 9.6.16, <table:data-pilot-groups> 9.6.17 and <table:data-pilot-level> 9.6.8.

## 9.6.8 <table:data-pilot-level>

The <table:data-pilot-level> element acts as a container for elements that specify addition information about a data pilot field.

The `<table: data-pilot-level>` element is usable within the following element:  
`<table: data-pilot-field>` 9.6.7.

The `<table: data-pilot-level>` element has the following attribute: `table: show-empty` 19.714.

The `<table: data-pilot-level>` element has the following child elements:  
`<table: data-pilot-display-info>` 9.6.13, `<table: data-pilot-layout-info>` 9.6.15, `<table: data-pilot-members>` 9.6.11, `<table: data-pilot-sort-info>` 9.6.14 and `<table: data-pilot-subtotals>` 9.6.9.

## 9.6.9 `<table: data-pilot-subtotals>`

The `<table: data-pilot-subtotals>` element contains the `<table: data-pilot-subtotal>` elements for a data pilot table.

The `<table: data-pilot-subtotals>` element is usable within the following element:  
`<table: data-pilot-level>` 9.6.8.

The `<table: data-pilot-subtotals>` element has no attributes.

The `<table: data-pilot-subtotals>` element has the following child element:  
`<table: data-pilot-subtotal>` 9.6.10.

## 9.6.10 `<table: data-pilot-subtotal>`

The `<table: data-pilot-subtotal>` element contains the results of a single subtotal calculation.

The `<table: data-pilot-subtotal>` element is usable within the following element:  
`<table: data-pilot-subtotals>` 9.6.9.

The `<table: data-pilot-subtotal>` element has the following attribute:  
`table: function` 19.647.

The `<table: data-pilot-subtotal>` element has no child elements.

## 9.6.11 `<table: data-pilot-members>`

The `<table: data-pilot-members>` element specifies whether or not to display category columns in data pilot tables or information for category columns.

The `<table: data-pilot-members>` element is usable within the following element:  
`<table: data-pilot-level>` 9.6.8.

The `<table: data-pilot-members>` element has no attributes.

The `<table: data-pilot-members>` element has the following child element:  
`<table: data-pilot-member>` 9.6.12.

## 9.6.12 `<table: data-pilot-member>`

The `<table: data-pilot-member>` element specifies what information is displayed for a member.

The `<table: data-pilot-member>` element is usable within the following element:  
`<table: data-pilot-members>` 9.6.11.

The `<table: data-pilot-member>` element has the following attributes: `table: display` 19.621, `table: name` 19.677 and `table: show-details` 19.713.

The `<table: data-pilot-member>` element has no child elements.

### 9.6.13 <table:data-pilot-display-info>

The `<table:data-pilot-display-info>` element restricts the number rows that are displayed for a category column to a specific number of values of a data field.

The `<table:data-pilot-display-info>` element is usable within the following element: `<table:data-pilot-level> 9.6.8`.

The `<table:data-pilot-display-info>` element has the following attributes: `table:data-field 19.611`, `table:display-member-mode 19.626`, `table:enabled 19.629` and `table:member-count 19.671`.

The `<table:data-pilot-display-info>` element has no child elements.

### 9.6.14 <table:data-pilot-sort-info>

The `<table:data-pilot-sort-info>` element specifies how the members of a category column are sorted.

The `<table:data-pilot-sort-info>` element is usable within the following element: `<table:data-pilot-level> 9.6.8`.

The `<table:data-pilot-sort-info>` element has the following attributes: `table:data-field 19.611`, `table:order 19.689`, `table:sort-mode 19.716`.

The `<table:data-pilot-sort-info>` element has no child elements.

### 9.6.15 <table:data-pilot-layout-info>

The `<table:data-pilot-layout-info>` element specifies how to layout a field.

The `<table:data-pilot-layout-info>` element is usable within the following element: `<table:data-pilot-level> 9.6.8`.

The `<table:data-pilot-layout-info>` element has the following attributes: `table:add-empty-lines 19.586` and `table:layout-mode 19.666`.

The `<table:data-pilot-layout-info>` element has no child elements.

### 9.6.16 <table:data-pilot-field-reference>

The `<table:data-pilot-field-reference>` element specifies data which can be used to modify the displayed values of data fields.

The `<table:data-pilot-field-reference>` element is usable within the following element: `<table:data-pilot-field> 9.6.7`.

The `<table:data-pilot-field-reference>` element has the following attributes: `table:field-name 19.640`, `table:member-name 19.672`, `table:member-type 19.673` and `table:type 19.740`.

The `<table:data-pilot-field-reference>` element has no child elements.

### 9.6.17 <table:data-pilot-groups>

The `<table:data-pilot-groups>` element specifies that a data pilot field is a group field. A group field allows grouping of other fields.

Grouping may also take place for numeric or date values.

The `<table:data-pilot-groups>` element is usable within the following element: `<table:data-pilot-field> 9.6.7`.

The `<table:data-pilot-groups>` element has the following attributes: `table:date-end` 19.616, `table:date-start` 19.617, `table:end` 19.630, `table:grouped-by` 19.650, `table:source-field-name` 19.718, `table:start` 19.721 and `table:step` 19.727.

The `<table:data-pilot-groups>` element has the following child element:  
`<table:data-pilot-group>` 9.6.18.

### 9.6.18 `<table:data-pilot-group>`

The `<table:data-pilot-group>` element specifies names of groups if grouping takes place by specifying the member names.

The `<table:data-pilot-group>` element is usable within the following element:  
`<table:data-pilot-groups>` 9.6.17.

The `<table:data-pilot-group>` element has the following attribute: `table:name` 19.677.

The `<table:data-pilot-group>` element has the following child element:  
`<table:data-pilot-group-member>` 9.6.19.

### 9.6.19 `<table:data-pilot-group-member>`

The `<table:data-pilot-group-member>` element specifies the name of a single group member.

The `<table:data-pilot-group-member>` element is usable within the following element:  
`<table:data-pilot-group>` 9.6.18.

The `<table:data-pilot-group-member>` element has the following attribute: `table:name` 19.677.

The `<table:data-pilot-group-member>` element has no child elements.

## 9.7 `<table:consolidation>`

The `<table:consolidation>` element defines the consolidation of data from multiple table ranges. Consolidation means that a new table range is filled with values calculated by applying a mathematical function to all cells in the source table ranges that have the same relative address within these ranges.

The `<table:consolidation>` element is usable within the following elements:  
`<office:chart>` 3.8, `<office:drawing>` 3.5, `<office:presentation>` 3.6, `<office:spreadsheet>` 3.7 and `<office:text>` 3.4.

The `<table:consolidation>` element has the following attributes: `table:function` 19.647, `table:link-to-source-data` 19.667, `table:source-cell-range-addresses` 19.717, `table:target-cell-address` 19.735 and `table:use-labels` 19.747.

The `<table:consolidation>` element has no child elements.

## 9.8 `<table:dde-links>`

The `<table:dde-links>` container element stores all DDE links for use in spreadsheet formulas. Every link contains the DDE Source and the data of the last connection.

See 14.7 for the use of DDE connections.

The `<table:dde-links>` element is usable within the following elements: `<office:chart>` 3.8, `<office:drawing>` 3.5, `<office:presentation>` 3.6, `<office:spreadsheet>` 3.7 and `<office:text>` 3.4.

The `<table:dde-links>` element has no attributes.

The `<table:dde-links>` element has the following child element: `<table:dde-link>` 14.7.4.

## 9.9 Change Tracking in Spreadsheets

### 9.9.1 General

All changes that have been applied to a spreadsheet document are stored in a list. The list contains an element for each change made to the table.

**Note:** See 5.5.1 for general remarks about implementing change tracking.

### 9.9.2 `<table:tracked-changes>`

The `<table:tracked-changes>` element is a container for tracked changes of a spreadsheet. Its presence further enables change tracking for a spreadsheet.

The `<table:tracked-changes>` element is usable within the following element: `<office:spreadsheet>` 3.7.

The `<table:tracked-changes>` element has the following attribute: `table:track-changes` 19.739.

The `<table:tracked-changes>` element has the following child elements: `<table:cell-content-change>` 9.9.17, `<table:deletion>` 9.9.9, `<table:insertion>` 9.9.3 and `<table:movement>` 9.9.13.

### 9.9.3 `<table:insertion>`

The `<table:insertion>` element contains the information that identifies any insertion of content. This content can be one or more rows, one or more columns, or a table.

The `<table:insertion>` element is usable within the following element: `<table:tracked-changes>` 9.9.2.

The `<table:insertion>` element has the following attributes: `table:acceptance-state` 19.584, `table:count` 19.609, `table:id` 19.652, `table:position` 19.695, `table:rejecting-change-id` 19.706, `table:table` 19.732 and `table:type` 19.740.

The `<table:insertion>` element has the following child elements: `<office:change-info>` 5.5.7, `<table:deletions>` 9.9.6 and `<table:dependencies>` 9.9.4.

### 9.9.4 `<table:dependencies>`

The `<table:dependencies>` element contains the information on which other tracked changes the current change depends.

The `<table:dependencies>` element is usable within the following elements: `<table:cell-content-change>` 9.9.17, `<table:deletion>` 9.9.9, `<table:insertion>` 9.9.3 and `<table:movement>` 9.9.13.

The `<table:dependencies>` element has no attributes.

The `<table:dependencies>` element has the following child element: `<table:dependency>` 9.9.5.

### 9.9.5 <table:dependency>

The `<table:dependency>` element contains the information about one change action on which on which the current tracked change depends. The change action on which the current change depends is referenced by its `table:id` attribute

The `<table:dependency>` element is usable within the following element:  
`<table:dependencies>` 9.9.4.

The `<table:dependency>` element has the following attribute: `table:id` 19.652.

The `<table:dependency>` element has no child elements.

### 9.9.6 <table:deletions>

The `<table:deletions>` element contains all deletions which are performed while tracking a single change to a table.

The `<table:deletions>` element is usable within the following elements: `<table:cell-content-change>` 9.9.17, `<table:deletion>` 9.9.9, `<table:insertion>` 9.9.3 and `<table:movement>` 9.9.13.

The `<table:deletions>` element has no attributes.

The `<table:deletions>` element has the following child elements: `<table:cell-content-deletion>` 9.9.7 and `<table:change-deletion>` 9.9.8.

### 9.9.7 <table:cell-content-deletion>

The `<table:cell-content-deletion>` element specifies that a cell content has been deleted. It contains the address of the effected cell and its former content. If a `table:id` attribute is present, it specifies the id of a previously tracked change for the cell that gets deleted by the current change.

The `<table:cell-content-deletion>` element is usable within the following element:  
`<table:deletions>` 9.9.6.

The `<table:cell-content-deletion>` element has the following attribute: `table:id` 19.652.

The `<table:cell-content-deletion>` element has the following child elements:  
`<table:cell-address>` 9.9.18 and `<table:change-track-table-cell>` 9.9.16.

### 9.9.8 <table:change-deletion>

The `<table:change-deletion>` element specifies the value of the `table:id` of a previously tracked change that is deleted by the current change.

The `<table:change-deletion>` element is usable within the following element:  
`<table:deletions>` 9.9.6.

The `<table:change-deletion>` element has the following attribute: `table:id` 19.652.

The `<table:change-deletion>` element has no child elements.

### 9.9.9 <table:deletion>

A `<table:deletion>` element contains content that was deleted while change tracking was enabled. The content of a cell that was deleted is contained in either a `<table:dependencies>`, or a `<table:deletions>` element.

The `<table:deletion>` element is usable within the following element:  
`<table:tracked-changes>` 9.9.2.

The `<table:deletion>` element has the following attributes: `table:acceptance-state 19.584`, `table:id 19.652`, `table:multi-deletion-spanned 19.676`, `table:position 19.695`, `table:rejecting-change-id 19.706`, `table:table 19.732` and `table:type 19.740`.

The `<table:deletion>` element has the following child elements: `<office:change-info> 5.5.7`, `<table:cut-offs> 9.9.10`, `<table:deletions> 9.9.6` and `<table:dependencies> 9.9.4`.

## 9.9.10 `<table:cut-offs>`

A `<table:cut-offs>` element contains elements that specify the location of tracked but deleted insertions or movements.

The `<table:cut-offs>` element is usable within the following element: `<table:deletion> 9.9.9`.

The `<table:cut-offs>` element has no attributes.

The `<table:cut-offs>` element has the following child elements: `<table:insertion-cut-off> 9.9.11`, `<table:movement-cut-off> 9.9.12`.

## 9.9.11 `<table:insertion-cut-off>`

The `<table:insertion-cut-off>` element specifies where an insertion was deleted.

The `<table:insertion-cut-off>` element is usable within the following element: `<table:cut-offs> 9.9.10`.

The `<table:insertion-cut-off>` element has the following attributes: `table:id 19.652` and `table:position 19.695`.

The `<table:insertion-cut-off>` element has no child elements.

## 9.9.12 `<table:movement-cut-off>`

The `<table:movement-cut-off>` element specifies the deletion of a movement.

The `<table:movement-cut-off>` element is usable within the following element: `<table:cut-offs> 9.9.10`.

The `<table:movement-cut-off>` element has the following attributes: `table:end-position 19.633`, `table:position 19.695` and `table:start-position 19.723`.

The `<table:movement-cut-off>` element has no child elements.

## 9.9.13 `<table:movement>`

A `<table:movement>` element specifies the information that identifies movement of content. This content can be a cell content or a cell range content.

The `<table:movement>` element is usable within the following element: `<table:tracked-changes> 9.9.2`.

The `<table:movement>` element has the following attributes: `table:acceptance-state 19.584`, `table:id 19.652` and `table:rejecting-change-id 19.706`.

The `<table:movement>` element has the following child elements: `<office:change-info> 5.5.7`, `<table:deletions> 9.9.6`, `<table:dependencies> 9.9.4`, `<table:source-range-address> 9.9.14` and `<table:target-range-address> 9.9.15`.

### 9.9.14 <table:source-range-address>

The `<table:source-range-address>` element specifies the source or cell range address of a movement.

The `<table:source-range-address>` element is usable within the following element:  
`<table:movement>` 9.9.13.

The `<table:source-range-address>` element has the following attributes:  
`table:column 19.598`, `table:end-column 19.632`, `table:end-row 19.634`,  
`table:end-table 19.635`, `table:row 19.708`, `table:start-column 19.722`,  
`table:start-row 19.724`, `table:start-table 19.725` and `table:table 19.732`.

The `<table:source-range-address>` element has no child elements.

### 9.9.15 <table:target-range-address>

The `<table:target-range-address>` element specifies the target cell address or cell range address of a movement.

The `<table:target-range-address>` element is usable within the following element:  
`<table:movement>` 9.9.13.

The `<table:target-range-address>` element has the following attributes:  
`table:column 19.598`, `table:end-column 19.632`, `table:end-row 19.634`,  
`table:end-table 19.635`, `table:row 19.708`, `table:start-column 19.722`,  
`table:start-row 19.724`, `table:start-table 19.725` and `table:table 19.732`.

The `<table:target-range-address>` element has no child elements.

### 9.9.16 <table:change-track-table-cell>

The `<table:change-track-table-cell>` element specifies the information that tracks changes to a table cell.

The `<table:change-track-table-cell>` element is usable within the following elements: `<table:cell-content-deletion>` 9.9.7 and `<table:previous>` 9.9.19.

The `<table:change-track-table-cell>` element has the following attributes:  
`office:boolean-value 19.371`, `office:currency 19.373`, `office:date-value 19.374`, `office:string-value 19.383`, `office:time-value 19.386`, `office:value 19.388`, `office:value-type 19.389`, `table:cell-address 19.595`, `table:formula 19.646`, `table:matrix-covered 19.669`, `table:number-matrix-columns-spanned 19.683` and `table:number-matrix-rows-spanned 19.684`.

The `<table:change-track-table-cell>` element has the following child element:  
`<text:p>` 5.1.3.

### 9.9.17 <table:cell-content-change>

A `<table:cell-content-change>` element specifies the information that identifies changes of the cell content.

The `<table:cell-content-change>` element is usable within the following element:  
`<table:tracked-changes>` 9.9.2.

The `<table:cell-content-change>` element has the following attributes:  
`table:acceptance-state 19.584`, `table:id 19.652` and `table:rejecting-change-id 19.706`.

The `<table:cell-content-change>` element has the following child elements: `<office:change-info>` 5.5.7, `<table:cell-address>` 9.9.18, `<table:deletions>` 9.9.6, `<table:dependencies>` 9.9.4 and `<table:previous>` 9.9.19.

## 9.9.18 `<table:cell-address>`

The `<table:cell-address>` element contains the address of a cell that has changed. Unlike other cell addresses, the cell address consists of the row, column and table number of the cell. This allows the specification of addresses that are outside a valid cell address range.

The `<table:cell-address>` element is usable within the following elements: `<table:cell-content-change>` 9.9.17 and `<table:cell-content-deletion>` 9.9.7.

The `<table:cell-address>` element has the following attributes: `table:column` 19.598, `table:row` 19.708 and `table:table` 19.732.

The `<table:cell-address>` element has no child elements.

## 9.9.19 `<table:previous>`

The `<table:previous>` element contains the previous cell content which is overwritten by the current change. If a `table:id` attribute is present, it specifies the id of a previously tracked change for the cell that is changed again by the current change.

The `<table:previous>` element is usable within the following element: `<table:cell-content-change>` 9.9.17.

The `<table:previous>` element has the following attribute: `table:id` 19.652.

The `<table:previous>` element has the following child element: `<table:change-track-table-cell>` 9.9.16.

# 10 Graphic Content

## 10.1 General

This chapter specifies the elements that represent graphic objects and elements used in the representation of graphical objects.

## 10.2 Enhanced Page Features for Graphical Applications

### 10.2.1 <style:handout-master>

The `<style:handout-master>` element represents a template for the generation of handout pages.

The `<style:handout-master>` element is usable within the following element:  
`<office:master-styles>` 3.15.4.

The `<style:handout-master>` element has the following attributes: `draw:style-name` 19.219, `presentation:presentation-page-layout-name` 19.413, `presentation:use-date-time-name` 19.428, `presentation:use-footer-name` 19.429, `presentation:use-header-name` 19.430 and `style:page-layout-name` 19.508.

The `<style:handout-master>` element has the following child elements: `<dr3d:scene>` 10.5.2, `<draw:a>` 10.4.12, `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:connector>` 10.3.10, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:g>` 10.3.15, `<draw:line>` 10.3.3, `<draw:measure>` 10.3.12, `<draw:page-thumbnail>` 10.3.14, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2 and `<draw:regular-polygon>` 10.3.6.

### 10.2.2 <draw:layer-set>

The `<draw:layer-set>` element defines a set of layers. If placed inside a `<style:master-page>` or `<draw:page>` element it defines a set of layers for that page. If placed inside the `<office:master-styles>` element it defines a set of layers for all pages that do not have their own set of layers.

Layers group drawing objects. Assigning a shape to a layer does not change its rendering order as defined by its `draw:z-index` attribute. 19.231

The `<draw:layer-set>` element is usable within the following elements: `<draw:page>` 10.2.4, `<office:master-styles>` 3.15.4 and `<style:master-page>` 16.9.

The `<draw:layer-set>` element has no attributes.

The `<draw:layer-set>` element has the following child element: `<draw:layer>` 10.2.3.

### 10.2.3 <draw:layer>

The `<draw:layer>` element defines a single layer. Layers group drawing objects.

Each drawing object inside a drawing or presentation document can be assigned to a layer. Each object that is assigned to a layer inherits the settings of that layer. Drawing objects may be assigned to these layers by the value of their `draw:layer` 19.189 attribute.

The `<draw:layer>` element is usable within the following element: `<draw:layer-set>` 10.2.2.

The `<draw:layer>` element has the following attributes: `draw:display` 19.132, `draw:name` 19.197 and `draw:protected` 19.207.

The `<draw:layer>` element has the following child elements: `<svg:desc>` 10.3.18 and `<svg:title>` 10.3.17.

#### 10.2.4 `<draw:page>`

The `<draw:page>` element is a container for content in a drawing or presentation document. This element is used for the following:

- Layers 10.2.2
- Forms 13.2
- Drawing Objects 10.3
- Frames 10.4
- Presentation Animations 10.9
- Presentation Notes 16.19

The `<draw:page>` element is usable within the following elements: `<office:drawing>` 3.5 and `<office:presentation>` 3.6.

The `<draw:page>` element has the following attributes: `draw:id` 19.187, `draw:master-page-name` 19.194, `draw:name` 19.197, `draw:nav-order` 19.198, `draw:style-name` 19.219, `presentation:presentation-page-layout-name` 19.413, `presentation:use-date-time-name` 19.428, `presentation:use-footer-name` 19.429, `presentation:use-header-name` 19.430 and `xml:id` 19.920.

The `<draw:page>` element has the following child elements: `<anim:animate>` 15.2.2, `<anim:animateColor>` 15.2.6, `<anim:animateMotion>` 15.2.5, `<anim:animateTransform>` 15.2.3, `<anim:audio>` 15.5, `<anim:command>` 15.6.1, `<anim:iterate>` 15.4.4, `<anim:par>` 15.4.2, `<anim:seq>` 15.4.3, `<anim:set>` 15.2.4, `<anim:transitionFilter>` 15.2.7, `<dr3d:scene>` 10.5.2, `<draw:a>` 10.4.12, `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:connector>` 10.3.10, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:g>` 10.3.15, `<draw:layer-set>` 10.2.2, `<draw:line>` 10.3.3, `<draw:measure>` 10.3.12, `<draw:page-thumbnail>` 10.3.14, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2, `<draw:regular-polygon>` 10.3.6, `<office:annotation>` 14.1, `<office:forms>` 13.2, `<presentation:animations>` 10.8.1, `<presentation:notes>` 16.19, `<svg:desc>` 10.3.18 and `<svg:title>` 10.3.17.

#### 10.2.5 `<draw:equation>`

The `<draw:equation>` element specifies a formula for the calculation of a value and the name by which that formula can be referenced.

The `<draw:equation>` element is usable within the following element: `<draw:enhanced-geometry>` 10.6.2.

The `<draw:equation>` element has the following attributes: `draw:formula` 19.171 and `draw:name` 19.197.

The `<draw:equation>` element has no child elements.

## 10.3 Drawing Shapes

### 10.3.1 General

This section defines graphical objects that may occur within all types of documents.

### 10.3.2 <draw:rect>

The `<draw:rect>` element represents a rectangular drawing shape.

The `<draw:rect>` element is usable within the following elements: `<chart:chart> 11.1, <draw:a> 10.4.12, <draw:g> 10.3.15, <draw:page> 10.2.4, <draw:text-box> 10.4.3, <office:text> 3.4, <presentation:notes> 16.19, <style:handout-master> 10.2.1, <style:master-page> 16.9, <table:covered-table-cell> 9.1.5, <table:shapes> 9.2.8, <table:table-cell> 9.1.4, <text:a> 6.1.8, <text:deletion> 5.5.5, <text:h> 5.1.2, <text:index-body> 8.2.2, <text:index-title> 8.2.3, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:note-body> 6.3.4, <text:p> 5.1.3, <text:ruby-base> 6.4.2, <text:section> 5.4 and <text:span> 6.1.7.`

The `<draw:rect>` element has the following attributes: `draw:caption-id 19.115, draw:class-names 19.120, draw:corner-radius 19.127, draw:id 19.187, draw:layer 19.189, draw:name 19.197, draw:style-name 19.219, draw:text-style-name 19.227, draw:transform 19.228, draw:z-index 19.231, presentation:class-names 19.394, presentation:style-name 19.426, svg:height 19.543, svg:rx 19.554, svg:ry 19.555, svg:width 19.575, svg:x 19.577, svg:y 19.581, table:end-cell-address 19.631, table:end-x 19.636, table:end-y 19.637, table:table-background 19.733, text:anchor-page-number 19.758, text:anchor-type 19.759 and xml:id 19.920.`

The `<draw:rect>` element has the following child elements: `<draw:glue-point> 10.3.16, <office:event-listeners> 10.3.19, <svg:desc> 10.3.18, <svg:title> 10.3.17, <table:table> 9.1.2, <text:list> 5.3.1 and <text:p> 5.1.3.`

### 10.3.3 <draw:line>

The `<draw:line>` element represents a line.

The `<draw:line>` element is usable within the following elements: `<chart:chart> 11.1, <draw:a> 10.4.12, <draw:g> 10.3.15, <draw:page> 10.2.4, <draw:text-box> 10.4.3, <office:text> 3.4, <presentation:notes> 16.19, <style:handout-master> 10.2.1, <style:master-page> 16.9, <table:covered-table-cell> 9.1.5, <table:shapes> 9.2.8, <table:table-cell> 9.1.4, <text:a> 6.1.8, <text:deletion> 5.5.5, <text:h> 5.1.2, <text:index-body> 8.2.2, <text:index-title> 8.2.3, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:note-body> 6.3.4, <text:p> 5.1.3, <text:ruby-base> 6.4.2, <text:section> 5.4 and <text:span> 6.1.7.`

The `<draw:line>` element has the following attributes: `draw:caption-id 19.115, draw:class-names 19.120, draw:id 19.187, draw:layer 19.189, draw:name 19.197, draw:style-name 19.219, draw:text-style-name 19.227, draw:transform 19.228, draw:z-index 19.231, presentation:class-names 19.394, presentation:style-name 19.426, svg:x1 19.578, svg:x2 19.579, svg:y1 19.582, svg:y2 19.583, table:end-cell-address 19.631, table:end-x 19.636, table:end-y 19.637, table:table-background 19.733, text:anchor-page-number 19.758, text:anchor-type 19.759 and xml:id 19.920.`

The `<draw:line>` element has the following child elements: `<draw:glue-point>` 10.3.16, `<office:event-listeners>` 10.3.19, `<svg:desc>` 10.3.18, `<svg:title>` 10.3.17, `<table:table>` 9.1.2, `<text:list>` 5.3.1 and `<text:p>` 5.1.3.

#### 10.3.4 `<draw:polyline>`

The `<draw:polyline>` element represents a polyline drawing shape.

Consumers may ignore `svg:height` and `svg:width` attributes, and determine the size of a shape from the polygon vertices.

The `<draw:polyline>` element is usable within the following elements: `<chart:chart>` 11.1, `<draw:a>` 10.4.12, `<draw:g>` 10.3.15, `<draw:page>` 10.2.4, `<draw:text-box>` 10.4.3, `<office:text>` 3.4, `<presentation:notes>` 16.19, `<style:handout-master>` 10.2.1, `<style:master-page>` 16.9, `<table:covered-table-cell>` 9.1.5, `<table:shapes>` 9.2.8, `<table:table-cell>` 9.1.4, `<text:a>` 6.1.8, `<text:deletion>` 5.5.5, `<text:h>` 5.1.2, `<text:index-body>` 8.2.2, `<text:index-title>` 8.2.3, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:note-body>` 6.3.4, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2, `<text:section>` 5.4 and `<text:span>` 6.1.7.

The `<draw:polyline>` element has the following attributes: `draw:caption-id` 19.115, `draw:class-names` 19.120, `draw:id` 19.187, `draw:layer` 19.189, `draw:name` 19.197, `draw:points` 19.206, `draw:style-name` 19.219, `draw:text-style-name` 19.227, `draw:transform` 19.228, `draw:z-index` 19.231, `presentation:class-names` 19.394, `presentation:style-name` 19.426, `svg:height` 19.543, `svg:viewBox` 19.574, `svg:width` 19.575, `svg:x` 19.577, `svg:y` 19.581, `table:end-cell-address` 19.631, `table:end-x` 19.636, `table:end-y` 19.637, `table:table-background` 19.733, `text:anchor-page-number` 19.758, `text:anchor-type` 19.759 and `xml:id` 19.920.

The `<draw:polyline>` element has the following child elements: `<draw:glue-point>` 10.3.16, `<office:event-listeners>` 10.3.19, `<svg:desc>` 10.3.18, `<svg:title>` 10.3.17, `<table:table>` 9.1.2, `<text:list>` 5.3.1 and `<text:p>` 5.1.3.

#### 10.3.5 `<draw:polygon>`

The `<draw:polygon>` element represents a polygon. A polygon is a closed set of straight lines.

Consumers may ignore `svg:height` and `svg:width` attributes, and determine the size of a shape from the polygon vertices.

The `<draw:polygon>` element is usable within the following elements: `<chart:chart>` 11.1, `<draw:a>` 10.4.12, `<draw:g>` 10.3.15, `<draw:page>` 10.2.4, `<draw:text-box>` 10.4.3, `<office:text>` 3.4, `<presentation:notes>` 16.19, `<style:handout-master>` 10.2.1, `<style:master-page>` 16.9, `<table:covered-table-cell>` 9.1.5, `<table:shapes>` 9.2.8, `<table:table-cell>` 9.1.4, `<text:a>` 6.1.8, `<text:deletion>` 5.5.5, `<text:h>` 5.1.2, `<text:index-body>` 8.2.2, `<text:index-title>` 8.2.3, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:note-body>` 6.3.4, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2, `<text:section>` 5.4 and `<text:span>` 6.1.7.

The `<draw:polygon>` element has the following attributes: `draw:caption-id` 19.115, `draw:class-names` 19.120, `draw:id` 19.187, `draw:layer` 19.189, `draw:name` 19.197, `draw:points` 19.206, `draw:style-name` 19.219, `draw:text-style-name` 19.227, `draw:transform` 19.228, `draw:z-index` 19.231, `presentation:class-names` 19.394, `presentation:style-name` 19.426, `svg:height` 19.543, `svg:viewBox` 19.574, `svg:width` 19.575, `svg:x` 19.577, `svg:y` 19.581, `table:end-cell-address`

19.631, table:end-x 19.636, table:end-y 19.637, table:table-background 19.733, text:anchor-page-number 19.758, text:anchor-type 19.759 and xml:id 19.920.

The `<draw:polygon>` element has the following child elements: `<draw:glue-point>` 10.3.16, `<office:event-listeners>` 10.3.19, `<svg:desc>` 10.3.18, `<svg:title>` 10.3.17, `<table:table>` 9.1.2, `<text:list>` 5.3.1 and `<text:p>` 5.1.3.

### 10.3.6 `<draw:regular-polygon>`

The `<draw:regular-polygon>` element represents a regular polygon. A regular polygon is a polygon that is specified by its number of edges (that is equal to the number of its corners), rather than by arbitrary points.

The `<draw:regular-polygon>` element is usable within the following elements: `<chart:chart>` 11.1, `<draw:a>` 10.4.12, `<draw:g>` 10.3.15, `<draw:page>` 10.2.4, `<draw:text-box>` 10.4.3, `<office:text>` 3.4, `<presentation:notes>` 16.19, `<style:handout-master>` 10.2.1, `<style:master-page>` 16.9, `<table:covered-table-cell>` 9.1.5, `<table:shapes>` 9.2.8, `<table:table-cell>` 9.1.4, `<text:a>` 6.1.8, `<text:deletion>` 5.5.5, `<text:h>` 5.1.2, `<text:index-body>` 8.2.2, `<text:index-title>` 8.2.3, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:note-body>` 6.3.4, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2, `<text:section>` 5.4 and `<text:span>` 6.1.7.

The `<draw:regular-polygon>` element has the following attributes: `draw:caption-id` 19.115, `draw:class-names` 19.120, `draw:concave` 19.123, `draw:corners` 19.128, `draw:id` 19.187, `draw:layer` 19.189, `draw:name` 19.197, `draw:sharpness` 19.211, `draw:style-name` 19.219, `draw:text-style-name` 19.227, `draw:transform` 19.228, `draw:z-index` 19.231, `presentation:class-names` 19.394, `presentation:style-name` 19.426, `svg:height` 19.543, `svg:width` 19.575, `svg:x` 19.577, `svg:y` 19.581, `table:end-cell-address` 19.631, `table:end-x` 19.636, `table:end-y` 19.637, `table:table-background` 19.733, `text:anchor-page-number` 19.758, `text:anchor-type` 19.759 and `xml:id` 19.920.

The `<draw:regular-polygon>` element has the following child elements: `<draw:glue-point>` 10.3.16, `<office:event-listeners>` 10.3.19, `<svg:desc>` 10.3.18, `<svg:title>` 10.3.17, `<table:table>` 9.1.2, `<text:list>` 5.3.1 and `<text:p>` 5.1.3.

### 10.3.7 `<draw:path>`

The `<draw:path>` element represents a path. A path is a shape with a user-defined outline. The outline is defined by the `svg:d` attribute. 19.530

The `<draw:path>` element is usable within the following elements: `<chart:chart>` 11.1, `<draw:a>` 10.4.12, `<draw:g>` 10.3.15, `<draw:page>` 10.2.4, `<draw:text-box>` 10.4.3, `<office:text>` 3.4, `<presentation:notes>` 16.19, `<style:handout-master>` 10.2.1, `<style:master-page>` 16.9, `<table:covered-table-cell>` 9.1.5, `<table:shapes>` 9.2.8, `<table:table-cell>` 9.1.4, `<text:a>` 6.1.8, `<text:deletion>` 5.5.5, `<text:h>` 5.1.2, `<text:index-body>` 8.2.2, `<text:index-title>` 8.2.3, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:note-body>` 6.3.4, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2, `<text:section>` 5.4 and `<text:span>` 6.1.7.

The `<draw:path>` element has the following attributes: `draw:caption-id` 19.115, `draw:class-names` 19.120, `draw:id` 19.187, `draw:layer` 19.189, `draw:name` 19.197, `draw:style-name` 19.219, `draw:text-style-name` 19.227, `draw:transform` 19.228, `draw:z-index` 19.231, `presentation:class-names` 19.394, `presentation:style-name` 19.426, `svg:d` 19.530, `svg:height` 19.543, `svg:viewBox` 19.574, `svg:width` 19.575, `svg:x` 19.577, `svg:y` 19.581, `table:end-cell-address` 19.631, `table:end-x`

19.636, table:end-y 19.637, table:table-background 19.733, text:anchor-page-number 19.758, text:anchor-type 19.759 and xml:id 19.920.

The `<draw:path>` element has the following child elements: `<draw:glue-point>` 10.3.16, `<office:event-listeners>` 10.3.19, `<svg:desc>` 10.3.18, `<svg:title>` 10.3.17, `<table:table>` 9.1.2, `<text:list>` 5.3.1 and `<text:p>` 5.1.3.

### 10.3.8 `<draw:circle>`

The `<draw:circle>` element represents a circular drawing shape.

The `<draw:circle>` element is usable within the following elements: `<chart:chart>` 11.1, `<draw:a>` 10.4.12, `<draw:g>` 10.3.15, `<draw:page>` 10.2.4, `<draw:text-box>` 10.4.3, `<office:text>` 3.4, `<presentation:notes>` 16.19, `<style:handout-master>` 10.2.1, `<style:master-page>` 16.9, `<table:covered-table-cell>` 9.1.5, `<table:shapes>` 9.2.8, `<table:table-cell>` 9.1.4, `<text:a>` 6.1.8, `<text:deletion>` 5.5.5, `<text:h>` 5.1.2, `<text:index-body>` 8.2.2, `<text:index-title>` 8.2.3, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:note-body>` 6.3.4, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2, `<text:section>` 5.4 and `<text:span>` 6.1.7.

The `<draw:circle>` element has the following attributes: `draw:caption-id` 19.115, `draw:class-names` 19.120, `draw:end-angle` 19.140, `draw:id` 19.187, `draw:kind` 19.188, `draw:layer` 19.189, `draw:name` 19.197, `draw:start-angle` 19.213, `draw:style-name` 19.219, `draw:text-style-name` 19.227, `draw:transform` 19.228, `draw:z-index` 19.231, `presentation:class-names` 19.394, `presentation:style-name` 19.426, `svg:cx` 19.528, `svg:cy` 19.529, `svg:height` 19.543, `svg:r` 19.553, `svg:width` 19.575, `svg:x` 19.577, `svg:y` 19.581, `table:end-cell-address` 19.631, `table:end-x` 19.636, `table:end-y` 19.637, `table:table-background` 19.733, `text:anchor-page-number` 19.758, `text:anchor-type` 19.759 and `xml:id` 19.920.

The `<draw:circle>` element has the following child elements: `<draw:glue-point>` 10.3.16, `<office:event-listeners>` 10.3.19, `<svg:desc>` 10.3.18, `<svg:title>` 10.3.17, `<table:table>` 9.1.2, `<text:list>` 5.3.1 and `<text:p>` 5.1.3.

### 10.3.9 `<draw:ellipse>`

The `<draw:ellipse>` element represents an ellipse.

The `<draw:ellipse>` element is usable within the following elements: `<chart:chart>` 11.1, `<draw:a>` 10.4.12, `<draw:g>` 10.3.15, `<draw:page>` 10.2.4, `<draw:text-box>` 10.4.3, `<office:text>` 3.4, `<presentation:notes>` 16.19, `<style:handout-master>` 10.2.1, `<style:master-page>` 16.9, `<table:covered-table-cell>` 9.1.5, `<table:shapes>` 9.2.8, `<table:table-cell>` 9.1.4, `<text:a>` 6.1.8, `<text:deletion>` 5.5.5, `<text:h>` 5.1.2, `<text:index-body>` 8.2.2, `<text:index-title>` 8.2.3, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:note-body>` 6.3.4, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2, `<text:section>` 5.4 and `<text:span>` 6.1.7.

The `<draw:ellipse>` element has the following attributes: `draw:caption-id` 19.115, `draw:class-names` 19.120, `draw:end-angle` 19.140, `draw:id` 19.187, `draw:kind` 19.188, `draw:layer` 19.189, `draw:name` 19.197, `draw:start-angle` 19.213, `draw:style-name` 19.219, `draw:text-style-name` 19.227, `draw:transform` 19.228, `draw:z-index` 19.231, `presentation:class-names` 19.394, `presentation:style-name` 19.426, `svg:cx` 19.528, `svg:cy` 19.529, `svg:height` 19.543, `svg:rx` 19.554, `svg:ry` 19.555, `svg:width` 19.575, `svg:x` 19.577, `svg:y` 19.581, `table:end-cell-address` 19.631, `table:end-x` 19.636, `table:end-y` 19.637, `table:table-background` 19.733, `text:anchor-page-number` 19.758, `text:anchor-type` 19.759 and `xml:id` 19.920.

The `<draw:ellipse>` element has the following child elements: `<draw:glue-point>` 10.3.16, `<office:event-listeners>` 10.3.19, `<svg:desc>` 10.3.18, `<svg:title>` 10.3.17, `<table:table>` 9.1.2, `<text:list>` 5.3.1 and `<text:p>` 5.1.3.

### 10.3.10 `<draw:connector>`

The `<draw:connector>` element represents a connected set of one or more lines that visually connects a start and an end point.

Start and/or end points can be defined by references to glue points 10.3.16 or as absolute positions. The connector's geometric path is defined by the `svg:d` 19.530 attribute. Consumers may also compute the connector's geometric path considering the `draw:type` 19.229.2 and `draw:line-skew` 19.190 attributes, and the formatting properties defined for connectors.

If the `svg:d` attribute is not present, the connector's geometric path is implementation-dependent. Producers should export the connector's geometric path using the `svg:d` attribute.

**Note:** This assists consumers that are not capable of computing a connector path.

The `<draw:connector>` element is usable within the following elements: `<chart:chart>` 11.1, `<draw:a>` 10.4.12, `<draw:g>` 10.3.15, `<draw:page>` 10.2.4, `<draw:text-box>` 10.4.3, `<office:text>` 3.4, `<presentation:notes>` 16.19, `<style:handout-master>` 10.2.1, `<style:master-page>` 16.9, `<table:covered-table-cell>` 9.1.5, `<table:shapes>` 9.2.8, `<table:table-cell>` 9.1.4, `<text:a>` 6.1.8, `<text:deletion>` 5.5.5, `<text:h>` 5.1.2, `<text:index-body>` 8.2.2, `<text:index-title>` 8.2.3, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:note-body>` 6.3.4, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2, `<text:section>` 5.4 and `<text:span>` 6.1.7.

The `<draw:connector>` element has the following attributes: `draw:caption-id` 19.115, `draw:class-names` 19.120, `draw:end-glue-point` 19.142, `draw:end-shape` 19.144, `draw:id` 19.187, `draw:layer` 19.189, `draw:line-skew` 19.190, `draw:name` 19.197, `draw:start-glue-point` 19.215, `draw:start-shape` 19.217, `draw:style-name` 19.219, `draw:text-style-name` 19.227, `draw:transform` 19.228, `draw:type` 19.229, `draw:z-index` 19.231, `presentation:class-names` 19.394, `presentation:style-name` 19.426, `svg:d` 19.530, `svg:viewBox` 19.574, `svg:x1` 19.578, `svg:x2` 19.579, `svg:y1` 19.582, `svg:y2` 19.583, `table:end-cell-address` 19.631, `table:end-x` 19.636, `table:end-y` 19.637, `table:table-background` 19.733, `text:anchor-page-number` 19.758, `text:anchor-type` 19.759 and `xml:id` 19.920.

The `<draw:connector>` element has the following child elements: `<draw:glue-point>` 10.3.16, `<office:event-listeners>` 10.3.19, `<svg:desc>` 10.3.18, `<svg:title>` 10.3.17, `<table:table>` 9.1.2, `<text:list>` 5.3.1 and `<text:p>` 5.1.3.

### 10.3.11 `<draw:caption>`

The `<draw:caption>` element represents a description attached to a fixed point. It consists of rectangular drawing shape with an additional set of connected lines that connect the rectangle with the fixed point.

The `<draw:caption>` element is usable within the following elements: `<chart:chart>` 11.1, `<draw:a>` 10.4.12, `<draw:g>` 10.3.15, `<draw:page>` 10.2.4, `<draw:text-box>` 10.4.3, `<office:text>` 3.4, `<presentation:notes>` 16.19, `<style:handout-master>` 10.2.1, `<style:master-page>` 16.9, `<table:covered-table-cell>` 9.1.5, `<table:shapes>` 9.2.8, `<table:table-cell>` 9.1.4, `<text:a>` 6.1.8, `<text:deletion>` 5.5.5, `<text:h>` 5.1.2, `<text:index-body>` 8.2.2, `<text:index-`

`title> 8.2.3, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:note-body> 6.3.4, <text:p> 5.1.3, <text:ruby-base> 6.4.2, <text:section> 5.4 and <text:span> 6.1.7.`

The `<draw:caption>` element has the following attributes: `draw:caption-id` 19.115, `draw:caption-point-x` 19.116, `draw:caption-point-y` 19.117, `draw:class-names` 19.120, `draw:corner-radius` 19.127, `draw:id` 19.187, `draw:layer` 19.189, `draw:name` 19.197, `draw:style-name` 19.219, `draw:text-style-name` 19.227, `draw:transform` 19.228, `draw:z-index` 19.231, `presentation:class-names` 19.394, `presentation:style-name` 19.426, `svg:height` 19.543, `svg:rx` 19.554, `svg:ry` 19.555, `svg:width` 19.575, `svg:x` 19.577, `svg:y` 19.581, `table:end-cell-address` 19.631, `table:end-x` 19.636, `table:end-y` 19.637, `table:table-background` 19.733, `text:anchor-page-number` 19.758, `text:anchor-type` 19.759 and `xml:id` 19.920.

The `<draw:caption>` element has the following child elements: `<draw:glue-point>` 10.3.16, `<office:event-listeners>` 10.3.19, `<svg:desc>` 10.3.18, `<svg:title>` 10.3.17, `<table:table>` 9.1.2, `<text:list>` 5.3.1 and `<text:p>` 5.1.3.

### 10.3.12 `<draw:measure>`

The `<draw:measure>` element represents a shape that is used to measure distances in drawings.

A measure shape consists of two parallel lines, called *extension lines*, and a line perpendicular to the extension lines, called a *dimension line*. The dimension line is displayed between the extension lines and terminated by them. The dimension line has markers at its endpoints. A marker at the start of the dimension line is defined by the `draw:marker-start` attribute and at the end of the dimension line by the `draw:marker-end` attribute of the graphic style, which is applied to the measure shape. Where space is limited, reversed markers at the opposite side of the extension line may be used.

The positions of extension lines and the dimension line are determined by two reference points, called *start* and *end reference points*. The start reference point is defined by the attributes `svg:x1` and `svg:y1`. The end reference point is defined by the attributes `svg:x2`, and `svg:y2`.

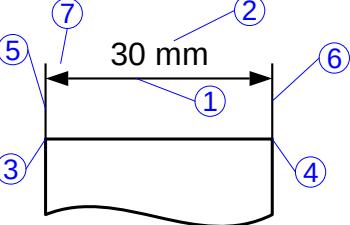
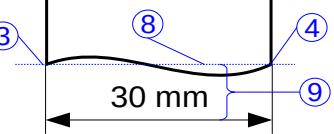
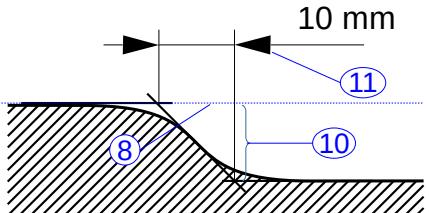
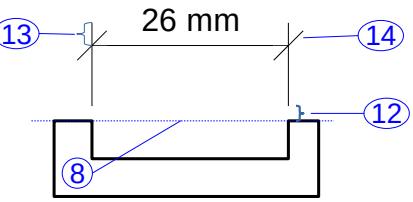
A *dimension value* specified by child elements `<text:measure>` of the `<text:p>` child element of the `<draw:measure>` element may be bound to the length of the dimension line by producers. The position of the dimension value relative to the dimension line is specified by the attributes `draw:measure-align` and `draw:measure-vertical-align`.

The `draw:line-distance` style attribute defines the distance between the (virtual) line between the reference points, called *reference line*, and the dimension line. The `draw:placing` style attributes defines if the measure shape is placed above or below the reference line.

The first extension line is drawn on a line as defined by the vector from the start reference point to the start point of the dimension line. The extension line will start along this vector at a distance from the start reference point that is defined by the `draw:guide-distance` style attribute minus the value of `draw:start-guide` style attribute.

The second extension line is drawn on a line as defined by the vector from the end reference point to the end point of the dimension line. The extension line will start along this vector at the distance from the end reference point that is defined by the `draw:guide-distance` style attribute minus the value of `draw:end-guide` style attribute.

The end points of the extension lines are at the distance defined by the `draw:guide-overhang` style attribute after the intersection with the dimension line.

	<b>Key</b>
 <p><i>Figure 1: measure line placed above reference line</i></p>	1 dimension line 2 dimension value 3 start reference point 4 end reference point 5 first extension line 6 second extension line 7 arrowhead as marker
 <p><i>Figure 2: measure line placed below reference line</i></p>	8 virtual reference line 9 distance between reference line and dimension line
 <p><i>Figure 3:</i></p>	10 attribute <code>draw:end-guide</code> with positive value, attribute <code>draw:guide-distance</code> with value zero 11 reversed dimension line markers and dimension value <code>measure-align</code> <code>right-outside</code>
 <p><i>Figure 4:</i></p>	12 gap specified by attribute <code>draw:guide-distance</code> 13 length specified by attribute <code>draw:guide-overhang</code> 14 Note: Certain technical fields might need special markers for dimension lines. The ODF standard does not restrict the shape of the used markers.

The `<draw:measure>` element is usable within the following elements: `<chart:chart>` 11.1, `<draw:a>` 10.4.12, `<draw:g>` 10.3.15, `<draw:page>` 10.2.4, `<draw:text-box>` 10.4.3, `<office:text>` 3.4, `<presentation:notes>` 16.19, `<style:handout-master>` 10.2.1, `<style:master-page>` 16.9, `<table:covered-table-cell>` 9.1.5, `<table:shapes>` 9.2.8, `<table:table-cell>` 9.1.4, `<text:a>` 6.1.8, `<text:deletion>` 5.5.5, `<text:h>` 5.1.2, `<text:index-body>` 8.2.2, `<text:index-title>` 8.2.3, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:note-body>`

6.3.4, <text:p> 5.1.3, <text:ruby-base> 6.4.2, <text:section> 5.4 and <text:span> 6.1.7.

The <draw:measure> element has the following attributes: draw:caption-id 19.115, draw:class-names 19.120, draw:id 19.187, draw:layer 19.189, draw:name 19.197, draw:style-name 19.219, draw:text-style-name 19.227, draw:transform 19.228, draw:z-index 19.231, presentation:class-names 19.394, presentation:style-name 19.426, svg:x1 19.578, svg:x2 19.579, svg:y1 19.582, svg:y2 19.583, table:end-cell-address 19.631, table:end-x 19.636, table:end-y 19.637, table:table-background 19.733, text:anchor-page-number 19.758, text:anchor-type 19.759 and xml:id 19.920.

The <draw:measure> element has the following child elements: <draw:glue-point> 10.3.16, <office:event-listeners> 10.3.19, <svg:desc> 10.3.18, <svg:title> 10.3.17, <table:table> 9.1.2, <text:list> 5.3.1 and <text:p> 5.1.3.

### 10.3.13 <draw:control>

The <draw:control> element represents a shape that is linked to a control inside an <office:forms> element.

The <draw:control> element is usable within the following elements: <chart:chart> 11.1, <draw:a> 10.4.12, <draw:g> 10.3.15, <draw:page> 10.2.4, <draw:text-box> 10.4.3, <office:text> 3.4, <presentation:notes> 16.19, <style:handout-master> 10.2.1, <style:master-page> 16.9, <table:covered-table-cell> 9.1.5, <table:shapes> 9.2.8, <table:table-cell> 9.1.4, <text:a> 6.1.8, <text:deletion> 5.5.5, <text:h> 5.1.2, <text:index-body> 8.2.2, <text:index-title> 8.2.3, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:note-body> 6.3.4, <text:p> 5.1.3, <text:ruby-base> 6.4.2, <text:section> 5.4 and <text:span> 6.1.7.

The <draw:control> element has the following attributes: draw:caption-id 19.115, draw:class-names 19.120, draw:control 19.125, draw:id 19.187, draw:layer 19.189, draw:name 19.197, draw:style-name 19.219, draw:text-style-name 19.227, draw:transform 19.228, draw:z-index 19.231, presentation:class-names 19.394, presentation:style-name 19.426, svg:height 19.543, svg:width 19.575, svg:x 19.577, svg:y 19.581, table:end-cell-address 19.631, table:end-x 19.636, table:end-y 19.637, table:table-background 19.733, text:anchor-page-number 19.758, text:anchor-type 19.759 and xml:id 19.920.

The <draw:control> element has the following child elements: <draw:glue-point> 10.3.16, <svg:desc> 10.3.18 and <svg:title> 10.3.17.

### 10.3.14 <draw:page-thumbnail>

The <draw:page-thumbnail> element represents a rectangular area that displays the thumbnail of a drawing page.

The <draw:page-thumbnail> element is usable within the following elements: <chart:chart> 11.1, <draw:a> 10.4.12, <draw:g> 10.3.15, <draw:page> 10.2.4, <draw:text-box> 10.4.3, <office:text> 3.4, <presentation:notes> 16.19, <style:handout-master> 10.2.1, <style:master-page> 16.9, <table:covered-table-cell> 9.1.5, <table:shapes> 9.2.8, <table:table-cell> 9.1.4, <text:a> 6.1.8, <text:deletion> 5.5.5, <text:h> 5.1.2, <text:index-body> 8.2.2, <text:index-title> 8.2.3, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:note-body> 6.3.4, <text:p> 5.1.3, <text:ruby-base> 6.4.2, <text:section> 5.4 and <text:span> 6.1.7.

The `<draw:page-thumbnail>` element has the following attributes: `draw:caption-id 19.115`, `draw:class-names 19.120`, `draw:id 19.187`, `draw:layer 19.189`, `draw:name 19.197`, `draw:page-number 19.203`, `draw:style-name 19.219`, `draw:transform 19.228`, `draw:z-index 19.231`, `presentation:class 19.393`, `presentation:class-names 19.394`, `presentation:placeholder 19.411`, `presentation:style-name 19.426`, `presentation:user-transformed 19.431`, `svg:height 19.543`, `svg:width 19.575`, `svg:x 19.577`, `svg:y 19.581`, `table:end-cell-address 19.631`, `table:end-x 19.636`, `table:end-y 19.637`, `table:table-background 19.733`, `text:anchor-page-number 19.758`, `text:anchor-type 19.759` and `xml:id 19.920`.

The `<draw:page-thumbnail>` element has the following child elements: `<svg:desc> 10.3.18` and `<svg:title> 10.3.17`.

### 10.3.15 `<draw:g>`

The `<draw:g>` element represents a group of drawing shapes.

Drawing shapes contained in a `<draw:g>` element, that is itself contained in a `<draw:a>` element act as hyperlinks using the `xlink:href 19.916` attribute of the containing `<draw:a>` element. If the included drawing shapes are themselves contained within `<draw:a>` elements, then the `xlink:href` attributes of those `<draw:a>` elements act as the hyperlink information for the shapes they contain.

The `<draw:g>` element is usable within the following elements: `<chart:chart> 11.1`, `<draw:a> 10.4.12`, `<draw:g> 10.3.15`, `<draw:page> 10.2.4`, `<draw:text-box> 10.4.3`, `<office:text> 3.4`, `<presentation:notes> 16.19`, `<style:handout-master> 10.2.1`, `<style:master-page> 16.9`, `<table:covered-table-cell> 9.1.5`, `<table:shapes> 9.2.8`, `<table:table-cell> 9.1.4`, `<text:a> 6.1.8`, `<text:deletion> 5.5.5`, `<text:h> 5.1.2`, `<text:index-body> 8.2.2`, `<text:index-title> 8.2.3`, `<text:meta> 6.1.9`, `<text:meta-field> 7.5.19`, `<text:note-body> 6.3.4`, `<text:p> 5.1.3`, `<text:ruby-base> 6.4.2`, `<text:section> 5.4` and `<text:span> 6.1.7`.

The `<draw:g>` element has the following attributes: `draw:caption-id 19.115`, `draw:class-names 19.120`, `draw:id 19.187`, `draw:name 19.197`, `draw:style-name 19.219`, `draw:z-index 19.231`, `presentation:class-names 19.394`, `presentation:style-name 19.426`, `svg:y 19.581`, `table:end-cell-address 19.631`, `table:end-x 19.636`, `table:end-y 19.637`, `table:table-background 19.733`, `text:anchor-page-number 19.758`, `text:anchor-type 19.759` and `xml:id 19.920`.

The `<draw:g>` element has the following child elements: `<dr3d:scene> 10.5.2`, `<draw:a> 10.4.12`, `<draw:caption> 10.3.11`, `<draw:circle> 10.3.8`, `<draw:connector> 10.3.10`, `<draw:control> 10.3.13`, `<draw:custom-shape> 10.6.1`, `<draw:ellipse> 10.3.9`, `<draw:frame> 10.4.2`, `<draw:g> 10.3.15`, `<draw:glue-point> 10.3.16`, `<draw:line> 10.3.3`, `<draw:measure> 10.3.12`, `<draw:page-thumbnail> 10.3.14`, `<draw:path> 10.3.7`, `<draw:polygon> 10.3.5`, `<draw:polyline> 10.3.4`, `<draw:rect> 10.3.2`, `<draw:regular-polygon> 10.3.6`, `<office:event-listeners> 10.3.19`, `<svg:desc> 10.3.18` and `<svg:title> 10.3.17`.

### 10.3.16 `<draw:glue-point>`

The `<draw:glue-point>` element specifies a point in the area of a drawing object to which a connector shape can connect. All drawing objects have four standard glue points located at the center of the four edges of the object's bounding box. Additional glue points may be added to a drawing object by inserting one or more `<draw:glue-point>` elements into a drawing object element.

The `<draw:glue-point>` element is usable within the following elements: `<dr3d:scene>` 10.5.2, `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:connector>` 10.3.10, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:g>` 10.3.15, `<draw:line>` 10.3.3, `<draw:measure>` 10.3.12, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2 and `<draw:regular-polygon>` 10.3.6.

The `<draw:glue-point>` element has the following attributes: `draw:align` 19.111, `draw:escape-direction` 19.147, `draw:id` 19.187, `svg:x` 19.577 and `svg:y` 19.581.

The `<draw:glue-point>` element has no child elements.

### 10.3.17 `<svg:title>`

The `<svg:title>` element specifies a name for a graphic object.

The `<svg:title>` element is usable within the following elements: `<dr3d:scene>` 10.5.2, `<draw:area-circle>` 10.4.13.4, `<draw:area-polygon>` 10.4.13.5, `<draw:area-rectangle>` 10.4.13.3, `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:connector>` 10.3.10, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:g>` 10.3.15, `<draw:layer>` 10.2.3, `<draw:line>` 10.3.3, `<draw:measure>` 10.3.12, `<draw:page>` 10.2.4, `<draw:page-thumbnail>` 10.3.14, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2 and `<draw:regular-polygon>` 10.3.6.

The `<svg:title>` element has no attributes.

The `<svg:title>` element has no child elements.

The `<svg:title>` element has character data content.

### 10.3.18 `<svg:desc>`

The `<svg:desc>` element specifies a prose description of a graphic object that may be used to support accessibility. See appendix D.

The `<svg:desc>` element is usable within the following elements: `<dr3d:scene>` 10.5.2, `<draw:area-circle>` 10.4.13.4, `<draw:area-polygon>` 10.4.13.5, `<draw:area-rectangle>` 10.4.13.3, `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:connector>` 10.3.10, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:g>` 10.3.15, `<draw:layer>` 10.2.3, `<draw:line>` 10.3.3, `<draw:measure>` 10.3.12, `<draw:page>` 10.2.4, `<draw:page-thumbnail>` 10.3.14, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2 and `<draw:regular-polygon>` 10.3.6.

The `<svg:desc>` element has no attributes.

The `<svg:desc>` element has no child elements.

The `<svg:desc>` element has character data content.

### 10.3.19 `<office:event-listeners>`

The `<office:event-listeners>` element represents the attachment of an event listener to a drawing shape, form control or other event-driven object. See 14.5.

The `<office:event-listeners>` element is usable within the following elements: `<draw:area-circle>` 10.4.13.4, `<draw:area-polygon>` 10.4.13.5, `<draw:area-rectangle>` 10.4.13.3, `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:connector>` 10.3.10, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:g>` 10.3.15, `<draw:line>` 10.3.3, `<draw:measure>`

10.3.12, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2, `<draw:regular-polygon>` 10.3.6, `<form:button>` 13.5.15, `<form:checkbox>` 13.5.17, `<form:combobox>` 13.5.11, `<form:date>` 13.5.8, `<form:file>` 13.5.5, `<form:fixed-text>` 13.5.10, `<form:form>` 13.3, `<form:formatted-text>` 13.5.6, `<form:frame>` 13.5.19, `<form:generic-control>` 13.5.25, `<form:grid>` 13.5.22, `<form:hidden>` 13.5.21, `<form:image>` 13.5.16, `<form:image-frame>` 13.5.20, `<form:listbox>` 13.5.13, `<form:number>` 13.5.7, `<form:password>` 13.5.4, `<form:radio>` 13.5.18, `<form:text>` 13.5.2, `<form:textarea>` 13.5.3, `<form:time>` 13.5.9, `<form:value-range>` 13.5.24, `<office:scripts>` 3.12, `<table:content-validation>` 9.4.5, `<text:a>` 6.1.8 and `<text:execute-macro>` 7.7.10.

The `<office:event-listeners>` element has no attributes.

The `<office:event-listeners>` element has the following child elements:

`<presentation:event-listener>` 10.9.2 and `<script:event-listener>` 14.5.2.

## 10.4 Frames

### 10.4.1 General

A frame is a container for enhanced content like text boxes, images or objects. A frame may contain multiple renditions of content. A consumer may choose the representation that it supports best.

Multiple representations may share `<svg:desc>` and `<svg:title>` elements.

Each child element of a frame is a different representation of the same content. The order of content elements reflects the document author's preference for rendering, with the first child element being preferred. That means that consumers should render the first child element that they support. A frame may contain multiple content elements, but shall contain at least one content element.

Within text documents, frames are also used to position content outside the default text flow of a document.

### 10.4.2 `<draw:frame>`

The `<draw:frame>` element represents a frame and serves as the container for elements that may occur in a frame.

Frame formatting properties are stored in styles belonging to the graphic family.

The `<draw:frame>` element is usable within the following elements: `<chart:chart>` 11.1, `<draw:a>` 10.4.12, `<draw:g>` 10.3.15, `<draw:page>` 10.2.4, `<draw:text-box>` 10.4.3, `<office:image>` 3.9, `<office:text>` 3.4, `<presentation:notes>` 16.19, `<style:handout-master>` 10.2.1, `<style:master-page>` 16.9, `<table:covered-table-cell>` 9.1.5, `<table:shapes>` 9.2.8, `<table:table-cell>` 9.1.4, `<text:a>` 6.1.8, `<text:deletion>` 5.5.5, `<text:h>` 5.1.2, `<text:index-body>` 8.2.2, `<text:index-title>` 8.2.3, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:note-body>` 6.3.4, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2, `<text:section>` 5.4 and `<text:span>` 6.1.7.

The `<draw:frame>` element has the following attributes: `draw:caption-id` 19.115, `draw:class-names` 19.120, `draw:copy-of` 19.126, `draw:id` 19.187, `draw:layer` 19.189, `draw:name` 19.197, `draw:style-name` 19.219, `draw:text-style-name` 19.227, `draw:transform` 19.228, `draw:z-index` 19.231, `presentation:class` 19.393, `presentation:class-names` 19.394, `presentation:placeholder` 19.411, `presentation:style-name` 19.426, `presentation:user-transformed` 19.431, `style:rel-height` 19.513, `style:rel-width` 19.514, `svg:height` 19.543,

svg:width 19.575, svg:x 19.577, svg:y 19.581, table:end-cell-address 19.631, table:end-x 19.636, table:end-y 19.637, table:table-background 19.733, text:anchor-page-number 19.758, text:anchor-type 19.759 and xml:id 19.920.

The `<draw:frame>` element has the following child elements: `<draw:applet>` 10.4.7, `<draw:contour-path>` 10.4.11.3, `<draw:contour-polygon>` 10.4.11.2, `<draw:floating-frame>` 10.4.10, `<draw:glue-point>` 10.3.16, `<draw:image>` 10.4.4, `<draw:image-map>` 10.4.13.2, `<draw:object>` 10.4.6.2, `<draw:object-ole>` 10.4.6.3, `<draw:plugin>` 10.4.8, `<draw:text-box>` 10.4.3, `<office:event-listeners>` 10.3.19, `<svg:desc>` 10.3.18, `<svg:title>` 10.3.17 and `<table:table>` 9.1.2.

### 10.4.3 `<draw:text-box>`

The `<draw:text-box>` element represents a text box. This element may be used to place text in a container that is outside of the flow of the document.

The `<draw:text-box>` element is usable within the following element: `<draw:frame>` 10.4.2.

The `<draw:text-box>` element has the following attributes: `draw:chain-next-name` 19.118, `draw:corner-radius` 19.127, `fo:max-height` 19.238, `fo:max-width` 19.239, `fo:min-height` 19.240, `fo:min-width` 19.241, `svg:rx` 19.554, `svg:ry` 19.555, `text:id` 19.815 and `xml:id` 19.920.

The `<draw:text-box>` element has the following child elements: `<dr3d:scene>` 10.5.2, `<draw:a>` 10.4.12, `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:connector>` 10.3.10, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:g>` 10.3.15, `<draw:line>` 10.3.3, `<draw:measure>` 10.3.12, `<draw:page-thumbnail>` 10.3.14, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2, `<draw:regular-polygon>` 10.3.6, `<table:table>` 9.1.2, `<text:alphabetical-index>` 8.8, `<text:bibliography>` 8.9, `<text:change>` 5.5.8.4, `<text:change-end>` 5.5.8.3, `<text:change-start>` 5.5.8.2, `<text:h>` 5.1.2, `<text:illustration-index>` 8.4, `<text:list>` 5.3.1, `<text:numbered-paragraph>` 5.3.6, `<text:object-index>` 8.6, `<text:p>` 5.1.3, `<text:section>` 5.4, `<text:soft-page-break>` 5.6, `<text:table-index>` 8.5, `<text:table-of-content>` 8.3 and `<text:user-index>` 8.7.

### 10.4.4 `<draw:image>`

The `<draw:image>` element represents an image. An image can be either:

- A link to an external resource
- or
- Embedded in the document

The `<draw:image>` element may have text content. Text content is displayed in addition to the image data.

**Note:** While the image data may have an arbitrary format, vector graphics should be stored in the [SVG] format and bitmap graphics in the [PNG] format.

The `<draw:image>` element is usable within the following element: `<draw:frame>` 10.4.2.

The `<draw:image>` element has the following attributes: `draw:filter-name` 19.170, `draw:mime-type` 19.191, `xlink:actuate` 19.915, `xlink:href` 19.916, `xlink:show` 19.917, `xlink:type` 19.919 and `xml:id` 19.920.

The `<draw:image>` element has the following child elements: `<office:binary-data>` 10.4.5, `<table:table>` 9.1.2, `<text:list>` 5.3.1 and `<text:p>` 5.1.3.

## 10.4.5 `<office:binary-data>`

The `<office:binary-data>` element contains image data in BASE64 encoding (as defined in [RFC2045]). If this element is present, an `xlink:href` 19.916 attribute in its parent element shall be ignored.

**Note:** The image data may be in an arbitrary data format. Consumers have to analyze the image data to determine the data format that is used.

The `<office:binary-data>` element is usable within the following elements: `<draw:fill-image>` 16.42.6, `<draw:image>` 10.4.4, `<draw:object-ole>` 10.4.6.3, `<style:background-image>` 17.3 and `<text:list-level-style-image>` 16.35.

The `<office:binary-data>` element has no attributes.

The `<office:binary-data>` element has no child elements.

The `<office:binary-data>` element has content of data type `base64Binary` 18.2.

## 10.4.6 Embedded Objects

### 10.4.6.1 General

A document in OpenDocument format can contain two types of embedded objects, as follows:

- Objects that have an OpenDocument representation.
- Objects that do not have an OpenDocument representation.

**Note:** OLE objects are an example of objects without an OpenDocument representation. See [OLE].

### 10.4.6.2 `<draw:object>`

The `<draw:object>` element represents objects that have a OpenDocument representation.

Objects with an OpenDocument representation can either be:

- Contained in the same package as a document, in which case the Object is a sub-document within the package. The `xlink:href` attribute of this element references that folder. 1.3
- Contained in a separate OpenDocument document instance. The `xlink:href` attribute of this element references that document's package.
- Contained in a child `<office:document>` element of this element.
- Contained in a child `<math:math>` element of this element.

**Note:** An image representation of an object should be included in a frame in addition to the object itself.

The `<draw:object>` element is usable within the following element: `<draw:frame>` 10.4.2.

The `<draw:object>` element has the following attributes: `draw:notify-on-update-of-ranges` 19.200, `xlink:actuate` 19.915, `xlink:href` 19.916, `xlink:show` 19.917, `xlink:type` 19.919 and `xml:id` 19.920.

The `<draw:object>` element has the following child elements: `<math:math>` 14.6 and `<office:document>` 3.1.2.

### 10.4.6.3 <draw:object-ole>

The `<draw:object-ole>` element represents objects that do not have an OpenDocument representation.

Objects without an OpenDocument representation can either be:

- Contained in the same package as a document, in which case it is contained in a file within the package. The `xlink:href` attribute of this element references that file.
- Contained in a separate file. The `xlink:href` attribute of this element references that file.
- Contained in a child `<office:binary-data>` element of this element.

The `<draw:object-ole>` element is usable within the following element: `<draw:frame>` 10.4.2.

The `<draw:object-ole>` element has the following attributes: `draw:class-id` 19.119, `xlink:actuate` 19.915, `xlink:href` 19.916, `xlink:show` 19.917, `xlink:type` 19.919 and `xml:id` 19.920.

The `<draw:object-ole>` element has the following child element: `<office:binary-data>` 10.4.5.

### 10.4.7 <draw:applet> (Deprecated)

The `<draw:applet>` element represents an applet that is embedded in a document. Its semantics are the same as the `<applet>` element in HTML. §13.4 of [HTML4]. This element shall have either a `draw:code` or `draw:object` attribute.

The applet name, width, height, margins, title and description are specified by the `<draw:applet>` element's `<draw:frame>` parent element. The `<draw:applet>` element does not support character data content.

The `<draw:applet>` element is usable within the following element: `<draw:frame>` 10.4.2.

The `<draw:applet>` element has the following attributes: `draw:archive` 19.113, `draw:code` 19.121, `draw:may-script` 19.195, `draw:object` 19.201, `xlink:actuate` 19.915, `xlink:href` 19.916, `xlink:show` 19.917, `xlink:type` 19.919 and `xml:id` 19.920.

The `<draw:applet>` element has the following child element: `<draw:param>` 10.4.9.

### 10.4.8 <draw:plugin>

The `<draw:plugin>` element represents an external application that processes data with a specified media type.

How OpenDocument consumers interact with external applications is not defined by this specification.

**Note:** Consumers are responsible for any and all security issues that may arise from interaction with an external application referenced by a `<draw:plugin>` element.

The `<draw:plugin>` element is usable within the following element: `<draw:frame>` 10.4.2.

The `<draw:plugin>` element has the following attributes: `draw:mime-type` 19.191, `xlink:actuate` 19.915, `xlink:href` 19.916, `xlink:show` 19.917, `xlink:type` 19.919 and `xml:id` 19.920.

The `<draw:plugin>` element has the following child element: `<draw:param>` 10.4.9.

## 10.4.9 <draw:param>

The `<draw:param>` element contains parameters that are passed to an applet or plugin when they are initialized.

The `<draw:param>` element is usable within the following elements: `<draw:applet>` 10.4.7 and `<draw:plugin>` 10.4.8.

The `<draw:param>` element has the following attributes: `draw:name` 19.197 and `draw:value` 19.230.

The `<draw:param>` element has no child elements.

## 10.4.10 <draw:floating-frame>

The `<draw:floating-frame>` element represents a frame that is embedded in a document.

Floating frames cannot be transformed as described in section 19.228.

The `<draw:floating-frame>` element is usable within the following element: `<draw:frame>` 10.4.2.

The `<draw:floating-frame>` element has the following attributes: `draw:frame-name` 19.172, `xlink:actuate` 19.915, `xlink:href` 19.916, `xlink:show` 19.917, `xlink:type` 19.919 and `xml:id` 19.920.

The `<draw:floating-frame>` element has no child elements.

## 10.4.11 Contours

### 10.4.11.1 General

The `<draw:contour-polygon>` and `<draw:contour-path>` elements are used to specify contours to be applied to object or images.

Consumers should support pixel lengths for the `svg:width` and `svg:height` attributes of the `<draw:contour-polygon>` and `<draw:contour-path>` elements.

**Note:** A contour of an object or image establishes an outer boundary around an object or image. One use would be the wrapping of text around a contour specified for an image, if `style:wrap-contour="true"` and a contour is specified.

### 10.4.11.2 <draw:contour-polygon>

The `<draw:contour-polygon>` element specifies a contour using a polygon.

The `<draw:contour-polygon>` element is usable within the following element: `<draw:frame>` 10.4.2.

The `<draw:contour-polygon>` element has the following attributes: `draw:points` 19.206, `draw:recreate-on-edit` 19.208, `svg:height` 19.543, `svg:viewBox` 19.574 and `svg:width` 19.575.

The `<draw:contour-polygon>` element has no child elements.

### 10.4.11.3 <draw:contour-path>

The `<draw:contour-path>` element specifies a contour using a path.

The `<draw:contour-path>` element is usable within the following element: `<draw:frame>` 10.4.2.

The `<draw:contour-path>` element has the following attributes: `draw:recreate-on-edit` 19.208, `svg:d` 19.530, `svg:height` 19.543, `svg:viewBox` 19.574 and `svg:width` 19.575.

The `<draw:contour-path>` element has no child elements.

### 10.4.12 `<draw:a>`

The `<draw:a>` element enables both frames and drawing shapes to act as hyperlinks.

The `<draw:a>` element contains a drawing shape that should behave as a hyperlink.

The `<draw:a>` element is usable within the following elements: `<chart:chart>` 11.1, `<draw:g>` 10.3.15, `<draw:page>` 10.2.4, `<draw:text-box>` 10.4.3, `<office:text>` 3.4, `<presentation:notes>` 16.19, `<style:handout-master>` 10.2.1, `<style:master-page>` 16.9, `<table:covered-table-cell>` 9.1.5, `<table:shapes>` 9.2.8, `<table:table-cell>` 9.1.4, `<text:a>` 6.1.8, `<text:deletion>` 5.5.5, `<text:h>` 5.1.2, `<text:index-body>` 8.2.2, `<text:index-title>` 8.2.3, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:note-body>` 6.3.4, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2, `<text:section>` 5.4 and `<text:span>` 6.1.7.

The `<draw:a>` element has the following attributes: `office:name` 19.380, `office:server-map` 19.382, `office:target-frame-name` 19.385, `office:title` 19.387, `xlink:actuate` 19.915, `xlink:href` 19.916, `xlink:show` 19.917, `xlink:type` 19.919 and `xml:id` 19.920.

The `<draw:a>` element has the following child elements: `<dr3d:scene>` 10.5.2, `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:connector>` 10.3.10, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:g>` 10.3.15, `<draw:line>` 10.3.3, `<draw:measure>` 10.3.12, `<draw:page-thumbnail>` 10.3.14, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2 and `<draw:regular-polygon>` 10.3.6.

### 10.4.13 Client Side Image Maps

#### 10.4.13.1 General

A client side image map is a collection of hyperlinks that are associated with graphic elements. The image map is a sequence of image map elements. Each image map element associates a hyperlink with an area. The area can be a rectangle, circle or polygon.

#### 10.4.13.2 `<draw:image-map>`

The `<draw:image-map>` element represents an image map.

Image map elements are specified at absolute positions relative to the original size of its associated graphical element. OpenDocument consumer shall scale the image map to match the current size of the image.

The `<draw:image-map>` element is usable within the following element: `<draw:frame>` 10.4.2.

The `<draw:image-map>` element has no attributes.

The `<draw:image-map>` element has the following child elements: `<draw:area-circle>` 10.4.13.4, `<draw:area-polygon>` 10.4.13.5 and `<draw:area-rectangle>` 10.4.13.3.

#### 10.4.13.3 `<draw:area-rectangle>`

The `<draw:area-rectangle>` element specifies a rectangular image map area.

The `<draw:area-rectangle>` element is usable within the following element:  
`<draw:image-map>` 10.4.13.2.

The `<draw:area-rectangle>` element has the following attributes: `draw:nohref` 19.199, `office:name` 19.380, `office:target-frame-name` 19.385, `svg:height` 19.543, `svg:width` 19.575, `svg:x` 19.577, `svg:y` 19.581, `xlink:href` 19.916, `xlink:show` 19.917 and `xlink:type` 19.919.

The `<draw:area-rectangle>` element has the following child elements:  
`<office:event-listeners>` 10.3.19, `<svg:desc>` 10.3.18 and `<svg:title>` 10.3.17.

#### 10.4.13.4 `<draw:area-circle>`

The `<draw:area-circle>` element represents a circular image map area.

The `<draw:area-circle>` element is usable within the following element: `<draw:image-map>` 10.4.13.2.

The `<draw:area-circle>` element has the following attributes: `draw:nohref` 19.199, `office:name` 19.380, `office:target-frame-name` 19.385, `svg:cx` 19.528, `svg:cy` 19.529, `svg:r` 19.553, `xlink:href` 19.916, `xlink:show` 19.917 and `xlink:type` 19.919.

The `<draw:area-circle>` element has the following child elements: `<office:event-listeners>` 10.3.19, `<svg:desc>` 10.3.18 and `<svg:title>` 10.3.17.

#### 10.4.13.5 `<draw:area-polygon>`

The `<draw:area-polygon>` element specifies a polygonal image map area. A polygonal image map area has the following components:

- A bounding box.  
The bounding box establishes the reference frame for the view box and the polygon point sequence. The reference frame enables the coordinates to be translated into absolute coordinates.
- A view box.  
The view box attribute establishes a coordinate system for the point sequence. The view box avoids the need to record every point of the point sequence as absolute coordinates with length and unit of measurement.
- A sequence of points in view-box coordinates in the `draw:points` attribute.

**Note:** For more information about how to represent polygons, see  
`<draw:polygon>` 10.3.5.

The `<draw:area-polygon>` element is usable within the following element:  
`<draw:image-map>` 10.4.13.2.

The `<draw:area-polygon>` element has the following attributes: `draw:nohref` 19.199, `draw:points` 19.206, `office:name` 19.380, `office:target-frame-name` 19.385, `svg:height` 19.543, `svg:viewBox` 19.574, `svg:width` 19.575, `svg:x` 19.577, `svg:y` 19.581, `xlink:href` 19.916, `xlink:show` 19.917 and `xlink:type` 19.919.

The `<draw:area-polygon>` element has the following child elements: `<office:event-listeners>` 10.3.19, `<svg:desc>` 10.3.18 and `<svg:title>` 10.3.17.

## 10.5 3D Shapes

### 10.5.1 General

3D shapes are used to define three-dimensional coordinate systems and geometrical figures within them. All OpenDocument 3D coordinate systems are right-handed. Geometry is represented by a tree of scenes. Each scene may contain zero or more 3D shapes, with a scene being a special case of shape. The root scene defines the world coordinate system. Each 3D shape may define a local coordinate system relative to its parent 3D scene.

### 10.5.2 <dr3d:scene>

The `<dr3d:scene>` element is the only element that can contain three-dimensional shapes. Like the `<draw:g>` element it groups shapes, but it also defines the projection, lighting, and other rendering details for the shapes inside the scene.

The attributes `svg:x`, `svg:y`, `svg:width` and `svg:height` are only evaluated for the outermost `<dr3d:scene>` element. The attributes `svg:x`, `svg:y`, `svg:width` and `svg:height` specify together a rectangular viewport into which the projection of the scene is drawn. The projection is scaled non-uniformly such that the bounding box of the projected elements matches the viewport rectangle.

The attributes `svg:x` and `svg:y` refer to the coordinate system of the parent element. They specify the top-left point of the viewport rectangle.

**Note:** To display a uniformly scaled projection, the producer has to generate suitable values for `svg:width` and `svg:height`.

The `<dr3d:scene>` element is usable within the following elements: `<chart:chart>` 11.1, `<dr3d:scene>` 10.5.2, `<draw:a>` 10.4.12, `<draw:g>` 10.3.15, `<draw:page>` 10.2.4, `<draw:text-box>` 10.4.3, `<office:text>` 3.4, `<presentation:notes>` 16.19, `<style:handout-master>` 10.2.1, `<style:master-page>` 16.9, `<table:covered-table-cell>` 9.1.5, `<table:shapes>` 9.2.8, `<table:table-cell>` 9.1.4, `<text:a>` 6.1.8, `<text:deletion>` 5.5.5, `<text:h>` 5.1.2, `<text:index-body>` 8.2.2, `<text:index-title>` 8.2.3, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:note-body>` 6.3.4, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2, `<text:section>` 5.4 and `<text:span>` 6.1.7.

The `<dr3d:scene>` element has the following attributes: `dr3d:ambient-color` 19.92, `dr3d:distance` 19.96, `dr3d:focal-length` 19.98, `dr3d:lighting-mode` 19.99, `dr3d:projection` 19.102, `dr3d:shade-mode` 19.103, `dr3d:shadow-slant` 19.104, `dr3d:transform` 19.107, `dr3d:vpn` 19.108, `dr3d:vrp` 19.109, `dr3d:vup` 19.110, `draw:caption-id` 19.115, `draw:class-names` 19.120, `draw:id` 19.187, `draw:layer` 19.189, `draw:name` 19.197, `draw:style-name` 19.219, `draw:z-index` 19.231, `presentation:class-names` 19.394, `presentation:style-name` 19.426, `svg:height` 19.543, `svg:width` 19.575, `svg:x` 19.577, `svg:y` 19.581, `table:end-cell-address` 19.631, `table:end-x` 19.636, `table:end-y` 19.637, `table:table-background` 19.733, `text:anchor-page-number` 19.758, `text:anchor-type` 19.759 and `xml:id` 19.920.

The `<dr3d:scene>` element has the following child elements: `<dr3d:cube>` 10.5.4, `<dr3d:extrude>` 10.5.6, `<dr3d:light>` 10.5.3, `<dr3d:rotate>` 10.5.7, `<dr3d:scene>` 10.5.2, `<dr3d:sphere>` 10.5.5, `<draw:glue-point>` 10.3.16, `<svg:desc>` 10.3.18 and `<svg:title>` 10.3.17.

### 10.5.3 <dr3d:light>

The `<dr3d:light>` element represents a light inside a scene.

**Note:** There may be several lights, but consumers may limit the number of lights per scene to 8.

The `<dr3d:light>` element is usable within the following elements: `<chart:plot-area>` 11.5 and `<dr3d:scene>` 10.5.2.

The `<dr3d:light>` element has the following attributes: `dr3d:diffuse-color` 19.94, `dr3d:direction` 19.95, `dr3d:enabled` 19.97 and `dr3d:specular` 19.106.

The `<dr3d:light>` element has no child elements.

#### 10.5.4 `<dr3d:cube>`

The `<dr3d:cube>` element represents a three-dimensional cube shape.

The `<dr3d:cube>` element is usable within the following element: `<dr3d:scene>` 10.5.2.

The `<dr3d:cube>` element has the following attributes: `dr3d:max-edge` 19.100, `dr3d:min-edge` 19.101, `dr3d:transform` 19.107, `draw:class-names` 19.120, `draw:id` 19.187, `draw:layer` 19.189, `draw:name` 19.197, `draw:style-name` 19.219, `draw:z-index` 19.231, `presentation:class-names` 19.394, `presentation:style-name` 19.426 and `xml:id` 19.920.

The `<dr3d:cube>` element has no child elements.

#### 10.5.5 `<dr3d:sphere>`

The `<dr3d:sphere>` element represents a three-dimensional sphere shape.

The `<dr3d:sphere>` element is usable within the following element: `<dr3d:scene>` 10.5.2.

The `<dr3d:sphere>` element has the following attributes: `dr3d:center` 19.93, `dr3d:size` 19.105, `dr3d:transform` 19.107, `draw:class-names` 19.120, `draw:id` 19.187, `draw:layer` 19.189, `draw:name` 19.197, `draw:style-name` 19.219, `draw:z-index` 19.231, `presentation:class-names` 19.394, `presentation:style-name` 19.426 and `xml:id` 19.920.

The `<dr3d:sphere>` element has no child elements.

#### 10.5.6 `<dr3d:extrude>`

The `<dr3d:extrude>` element represents a three-dimensional extruded object based on the given polygon. The 3D geometry is formed by extruding a polygon perpendicular to the plane of the given polygon by a depth (see `dr3d:depth`, 20.79), creating a front face, a back face and side faces. The side faces are formed by connecting all corresponding single edges of the front and back faces. When no depth is given, the polygon forms a shape without depth. The closing of the front face depends on the value of the `dr3d:close-front` attribute 20.78. The closing of the back face depends on the value of the `dr3d:close-back` attribute 20.77. The size of the back face can be relative to the size of the front face when the `dr3d:back-scale` attribute is used. The side faces defined by the extrusion of the shape are always closed.

The `<dr3d:extrude>` element is usable within the following element: `<dr3d:scene>` 10.5.2.

The `<dr3d:extrude>` element has the following attributes: `dr3d:transform` 19.107, `draw:class-names` 19.120, `draw:id` 19.187, `draw:layer` 19.189, `draw:name` 19.197, `draw:style-name` 19.219, `draw:z-index` 19.231, `presentation:class-names` 19.394, `presentation:style-name` 19.426, `svg:d` 19.530, `svg:viewBox` 19.574 and `xml:id` 19.920.

The `<dr3d:extrude>` element has no child elements.

### 10.5.7 <dr3d:rotate>

The `<dr3d:rotate>` element represents a three-dimensional rotation shape based on the given polygon. The 3D geometry is defined by rotating the XY-plane with the polygon around the Y-axis of the local coordinate system. This rotation is linearly interpolated in the given number of steps (`dr3d:horizontal-segments` 20.85) around the given angle (`dr3d:end-angle` 20.84). The side faces are formed by connecting all corresponding single edges of a neighboring pairs of rotated polygons. By default, the rotation uses a full rotation (360 degrees, `dr3d:end-angle` attribute 20.84). In that case the 3D geometry is implicitly closed and no front face or back face is created (unless they are of different sizes). If the rotation is different, a front face is created using the first, unrotated polygon and a back face is created using the last rotated polygon. The closing of a front face also depends on the value of the `dr3d:close-front` attribute 20.78 and the closing of a back face on the value of the `dr3d:close-back` 20.77 attribute. The size of the back face can be different than the size of the front face when the `dr3d:back-scale` attribute 20.75 is used.

The `<dr3d:rotate>` element is usable within the following element: `<dr3d:scene>` 10.5.2.

The `<dr3d:rotate>` element has the following attributes: `dr3d:transform` 19.107, `draw:class-names` 19.120, `draw:id` 19.187, `draw:layer` 19.189, `draw:name` 19.197, `draw:style-name` 19.219, `draw:z-index` 19.231, `presentation:class-names` 19.394, `presentation:style-name` 19.426, `svg:d` 19.530, `svg:viewBox` 19.574 and `xml:id` 19.920.

The `<dr3d:rotate>` element has no child elements.

## 10.6 Custom Shape

### 10.6.1 <draw:custom-shape>

The `<draw:custom-shape>` element represents a complex figure. It supports font geometry effects and extrusion. A custom shape may have a geometry that influences its shape.

The `<draw:custom-shape>` element is usable within the following elements:

`<chart:chart>` 11.1, `<draw:a>` 10.4.12, `<draw:g>` 10.3.15, `<draw:page>` 10.2.4, `<draw:text-box>` 10.4.3, `<office:text>` 3.4, `<presentation:notes>` 16.19, `<style:handout-master>` 10.2.1, `<style:master-page>` 16.9, `<table:covered-table-cell>` 9.1.5, `<table:shapes>` 9.2.8, `<table:table-cell>` 9.1.4, `<text:a>` 6.1.8, `<text:deletion>` 5.5.5, `<text:h>` 5.1.2, `<text:index-body>` 8.2.2, `<text:index-title>` 8.2.3, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:note-body>` 6.3.4, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2, `<text:section>` 5.4 and `<text:span>` 6.1.7.

The `<draw:custom-shape>` element has the following attributes: `draw:caption-id` 19.115, `draw:class-names` 19.120, `draw:data` 19.131, `draw:engine` 19.146, `draw:id` 19.187, `draw:layer` 19.189, `draw:name` 19.197, `draw:style-name` 19.219, `draw:text-style-name` 19.227, `draw:transform` 19.228, `draw:z-index` 19.231, `presentation:class-names` 19.394, `presentation:style-name` 19.426, `svg:height` 19.543, `svg:width` 19.575, `svg:x` 19.577, `svg:y` 19.581, `table:end-cell-address` 19.631, `table:end-x` 19.636, `table:end-y` 19.637, `table:table-background` 19.733, `text:anchor-page-number` 19.758, `text:anchor-type` 19.759 and `xml:id` 19.920.

The `<draw:custom-shape>` element has the following child elements: `<draw:enhanced-geometry>` 10.6.2, `<draw:glue-point>` 10.3.16, `<office:event-listeners>` 10.3.19, `<svg:desc>` 10.3.18, `<svg:title>` 10.3.17, `<table:table>` 9.1.2, `<text:list>` 5.3.1 and `<text:p>` 5.1.3.

## 10.6.2 <draw:enhanced-geometry>

The `<draw:enhanced-geometry>` element contains the geometry for a `<draw:custom-shape>` element if its `draw:engine` attribute has been omitted.

The `<draw:enhanced-geometry>` element is usable within the following element:  
`<draw:custom-shape>` 10.6.1.

The `<draw:enhanced-geometry>` element has the following attributes:  
`dr3d:projection` 19.102, `dr3d:shade-mode` 19.103, `draw:concentric-gradient-fill-allowed` 19.124, `draw:enhanced-path` 19.145, `draw:extrusion` 19.148, `draw:extrusion-allowed` 19.149, `draw:extrusion-brightness` 19.150, `draw:extrusion-color` 19.151, `draw:extrusion-depth` 19.152, `draw:extrusion-diffusion` 19.153, `draw:extrusion-first-light-direction` 19.154, `draw:extrusion-first-light-harsh` 19.155, `draw:extrusion-first-light-level` 19.156, `draw:extrusion-light-face` 19.157, `draw:extrusion-metal` 19.158, `draw:extrusion-metal-type` 19.923, `draw:extrusion-number-of-line-segments` 19.159, `draw:extrusion-origin` 19.160, `draw:extrusion-rotation-angle` 19.161, `draw:extrusion-rotation-center` 19.162, `draw:extrusion-second-light-direction` 19.163, `draw:extrusion-second-light-harsh` 19.164, `draw:extrusion-second-light-level` 19.165, `draw:extrusion-shininess` 19.166, `draw:extrusion-skew` 19.167, `draw:extrusion-specularity` 19.168, `draw:extrusion-viewpoint` 19.169, `draw:glue-point-leaving-directions` 19.173, `draw:glue-point-type` 19.174, `draw:glue-points` 19.175, `draw:mirror-horizontal` 19.192, `draw:mirror-vertical` 19.193, `draw:modifiers` 19.196, `draw:path-stretchpoint-x` 19.204, `draw:path-stretchpoint-y` 19.205, `draw:text-areas` 19.220, `draw:text-path` 19.221, `draw:text-path-allowed` 19.222, `draw:text-path-mode` 19.223, `draw:text-path-same-letter-heights` 19.224, `draw:text-path-scale` 19.225, `draw:text-rotate-angle` 19.226, `draw:type` 19.229 and `svg:viewBox` 19.574.

The `<draw:enhanced-geometry>` element has the following child elements:  
`<draw:equation>` 10.2.5 and `<draw:handle>` 10.6.3.

## 10.6.3 <draw:handle>

The `<draw:handle>` element represents a single interaction handle for a drawing shape. It is either an XY-handle or a polar-handle. If a `draw:handle-position-x` 19.924 attribute together with a `draw:handle-position-y` 19.925 attribute is present, it is an XY-handle. If a `draw:handle-polar-pole-x` 19.926 attribute together with a `draw:handle-polar-pole-y` 19.927 attribute is present, it is a polar-handle. Exactly one of these two pairs shall exist.

The position of an XY-handle is specified in Cartesian coordinates. The x-coordinate is specified by the `draw:handle-position-x` attribute, the y-coordinate is specified by the `draw:handle-position-y` attribute.

The position of a polar-handle is specified as tuple of radius and polar angle in a polar coordinate system. The `draw:handle-polar-pole-x` attribute specifies the x-coordinate and the `draw:handle-polar-pole-y` attribute specifies the y-coordinate of the pole of the polar coordinate system. The `draw:handle-polar-radius` 19.928 attribute specifies the radius and the `draw:handle-polar-angle` 19.929 attribute defines the polar angle of the position of the polar-handle.

The polar axis of the polar coordinate system points horizontally from the pole to the right. The polar angle is measured clockwise from the polar axis.

Note: The y-axis in the enhanced geometry of a custom shape points down. Therefore the measurement is clockwise and not counter-clockwise.

Parameters in the values of the `draw:handle-position-x`, `draw:handle-position-y`, `draw:handle-polar-pole-x`, `draw:handle-polar-pole-y`, `draw:handle-polar-radius` and `draw:handle-polar-angle` attributes use a common syntax.

A handle-parameter can be a number or it can have one of the following enhancements:

- A “?” (U+003F, QUESTION MARK) is used to mark the beginning of a formula name. The result of the `<draw:equation>` 10.2.5 element's `draw:formula` 19.171 attribute whose `draw:name` 19.197.8 attribute is the formula name that is used as the parameter value.
- If “\$” (U+0024, DOLLAR SIGN) precedes an integer value, the value is an index to a `draw:modifiers` 19.196 attribute. The corresponding modifier value is used as parameter value.
- Instead of a number a value can also be one of the following constants:

<b>Constant</b>	<b>Description</b>
<code>bottom</code>	The bottom coordinate of the view box as specified by the <code>svg:viewBox</code> 19.574 attribute is used.
<code>hasfill</code>	If the shape has a fill style, a value of 1 is used.
<code>hasstroke</code>	If the shape has a line style, a value of 1 is used.
<code>height</code>	The view box height as specified by the <code>svg:viewBox</code> 19.574 attribute is used.
<code>logheight</code>	The height in 1/100th mm as specified by the <code>svg:height</code> 19.543 attribute is used.
<code>logwidth</code>	The width in 1/100th mm as specified by the <code>svg:width</code> 19.575 attribute is used.
<code>left</code>	The left coordinate of the view box as specified by the <code>svg:viewBox</code> 19.574 attribute is used.
<code>right</code>	The right coordinate of the view box as specified by the <code>svg:viewBox</code> 19.574 attribute is used.
<code>top</code>	The top coordinate of the view box as specified by the <code>svg:viewBox</code> 19.574 attribute is used.
<code>width</code>	The view box width as specified by the <code>svg:viewBox</code> 19.574 attribute is used.
<code>xstretch</code>	The value of the <code>draw:path-stretchpoint-x</code> 19.204 attribute is used.
<code>ystretch</code>	The value of the <code>draw:path-stretchpoint-y</code> 19.205 attribute is used.

A handle-parameter is formally defined as:

```
handle-parameter ::= formula | modifier | constant | number
formula ::= '?' name
modifier ::= '$' integer
constant ::= 'left' | 'top' | 'right' | 'bottom' |
  'xstretch' | 'ystretch' | 'hasstroke' | 'hasfill' |
  'width' | 'height' | 'logwidth' | 'logheight'
```

```

number ::= sign? float | sign? integer
float ::= fractional exponent? | integer exponent
fractional ::= integer? '.' integer | integer '.'
exponent ::= ( 'e' | 'E' ) sign? integer
sign ::= '+' | '-'
integer ::= [0-9] +
name ::= [^ ] +

```

The `<draw:handle>` element is usable within the following element: `<draw:enhanced-geometry>` 10.6.2.

The `<draw:handle>` element has the following attributes: `draw:handle-mirror-horizontal` 19.176, `draw:handle-mirror-vertical` 19.177, `draw:handle-polar` 19.178, `draw:handle-polar-angle` 19.929, `draw:handle-polar-pole-x` 19.926, `draw:handle-polar-pole-y` 19.927, `draw:handle-polar-radius` 19.928, `draw:handle-position` 19.179, `draw:handle-position-x` 19.924, `draw:handle-position-y` 19.925, `draw:handle-radius-range-maximum` 19.180, `draw:handle-radius-range-minimum` 19.181, `draw:handle-range-x-maximum` 19.182, `draw:handle-range-x-minimum` 19.183, `draw:handle-range-y-maximum` 19.184, `draw:handle-range-y-minimum` 19.185 and `draw:handle-switched` 19.186.

The `<draw:handle>` element has no child elements.

## 10.7 Presentation Shapes

Presentation shapes are shapes on a draw page that are part of a presentation page layout 16.43 element. They use styles with a style family value of `presentation` 19.394, unlike drawing shapes which use styles with a style family value of `graphic` 19.120. Presentation shapes can be empty, acting only as placeholders. If a draw page's presentation layout is changed, all presentation shapes contained in that draw page are adapted automatically.

Standard drawing shapes can also be used in presentations. The `presentation:class` 19.393 attribute distinguishes presentation shapes from drawing shapes. Unlike presentation shapes, standard drawing shapes are not adapted if the presentation page layout is changed.

## 10.8 Presentation Animations (deprecated)

### 10.8.1 `<presentation:animations>` (deprecated)

The `<presentation:animations>` element is a container for animation effects. Animation effects are executed when a page represented by a `<draw:page>` 10.2.4 element containing this element is displayed.

A single shape may be the subject of multiple effects.

The `<presentation:animations>` element is usable within the following element: `<draw:page>` 10.2.4.

The `<presentation:animations>` element has no attributes.

The `<presentation:animations>` element has the following child elements: `<presentation:animation-group>` 10.8.9, `<presentation:dim>` 10.8.7, `<presentation:hide-shape>` 10.8.5, `<presentation:hide-text>` 10.8.6, `<presentation:play>` 10.8.8, `<presentation:show-shape>` 10.8.3 and `<presentation:show-text>` 10.8.4.

## 10.8.2 <presentation:sound> (moved to 10.9.3)

### 10.8.3 <presentation:show-shape> (deprecated)

The <presentation:show-shape> element specifies when and how a shape becomes visible. Prior to execution of the effect defined by this element the shape is invisible.

The <presentation:show-shape> element is usable within the following elements: <presentation:animation-group> 10.8.9 and <presentation:animations> 10.8.1.

The <presentation:show-shape> element has the following attributes: draw:shape-id 19.210, presentation:delay 19.395, presentation:direction 19.396, presentation:effect 19.398, presentation:path-id 19.409, presentation:speed 19.421 and presentation:start-scale 19.422.

The <presentation:show-shape> element has the following child element: <presentation:sound> 10.8.2.

### 10.8.4 <presentation:show-text> (deprecated)

The <presentation:show-text> element specifies when and how the text within a shape becomes visible. Prior to execution of the effect defined by this element the shape and the text it contains are invisible.

The <presentation:show-text> element is usable within the following elements: <presentation:animation-group> 10.8.9 and <presentation:animations> 10.8.1.

The <presentation:show-text> element has the following attributes: draw:shape-id 19.210, presentation:delay 19.395, presentation:direction 19.396, presentation:effect 19.398, presentation:path-id 19.409, presentation:speed 19.421 and presentation:start-scale 19.422.

The <presentation:show-text> element has the following child element: <presentation:sound> 10.8.2.

### 10.8.5 <presentation:hide-shape> (deprecated)

The <presentation:hide-shape> element specifies when and how a shape becomes invisible. Prior to execution of the effect defined by this element the shape is visible.

The <presentation:hide-shape> element is usable within the following elements: <presentation:animation-group> 10.8.9 and <presentation:animations> 10.8.1.

The <presentation:hide-shape> element has the following attributes: draw:shape-id 19.210, presentation:delay 19.395, presentation:direction 19.396, presentation:effect 19.398, presentation:path-id 19.409, presentation:speed 19.421 and presentation:start-scale 19.422.

The <presentation:hide-shape> element has the following child element: <presentation:sound> 10.8.2.

### 10.8.6 <presentation:hide-text> (deprecated)

The <presentation:hide-text> element specifies when and how the text within a shape becomes invisible. Prior to execution of the effect specified by this element the shape and the text it contains are visible.

The `<presentation:hide-text>` element is usable within the following elements: `<presentation:animation-group>` 10.8.9 and `<presentation:animations>` 10.8.1.

The `<presentation:hide-text>` element has the following attributes: `draw:shape-id` 19.210, `presentation:delay` 19.395, `presentation:direction` 19.396, `presentation:effect` 19.398, `presentation:path-id` 19.409, `presentation:speed` 19.421 and `presentation:start-scale` 19.422.

The `<presentation:hide-text>` element has the following child element: `<presentation:sound>` 10.8.2.

## 10.8.7 `<presentation:dim>` (deprecated)

The `<presentation:dim>` element specifies a fill color for a shape.

The `<presentation:dim>` element is usable within the following elements: `<presentation:animation-group>` 10.8.9 and `<presentation:animations>` 10.8.1.

The `<presentation:dim>` element has the following attributes: `draw:color` 19.122 and `draw:shape-id` 19.210.

The `<presentation:dim>` element has the following child element: `<presentation:sound>` 10.8.2.

## 10.8.8 `<presentation:play>` (deprecated)

The `<presentation:play>` element specifies the presentation speed of animation of a shape.

The `<presentation:play>` element is usable within the following elements: `<presentation:animation-group>` 10.8.9 and `<presentation:animations>` 10.8.1.

The `<presentation:play>` element has the following attributes: `draw:shape-id` 19.210 and `presentation:speed` 19.421.

The `<presentation:play>` element has no child elements.

## 10.8.9 `<presentation:animation-group>` (deprecated)

The `<presentation:animation-group>` element contains elements that specify multiple animation effects. Containment results in all effects occurring at the same time.

The `<presentation:animation-group>` element is usable within the following element: `<presentation:animations>` 10.8.1.

The `<presentation:animation-group>` element has no attributes.

The `<presentation:animation-group>` element has the following child elements: `<presentation:dim>` 10.8.7, `<presentation:hide-shape>` 10.8.5, `<presentation:hide-text>` 10.8.6, `<presentation:play>` 10.8.8, `<presentation:show-shape>` 10.8.3 and `<presentation:show-text>` 10.8.4.

# 10.9 SMIL Presentation Animations

## 10.9.1 General

SMIL based shape animations are specified for presentation documents. Producers should not use animations specified by the deprecated `<presentation:animations>` elements.

Compared with animations specified by `<presentation:animations>` elements, SMIL based shape animations allow in addition:

- Multiple animations per shape.
- A mixture of animations starting on user interaction and starting automatically per page.
- Multiple animations running at the same time.
- Additional effects "programmed" in XML by combining animation elements.
- Document transformations to SVG.

An *effect* is a combination of one or more animation elements that animate a single shape and or a shape's paragraphs.

### 10.9.2 `<presentation:event-listener>`

The `<presentation:event-listener>` element defines the trigger for a presentation event.

The `<presentation:event-listener>` element is usable within the following element: `<office:event-listeners>` 10.3.19.

The `<presentation:event-listener>` element has the following attributes: `presentation:action` 19.391, `presentation:direction` 19.396, `presentation:effect` 19.398, `presentation:speed` 19.421, `presentation:start-scale` 19.422, `presentation:verb` 19.432, `script:event-name` 19.433, `xlink:actuate` 19.915, `xlink:href` 19.916, `xlink:show` 19.917 and `xlink:type` 19.919.

The `<presentation:event-listener>` element has the following child element: `<presentation:sound>` 10.8.2.

### 10.9.3 `<presentation:sound>`

The `<presentation:sound>` element represents a sound.

The `<presentation:sound>` element is usable within the following elements: `<presentation:dim>` 10.8.7, `<presentation:event-listener>` 10.9.2, `<presentation:hide-shape>` 10.8.5, `<presentation:hide-text>` 10.8.6, `<presentation:show-shape>` 10.8.3, `<presentation:show-text>` 10.8.4 and `<style:drawing-page-properties>` 17.25.

The `<presentation:sound>` element has the following attributes: `presentation:play-full` 19.412, `xlink:actuate` 19.915, `xlink:href` 19.916, `xlink:show` 19.917, `xlink:type` 19.919 and `xml:id` 19.920.

The `<presentation:sound>` element has no child elements.

## 10.9.4 Presentation Document Content

### 10.9.4.1 `<presentation:header>`

The `<presentation:header>` element defines a field that contains a header for a presentation. Which header field declaration is used is specified by the `presentation:use-header-name` 19.430 attribute of the `<draw:page>` 10.2.4 where the field occurs. In a presentation shape inside a master page 19.393, the `presentation:use-header-name` attribute of the `<draw:page>` element for which the drawing shape is displayed is used.

**Note:** This field is mainly used inside master pages. Since its value may differ for the individual drawing pages that make use of a master page, the current field value is not available.

The `<presentation:header>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<presentation:header>` element has no attributes.

The `<presentation:header>` element has no child elements.

#### 10.9.4.2 `<presentation:header-decl>`

The `<presentation:header-decl>` element specifies the text of a header field.

The `<presentation:header-decl>` element is usable within the following element: `<office:presentation>` 3.6.

The `<presentation:header-decl>` element has the following attribute: `presentation:name` 19.405.

The `<presentation:header-decl>` element has no child elements.

The `<presentation:header-decl>` element has character data content.

#### 10.9.4.3 `<presentation:footer>`

The `<presentation:footer>` element defines a field that contains a footer for a presentation. Footer fields display a footer text specified in a footer field declaration. See 10.9.4.4. Which footer field declaration is used is specified by the `presentation:use-footer-name` attribute of the draw page where the field occurs. In a presentation drawing shape inside a master page 19.393, the `presentation:use-footer-name` attribute of the `<draw:page>` for which the drawing shape is displayed is used.

**Note:** This field is mainly used inside master pages. Since its value may differ for the individual drawing pages that make use of a master page, the current field value is not available.

The `<presentation:footer>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<presentation:footer>` element has no attributes.

The `<presentation:footer>` element has no child elements.

#### 10.9.4.4 `<presentation:footer-decl>`

The `<presentation:footer-decl>` element specifies the text of a footer field.

The `<presentation:footer-decl>` element is usable within the following element: `<office:presentation>` 3.6.

The `<presentation:footer-decl>` element has the following attribute: `presentation:name` 19.405.

The `<presentation:footer-decl>` element has no child elements.

The `<presentation:footer-decl>` element has character data content.

#### 10.9.4.5 `<presentation:date-time>`

The `<presentation:date-time>` element defines a field that contains a date/time text for a presentation. Date/time fields display a date/time text as specified in the date/time field declaration. See 10.9.4.6. Which date/time field declaration is used is specified by the `presentation:use-date-time-name` attribute of the draw page where the field occurs. In

a presentation drawing shape inside a master page 19.393, the `presentation:use-date-time-name` attribute of the drawing page for which the drawing shape is displayed is used.

**Note:** This field is mainly used inside master pages. Since its value may differ for the individual drawing pages that make use of a master page, the current field value is not available.

The `<presentation:date-time>` element is usable within the following elements: `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<presentation:date-time>` element has no attributes.

The `<presentation:date-time>` element has no child elements.

#### 10.9.4.6 `<presentation:date-time-decl>`

The `<presentation:date-time-decl>` element specifies the text of a date/time field.

The `<presentation:date-time-decl>` element is usable within the following element: `<office:presentation>` 3.6.

The `<presentation:date-time-decl>` element has the following attributes: `presentation:name` 19.405, `presentation:source` 19.420 and `style:data-style-name` 19.473.

The `<presentation:date-time-decl>` element has no child elements.

The `<presentation:date-time-decl>` element has character data content.

#### 10.9.4.7 `<presentation:settings>`

The `<presentation:settings>` element is a container for the settings for a presentation in a document.

The `<presentation:settings>` element is usable within the following element: `<office:presentation>` 3.6.

The `<presentation:settings>` element has the following attributes: `presentation:animations` 19.392, `presentation:endless` 19.397, `presentation:force-manual` 19.399, `presentation:full-screen` 19.400, `presentation:mouse-as-pen` 19.403, `presentation:mouse-visible` 19.404, `presentation:pause` 19.410, `presentation:show` 19.417, `presentation:show-end-of-presentation-slide` 19.418, `presentation:show-logo` 19.419, `presentation:start-page` 19.423, `presentation:start-with-navigator` 19.424, `presentation:stay-on-top` 19.425 and `presentation:transition-on-click` 19.427.

The `<presentation:settings>` element has the following child element: `<presentation:show>` 10.9.4.8.

#### 10.9.4.8 `<presentation:show>`

The `<presentation:show>` element specifies the order in which pages are displayed during a presentation.

**Note:** The order of pages can be used to omit or repeat pages in a presentation.

The `<presentation:show>` element is usable within the following element: `<presentation:settings>` 10.9.4.7.

The `<presentation:show>` element has the following attributes: `presentation:name` 19.405 and `presentation:pages` 19.408.

The `<presentation:show>` element has no child elements.

# 11 Chart Content

## 11.1 <chart:chart>

The `<chart:chart>` element represents a chart.

The `<chart:chart>` element is usable within the following element: `<office:chart>` 3.8.

The `<chart:chart>` element has the following attributes: `chart:class` 19.15, `chart:column-mapping` 19.16, `chart:row-mapping` 19.26, `chart:style-name` 19.27, `svg:height` 19.543, `svg:width` 19.575, `xlink:href` 19.916, `xlink:type` 19.919 and `xml:id` 19.920.

The `<chart:chart>` element has the following child elements: `<chart:footer>` 11.2.3, `<chart:legend>` 11.4, `<chart:plot-area>` 11.5, `<chart:subtitle>` 11.2.2, `<chart:title>` 11.2.1, `<dr3d:scene>` 10.5.2, `<draw:a>` 10.4.12, `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:connector>` 10.3.10, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:g>` 10.3.15, `<draw:line>` 10.3.3, `<draw:measure>` 10.3.12, `<draw:page-thumbnail>` 10.3.14, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2, `<draw:regular-polygon>` 10.3.6 and `<table:table>` 9.1.2.

## 11.2 Title, Subtitle and Footer

### 11.2.1 <chart:title>

The `<chart:title>` element represents a title in a chart document.

The text of a title is specified by a child `<text:p>` element or a `table:cell-range` attribute. If a child `<text:p>` element and a `table:cell-range` attribute are both present, the text specified by the `table:cell-range` attribute is used as the title. By default, the content of a child `<text:p>` element is displayed as the title.

This element can also be a sub-element of `<chart:axis>`. 11.9 In that case the title is displayed beside the axis specified by the `<chart:axis>` element.

The `<chart:title>` element is usable within the following elements: `<chart:axis>` 11.9 and `<chart:chart>` 11.1.

The `<chart:title>` element has the following attributes: `chart:style-name` 19.27, `svg:x` 19.577, `svg:y` 19.581 and `table:cell-range` 19.596.

The `<chart:title>` element has the following child element: `<text:p>` 5.1.3.

### 11.2.2 <chart:subtitle>

The `<chart:subtitle>` element represents a subtitle for a chart.

The text of a subtitle is specified by a child `<text:p>` element or a `table:cell-range` attribute. If a child `<text:p>` element and a `table:cell-range` attribute are both present, the text specified by the `table:cell-range` attribute is used as the subtitle. By default, the content of a child `<text:p>` element is displayed as the title.

The `<chart:subtitle>` element is usable within the following element: `<chart:chart>` 11.1.

The `<chart:subtitle>` element has the following attributes: `chart:style-name` 19.27, `svg:x` 19.577, `svg:y` 19.581 and `table:cell-range` 19.596.

The `<chart:subtitle>` element has the following child element: `<text:p>` 5.1.3.

### 11.2.3 `<chart:footer>`

The `<chart:footer>` element represents a footer below a chart's plot area.

The text of a footer is specified by a child `<text:p>` element or a `table:cell-range` attribute. If a child `<text:p>` element and a `table:cell-range` attribute are both present, the text specified by the `table:cell-range` attribute is used as the footer. By default, the content of a child `<text:p>` element is displayed as the footer.

The `<chart:footer>` element is usable within the following element: `<chart:chart>` 11.1.

The `<chart:footer>` element has the following attributes: `chart:style-name` 19.27, `svg:x` 19.577, `svg:y` 19.581 and `table:cell-range` 19.596.

The `<chart:footer>` element has the following child element: `<text:p>` 5.1.3.

## 11.3 `<chart:coordinate-region>`

The `<chart:coordinate-region>` element specifies a positioning rectangle.

For charts with a two-dimensional Cartesian coordinate system, the used intervals on the axes span a rectangular coordinate-region. Axis labels, tick marks, axis titles and data labels are not considered in determining the coordinate-region. The coordinate system is scaled so that the coordinate-region matches width and height of the positioning rectangle. The entire chart is located so that the coordinate-region matches the positioning rectangle.

Charts of the predefined classes `chart:radar`, `chart:filled-radar`, `chart:circle` and `chart:ring` do not have a Cartesian coordinate system. For such charts the smallest bounding circle is considered ignoring all axis labels, tick marks, axis titles and data labels. The rectangular bounding box of this circle is the coordinate-region. Where single data points have a `chart:pie-offset` greater than zero that offset shall not contribute to an expansion of the coordinate-region; thus a pulled out pie segment is allowed to protrude from the coordinate-region. A `chart:pie-offset` at the series element in contrast shall contribute to the coordinate region.

Circle or ring or the polar coordinate system of radar-charts respectively is uniformly scaled so that the coordinate-region has maximal size, but does not exceed the size of the positioning rectangle. The entire chart is located so that the coordinate-region is centered in the positioning rectangle. Producers should specify a square positioning rectangle for charts of these classes.

The behavior for the three-dimensional charts is described in section 11.5.

The behavior is implementation-defined for charts of non-predefined chart classes, which have no Cartesian coordinate system.

The `<chart:coordinate-region>` element is usable within the following element: `<chart:plot-area>` 11.5.

The `<chart:coordinate-region>` element has the following attributes: `svg:height` 19.543, `svg:width` 19.575, `svg:x` 19.577 and `svg:y` 19.581.

The `<chart:coordinate-region>` element has no child elements.

## 11.4 `<chart:legend>`

The `<chart:legend>` element represents a legend for a chart. If there is no `<chart:legend>` element for a chart, no legend is displayed.

The labels used by a legend are defined by the `<chart:categories>` 11.10 element on the x-axis. They are paired with matching data points in the order in which they appear.

The `<chart:legend>` element may contain a `<text:p>` element. If present, it defines a title for the legend.

Graphical components of chart legends are defined by the `chart:class` 19.15 attribute.

The `<chart:legend>` element is usable within the following element: `<chart:chart>` 11.1.

The `<chart:legend>` element has the following attributes: `chart:legend-align` 19.22, `chart:legend-position` 19.23, `chart:style-name` 19.27, `style:legend-expansion` 19.494, `style:legend-expansion-aspect-ratio` 19.495, `svg:height` 19.543, `svg:width` 19.575, `svg:x` 19.577 and `svg:y` 19.581.

The `<chart:legend>` element has the following child element: `<text:p>` 5.1.3.

## 11.5 `<chart:plot-area>`

The `<chart:plot-area>` element represents a coordinate system in which data is plotted including defined axes.

The plot area may be displayed as a 3D scene as specified in section 10.5.2. All 3D attributes that can be applied to the `<dr3d:scene>` element can be applied to the `<chart:plot-area>` element. This includes the `dr3d:transform` attribute that specifies the rotation of the three-dimensional plot area. 10.5.2 The `<chart:plot-area>` element may contain a `<dr3d:light>` element as specified in section 10.5.3.

The `svg:x`, `svg:y`, `svg:width` and `svg:height` attributes specify together a positioning rectangle that serves as the viewport for the projection of the scene. Consumers shall use uniform scaling, so that the projection has maximal size but does not exceed the size of the positioning rectangle. The projection of the scene is centered inside the positioning rectangle.

If a `<chart:plot-area>` element contains a `<chart:coordinate-region>` element, the position and size values of the `<chart:coordinate-region>` element are used instead of those from the `<chart:plot-area>` to determine the positioning rectangle.

If the attributes `chart:auto-position` and `chart:auto-size` within a chart style applied to the `chart:plot-area` have the value `true`, all positioning attributes including those within child element `<chart:coordinate-region>` shall be ignored.

For three-dimensional charts of classes `chart:circle` and `chart:ring` the content is extruded. The bounding box as described in 11.3 is extruded the same way and used instead of the entire scene to position and scale the projection.

For charts with a three-dimensional Cartesian coordinate system, the used intervals on the axes generate a cuboid. This is used instead of the entire scene to position and scale the projection.

For other three-dimensional charts the behavior is implementation-defined.

The `<chart:plot-area>` element is usable within the following element: `<chart:chart>` 11.1.

The `<chart:plot-area>` element has the following attributes: `chart:data-source-has-labels` 19.17, `chart:style-name` 19.27, `dr3d:ambient-color` 19.92, `dr3d:distance` 19.96, `dr3d:focal-length` 19.98, `dr3d:lighting-mode` 19.99, `dr3d:projection` 19.102, `dr3d:shade-mode` 19.103, `dr3d:shadow-slant` 19.104, `dr3d:transform` 19.107, `dr3d:vpn` 19.108, `dr3d:vrp` 19.109, `dr3d:vup` 19.110,

svg:height 19.543, svg:width 19.575, svg:x 19.577, svg:y 19.581 and xml:id 19.920.

The `<chart:plot-area>` element has the following child elements: `<chart:axis> 11.9, <chart:coordinate-region> 11.3, <chart:floor> 11.8, <chart:series> 11.12, <chart:stock-gain-marker> 11.20, <chart:stock-loss-marker> 11.21, <chart:stock-range-line> 11.22, <chart:wall> 11.7` and `<dr3d:light> 10.5.3.`

## 11.6 3D Plot Area

The plot area may be displayed as an 3D scene as specified in section 10.5.2. All 3D attributes that can be applied to the `<dr3d:scene>` element can be applied to the `<chart:plot-area>` element. This includes the `dr3d:transform` attribute that specifies the rotation of the three-dimensional plot area. 10.5.2 The `<chart:plot-area>` element may contain a `<dr3d:light>` element as specified in section 10.5.3.

## 11.7 <chart:wall>

The `<chart:wall>` element specifies the wall of a chart. For two-dimensional charts, the wall spans the entire plot area. For three-dimensional charts, the wall consists of two perpendicular rectangles.

The `svg:width` attribute specifies the thickness of a wall for three-dimensional charts.

The `<chart:wall>` element is usable within the following element: `<chart:plot-area> 11.5.`

The `<chart:wall>` element has the following attributes: `chart:style-name 19.27` and `svg:width 19.575.`

The `<chart:wall>` element has no child elements.

## 11.8 <chart:floor>

The `<chart:floor>` element specifies the floor of a chart. For three-dimensional charts, the `<chart:floor>` element shall be present in addition to the `<chart:wall>` element.

The `svg:width` attribute specifies the thickness of the floor.

The `<chart:floor>` element is usable within the following element: `<chart:plot-area> 11.5.`

The `<chart:floor>` element has the following attributes: `chart:style-name 19.27` and `svg:width 19.575.`

The `<chart:floor>` element has no child elements.

## 11.9 <chart:axis>

The `<chart:axis>` element specifies an axis for a chart.

The `<chart:axis>` element is usable within the following element: `<chart:plot-area> 11.5.`

The `<chart:axis>` element has the following attributes: `chart:dimension 19.18, chart:name 19.24` and `chart:style-name 19.27.`

The `<chart:axis>` element has the following child elements: `<chart:categories> 11.10, <chart:grid> 11.11` and `<chart:title> 11.2.1.`

## 11.10 <chart:categories>

The <chart:categories> element represents labels that are displayed on a category-axis.

This element may have a `table:cell-range-address` attribute that specifies a range from which category labels are taken. If this attribute or the <chart:categories> element itself is omitted, the `chart:data-source-has-labels` attribute of the <chart:plot-area> element should be evaluated for labels to display on a category-axis.

The <chart:categories> element is usable within the following element: <chart:axis> 11.9.

The <chart:categories> element has the following attribute: `table:cell-range-address` 19.597.

The <chart:categories> element has no child elements.

## 11.11 <chart:grid>

The <chart:grid> element specifies a grid for an axis.

The <chart:grid> element is usable within the following element: <chart:axis> 11.9.

The <chart:grid> element has the following attributes: `chart:class` 19.15 and `chart:style-name` 19.27.

The <chart:grid> element has no child elements.

## 11.12 <chart:series>

The <chart:series> element represents a data series in a chart. If the chart requires more input data, it is the case for scatter and bubble charts, <chart:domain> sub-elements shall be defined that contain the cell range addresses of the corresponding data.

The <chart:series> element is usable within the following element: <chart:plot-area> 11.5.

The <chart:series> element has the following attributes: `chart:attached-axis` 19.13, `chart:class` 19.15, `chart:label-cell-address` 19.21, `chart:style-name` 19.27, `chart:values-cell-range-address` 19.28 and `xml:id` 19.920.

The <chart:series> element has the following child elements: <chart:label> 11.15, <chart:point> 11.14, <chart:domain> 11.13, <chart:error-indicator> 11.17, <chart:mean-value> 11.16 and <chart:regression-curve> 11.18.

## 11.13 <chart:domain>

The <chart:domain> element specifies coordinate values required by particular chart types.

For scatter charts, one <chart:domain> element shall exist. Its `table:cell-range-address` attribute references the x-coordinate values for the scatter chart.

For bubble charts, two <chart:domain> elements shall exist. The values for the y-coordinates are given by the first <chart:domain> element. The values for the x-coordinates are given by the second <chart:domain> element.

For surface charts, up to two <chart:domain> elements are allowed to exist. The values for the y-coordinates are given by the first <chart:domain> element. The values for the x-coordinates are given by the second <chart:domain> element.

At least one <chart:series> element of a given `chart:class` shall have the necessary number of <chart:domain> sub-elements. All other <chart:series> elements with the

same `chart:class` may omit the `<chart:domain>` sub-elements and use the previously-defined values for the same `chart:class` value.

The `<chart:domain>` element is usable within the following element: `<chart:series>` 11.12.

The `<chart:domain>` element has the following attribute: `table:cell-range-address` 19.597.

The `<chart:domain>` element has no child elements.

## 11.14 `<chart:data-point>`

The `<chart:data-point>` element specifies a style for a single data point in a data series.

**Note:** This element would typically be used if a single data point has a special or distinct appearance.

The `<chart:data-point>` element is usable within the following element: `<chart:series>` 11.12.

The `<chart:data-point>` element has the following attributes: `chart:repeated` 19.25, `chart:style-name` 19.27 and `xml:id` 19.920.

The `<chart:data-point>` element has the following child element: `<chart:label>` 11.15.

## 11.15 `<chart:data-label>`

The `<chart:data-label>` element represents the data label of a data point.

The `<chart:data-label>` element can also be a sub-element of a data series. In that case, the data label serves as default for all the data points of this series.

The `<chart:data-label>` element is usable within the following elements: `<chart:point>` 11.14 and `<chart:series>` 11.12.

The `<chart:data-label>` element has the following attributes: `chart:style-name` 19.27, `svg:x` 19.577 and `svg:y` 19.581.

The `<chart:data-label>` element has the following child element: `<text:p>` 5.1.3.

## 11.16 `<chart:mean-value>`

The `<chart:mean-value>` element specifies a style for a mean-value line.

The `<chart:mean-value>` element is usable within the following element: `<chart:series>` 11.12.

The `<chart:mean-value>` element has the following attribute: `chart:style-name` 19.27.

The `<chart:mean-value>` element has no child elements.

## 11.17 `<chart:error-indicator>`

The `<chart:error-indicator>` element specifies a style for error indicators.

The `<chart:error-indicator>` element is usable within the following element: `<chart:series>` 11.12.

The `<chart:error-indicator>` element has the following attributes: `chart:dimension` 19.18 and `chart:style-name` 19.27.

The `<chart:error-indicator>` element has no child elements.

## 11.18 `<chart:regression-curve>`

The `<chart:regression-curve>` element specifies the properties of regression curves.

The `<chart:regression-curve>` element is usable within the following element:  
`<chart:series>` 11.12.

The `<chart:regression-curve>` element has the following attribute: `chart:style-name` 19.27.

The `<chart:regression-curve>` element has the following child element:  
`<chart:equation>` 11.19.

## 11.19 `<chart:equation>`

The `<chart:equation>` element represents the equation for a containing regression curve.

The `<chart:equation>` element is usable within the following element:  
`<chart:regression-curve>` 11.18.

The `<chart:equation>` element has the following attributes: `chart:automatic-content` 19.14, `chart:display-equation` 19.19, `chart:display-r-square` 19.20, `chart:style-name` 19.27, `svg:x` 19.577 and `svg:y` 19.581.

The `<chart:equation>` element has the following child element: `<text:p>` 5.1.3.

## 11.20 `<chart:stock-gain-marker>`

The `<chart:stock-gain-marker>` element specifies a style for candlestick-bars in a stock chart that have a higher closing value than opening value.

The `<chart:stock-gain-marker>` element is usable within the following element:  
`<chart:plot-area>` 11.5.

The `<chart:stock-gain-marker>` element has the following attribute: `chart:style-name` 19.27.

The `<chart:stock-gain-marker>` element has no child elements.

## 11.21 `<chart:stock-loss-marker>`

The `<chart:stock-loss-marker>` element specifies the style for candlestick-bars 19.15.1 in a stock chart that have a lower closing value than opening value.

The `<chart:stock-loss-marker>` element is usable within the following element:  
`<chart:plot-area>` 11.5.

The `<chart:stock-loss-marker>` element has the following attribute: `chart:style-name` 19.27.

The `<chart:stock-loss-marker>` element has no child elements.

## 11.22 `<chart:stock-range-line>`

The `<chart:stock-range-line>` element specifies a style for the range-lines 19.15.1 in a stock chart. A range-line is a line connecting the minimum value with the maximum value.

The `<chart:stock-range-line>` element is usable within the following element:  
`<chart:plot-area>` 11.5.

The `<chart:stock-range-line>` element has the following attribute: `chart:style-name 19.27`.

The `<chart:stock-range-line>` element has no child elements.

## 12 Database Front-end Document Content

### 12.1 <office:database>

A <office:database> element is a container of database elements.

The <office:database> element is usable within the following element: <office:body> 3.3.

The <office:database> element has no attributes.

The <office:database> element has the following child elements: <db:data-source> 12.2, <db:forms> 12.25.2, <db:queries> 12.26, <db:reports> 12.25.3, <db:schema-definition> 12.36 and <db:table-representations> 12.32.

### 12.2 <db:data-source>

A <db:data-source> element is a container for data source specific elements that are needed to create a connection to a database.

The <db:data-source> element is usable within the following element: <office:database> 12.1.

The <db:data-source> element has no attributes.

The <db:data-source> element has the following child elements: <db:application-connection-settings> 12.15, <db:connection-data> 12.3 and <db:driver-settings> 12.9.

### 12.3 <db:connection-data>

The <db:connection-data> element specifies the information necessary to connect to a database.

The <db:connection-data> element is usable within the following element: <db:data-source> 12.2.

The <db:connection-data> element has no attributes.

The <db:connection-data> element has the following child elements: <db:connection-resource> 12.7, <db:database-description> 12.4 and <db:login> 12.8.

### 12.4 <db:database-description>

The <db:database-description> element specifies a database resource by its type and access parameters.

The <db:database-description> element is usable within the following element: <db:connection-data> 12.3.

The <db:database-description> element has no attributes.

The <db:database-description> element has the following child elements: <db:file-based-database> 12.5 and <db:server-database> 12.6.

## 12.5 <db:file-based-database>

The <db:file-based-database> element specifies a database which is composed of one or more files in an arbitrary file system.

The <db:file-based-database> element is usable within the following element:  
<db:database-description> 12.4.

The <db:file-based-database> element has the following attributes: db:extension 19.52, db:media-type 19.69, xlink:href 19.916 and xlink:type 19.919.

The <db:file-based-database> element has no child elements.

## 12.6 <db:server-database>

The <db:server-database> element specifies connection information for a server-based database.

The <db:server-database> element is usable within the following element:  
<db:database-description> 12.4.

The <db:server-database> element has the following attributes: db:database-name 19.43, db:hostname 19.54, db:local-socket 19.67, db:port 19.72 and db:type 19.85.

The <db:server-database> element has no child elements.

## 12.7 <db:connection-resource>

The <db:connection-resource> element specifies a database connection by an IRI.

The <db:connection-resource> element is usable within the following element:  
<db:connection-data> 12.3.

The <db:connection-resource> element has the following attributes: xlink:actuate 19.915, xlink:href 19.916, xlink:show 19.917 and xlink:type 19.919.

The <db:connection-resource> element has no child elements.

## 12.8 <db:login>

The <db:login> element specifies the information needed to establish a connection to a database.

**Note:** For security reasons, the <db:login> element does not have an attribute to store a password that may be required to establish a connection.

The <db:login> element is usable within the following element: <db:connection-data> 12.3.

The <db:login> element has the following attributes: db:is-password-required 19.63, db:login-timeout 19.61, db:use-system-user 19.89 and db:user-name 19.90.

The <db:login> element has no child elements.

## 12.9 <db:driver-settings>

The <db:driver-settings> element specifies settings for a driver establishing the database connection.

The <db:driver-settings> element is usable within the following element: <db:datasource> 12.2.

The `<db:driver-settings>` element has the following attributes: `db:base-dn` 19.35, `db:is-first-row-header-line` 19.60, `db:parameter-name-substitution` 19.71, `db:show-deleted` 19.78 and `db:system-driver-settings` 19.82.

The `<db:driver-settings>` element has the following child elements: `<db:auto-increment>` 12.10, `<db:character-set>` 12.12, `<db:delimiter>` 12.11 and `<db:table-settings>` 12.13.

## 12.10 `<db:auto-increment>`

The `<db:auto-increment>` element specifies SQL statements to retrieve auto-generated row values and to set AUTO\_INCREMENT values for columns.

The `<db:auto-increment>` element is usable within the following element: `<db:driver-settings>` 12.9.

The `<db:auto-increment>` element has the following attributes: `db:additional-column-statement` 19.31 and `db:row-retrieving-statement` 19.75.

The `<db:auto-increment>` element has no child elements.

## 12.11 `<db:delimiter>`

The `<db:delimiter>` element defines a delimiter for parsing data.

The `<db:delimiter>` element is usable within the following elements: `<db:driver-settings>` 12.9 and `<db:table-setting>` 12.14.

The `<db:delimiter>` element has the following attributes: `db:decimal` 19.44, `db:field` 19.53, `db:string` 19.79 and `db:thousand` 19.83.

The `<db:delimiter>` element has no child elements.

## 12.12 `<db:character-set>`

The `<db:character-set>` element defines the text encoding that is used to transform string data obtained from a database.

The `<db:character-set>` element is usable within the following elements: `<db:driver-settings>` 12.9 and `<db:table-setting>` 12.14.

The `<db:character-set>` element has the following attribute: `db:encoding` 19.50.

The `<db:character-set>` element has no child elements.

## 12.13 `<db:table-settings>`

The `<db:table-settings>` element is a container for `<db:table-setting>` elements.

The `<db:table-settings>` element is usable within the following element: `<db:driver-settings>` 12.9.

The `<db:table-settings>` element has no attributes.

The `<db:table-settings>` element has the following child element: `<db:table-setting>` 12.14.

## 12.14 `<db:table-setting>`

The `<db:table-setting>` element specifies driver-level settings which are used by a database driver for defining a single database table.

The `<db:table-setting>` element is usable within the following element: `<db:table-settings>` 12.13.

The `<db:table-setting>` element has the following attributes: `db:is-first-row-header-line` 19.60 and `db:show-deleted` 19.78.

The `<db:table-setting>` element has the following child elements: `<db:character-set>` 12.12 and `<db:delimiter>` 12.11.

## 12.15 <db:application-connection-settings>

The `<db:application-connection-settings>` element specifies settings which define how a consumer uses a database connection.

The `<db:application-connection-settings>` element is usable within the following element: `<db:datasource>` 12.2.

The `<db:application-connection-settings>` element has the following attributes: `db:append-table-alias-name` 19.32, `db:boolean-comparison-mode` 19.36, `db:enable-sql92-check` 19.49, `db:ignore-driver-privileges` 19.55, `db:is-table-name-length-limited` 19.66, `db:max-row-count` 19.68, `db:suppress-version-columns` 19.81 and `db:use-catalog` 19.88.

The `<db:application-connection-settings>` element has the following child elements: `<db:datasource-settings>` 12.22, `<db:table-filter>` 12.16 and `<db:table-type-filter>` 12.20.

## 12.16 <db:table-filter>

A `<db:table-filter>` element defines a list of filter patterns which are applied on the table names returned by a database. Only tables which match at least one include filter, and do not match any exclude filter, will be displayed.

The `<db:table-filter>` element is usable within the following element: `<db:application-connection-settings>` 12.15.

The `<db:table-filter>` element has no attributes.

The `<db:table-filter>` element has the following child elements: `<db:table-exclude-filter>` 12.18 and `<db:table-include-filter>` 12.17.

## 12.17 <db:table-include-filter>

The `<db:table-include-filter>` element specifies a list of filter patterns which determine inclusion of tables in a consumer's table display.

The `<db:table-include-filter>` element is usable within the following element: `<db:table-filter>` 12.16.

The `<db:table-include-filter>` element has no attributes.

The `<db:table-include-filter>` element has the following child element: `<db:table-filter-pattern>` 12.19.

## 12.18 <db:table-exclude-filter>

The `<db:table-exclude-filter>` element specifies a list of filter patterns which determine exclusion of tables in a consumer's table display.

The `<db:table-exclude-filter>` element is usable within the following element: `<db:table-filter>` 12.16.

The `<db:table-exclude-filter>` element has no attributes.

The `<db:table-exclude-filter>` element has the following child element: `<db:table-filter-pattern>` 12.19.

## 12.19 `<db:table-filter-pattern>`

The `<db:table-filter-pattern>` element specifies a filter pattern.

The "%" (U+0025. PERCENT SIGN) character is used as wildcard.

The `<db:table-filter-pattern>` element is usable within the following elements: `<db:table-exclude-filter>` 12.18 and `<db:table-include-filter>` 12.17.

The `<db:table-filter-pattern>` element has no attributes.

The `<db:table-filter-pattern>` element has no child elements.

The `<db:table-filter-pattern>` element has content of data type `string` 18.2.

## 12.20 `<db:table-type-filter>`

A `<db:table-type-filter>` element defines a list of table types which a database uses to filter tables.

The `<db:table-type-filter>` element is usable within the following element: `<db:application-connection-settings>` 12.15.

The `<db:table-type-filter>` element has no attributes.

The `<db:table-type-filter>` element has the following child element: `<db:table-type>` 12.21.

## 12.21 `<db:table-type>`

The `<db:table-type>` element specifies a table type.

**Note:** Table types are database-specific.

The `<db:table-type>` element is usable within the following element: `<db:table-type-filter>` 12.20.

The `<db:table-type>` element has no attributes.

The `<db:table-type>` element has no child elements.

The `<db:table-type>` element has content of data type `string` 18.2.

## 12.22 `<db:data-source-settings>`

The `<db:data-source-settings>` element is a container element for `<db:data-source-setting>` elements.

The `<db:data-source-settings>` element is usable within the following element: `<db:application-connection-settings>` 12.15.

The `<db:data-source-settings>` element has no attributes.

The `<db:data-source-settings>` element has the following child element: `<db:data-source-setting>` 12.23.

## 12.23 <db:data-source-setting>

The <db:data-source-setting> element specifies the name of a data-source-setting, a data-source-setting type, and whether a data-source-setting contains a list of data or single data.

The <db:data-source-setting> element is usable within the following element: <db:data-source-settings> 12.22.

The <db:data-source-setting> element has the following attributes: db:data-source-setting-is-list 19.39, db:data-source-setting-name 19.40 and db:data-source-setting-type 19.41.

The <db:data-source-setting> element has the following child element: <db:data-source-setting-value> 12.24.

## 12.24 <db:data-source-setting-value>

The <db:data-source-setting-value> element specifies the content of a data-source-setting.

The <db:data-source-setting-value> element is usable within the following element: <db:data-source-setting> 12.23.

The <db:data-source-setting-value> element has no attributes.

The <db:data-source-setting-value> element has no child elements.

The <db:data-source-setting-value> element has content of data type string 18.2.

## 12.25 Forms and Reports

### 12.25.1 General

Database documents can contain forms and reports as sub-documents. They're included in database front-end documents by XLinks. Alternatively, the content of sub-documents may be included in the <db:component> elements as child elements.

### 12.25.2 <db:forms>

The <db:forms> element may contain <db:component> or <db:component-collection> elements or both. It specifies a list of form documents and sub-collections.

The <db:forms> element is usable within the following element: <office:database> 12.1.

The <db:forms> element has no attributes.

The <db:forms> element has the following child elements: <db:component> 12.25.5 and <db:component-collection> 12.25.4.

### 12.25.3 <db:reports>

The <db:reports> element may contain <db:component> or <db:component-collection> elements or both. It specifies a list of report documents and sub-collections.

The <db:reports> element is usable within the following element: <office:database> 12.1.

The <db:reports> element has no attributes.

The <db:reports> element has the following child elements: <db:component> 12.25.5 and <db:component-collection> 12.25.4.

#### 12.25.4 <db:component-collection>

The `<db:component-collection>` element specifies a collection of `<db:component>` elements and `<db:component-collection>` elements.

The `<db:component-collection>` element is usable within the following elements: `<db:component-collection>` 12.25.4, `<db:forms>` 12.25.2 and `<db:reports>` 12.25.3.

The `<db:component-collection>` element has the following attributes: `db:description` 19.48, `db:name` 19.70 and `db:title` 19.84.

The `<db:component-collection>` element has the following child elements: `<db:component>` 12.25.5 and `<db:component-collection>` 12.25.4.

#### 12.25.5 <db:component>

The `<db:component>` element specifies a database component which can be either a form or a report.

A database component is referenced using the `xlink:href` attribute or it is contained in a child element of either a `<office:document>` or `<math:math>` element.

The `<db:component>` element is usable within the following elements: `<db:component-collection>` 12.25.4, `<db:forms>` 12.25.2 and `<db:reports>` 12.25.3.

The `<db:component>` element has the following attributes: `db:as-template` 19.34, `db:description` 19.48, `db:name` 19.70, `db:title` 19.84, `xlink:actuate` 19.915, `xlink:href` 19.916, `xlink:show` 19.917 and `xlink:type` 19.919.

The `<db:component>` element has the following child elements: `<math:math>` 14.6 and `<office:document>` 3.1.2.

#### 12.26 <db:queries>

The `<db:queries>` element may contain `<db:query>` or `<db:query-collection>` elements or both.

The `<db:queries>` element is usable within the following element: `<office:database>` 12.1.

The `<db:queries>` element has no attributes.

The `<db:queries>` element has the following child elements: `<db:query>` 12.28 and `<db:query-collection>` 12.27.

#### 12.27 <db:query-collection>

The `<db:query-collection>` element may contain `<db:query>` or `<db:query-collection>` elements or both.

The `<db:query-collection>` element is usable within the following elements: `<db:queries>` 12.26 and `<db:query-collection>` 12.27.

The `<db:query-collection>` element has the following attributes: `db:description` 19.48, `db:name` 19.70 and `db:title` 19.84.

The `<db:query-collection>` element has the following child elements: `<db:query>` 12.28 and `<db:query-collection>` 12.27.

#### 12.28 <db:query>

The `<db:query>` element specifies a database query.

The `<db:query>` element is usable within the following elements: `<db:queries>` 12.26 and `<db:query-collection>` 12.27.

The `<db:query>` element has the following attributes: `db:command` 19.38, `db:default-row-style-name` 19.46, `db:description` 19.48, `db:escape-processing` 19.51, `db:name` 19.70, `db:style-name` 19.80 and `db:title` 19.84.

The `<db:query>` element has the following child elements: `<db:columns>` 12.34, `<db:filter-statement>` 12.30, `<db:order-statement>` 12.29 and `<db:update-table>` 12.31.

## 12.29 `<db:order-statement>`

The `<db:order-statement>` element specifies a sort ("ORDER BY") [SQL] clause which is to be applied on top of another [SQL] statement producing a result set.

The `<db:order-statement>` element is usable within the following elements: `<db:query>` 12.28 and `<db:table-representation>` 12.33.

The `<db:order-statement>` element has the following attributes: `db:apply-command` 19.33 and `db:command` 19.38.

The `<db:order-statement>` element has no child elements.

## 12.30 `<db:filter-statement>`

The `<db:filter-statement>` element specifies a filter ("WHERE") [SQL] clause which is to be applied on top of another [SQL]-statement producing a result set.

The `<db:filter-statement>` element is usable within the following elements: `<db:query>` 12.28 and `<db:table-representation>` 12.33.

The `<db:filter-statement>` element has the following attributes: `db:apply-command` 19.33 and `db:command` 19.38.

The `<db:filter-statement>` element has no child elements.

## 12.31 `<db:update-table>`

The `<db:update-table>` element specifies the name of the table which will be updated when an UPDATE, INSERT or DELETE [SQL] statement is executed. It defines the table which should be updated when a SELECT [SQL] statement contains more than one table.

The `<db:update-table>` element is usable within the following element: `<db:query>` 12.28.

The `<db:update-table>` element has the following attributes: `db:catalog-name` 19.37, `db:name` 19.70 and `db:schema-name` 19.77.

The `<db:update-table>` element has no child elements.

## 12.32 `<db:table-representations>`

The `<db:table-representations>` element specifies a collection of tables.

The `<db:table-representations>` element is usable within the following element: `<office:database>` 12.1.

The `<db:table-representations>` element has no attributes.

The `<db:table-representations>` element has the following child element: `<db:table-representation>` 12.33.

## 12.33 <db:table-representation>

A <db:table-representation> element specifies how a consumer displays a table to the user.

The <db:table-representation> element is usable within the following element:  
<db:table-representations> 12.32.

The <db:table-representation> element has the following attributes: db:catalog-name 19.37, db:default-row-style-name 19.46, db:description 19.48, db:name 19.70, db:schema-name 19.77, db:style-name 19.80 and db:title 19.84.

The <db:table-representation> element has the following child elements:  
<db:columns> 12.34, <db:filter-statement> 12.30 and <db:order-statement> 12.29.

## 12.34 <db:columns>

The <db:columns> element is a collection of <db:column> elements.

The <db:columns> element is usable within the following elements: <db:query> 12.28 and <db:table-representation> 12.33.

The <db:columns> element has no attributes.

The <db:columns> element has the following child element: <db:column> 12.35.

## 12.35 <db:column>

The <db:column> element specifies settings for a column.

The <db:column> element is usable within the following element: <db:columns> 12.34.

The <db:column> element has the following attributes: db:default-cell-style-name 19.45, db:description 19.48, db:name 19.70, db:style-name 19.80, db:title 19.84, db:visible 19.91, office:boolean-value 19.371, office:currency 19.373, office:date-value 19.374, office:string-value 19.383, office:time-value 19.386, office:value 19.388, office:value-type 19.389.

The <db:column> element has no child elements.

## 12.36 <db:schema-definition>

The <db:schema-definition> element specifies a database schema.

The <db:schema-definition> element is usable within the following element:  
<office:database> 12.1.

The <db:schema-definition> element has no attributes.

The <db:schema-definition> element has the following child element: <db:table-definitions> 12.37.

## 12.37 <db:table-definitions>

The <db:table-definitions> element is a container for multiple <db:table-definition> elements.

The <db:table-definitions> element is usable within the following element:  
<db:schema-definition> 12.36.

The <db:table-definitions> element has no attributes.

The `<db:table-definitions>` element has the following child element: `<db:table-definition>` 12.38.

## 12.38 `<db:table-definition>`

The `<db:table-definition>` element specifies the schema of a single table.

The `<db:table-definition>` element is usable within the following element: `<db:table-definitions>` 12.37.

The `<db:table-definition>` element has the following attributes: `db:catalog-name` 19.37, `db:name` 19.70, `db:schema-name` 19.77 and `db:type` 19.85.

The `<db:table-definition>` element has the following child elements: `<db:column-definitions>` 12.39, `<db:indices>` 12.45 and `<db:keys>` 12.41.

## 12.39 `<db:column-definitions>`

The `<db:column-definitions>` element is a container for `<db:column-definition>` elements.

The `<db:column-definitions>` element is usable within the following element: `<db:table-definition>` 12.38.

The `<db:column-definitions>` element has no attributes.

The `<db:column-definitions>` element has the following child element: `<db:column-definition>` 12.40.

## 12.40 `<db:column-definition>`

The `<db:column-definition>` element defines a table column.

The `<db:column-definition>` element is usable within the following element: `<db:column-definitions>` 12.39.

The `<db:column-definition>` element has the following attributes: `db:data-type` 19.42, `db:is-autoincrement` 19.57, `db:is-empty-allowed` 19.59, `db:is-nullable` 19.62, `db:name` 19.70, `db:precision` 19.65, `db:scale` 19.76, `db:type-name` 19.86, `office:boolean-value` 19.371, `office:currency` 19.373, `office:date-value` 19.374, `office:string-value` 19.383, `office:time-value` 19.386, `office:value` 19.388, `office:value-type` 19.389.

The `<db:column-definition>` element has no child elements.

## 12.41 `<db:keys>`

The `<db:keys>` element is a container for one or more `<db:key>` elements.

The `<db:keys>` element is usable within the following element: `<db:table-definition>` 12.38.

The `<db:keys>` element has no attributes.

The `<db:keys>` element has the following child element: `<db:key>` 12.42.

## 12.42 `<db:key>`

The `<db:key>` element specifies a key for a table.

The `<db:key>` element is usable within the following element: `<db:keys>` 12.41.

The `<db:key>` element has the following attributes: `db:delete-rule` 19.47, `db:name` 19.70, `db:referenced-table-name` 19.73, `db:type` 19.85 and `db:update-rule` 19.87.

The `<db:key>` element has the following child element: `<db:key-columns>` 12.43.

## 12.43 `<db:key-columns>`

The `<db:key-columns>` element specifies the columns of a key.

The `<db:key-columns>` element is usable within the following element: `<db:key>` 12.42.

The `<db:key-columns>` element has no attributes.

The `<db:key-columns>` element has the following child element: `<db:key-column>` 12.44.

## 12.44 `<db:key-column>`

The `<db:key-column>` element specifies a description of a column of a table key.

The `<db:key-column>` element is usable within the following element: `<db:key-columns>` 12.43.

The `<db:key-column>` element has the following attributes: `db:name` 19.70 and `db:related-column-name` 19.74.

The `<db:key-column>` element has no child elements.

## 12.45 `<db:indices>`

The `<db:indices>` element defines the indexes of a table.

The `<db:indices>` element is usable within the following element: `<db:table-definition>` 12.38.

The `<db:indices>` element has no attributes.

The `<db:indices>` element has the following child element: `<db:index>` 12.46.

## 12.46 `<db:index>`

The `<db:index>` element defines the characteristics of a table index.

The `<db:index>` element is usable within the following element: `<db:indices>` 12.45.

The `<db:index>` element has the following attributes: `db:catalog-name` 19.37, `db:is-clustered` 19.58, `db:is-unique` 19.64 and `db:name` 19.70.

The `<db:index>` element has the following child element: `<db:index-columns>` 12.47.

## 12.47 `<db:index-columns>`

The `<db:index-columns>` element specifies the columns of a table index.

The `<db:index-columns>` element is usable within the following element: `<db:index>` 12.46.

The `<db:index-columns>` element has no attributes.

The `<db:index-columns>` element has the following child element: `<db:index-column>` 12.48.

## 12.48 <db:index-column>

The <db:index-column> element specifies the description of a column of a table index.

The <db:index-column> element is usable within the following element: <db:index-columns> 12.47.

The <db:index-column> element has the following attributes: db:is-ascending 19.56 and db:name 19.70.

The <db:index-column> element has no child elements.

# 13 Form Content

## 13.1 General

A form is a container for user-interface controls.

**Note:** Buttons, text boxes, check boxes, and drop-down lists are user interface controls that can be contained in a form.

In the OpenDocument format, the following rules apply to user interface controls and forms:

- All controls shall be located in a form. 13.5
- All controls that are not hidden are assigned an absolute or relative position. These visual aspects of a control are represented by drawing shapes that contain a reference to the control.
- Forms may be nested. The containing form provides a context for the contained form.
- Forms are not connected with the text flow and layout of a document.
- Forms can be data-aware. A form is data-aware if the values subject to its controls are populated from and updated to a database.

Form behavior is defined for submission and connection to a data source only for forms defined by a `<form:form>` element.

The result set of values subject to the controls of the form is defined by the `form:datasource`, `form:command`, and `form:command-type` attributes. At any point in time, only one row of the result set is subject to the forms controls.

Form behavior is not defined for forms defined by a `<xforms:model>` element.

## 13.2 <office:forms>

The `<office:forms>` element is a container for `<form:form>` or `<xforms:model>` elements.

The `<office:forms>` element is usable within the following elements: `<draw:page>` 10.2.4, `<office:text>` 3.4, `<presentation:notes>` 16.19, `<style:master-page>` 16.9 and `<table:table>` 9.1.2.

The `<office:forms>` element has the following attributes: `form:apply-design-mode` 19.250 and `form:automatic-focus` 19.253.

The `<office:forms>` element has the following child elements: `<form:form>` 13.3 and `<xforms:model>` 13.4.

## 13.3 <form:form>

The `<form:form>` element specifies a user-interface form and defines the contents and properties of the form.

The `<form:form>` element is usable within the following elements: `<form:form>` 13.3 and `<office:forms>` 13.2.

The `<form:form>` element has the following attributes: `form:allow-deletes` 19.247, `form:allow-inserts` 19.248, `form:allow-updates` 19.249, `form:apply-filter` 19.251, `form:command` 19.256, `form:command-type` 19.257, `form:control-implementation` 19.258, `form:datasource` 19.264, `form:detail-fields` 19.267, `form:enctype` 19.271, `form:escape-processing` 19.272, `form:filter` 19.273,

form:ignore-result 19.277, form:master-fields 19.287, form:method 19.291, form:name 19.294, form:navigation-mode 19.295, form:order 19.296, form:tab-cycle 19.309, office:target-frame 19.384, xlink:actuate 19.915, xlink:href 19.916 and xlink:type 19.919.

The `<form:form>` element has the following child elements: `<form:button>` 13.5.15, `<form:checkbox>` 13.5.17, `<form:combobox>` 13.5.11, `<form:connection-resource>` 7.6.2, `<form:date>` 13.5.8, `<form:file>` 13.5.5, `<form:fixed-text>` 13.5.10, `<form:form>` 13.3, `<form:formatted-text>` 13.5.6, `<form:frame>` 13.5.19, `<form:generic-control>` 13.5.25, `<form:grid>` 13.5.22, `<form:hidden>` 13.5.21, `<form:image>` 13.5.16, `<form:image-frame>` 13.5.20, `<form:listbox>` 13.5.13, `<form:number>` 13.5.7, `<form:password>` 13.5.4, `<form:properties>` 13.7, `<form:radio>` 13.5.18, `<form:text>` 13.5.2, `<form:textarea>` 13.5.3, `<form:time>` 13.5.9, `<form:value-range>` 13.5.24 and `<office:event-listeners>` 10.3.19.

## 13.4 `<xforms:model>`

The `<xforms:model>` element implements [XForms] and is embedded in the `<office:forms>` element. XForms consists of two major parts, the XForms model which contains the form logic plus form data, and XForms controls, which can be bound to a data model.

Controls defined by OpenDocument (see 13.5) may be bound to an XForms model by an `xforms:bind` attribute.

The `<xforms:model>` element has the attributes and content defined in §3.3.1 of [XForms].

The `<xforms:model>` element is usable within the following element: `<office:forms>` 13.2.

## 13.5 Controls

### 13.5.1 General

Controls are used to interact with forms. Each control in a form is identified by a name. The name of a control need not be unique in a form.

Controls are connected to the surrounding document (and its text flow, if applicable) by binding them to a shape that acts as a placeholder for the control. 10.3.13

In addition to the attributes defined in this standard, controls may have implementation-dependent attributes. Those attributes are stored in a `<form:properties>` element in each control. Control events are specified in a `<office:event-listeners>` element.

### 13.5.2 `<form:text>`

The `<form:text>` element defines a control for displaying and inputting text on a single line.

The `<form:text>` element is usable within the following elements: `<form:column>` 13.5.23 and `<form:form>` 13.3.

The `<form:text>` element has the following attributes: `form:control-implementation` 19.258, `form:convert-empty-to-null` 19.259, `form:current-value` 19.262, `form:data-field` 19.263, `form:disabled` 19.268, `form:id` 19.276, `form:input-required` 19.921, `form:linked-cell` 19.283, `form:max-length` 19.288, `form:name` 19.294, `form:printable` 19.299, `form:readonly` 19.301, `form:tab-index` 19.310, `form:tab-stop` 19.311, `form:title` 19.313, `form:value` 19.316, `xforms:bind` 19.910 and `xml:id` 19.920.

The `<form:text>` element has the following child elements: `<form:properties>` 13.7 and `<office:event-listeners>` 10.3.19.

### 13.5.3 <form:textarea>

The `<form:textarea>` element defines a control for displaying and inputting text on multiple lines.

The `<form:textarea>` element may be used with plain-text values (specified by the `form:current-value` attribute) as well as with formatted text (specified as paragraph content). If both a `form:current-value` attribute and one or more `<text:p>` elements are present, it is implementation-dependent which text is used.

The `<form:textarea>` element is usable within the following elements: `<form:column>` 13.5.23 and `<form:form>` 13.3.

The `<form:textarea>` element has the following attributes: `form:control-implementation` 19.258, `form:convert-empty-to-null` 19.259, `form:current-value` 19.262, `form:data-field` 19.263, `form:disabled` 19.268, `form:id` 19.276, `form:input-required` 19.921, `form:linked-cell` 19.283, `form:max-length` 19.288, `form:name` 19.294, `form:printable` 19.299, `form:readonly` 19.301, `form:tab-index` 19.310, `form:tab-stop` 19.311, `form:title` 19.313, `form:value` 19.316, `xforms:bind` 19.910 and `xml:id` 19.920.

The `<form:textarea>` element has the following child elements: `<form:properties>` 13.7, `<office:event-listeners>` 10.3.19 and `<text:p>` 5.1.3.

### 13.5.4 <form:password>

The `<form:password>` element defines a control that uses an echo character to hide password input by a user.

The `<form:password>` element is usable within the following element: `<form:form>` 13.3.

The `<form:password>` element has the following attributes: `form:control-implementation` 19.258, `form:convert-empty-to-null` 19.259, `form:disabled` 19.268, `form:echo-char` 19.270, `form:id` 19.276, `form:input-required` 19.921, `form:linked-cell` 19.283, `form:max-length` 19.288, `form:name` 19.294, `form:printable` 19.299, `form:tab-index` 19.310, `form:tab-stop` 19.311, `form:title` 19.313, `form:value` 19.316, `xforms:bind` 19.910 and `xml:id` 19.920.

The `<form:password>` element has the following child elements: `<form:properties>` 13.7 and `<office:event-listeners>` 10.3.19.

### 13.5.5 <form:file>

The `<form:file>` element defines a control for selecting a file.

The `<form:file>` element is usable within the following element: `<form:form>` 13.3.

The `<form:file>` element has the following attributes: `form:control-implementation` 19.258, `form:current-value` 19.262, `form:disabled` 19.268, `form:id` 19.276, `form:linked-cell` 19.283, `form:max-length` 19.288, `form:name` 19.294, `form:printable` 19.299, `form:readonly` 19.301, `form:tab-index` 19.310, `form:tab-stop` 19.311, `form:title` 19.313, `form:value` 19.316, `xforms:bind` 19.910 and `xml:id` 19.920.

The `<form:file>` element has the following child elements: `<form:properties>` 13.7 and `<office:event-listeners>` 10.3.19.

### 13.5.6 <form:formatted-text>

The `<form:formatted-text>` element defines a control for inputting text, which follows the format defined by a data style that is assigned to the control's graphical shape.

The `<form:formatted-text>` element is usable within the following elements: `<form:column>` 13.5.23 and `<form:form>` 13.3.

The `<form:formatted-text>` element has the following attributes: `form:control-implementation` 19.258, `form:convert-empty-to-null` 19.259, `form:current-value` 19.262, `form:data-field` 19.263, `form:delay-for-repeat` 19.266, `form:disabled` 19.268, `form:id` 19.276, `form:input-required` 19.921, `form:linked-cell` 19.283, `form:max-length` 19.288, `form:max-value` 19.289, `form:min-value` 19.290, `form:name` 19.294, `form:printable` 19.299, `form:readonly` 19.301, `form:repeat` 19.302, `form:spin-button` 19.306, `form:tab-index` 19.310, `form:tab-stop` 19.311, `form:title` 19.313, `form:validation` 19.315, `form:value` 19.316, `xforms:bind` 19.910 and `xml:id` 19.920.

The `<form:formatted-text>` element has the following child elements: `<form:properties>` 13.7 and `<office:event-listeners>` 10.3.19.

### 13.5.7 `<form:number>`

The `<form:number>` element defines a control which allows the user to enter a floating-point number.

The `<form:number>` element is usable within the following elements: `<form:column>` 13.5.23 and `<form:form>` 13.3.

The `<form:number>` element has the following attributes: `form:control-implementation` 19.258, `form:convert-empty-to-null` 19.259, `form:current-value` 19.262, `form:data-field` 19.263, `form:delay-for-repeat` 19.266, `form:disabled` 19.268, `form:id` 19.276, `form:linked-cell` 19.283, `form:max-length` 19.288, `form:max-value` 19.289, `form:min-value` 19.290, `form:name` 19.294, `form:printable` 19.299, `form:readonly` 19.301, `form:repeat` 19.302, `form:spin-button` 19.306, `form:tab-index` 19.310, `form:tab-stop` 19.311, `form:title` 19.313, `form:value` 19.316, `xforms:bind` 19.910 and `xml:id` 19.920.

The `<form:number>` element has the following child elements: `<form:properties>` 13.7 and `<office:event-listeners>` 10.3.19.

### 13.5.8 `<form:date>`

The `<form:date>` element defines a control for inputting date data.

The `<form:date>` element is usable within the following elements: `<form:column>` 13.5.23 and `<form:form>` 13.3.

The `<form:date>` element has the following attributes: `form:control-implementation` 19.258, `form:convert-empty-to-null` 19.259, `form:current-value` 19.262, `form:data-field` 19.263, `form:delay-for-repeat` 19.266, `form:disabled` 19.268, `form:id` 19.276, `form:input-required` 19.921, `form:linked-cell` 19.283, `form:max-length` 19.288, `form:max-value` 19.289, `form:min-value` 19.290, `form:name` 19.294, `form:printable` 19.299, `form:readonly` 19.301, `form:repeat` 19.302, `form:spin-button` 19.306, `form:tab-index` 19.310, `form:tab-stop` 19.311, `form:title` 19.313, `form:value` 19.316, `xforms:bind` 19.910 and `xml:id` 19.920.

The `<form:date>` element has the following child elements: `<form:properties>` 13.7 and `<office:event-listeners>` 10.3.19.

### 13.5.9 `<form:time>`

The `<form:time>` element defines a control for inputting time data.

The `<form:time>` element is usable within the following elements: `<form:column>` 13.5.23 and `<form:form>` 13.3.

The `<form:time>` element has the following attributes: `form:control-implementation` 19.258, `form:convert-empty-to-null` 19.259, `form:current-value` 19.262, `form:data-field` 19.263, `form:delay-for-repeat` 19.266, `form:disabled` 19.268, `form:id` 19.276, `form:input-required` 19.921, `form:linked-cell` 19.283, `form:maxLength` 19.288, `form:max-value` 19.289, `form:min-value` 19.290, `form:name` 19.294, `form:printable` 19.299, `form:readonly` 19.301, `form:repeat` 19.302, `form:spin-button` 19.306, `form:tab-index` 19.310, `form:tab-stop` 19.311, `form:title` 19.313, `form:value` 19.316, `xforms:bind` 19.910 and `xml:id` 19.920.

The `<form:time>` element has the following child elements: `<form:properties>` 13.7 and `<office:event-listeners>` 10.3.19.

### 13.5.10 `<form:fixed-text>`

The `<form:fixed-text>` element defines a control which attaches additional information to controls, or displays information. Only one label may be associated with a control.

The `<form:fixed-text>` element is usable within the following element: `<form:form>` 13.3.

The `<form:fixed-text>` element has the following attributes: `form:control-implementation` 19.258, `form:disabled` 19.268, `form:for` 19.275, `form:id` 19.276, `form:label` 19.282, `form:multi-line` 19.292, `form:name` 19.294, `form:printable` 19.299, `form:title` 19.313, `xforms:bind` 19.910 and `xml:id` 19.920.

The `<form:fixed-text>` element has the following child elements: `<form:properties>` 13.7 and `<office:event-listeners>` 10.3.19.

### 13.5.11 `<form:combobox>`

The `<form:combobox>` element defines a control which allows displaying and editing of text, and contains a list of possible values for that text.

The `<form:combobox>` element is usable within the following elements: `<form:column>` 13.5.23 and `<form:form>` 13.3.

The `<form:combobox>` element has the following attributes: `form:auto-complete` 19.252, `form:control-implementation` 19.258, `form:convert-empty-to-null` 19.259, `form:current-value` 19.262, `form:data-field` 19.263, `form:disabled` 19.268, `form:dropdown` 19.269, `form:id` 19.276, `form:input-required` 19.921, `form:linked-cell` 19.283, `form:list-source` 19.285, `form:list-source-type` 19.286, `form:maxLength` 19.288, `form:name` 19.294, `form:printable` 19.299, `form:readonly` 19.301, `form:size` 19.304, `form:source-cell-range` 19.305, `form:tab-index` 19.310, `form:tab-stop` 19.311, `form:title` 19.313, `form:value` 19.316, `xforms:bind` 19.910 and `xml:id` 19.920.

The `<form:combobox>` element has the following child elements: `<form:item>` 13.5.12, `<form:properties>` 13.7 and `<office:event-listeners>` 10.3.19.

### 13.5.12 `<form:item>`

The `<form:item>` element defines a list item for a `<form:combobox>` control.

The `<form:item>` element is usable within the following element: `<form:combobox>` 13.5.11.

The `<form:item>` element has the following attribute: `form:label` 19.282.

The `<form:item>` element has no child elements.

The `<form:item>` element has character data content.

### 13.5.13 `<form:listbox>`

The `<form:listbox>` element defines an input control that allows a user to select one or more items from a list. It is an alternative representation for a group of radio buttons.

The `<form:listbox>` element is usable within the following elements: `<form:column>` 13.5.23 and `<form:form>` 13.3.

The `<form:listbox>` element has the following attributes: `form:bound-column` 19.254, `form:control-implementation` 19.258, `form:data-field` 19.263, `form:disabled` 19.268, `form:dropdown` 19.269, `form:id` 19.276, `form:input-required` 19.921, `form:linked-cell` 19.283, `form:list-linkage-type` 19.284, `form:list-source` 19.285, `form:list-source-type` 19.286, `form:multiple` 19.293, `form:name` 19.294, `form:printable` 19.299, `form:size` 19.304, `form:source-cell-range` 19.305, `form:tab-index` 19.310, `form:tab-stop` 19.311, `form:title` 19.313, `form:xforms-list-source` 19.318, `xforms:bind` 19.910 and `xml:id` 19.920.

The `<form:listbox>` element has the following child elements: `<form:option>` 13.5.14, `<form:properties>` 13.7 and `<office:event-listeners>` 10.3.19.

### 13.5.14 `<form:option>`

The `<form:option>` element defines a list item for a `<form:listbox>` control.

The `<form:option>` element is usable within the following element: `<form:listbox>` 13.5.13.

The `<form:option>` element has the following attributes: `form:current-selected` 19.260, `form:label` 19.282, `form:selected` 19.303 and `form:value` 19.316.

The `<form:option>` element has no child elements.

The `<form:option>` element has character data content.

### 13.5.15 `<form:button>`

The `<form:button>` element defines a button control.

The `<form:button>` element is usable within the following element: `<form:form>` 13.3.

The `<form:button>` element has the following attributes: `form:button-type` 19.255, `form:control-implementation` 19.258, `form:default-button` 19.265, `form:delay-for-repeat` 19.266, `form:disabled` 19.268, `form:focus-on-click` 19.274, `form:id` 19.276, `form:image-align` 19.278, `form:image-data` 19.279, `form:image-position` 19.280, `form:label` 19.282, `form:name` 19.294, `form:printable` 19.299, `form:repeat` 19.302, `form:tab-index` 19.310, `form:tab-stop` 19.311, `form:title` 19.313, `form:toggle` 19.314, `form:value` 19.316, `form:xforms-submission` 19.319, `office:target-frame` 19.384, `xforms:bind` 19.910, `xlink:href` 19.916 and `xml:id` 19.920.

The `<form:button>` element has the following child elements: `<form:properties>` 13.7 and `<office:event-listeners>` 10.3.19.

### 13.5.16 `<form:image>`

The `<form:image>` element defines a graphical button control.

**Note:** HTML 4.01 only allows the button type to be “submit” for an image button. In OpenDocument, an image button can be of any type.

The `<form:image>` element is usable within the following element: `<form:form>` 13.3.

The `<form:image>` element has the following attributes: `form:button-type` 19.255, `form:control-implementation` 19.258, `form:disabled` 19.268, `form:id` 19.276, `form:image-data` 19.279, `form:name` 19.294, `form:printable` 19.299, `form:tab-index` 19.310, `form:tab-stop` 19.311, `form:title` 19.313, `form:value` 19.316, `office:target-frame` 19.384, `xforms:bind` 19.910, `xlink:href` 19.916 and `xml:id` 19.920.

The `<form:image>` element has the following child elements: `<form:properties>` 13.7 and `<office:event-listeners>` 10.3.19.

### 13.5.17 `<form:checkbox>`

The `<form:checkbox>` element defines an on/off control. The control is on when the value of the `form:current-state` attribute associated with the control element is `checked`.

The `<form:checkbox>` element is usable within the following elements: `<form:column>` 13.5.23 and `<form:form>` 13.3.

The `<form:checkbox>` element has the following attributes: `form:control-implementation` 19.258, `form:current-state` 19.261, `form:data-field` 19.263, `form:disabled` 19.268, `form:id` 19.276, `form:image-align` 19.278, `form:image-position` 19.280, `form:input-required` 19.921, `form:is-tristate` 19.281, `form:label` 19.282, `form:linked-cell` 19.283, `form:name` 19.294, `form:printable` 19.299, `form:state` 19.307, `form:tab-index` 19.310, `form:tab-stop` 19.311, `form:title` 19.313, `form:value` 19.316, `form:visual-effect` 19.317, `xforms:bind` 19.910 and `xml:id` 19.920.

The `<form:checkbox>` element has the following child elements: `<form:properties>` 13.7 and `<office:event-listeners>` 10.3.19.

### 13.5.18 `<form:radio>`

The `<form:radio>` element defines a control which acts like a check box except that when multiple radio buttons belong to the same group they are mutually exclusive. When one button is on, all of the other buttons with the same name are off.

Radio buttons are defined to belong to the same group if they have the same control name, as specified by their `form:name` attribute.

If a group of radio buttons is bound to one database field, and a user selects any given button, the reference value of the selected radio button is written into its database field.

The `<form:radio>` element is usable within the following element: `<form:form>` 13.3.

The `<form:radio>` element has the following attributes: `form:control-implementation` 19.258, `form:current-selected` 19.260, `form:data-field` 19.263, `form:disabled` 19.268, `form:id` 19.276, `form:image-align` 19.278, `form:image-position` 19.280, `form:input-required` 19.921, `form:label` 19.282, `form:linked-cell` 19.283, `form:name` 19.294, `form:printable` 19.299, `form:selected` 19.303, `form:tab-index` 19.310, `form:tab-stop` 19.311, `form:title` 19.313, `form:value` 19.316, `form:visual-effect` 19.317, `xforms:bind` 19.910 and `xml:id` 19.920.

The `<form:radio>` element has the following child elements: `<form:properties>` 13.7 and `<office:event-listeners>` 10.3.19.

### 13.5.19 <form:frame>

The `<form:frame>` element defines a frame in which controls may be visually arranged.

The `<form:frame>` element is usable within the following element: `<form:form>` 13.3.

The `<form:frame>` element has the following attributes: `form:control-implementation` 19.258, `form:disabled` 19.268, `form:for` 19.275, `form:id` 19.276, `form:label` 19.282, `form:name` 19.294, `form:printable` 19.299, `form:title` 19.313, `xforms:bind` 19.910 and `xml:id` 19.920.

The `<form:frame>` element has the following child elements: `<form:properties>` 13.7 and `<office:event-listeners>` 10.3.19.

### 13.5.20 <form:image-frame>

The `<form:image-frame>` element defines a graphical control. The control displays an image, whose location is described in the control.

The `<form:image-frame>` element is usable within the following element: `<form:form>` 13.3.

The `<form:image-frame>` element has the following attributes: `form:control-implementation` 19.258, `form:data-field` 19.263, `form:disabled` 19.268, `form:id` 19.276, `form:image-data` 19.279, `form:input-required` 19.921, `form:name` 19.294, `form:printable` 19.299, `form:readonly` 19.301, `form:title` 19.313, `xforms:bind` 19.910 and `xml:id` 19.920.

The `<form:image-frame>` element has the following child elements: `<form:properties>` 13.7 and `<office:event-listeners>` 10.3.19.

### 13.5.21 <form:hidden>

The `<form:hidden>` element defines a control that does not have a visual representation.

**Note:** A control without visual representation can be used for information not displayed to the user but is submitted as part of a form.

The `<form:hidden>` element is usable within the following element: `<form:form>` 13.3.

The `<form:hidden>` element has the following attributes: `form:control-implementation` 19.258, `form:id` 19.276, `form:name` 19.294, `form:value` 19.316, `xforms:bind` 19.910 and `xml:id` 19.920.

The `<form:hidden>` element has the following child elements: `<form:properties>` 13.7 and `<office:event-listeners>` 10.3.19.

### 13.5.22 <form:grid>

The `<form:grid>` element defines a control that displays table data.

Each column in the grid is specified by a `<form:column>` element.

The `<form:grid>` element is usable within the following element: `<form:form>` 13.3.

The `<form:grid>` element has the following attributes: `form:control-implementation` 19.258, `form:disabled` 19.268, `form:id` 19.276, `form:name` 19.294, `form:printable` 19.299, `form:tab-index` 19.310, `form:tab-stop` 19.311, `form:title` 19.313, `xforms:bind` 19.910 and `xml:id` 19.920.

The `<form:grid>` element has the following child elements: `<form:column>` 13.5.23, `<form:properties>` 13.7 and `<office:event-listeners>` 10.3.19.

### 13.5.23 <form:column>

The `<form:column>` element defines a column in a grid control.

The `<form:column>` element is usable within the following element: `<form:grid>` 13.5.22.

The `<form:column>` element has the following attributes: `form:control-implementation` 19.258, `form:label` 19.282, `form:name` 19.294 and `form:text-style-name` 19.312.

The `<form:column>` element has the following child elements: `<form:checkbox>` 13.5.17, `<form:combobox>` 13.5.11, `<form:date>` 13.5.8, `<form:formatted-text>` 13.5.6, `<form:listbox>` 13.5.13, `<form:number>` 13.5.7, `<form:text>` 13.5.2, `<form:textarea>` 13.5.3 and `<form:time>` 13.5.9.

### 13.5.24 <form:value-range>

The `<form:value-range>` element defines a control which allows the user to select a value from a number range.

The `<form:value-range>` element is usable within the following element: `<form:form>` 13.3.

The `<form:value-range>` element has the following attributes: `form:control-implementation` 19.258, `form:delay-for-repeat` 19.266, `form:disabled` 19.268, `form:id` 19.276, `form:linked-cell` 19.283, `form:max-value` 19.289, `form:min-value` 19.290, `form:name` 19.294, `form:orientation` 19.297, `form:page-step-size` 19.298, `form:printable` 19.299, `form:repeat` 19.302, `form:step-size` 19.308, `form:tab-index` 19.310, `form:tab-stop` 19.311, `form:title` 19.313, `form:value` 19.316, `xforms:bind` 19.910 and `xml:id` 19.920.

The `<form:value-range>` element has the following child elements: `<form:properties>` 13.7 and `<office:event-listeners>` 10.3.19.

### 13.5.25 <form:generic-control>

The `<form:generic-control>` element defines an implementation-defined placeholder for a generic control. The generic control can contain any properties and any events.

The `<form:generic-control>` element is usable within the following element: `<form:form>` 13.3.

The `<form:generic-control>` element has the following attributes: `form:control-implementation` 19.258, `form:id` 19.276, `form:name` 19.294, `xforms:bind` 19.910 and `xml:id` 19.920.

The `<form:generic-control>` element has the following child elements: `<form:properties>` 13.7 and `<office:event-listeners>` 10.3.19.

## 13.6 Event Listeners

Forms and form controls may have event listeners attached. The event listeners that are attached to a control are represented by an event listener element as described in section 14.5. This element is contained within form or form control elements.

Section 19.433 contains guidelines for event names that may be used within forms and form controls. In addition to those, the events listed in Table 9 may be used in forms and form controls.

*Table 9 - Form control event names*

<b>Value of <code>script:event-name</code> Attribute</b>	<b>Applies To</b>	<b>Description of Event</b>
<code>form:approveaction</code>	Button or image.	Occurs before an “on perform action” event takes place. Allows a user to veto an action.
<code>form:performaction</code>	Button or image.	Occurs when the control action is to be performed. The common interpretation of this event is “pressing the button”.
<code>form:textchange</code>	All controls that allow text input.	Occurs when a user changes the text in a control.
<code>form:itemstatechange</code>	Check box or radio button.	Occurs when the state of a check box or radio button changes.
<code>form:mousedrag</code>	All controls.	Occurs when a user presses and holds one of the mouse buttons and moves the mouse pointer onto a control.
<code>form:approvereset</code>	same objects as for <code>form:on-reset</code>	Occurs before an “on-reset” event takes place. Allows a user to veto a reset event.
<code>form:approveupdate</code>	All controls that can be bound to a database field, that is controls that contain the <code>data-field</code> attribute.	Occurs before an “on update” event takes place. Allows a user to veto an update.
<code>form:update</code>	All controls that can be bound to a database field, that is controls that contain the <code>data-field</code> attribute.	Occurs when the content of a control that is bound to a database field is committed.
<code>form:load</code>	Forms.	Occurs when a form establishes a connection to the data source.
<code>form:startreload</code>	Forms.	Occurs when a form is about to refresh a data source connection.
<code>form:reload</code>	Forms.	Occurs when a form has refreshed a data source connection.
<code>form:startunload</code>	Forms.	Occurs when a form is about to drop a data source connection.
<code>form:unload</code>	Forms.	Occurs when a form has dropped a data source connection.
<code>form:confirmdelete</code>	Forms.	Occurs when a user is about to delete a record.
<code>form:approverowchange</code>	Forms.	Occurs before an “on row change” event takes place. Allows a user to veto a change.
<code>form:rowchange</code>	Forms.	Occurs after changes to a row are complete.

<b>Value of <i>script:event-name</i> Attribute</b>	<b>Applies To</b>	<b>Description of Event</b>
form:approvecursor-move	Forms.	Occurs before a form is moved to another row. Allows a user to veto a move.
form:cursormove	Forms.	Occurs after a form is moved to another row.
form:supplyparameter	Forms.	Occurs when a form asks for parameters to connect to a data source.
form:error	Forms, combo boxes and list boxes.	Occurs when a database-related error occurs.
form:adjust	Value Range	Occurs when the value of a Value Range element has been adjusted.

## 13.7 <form:properties>

The <form:properties> element is a container for <form:property> and <form:list-property> elements.

The <form:properties> element is usable within the following elements: <form:button> 13.5.15, <form:checkbox> 13.5.17, <form:combobox> 13.5.11, <form:date> 13.5.8, <form:file> 13.5.5, <form:fixed-text> 13.5.10, <form:form> 13.3, <form:formatted-text> 13.5.6, <form:frame> 13.5.19, <form:generic-control> 13.5.25, <form:grid> 13.5.22, <form:hidden> 13.5.21, <form:image> 13.5.16, <form:image-frame> 13.5.20, <form:listbox> 13.5.13, <form:number> 13.5.7, <form:password> 13.5.4, <form:radio> 13.5.18, <form:text> 13.5.2, <form:textarea> 13.5.3, <form:time> 13.5.9 and <form:value-range> 13.5.24.

The <form:properties> element has no attributes.

The <form:properties> element has the following child elements: <form:list-property> 13.9 and <form:property> 13.8.

## 13.8 <form:property>

The <form:property> element defines the name, type and value of a property.

The <form:property> element is usable within the following element: <form:properties> 13.7.

The <form:property> element has the following attributes: *form:property-name* 19.300, *office:boolean-value* 19.371, *office:currency* 19.373, *office:date-value* 19.374, *office:string-value* 19.383, *office:time-value* 19.386, *office:value* 19.388, *office:value-type* 19.389.

The <form:property> element has no child elements.

## 13.9 <form:list-property>

The <form:list-property> element container for <form:list-value> elements.

The <form:list-property> element has an *office:value-type* attribute (19.389) that acts as a constraint on the attributes that may appear on its child <form:list-value> element.

The `<form:list-property>` element is usable within the following element:  
`<form:properties>` 13.7.

The `<form:list-property>` element has the following attributes: `form:property-name 19.300`, `office:value-type 19.389`.

The `<form:list-property>` element has the following child elements: `<form:list-value>` 13.10.

## 13.10 `<form:list-value>`

The list value element contains value attributes for the value type given in the containing `<form:list-property>` element.

The `<form:list-value>` element is usable within the following element: `<form:list-property>` 13.9.

The `<form:list-value>` element has the following attributes: `office:boolean-value 19.371`, `office:currency 19.373`, `office:date-value 19.374`, `office:string-value 19.383`, `office:time-value 19.386`, `office:value 19.388`.

The `<form:list-value>` element has no child elements.

## 14 Common Content

### 14.1 <office:annotation>

The `<office:annotation>` element specifies an OpenDocument annotation. The annotation's text is contained in `<text:p>` and `<text:list>` elements.

The `<office:annotation>` element is usable within the following elements:

`<draw:page>` 10.2.4, `<table:covered-table-cell>` 9.1.5, `<table:table-cell>` 9.1.4, `<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<office:annotation>` element has the following attributes: `draw:caption-point-x` 19.116, `draw:caption-point-y` 19.117, `draw:class-names` 19.120, `draw:corner-radius` 19.127, `draw:id` 19.187, `draw:layer` 19.189, `draw:name` 19.197, `draw:style-name` 19.219, `draw:text-style-name` 19.227, `draw:transform` 19.228, `draw:z-index` 19.231, `office:display` 19.378, `office:name` 19.380, `presentation:class-names` 19.394, `presentation:style-name` 19.426, `svg:height` 19.543, `svg:rx` 19.554, `svg:ry` 19.555, `svg:width` 19.575, `svg:x` 19.577, `svg:y` 19.581, `table:end-cell-address` 19.631, `table:end-x` 19.636, `table:end-y` 19.637, `table:table-background` 19.733, `text:anchor-page-number` 19.758, `text:anchor-type` 19.759 and `xml:id` 19.920.

The `<office:annotation>` element has the following child elements: `<dc:creator>` 4.3.2.7, `<dc:date>` 4.3.2.10, `<meta:creator-initials>` 14.4, `<meta:date-string>` 14.3, `<text:list>` 5.3.1 and `<text:p>` 5.1.3.

### 14.2 <office:annotation-end>

The `<office:annotation-end>` element may be used to define the end of a text range of document content that spans element boundaries. In that case, an `<office:annotation>` element shall precede the `<office:annotation-end>` element. Both elements shall have the same value for their `office:name` attribute.

The `<office:annotation-end>` element shall be preceded by an `<office:annotation>` element that has the same value for its `office:name` attribute as the `<office:annotation-end>` element.

An `<office:annotation-end>` element without a preceding `<office:annotation>` element that has the same name assigned is ignored.

The `<office:annotation-end>` element is usable within the following elements:

`<text:a>` 6.1.8, `<text:h>` 5.1.2, `<text:meta>` 6.1.9, `<text:meta-field>` 7.5.19, `<text:p>` 5.1.3, `<text:ruby-base>` 6.4.2 and `<text:span>` 6.1.7.

The `<office:annotation-end>` element has the following attribute: `office:name` 19.380.

The `<office:annotation-end>` element has no child elements.

### 14.3 <meta:date-string> (deprecated)

The `<meta:date-string>` element contains a date that is of type string and not a date type. This value is used only for display as entered.

The `<meta:date-string>` element is usable within the following element: `<office:annotation>` 14.1.

The `<meta:date-string>` element has no attributes.  
The `<meta:date-string>` element has no child elements.  
The `<meta:date-string>` element has content of data type `string` 18.2.

## 14.4 <meta:creator initials>

The `<meta:creator initials>` element represents the initials of the author of the annotation represented by the parent `<office:annotation>` element.

The `<meta:creator initials>` element is usable within the following element:  
`<office:annotation>` 14.1.  
The `<meta:creator initials>` element has no attributes.  
The `<meta:creator initials>` element has no child elements.  
The `<meta:creator initials>` element has character data content.

## 14.5 Event Listener Tables

### 14.5.1 General

Controls, images, text boxes, or an entire document can support events. An event binds the occurrence of a condition to an action that is executed if the condition arises.

Events are represented as follows:

- All of the event elements that are associated with an object are located in an `<office:event-listeners>` element.
- Each event-to-action association is recorded in one child element.
- Depending on the type of action that the event triggers, the following child elements are used:
  - The `<script:event-listener>` 14.5.2 element represents events that are bound to a macro or script.
  - The `<presentation:event-listener>` 10.9.2 element represents events that are bound to an action that is specific to a presentation.

### 14.5.2 <script:event-listener>

The `<script:event-listener>` element binds an event to a macro.

The `<script:event-listener>` element is usable within the following element:  
`<office:event-listeners>` 10.3.19.  
The `<script:event-listener>` element has the following attributes: `script:event-name` 19.433, `script:language` 19.434, `script:macro-name` 19.435, `xlink:actuate` 19.915, `xlink:href` 19.916 and `xlink:type` 19.919.  
The `<script:event-listener>` element has no child elements.

### 14.5.3 Event Types

Event types are implementation-defined. See 19.433.

#### 14.5.4 Macro Name and Location

Macro code for an event may be specified by either an `xlink:href` 19.916 attribute as an IRI, or a `script:macro-name` 19.435 attribute by name. Macro names and applicable IRI schemes are script-language-dependent.

### 14.6 <math:math>

The `<math:math>` element serves as a container for content defined by the MathML standard. See [MathML].

The `<math:math>` element is usable within the following elements: `<db:component>` 12.25.5 and `<draw:object>` 10.4.6.2.

### 14.7 DDE Connections

#### 14.7.1 General

A Dynamic Data Exchange (DDE) connection consists of parameters for a DDE target application, a file name, and a command string. A DDE connection also takes a parameter that specifies whether it will be updated automatically or only on the user's request. Every DDE connection shall be named.

All elements making use of DDE connections shall contain their content (or its presentation), so that documents using DDE can still be properly displayed on machines which do not support the DDE mechanism, or where the DDE target is not available. Consumers should preserve the DDE connection information even if they cannot make use of it.

**Note:** DDE only is available on some operating systems. In order to create portable documents, authors are advised to use this feature in their documents with great care.

#### 14.7.2 <text:dde-connection-decls>

The `<text:dde-connection-decls>` element is a container element for `<text:dde-connection-decl>` elements in text documents.

The `<text:dde-connection-decls>` element is usable within the following elements: `<office:chart>` 3.8, `<office:drawing>` 3.5, `<office:presentation>` 3.6, `<office:spreadsheet>` 3.7, `<office:text>` 3.4, `<style:footer>` 16.12, `<style:footer-first>` 16.13, `<style:footer-left>` 16.15, `<style:header>` 16.10, `<style:header-first>` 16.11 and `<style:header-left>` 16.14.

The `<text:dde-connection-decls>` element has no attributes.

The `<text:dde-connection-decls>` element has the following child element: `<text:dde-connection-decl>` 14.7.3.

#### 14.7.3 <text:dde-connection-decl>

The `<text:dde-connection-decl>` element defines a DDE connection for use in DDE text fields. Multiple DDE fields can refer to one DDE connection by using the same name.

The `<text:dde-connection-decl>` element is usable within the following element: `<text:dde-connection-decls>` 14.7.2.

The `<text:dde-connection-decl>` element has the following attributes: `office:automatic-update` 19.370, `office:dde-application` 19.375, `office:dde-item` 19.376, `office:dde-topic` 19.377 and `office:name` 19.380.

The `<text:dde-connection-decl>` element has no child elements.

#### 14.7.4 <table:dde-link>

The `<table:dde-link>` element contains DDE source data for use in spreadsheet formulas. It contains a `<office:dde-source>` element and a simple table element that may be used to cache the data of a DDE source. The table does not need a name and does not contain style information. Only the data contained in the cell attributes is used. The cells themselves remain empty.

The `<table:dde-link>` element is usable within the following element: `<table:dde-links>` 9.8.

The `<table:dde-link>` element has no attributes.

The `<table:dde-link>` element has the following child elements: `<office:dde-source>` 14.7.5 and `<table:table>` 9.1.2.

#### 14.7.5 <office:dde-source>

The `<office:dde-source>` element defines a DDE connection.

If the element is a child element of a `<table:table>` element, the `<office:dde-source>` element specifies that the table is linked through DDE along the connection data.

If the element is a child element of a `<text:section>` element, the `<office:dde-source>` element specifies that the text section is linked through DDE along the connection data.

If the element is a child element of a `<table:dde-link>` element, the `<office:dde-source>` element specifies the connection data for a DDE link that can be used in spreadsheet formulas.

The `<office:dde-source>` element is usable within the following elements: `<table:dde-link>` 14.7.4, `<table:table>` 9.1.2 and `<text:section>` 5.4.

The `<office:dde-source>` element has the following attributes: `office:automatic-update` 19.370, `office:conversion-mode` 19.372, `office:dde-application` 19.375, `office:dde-item` 19.376, `office:dde-topic` 19.377 and `office:name` 19.380.

The `<office:dde-source>` element has no child elements.

# 15 SMIL Animations

## 15.1 General

This section specifies [SMIL20] based elements and attributes that can be used within the OpenDocument format for animation effects.

## 15.2 Basic Animation Elements

### 15.2.1 General

The animation elements are based upon §3.5 and §12.5 of [SMIL20], and section §19.2 of [SVG].

### 15.2.2 `<anim:animate>`

See §3.5.1 of [SMIL20].

The `<anim:animate>` element is usable within the following elements: `<anim:iterate>` 15.4.4, `<anim:par>` 15.4.2, `<anim:seq>` 15.4.3, `<draw:page>` 10.2.4 and `<style:master-page>` 16.9.

The `<anim:animate>` element has the following attributes: `anim:formula` 19.6, `anim:sub-item` 19.11, `smil:accelerate` 19.436, `smil:accumulate` 19.437, `smil:additive` 19.438, `smil:attributeName` 19.439, `smil:autoReverse` 19.440, `smil:begin` 19.441, `smil:by` 19.442, `smil:calcMode` 19.443, `smil:decelerate` 19.444, `smil:dur` 19.446, `smil:end` 19.447, `smil:fill` 19.450, `smil:fillDefault` 19.451, `smil:from` 19.452, `smil:keySplines` 19.453, `smil:keyTimes` 19.454, `smil:repeatCount` 19.456, `smil:repeatDur` 19.457, `smil:restart` 19.458, `smil:restartDefault` 19.459, `smil:targetElement` 19.461, `smil:to` 19.462 and `smil:values` 19.464.

The `<anim:animate>` element has no child elements.

### 15.2.3 `<anim:animateTransform>`

The `<anim:animateTransform>` element defines the transformation of an animation. It is based on the [SVG] `<svg:animateTransform>` element. See §19.2.14 of [SVG].

The `<anim:animateTransform>` element is usable within the following elements: `<anim:iterate>` 15.4.4, `<anim:par>` 15.4.2, `<anim:seq>` 15.4.3, `<draw:page>` 10.2.4 and `<style:master-page>` 16.9.

The `<anim:animateTransform>` element has the following attributes: `anim:formula` 19.6, `anim:sub-item` 19.11, `smil:accelerate` 19.436, `smil:accumulate` 19.437, `smil:additive` 19.438, `smil:attributeName` 19.439, `smil:autoReverse` 19.440, `smil:begin` 19.441, `smil:by` 19.442, `smil:decelerate` 19.444, `smil:dur` 19.446, `smil:end` 19.447, `smil:fill` 19.450, `smil:fillDefault` 19.451, `smil:from` 19.452, `smil:repeatCount` 19.456, `smil:repeatDur` 19.457, `smil:restart` 19.458, `smil:restartDefault` 19.459, `smil:targetElement` 19.461, `smil:to` 19.462, `smil:values` 19.464 and `svg:type` 19.565.

The `<anim:animateTransform>` element has no child elements.

#### 15.2.4 <anim:set>

See §3.5.2 of [SMIL20].

The `<anim:set>` element is usable within the following elements: `<anim:iterate>` 15.4.4, `<anim:par>` 15.4.2, `<anim:seq>` 15.4.3, `<draw:page>` 10.2.4 and `<style:master-page>` 16.9.

The `<anim:set>` element has the following attributes: `anim:sub-item` 19.11, `smil:accelerate` 19.436, `smil:accumulate` 19.437, `smil:additive` 19.438, `smil:attributeName` 19.439, `smil:autoReverse` 19.440, `smil:begin` 19.441, `smil:decelerate` 19.444, `smil:dur` 19.446, `smil:end` 19.447, `smil:fill` 19.450, `smil:fillDefault` 19.451, `smil:repeatCount` 19.456, `smil:repeatDur` 19.457, `smil:restart` 19.458, `smil:restartDefault` 19.459, `smil:targetElement` 19.461 and `smil:to` 19.462.

The `<anim:set>` element has no child elements.

#### 15.2.5 <anim:animateMotion>

See §19.2.12 of [SVG] and §3.5.3 of [SMIL20].

The `<anim:animateMotion>` element is usable within the following elements: `<anim:iterate>` 15.4.4, `<anim:par>` 15.4.2, `<anim:seq>` 15.4.3, `<draw:page>` 10.2.4 and `<style:master-page>` 16.9.

The `<anim:animateMotion>` element has the following attributes: `anim:formula` 19.6, `anim:sub-item` 19.11, `smil:accelerate` 19.436, `smil:accumulate` 19.437, `smil:additive` 19.438, `smil:attributeName` 19.439, `smil:autoReverse` 19.440, `smil:begin` 19.441, `smil:by` 19.442, `smil:calcMode` 19.443, `smil:decelerate` 19.444, `smil:dur` 19.446, `smil:end` 19.447, `smil:fill` 19.450, `smil:fillDefault` 19.451, `smil:from` 19.452, `smil:keySplines` 19.453, `smil:keyTimes` 19.454, `smil:repeatCount` 19.456, `smil:repeatDur` 19.457, `smil:restart` 19.458, `smil:restartDefault` 19.459, `smil:targetElement` 19.461, `smil:to` 19.462, `smil:values` 19.464, `svg:origin` 19.548 and `svg:path` 19.552.

The `<anim:animateMotion>` element has no child elements.

#### 15.2.6 <anim:animateColor>

See §3.5.4 of [SMIL20].

The `<anim:animateColor>` element is usable within the following elements: `<anim:iterate>` 15.4.4, `<anim:par>` 15.4.2, `<anim:seq>` 15.4.3, `<draw:page>` 10.2.4 and `<style:master-page>` 16.9.

The `<anim:animateColor>` element has the following attributes: `anim:color-interpolation` 19.2, `anim:color-interpolation-direction` 19.3, `anim:formula` 19.6, `anim:sub-item` 19.11, `smil:accelerate` 19.436, `smil:accumulate` 19.437, `smil:additive` 19.438, `smil:attributeName` 19.439, `smil:autoReverse` 19.440, `smil:begin` 19.441, `smil:by` 19.442, `smil:calcMode` 19.443, `smil:decelerate` 19.444, `smil:dur` 19.446, `smil:end` 19.447, `smil:fill` 19.450, `smil:fillDefault` 19.451, `smil:from` 19.452, `smil:keySplines` 19.453, `smil:keyTimes` 19.454, `smil:repeatCount` 19.456, `smil:repeatDur` 19.457, `smil:restart` 19.458, `smil:restartDefault` 19.459, `smil:targetElement` 19.461, `smil:to` 19.462 and `smil:values` 19.464.

The `<anim:animateColor>` element has no child elements.

### 15.2.7 <anim:transitionFilter>

The `<anim:transitionFilter>` element is based on the [SMIL20] `<smil:transitionFilter>` element. See §12.5.1 of [SMIL20].

The `<anim:transitionFilter>` element is usable within the following elements: `<anim:iterate>` 15.4.4, `<anim:par>` 15.4.2, `<anim:seq>` 15.4.3, `<draw:page>` 10.2.4 and `<style:master-page>` 16.9.

The `<anim:transitionFilter>` element has the following attributes: `anim:formula` 19.6, `anim:sub-item` 19.11, `smil:accelerate` 19.436, `smil:accumulate` 19.437, `smil:additive` 19.438, `smil:autoReverse` 19.440, `smil:begin` 19.441, `smil:by` 19.442, `smil:calcMode` 19.443, `smil:decelerate` 19.444, `smil:direction` 19.445, `smil:dur` 19.446, `smil:end` 19.447, `smil:fadeColor` 19.449, `smil:fill` 19.450, `smil:fillDefault` 19.451, `smil:from` 19.452, `smil:mode` 19.455, `smil:repeatCount` 19.456, `smil:repeatDur` 19.457, `smil:restart` 19.458, `smil:restartDefault` 19.459, `smil:subtype` 19.460, `smil:targetElement` 19.461, `smil:to` 19.462, `smil:type` 19.463 and `smil:values` 19.464.

The `<anim:transitionFilter>` element has no child elements.

## 15.3 Animation Model Attributes

### 15.3.1 General

The animation model uses the same concepts and syntax as specified in §3 of [SMIL20].

### 15.3.2 Simple Animation Functions

Simplified animations can be specified using the [SMIL20] `smil:from`, `smil:to` and `smil:by` attributes. See §3.4.4 of [SMIL20].

## 15.4 Animation Timing

### 15.4.1 General

Animation timing uses the same concepts and syntax as specified in §10 and §11 of [SMIL20].

### 15.4.2 <anim:par>

The `<anim:par>` element is a container of child animation nodes that have a common starting point.

The `<anim:par>` element is usable within the following elements: `<anim:iterate>` 15.4.4, `<anim:par>` 15.4.2, `<anim:seq>` 15.4.3, `<draw:page>` 10.2.4 and `<style:master-page>` 16.9.

The `<anim:par>` element has the following attributes: `anim:id` 19.7, `presentation:group-id` 19.401, `presentation:master-element` 19.402, `presentation:node-type` 19.406, `presentation:preset-class` 19.414, `presentation:preset-id` 19.415, `presentation:preset-sub-type` 19.416, `smil:accelerate` 19.436, `smil:autoReverse` 19.440, `smil:begin` 19.441, `smil:decelerate` 19.444, `smil:dur` 19.446, `smil:end` 19.447, `smil:endsync` 19.448, `smil:fill` 19.450, `smil:fillDefault` 19.451, `smil:repeatCount` 19.456, `smil:repeatDur` 19.457, `smil:restart` 19.458, `smil:restartDefault` 19.459 and `xml:id` 19.920.

The `<anim:par>` element has the following child elements: `<anim:animate>` 15.2.2, `<anim:animateColor>` 15.2.6, `<anim:animateMotion>` 15.2.5, `<anim:animateTransform>` 15.2.3, `<anim:audio>` 15.5, `<anim:command>` 15.6.1,

<anim:iterate> 15.4.4, <anim:par> 15.4.2, <anim:seq> 15.4.3, <anim:set> 15.2.4 and <anim:transitionFilter> 15.2.7.

### 15.4.3 <anim:seq>

The <anim:seq> element is a container for child animation nodes that start in a sequential fashion.

The <anim:seq> element is usable within the following elements: <anim:iterate> 15.4.4, <anim:par> 15.4.2, <anim:seq> 15.4.3, <draw:page> 10.2.4 and <style:master-page> 16.9.

The <anim:seq> element has the following attributes: anim:id 19.7, presentation:group-id 19.401, presentation:master-element 19.402, presentation:node-type 19.406, presentation:preset-class 19.414, presentation:preset-id 19.415, presentation:preset-sub-type 19.416, smil:accelerate 19.436, smil:autoReverse 19.440, smil:begin 19.441, smil:decelerate 19.444, smil:dur 19.446, smil:end 19.447, smil:endsync 19.448, smil:fill 19.450, smil:fillDefault 19.451, smil:repeatCount 19.456, smil:repeatDur 19.457, smil:restart 19.458, smil:restartDefault 19.459 and xml:id 19.920.

The <anim:seq> element has the following child elements: <anim:animate> 15.2.2, <anim:animateColor> 15.2.6, <anim:animateMotion> 15.2.5, <anim:animateTransform> 15.2.3, <anim:audio> 15.5, <anim:command> 15.6.1, <anim:iterate> 15.4.4, <anim:par> 15.4.2, <anim:seq> 15.4.3, <anim:set> 15.2.4 and <anim:transitionFilter> 15.2.7.

### 15.4.4 <anim:iterate>

The <anim:iterate> element defines a parallel time container. The animation effects are applied to the child objects of the target element. The effects for each child object are started one after another, where the delay between the start of the effects of two subsequent children is specified by the anim:iterate-interval attribute. 19.8

The <anim:iterate> element is usable within the following elements: <anim:iterate> 15.4.4, <anim:par> 15.4.2, <anim:seq> 15.4.3, <draw:page> 10.2.4 and <style:master-page> 16.9.

The <anim:iterate> element has the following attributes: anim:id 19.7, anim:iterate-interval 19.8, anim:iterate-type 19.9, anim:sub-item 19.11, presentation:group-id 19.401, presentation:master-element 19.402, presentation:node-type 19.406, presentation:preset-class 19.414, presentation:preset-id 19.415, presentation:preset-sub-type 19.416, smil:accelerate 19.436, smil:autoReverse 19.440, smil:begin 19.441, smil:decelerate 19.444, smil:dur 19.446, smil:end 19.447, smil:endsync 19.448, smil:fill 19.450, smil:fillDefault 19.451, smil:repeatCount 19.456, smil:repeatDur 19.457, smil:restart 19.458, smil:restartDefault 19.459, smil:targetElement 19.461 and xml:id 19.920.

The <anim:iterate> element has the following child elements: <anim:animate> 15.2.2, <anim:animateColor> 15.2.6, <anim:animateMotion> 15.2.5, <anim:animateTransform> 15.2.3, <anim:audio> 15.5, <anim:command> 15.6.1, <anim:iterate> 15.4.4, <anim:par> 15.4.2, <anim:seq> 15.4.3, <anim:set> 15.2.4 and <anim:transitionFilter> 15.2.7.

## 15.5 <anim:audio>

The `<anim:audio>` element is based on the [SMIL20] `<smil:audio>` element. It specifies the playback of audio streams during an animation. See §7.3.1 of [SMIL20].

The `<anim:audio>` element is usable within the following elements: `<anim:iterate>` 15.4.4, `<anim:par>` 15.4.2, `<anim:seq>` 15.4.3, `<draw:page>` 10.2.4 and `<style:master-page>` 16.9.

The `<anim:audio>` element has the following attributes: `anim:audio-level` 19.4, `anim:id` 19.7, `presentation:group-id` 19.401, `presentation:master-element` 19.402, `presentation:node-type` 19.406, `presentation:preset-class` 19.414, `presentation:preset-id` 19.415, `presentation:preset-sub-type` 19.416, `smil:begin` 19.441, `smil:dur` 19.446, `smil:end` 19.447, `smil:fill` 19.450, `smil:fillDefault` 19.451, `smil:repeatCount` 19.456, `smil:repeatDur` 19.457, `smil:restart` 19.458, `smil:restartDefault` 19.459, `xlink:href` 19.916 and `xml:id` 19.920.

The `<anim:audio>` element has no child elements.

## 15.6 Animation Command Elements

### 15.6.1 <anim:command>

The `<anim:command>` element specifies an animation command.

**Note:** See 19.5 for details about the element's usage in presentation documents.

The `<anim:command>` element is usable within the following elements: `<anim:iterate>` 15.4.4, `<anim:par>` 15.4.2, `<anim:seq>` 15.4.3, `<draw:page>` 10.2.4 and `<style:master-page>` 16.9.

The `<anim:command>` element has the following attributes: `anim:command` 19.5, `anim:id` 19.7, `anim:sub-item` 19.11, `presentation:group-id` 19.401, `presentation:master-element` 19.402, `presentation:node-type` 19.406, `presentation:preset-class` 19.414, `presentation:preset-id` 19.415, `presentation:preset-sub-type` 19.416, `smil:begin` 19.441, `smil:end` 19.447, `smil:targetElement` 19.461 and `xml:id` 19.920.

The `<anim:command>` element has the following child element: `<anim:param>` 15.6.2.

### 15.6.2 <anim:param>

The `<anim:param>` element specifies a parameter for an `<anim:command>` element.

The `<anim:param>` element is usable within the following element: `<anim:command>` 15.6.1.

The `<anim:param>` element has the following attributes: `anim:name` 19.10 and `anim:value` 19.12.

The `<anim:param>` element has no child elements.

---

# 16 Styles

## 16.1 General

Objects in an office document may have formatting properties. A formatting property influences the visual representation of an object.

In the OpenDocument format, formatting properties are stored in styles.

There are two reasons for using styles to store formatting properties:

- 1) Format information is separated from document content.
- 2) Styles enable consistent formatting and changing of formatting for objects subject to styles.

Styles and font face declarations are referenced by their `style:name` attribute. A referenced style or font face declaration should be defined in the same file as the reference, or in `styles.xml`.

## 16.2 `<style:style>`

The `<style:style>` element represents styles.

Styles defined by the `<style:style>` element use a hierarchical style model. The `<style:style>` element supports inheritance of formatting properties by a style from its parent style. A parent style is specified by the `style:parent-style-name` attribute on a `<style:style>` element.

The determination of the value of a formatting property begins with any style that is specified by an element. If the formatting property is present in that style, its value is used.

If that style does not specify a value for that formatting property and it has a parent style, the value of the formatting element is taken from the parent style, if present.

If the parent style does not have a value for the formatting property, the search for the formatting property value continues up parent styles until either the formatting property has been found or a style is found with no parent style.

If a search of the parent styles of a style does not result in a value for a formatting property, the determination of its value depends on the style family and the element to which a style is applied.

For styles with family `text` which are applied to elements which are contained in another element that specifies a style with family `text`, the search continues within the `text` style that is applied to the nearest ancestor element that specifies a style with family `text`, and continues in its parent styles.

For styles with family `text` which are applied to elements which are contained in a paragraph element 6.1.1, the search continues within the `paragraph` style that is applied to the paragraph element, and continues in its parent styles.

For styles with family `paragraph` which are applied to paragraph elements which are contained in a drawing shape or a chart element, the search continues within the style applied by a `draw:text-style-name` 19.227 attribute to the drawing shape or chart element and its parent styles, and then with graphic, presentation or chart style that is applied to the drawing object or chart element, and continues in its parent styles.

For styles with family `paragraph` which are applied to paragraph elements which are contained in a table cell, the search continues within the `table-cell` style that is applied to

the table-cell, and continues in its parent styles. If a value for the formatting property has not been found, the search continues as defined for styles with family table-cell.

For styles with family table-cell which are applied to a table cell, the search continues with the style specified by the table:default-cell-style-name attribute 19.619 of the table cell's `<table:table-row>` parent element, if present, and then with the style specified by the table:default-cell-style-name attribute of the `<table:table-column>` element associated with the table cell.

Styles of family drawing-page contain formatting for `<draw:page>` elements. First the style and its parent style chain is evaluated, which is directly referenced in the draw:style-name attribute of a draw element. If the formatting property is not present in this chain, then the search continues with the style and its parent style chain, that is referenced in the draw:style-name attribute of that `<style:master-page>` element, which is specified in the draw:master-page-name attribute of the `<draw:page>` element.

In all other cases, or if a value for the formatting property has not been found by any of the family specific rules, a default style 16.4 that has the same family as the style initially declared sets the value. If a value has not been found by these steps, but this specification defines a default value, then this default value is used. In all remaining cases an implementation-dependent value is used.

The `<style:style>` element is usable within the following elements: `<office:automatic-styles>` 3.15.3 and `<office:styles>` 3.15.2.

The `<style:style>` element has the following attributes: style:auto-update 19.467, style:class 19.470, style:data-style-name 19.473, style:default-outline-level 19.474, style:display-name 19.476, style:family 19.480, style:list-level 19.499, style:list-style-name 19.500, style:master-page-name 19.501, style:name 19.502, style:next-style-name 19.503, style:parent-style-name 19.510 and style:percentage-data-style-name 19.511.

The `<style:style>` element has the following child elements: `<style:chart-properties>` 17.22, `<style:drawing-page-properties>` 17.25, `<style:graphic-properties>` 17.21, `<style:map>` 16.3, `<style:paragraph-properties>` 17.6, `<style:ruby-properties>` 17.10, `<style:section-properties>` 17.11, `<style:table-cell-properties>` 17.18, `<style:table-column-properties>` 17.16, `<style:table-properties>` 17.15, `<style:table-row-properties>` 17.17, `<style:text-properties>` 16.29.29.

## 16.3 `<style:map>`

The `<style:map>` element represents a condition, which if met, results in the mapping of a style. A style that contains one or more mappings is called a conditional style. Each condition is represented by a unique `<style:map>` element.

Conditional styles are supported by:

- paragraph styles contained in text documents
- table cell styles contained in spreadsheets
- data styles

The `<style:map>` element is usable within the following elements: `<number:boolean-style>` 16.29.24, `<number:currency-style>` 16.29.8, `<number:date-style>` 16.29.11, `<number:number-style>` 16.29.2, `<number:percentage-style>` 16.29.10, `<number:text-style>` 16.29.26, `<number:time-style>` 16.29.19 and `<style:style>` 16.2.

The `<style:map>` element has the following attributes: style:apply-style-name 19.466, style:base-cell-address 19.468 and style:condition 19.472.

The `<style:map>` element has no child elements.

## 16.4 `<style:default-style>`

The `<style:default-style>` element represents default styles. A default style specifies default formatting properties for a style family. These defaults are used if a formatting property is neither specified by an automatic nor a common style. Default styles exist for all style families that are represented by the `<style:style>` element specified by the `style:family` attribute 19.480.

An OpenDocument document should contain the default styles of the style families for which are used in common or automatic styles in the document.

The `<style:default-style>` element is usable within the following element:  
`<office:styles>` 3.15.2.

The `<style:default-style>` element has the following attributes: `style:family` 19.480.

The `<style:default-style>` element has the following child elements: `<style:chart-properties>` 17.22, `<style:drawing-page-properties>` 17.25, `<style:graphic-properties>` 17.21, `<style:paragraph-properties>` 17.6, `<style:ruby-properties>` 17.10, `<style:section-properties>` 17.11, `<style:table-cell-properties>` 17.18, `<style:table-column-properties>` 17.16, `<style:table-properties>` 17.15, `<style:table-row-properties>` 17.17, `<style:text-properties>` 16.29.29.

## 16.5 `<style:page-layout>`

The `<style:page-layout>` element represents the styles that specify the formatting properties of a page.

The `<style:page-layout>` element is usable within the following element:  
`<office:automatic-styles>` 3.15.3.

The `<style:page-layout>` element has the following attributes: `style:name` 19.502 and `style:page-usage` 19.509.

The `<style:page-layout>` element has the following child elements: `<style:footer-style>` 16.7, `<style:header-style>` 16.6 and `<style:page-layout-properties>` 17.2.

## 16.6 `<style:header-style>`

The `<style:header-style>` element specifies the formatting properties for a header element.

The `<style:header-style>` element is usable within the following elements:  
`<style:default-page-layout>` 16.8 and `<style:page-layout>` 16.5.

The `<style:header-style>` element has no attributes.

The `<style:header-style>` element has the following child element: `<style:header-footer-properties>` 17.5.

## 16.7 `<style:footer-style>`

The `<style:footer-style>` element specifies the formatting properties for a footer element.

The `<style:footer-style>` element is usable within the following elements: `<style:default-page-layout>` 16.8 and `<style:page-layout>` 16.5.

The `<style:footer-style>` element has no attributes.

The `<style:footer-style>` element has the following child element: `<style:header-footer-properties>` 17.5.

## 16.8 `<style:default-page-layout>`

The `<style:default-page-layout>` element specifies default formatting properties for page layouts. These defaults are used if a formatting property is not specified in a `<style:page-layout>` 16.5 element. The `<style:default-page-layout>` element is usable within the following element: `<office:styles>` 3.15.2.

The `<style:default-page-layout>` element is usable within the following element: `<office:styles>` 3.15.2.

The `<style:default-page-layout>` element has no attributes.

The `<style:default-page-layout>` element has the following child elements: `<style:footer-style>` 16.7, `<style:header-style>` 16.6 and `<style:page-layout-properties>` 17.2.

## 16.9 `<style:master-page>`

In text and spreadsheet documents, the `<style:master-page>` element contains the content of headers and footers. For these types of documents, consumers may generate a sequence of pages by making use of a single master page or a set of master pages.

In drawing and presentation documents, the `<style:master-page>` element is used to define master pages as common backgrounds for drawing pages. Each drawing page is directly linked to one master page, which is specified by the `draw:master-page-name` attribute of the `<draw:page>` element..

Master pages are contained in the `<office:master-styles>` element.

All documents shall contain at least one master page element.

If a text or spreadsheet document is displayed in a paged layout, master pages are used to generate a sequence of pages containing the document content. When a page is created, an empty page is generated with the properties of the master page and the static content of the master page. The body of the page is then filled with content. A single master page can be used to create multiple pages within a document.

In text and spreadsheet documents, a master page can be assigned to paragraph and table styles using a `style:master-page-name` attribute. Each time the paragraph or table style is applied to text, a page break is inserted before the paragraph or table. A page that starts at the page break position uses the specified master page.

**Note:** The OpenDocument paging methodology differs significantly from the methodology used in [XSL]. In XSL, headers and footers are contained within page sequences that also contain the document content. In the OpenDocument format, headers and footers are contained in page styles. With either approach, the content of headers and footers can be changed or omitted without affecting the document content.

The `<style:master-page>` element is usable within the following element: `<office:master-styles>` 3.15.4.

The `<style:master-page>` element has the following attributes: `draw:style-name` 19.219, `style:display-name` 19.476, `style:name` 19.502, `style:next-style-name` 19.503 and `style:page-layout-name` 19.508.

The `<style:master-page>` element has the following child elements: `<anim:animate>` 15.2.2, `<anim:animateColor>` 15.2.6, `<anim:animateMotion>` 15.2.5, `<anim:animateTransform>` 15.2.3, `<anim:audio>` 15.5, `<anim:command>` 15.6.1, `<anim:iterate>` 15.4.4, `<anim:par>` 15.4.2, `<anim:seq>` 15.4.3, `<anim:set>` 15.2.4, `<anim:transitionFilter>` 15.2.7, `<dr3d:scene>` 10.5.2, `<draw:a>` 10.4.12, `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:connector>` 10.3.10, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:g>` 10.3.15, `<draw:layer-set>` 10.2.2, `<draw:line>` 10.3.3, `<draw:measure>` 10.3.12, `<draw:page-thumbnail>` 10.3.14, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2, `<draw:regular-polygon>` 10.3.6, `<office:forms>` 13.2, `<presentation:notes>` 16.19, `<style:footer>` 16.12, `<style:footer-first>` 16.13, `<style:footer-left>` 16.15, `<style:header>` 16.10, `<style:header-first>` 16.11 and `<style:header-left>` 16.14.

## 16.10 `<style:header>`

The `<style:header>` element represents the content of a header in a `<style:master-page>` element.

The `<style:header>` element is usable within the following element: `<style:master-page>` 16.9.

The `<style:header>` element has the following attribute: `style:display` 19.475.

The `<style:header>` element has the following child elements: `<style:region-center>` 16.17, `<style:region-left>` 16.16, `<style:region-right>` 16.18, `<table:table>` 9.1.2, `<text:alphabetical-index>` 8.8, `<text:alphabetical-index-auto-mark-file>` 8.8.3, `<text:bibliography>` 8.9, `<text:change>` 5.5.8.4, `<text:change-end>` 5.5.8.3, `<text:change-start>` 5.5.8.2, `<text:dde-connection-decls>` 14.7.2, `<text:h>` 5.1.2, `<text:illustration-index>` 8.4, `<text:index-title>` 8.2.3, `<text:list>` 5.3.1, `<text:object-index>` 8.6, `<text:p>` 5.1.3, `<text:section>` 5.4, `<text:sequence-decls>` 7.4.11, `<text:table-index>` 8.5, `<text:table-of-content>` 8.3, `<text:tracked-changes>` 5.5.2, `<text:user-field-decls>` 7.4.7, `<text:user-index>` 8.7 and `<text:variable-decls>` 7.4.2.

## 16.11 `<style:header-first>`

The `<style:header-first>` element represents the content of a header for a first page, if different from the left/right page in a `<style:master-page>` element.

The term "first page" means the first page to which the page style is applied, regardless of any numbering. If a different page style is applied in between two applications of a page style that has the `<style:header-first>` element, each of the applications of the page style has its own first page.

The `<style:header-first>` element is usable within the following element: `<style:master-page>` 16.9.

The `<style:header-first>` element has the following attribute: `style:display` 19.475.

The `<style:header-first>` element has the following child elements: `<style:region-center>` 16.17, `<style:region-left>` 16.16, `<style:region-right>` 16.18, `<table:table>` 9.1.2, `<text:alphabetical-index>` 8.8, `<text:alphabetical-index-auto-mark-file>` 8.8.3, `<text:bibliography>` 8.9, `<text:change>` 5.5.8.4, `<text:change-end>` 5.5.8.3, `<text:change-start>` 5.5.8.2, `<text:dde-connection-decls>` 14.7.2, `<text:h>` 5.1.2, `<text:illustration-index>` 8.4, `<text:index-title>` 8.2.3, `<text:list>` 5.3.1, `<text:object-index>` 8.6,

```
<text:p> 5.1.3, <text:section> 5.4, <text:sequence-decls> 7.4.11,  
<text:table-index> 8.5, <text:table-of-content> 8.3, <text:tracked-  
changes> 5.5.2, <text:user-field-decls> 7.4.7, <text:user-index> 8.7 and  
<text:variable-decls> 7.4.2.
```

## 16.12 <style:footer>

The `<style:footer>` element represents the content of a footer in a `<style:master-page>` element.

The `<style:footer>` element is usable within the following element: `<style:master-page>` 16.9.

The `<style:footer>` element has the following attribute: `style:display` 19.475.

The `<style:footer>` element has the following child elements: `<style:region-center>` 16.17, `<style:region-left>` 16.16, `<style:region-right>` 16.18, `<table:table>` 9.1.2, `<text:alphabetical-index>` 8.8, `<text:alphabetical-index-auto-mark-file>` 8.8.3, `<text:bibliography>` 8.9, `<text:change>` 5.5.8.4, `<text:change-end>` 5.5.8.3, `<text:change-start>` 5.5.8.2, `<text:dde-connection-decls>` 14.7.2, `<text:h>` 5.1.2, `<text:illustration-index>` 8.4, `<text:index-title>` 8.2.3, `<text:list>` 5.3.1, `<text:object-index>` 8.6, `<text:p>` 5.1.3, `<text:section>` 5.4, `<text:sequence-decls>` 7.4.11, `<text:table-index>` 8.5, `<text:table-of-content>` 8.3, `<text:tracked-changes>` 5.5.2, `<text:user-field-decls>` 7.4.7, `<text:user-index>` 8.7 and `<text:variable-decls>` 7.4.2.

## 16.13 <style:footer-first>

The `<style:footer-first>` element represents the content of a footer for a first page, if different from the left/right page in a `<style:master-page>` element.

The term "first page" means the first page to which the page style is applied, regardless of any numbering. If a different page style is applied in between two applications of the page style, that has the `<style:footer-first>` element, each of the applications of the page style has its own first page.

The `<style:footer-first>` element is usable within the following element: `<style:master-page>` 16.9.

The `<style:footer-first>` element has the following attribute: `style:display` 19.475.

The `<style:footer-first>` element has the following child elements: `<style:region-center>` 16.17, `<style:region-left>` 16.16, `<style:region-right>` 16.18, `<table:table>` 9.1.2, `<text:alphabetical-index>` 8.8, `<text:alphabetical-index-auto-mark-file>` 8.8.3, `<text:bibliography>` 8.9, `<text:change>` 5.5.8.4, `<text:change-end>` 5.5.8.3, `<text:change-start>` 5.5.8.2, `<text:dde-connection-decls>` 14.7.2, `<text:h>` 5.1.2, `<text:illustration-index>` 8.4, `<text:index-title>` 8.2.3, `<text:list>` 5.3.1, `<text:object-index>` 8.6, `<text:p>` 5.1.3, `<text:section>` 5.4, `<text:sequence-decls>` 7.4.11, `<text:table-index>` 8.5, `<text:table-of-content>` 8.3, `<text:tracked-changes>` 5.5.2, `<text:user-field-decls>` 7.4.7, `<text:user-index>` 8.7 and `<text:variable-decls>` 7.4.2.

## 16.14 <style:header-left>

The `<style:header-left>` element represents the content of a header for a left page, if different from the right page in a `<style:master-page>` element.

The `<style:header-left>` element is usable within the following element:  
`<style:master-page> 16.9.`

The `<style:header-left>` element has the following attribute: `style:display 19.475.`

The `<style:header-left>` element has the following child elements: `<style:region-center> 16.17, <style:region-left> 16.16, <style:region-right> 16.18, <table:table> 9.1.2, <text:alphabetical-index> 8.8, <text:alphabetical-index-auto-mark-file> 8.8.3, <text:bibliography> 8.9, <text:change> 5.5.8.4, <text:change-end> 5.5.8.3, <text:change-start> 5.5.8.2, <text:dde-connection-decls> 14.7.2, <text:h> 5.1.2, <text:illustration-index> 8.4, <text:index-title> 8.2.3, <text:list> 5.3.1, <text:object-index> 8.6, <text:p> 5.1.3, <text:section> 5.4, <text:sequence-decls> 7.4.11, <text:table-index> 8.5, <text:table-of-content> 8.3, <text:tracked-changes> 5.5.2, <text:user-field-decls> 7.4.7, <text:user-index> 8.7 and <text:variable-decls> 7.4.2.`

## 16.15 `<style:footer-left>`

The `<style:footer-left>` element represents the content of a footer for a left page, if different from the right page for a `<style:master-page>` element.

The `<style:footer-left>` element is usable within the following element:  
`<style:master-page> 16.9.`

The `<style:footer-left>` element has the following attribute: `style:display 19.475.`

The `<style:footer-left>` element has the following child elements: `<style:region-center> 16.17, <style:region-left> 16.16, <style:region-right> 16.18, <table:table> 9.1.2, <text:alphabetical-index> 8.8, <text:alphabetical-index-auto-mark-file> 8.8.3, <text:bibliography> 8.9, <text:change> 5.5.8.4, <text:change-end> 5.5.8.3, <text:change-start> 5.5.8.2, <text:dde-connection-decls> 14.7.2, <text:h> 5.1.2, <text:illustration-index> 8.4, <text:index-title> 8.2.3, <text:list> 5.3.1, <text:object-index> 8.6, <text:p> 5.1.3, <text:section> 5.4, <text:sequence-decls> 7.4.11, <text:table-index> 8.5, <text:table-of-content> 8.3, <text:tracked-changes> 5.5.2, <text:user-field-decls> 7.4.7, <text:user-index> 8.7 and <text:variable-decls> 7.4.2.`

## 16.16 `<style:region-left>`

The `<style:region-left>` element specifies a left-aligned portion of a header or footer. That region may contain a sequence of `<text:p>` elements.

The `<style:region-left>` element is usable within the following elements:  
`<style:footer> 16.12, <style:footer-first> 16.13, <style:footer-left> 16.15, <style:header> 16.10, <style:header-first> 16.11 and <style:header-left> 16.14.`

The `<style:region-left>` element has no attributes.

The `<style:region-left>` element has the following child element: `<text:p> 5.1.3.`

## 16.17 `<style:region-center>`

The `<style:region-center>` element specifies a center-aligned portion of a header or footer. That region may contain a sequence of `<text:p>` elements.

The `<style:region-center>` element is usable within the following elements:  
`<style:footer> 16.12, <style:footer-first> 16.13, <style:footer-left>`

16.15, `<style:header>` 16.10, `<style:header-first>` 16.11 and `<style:header-left>` 16.14.

The `<style:region-center>` element has no attributes.

The `<style:region-center>` element has the following child element: `<text:p>` 5.1.3.

## 16.18 `<style:region-right>`

The `<style:region-right>` element specifies a right-aligned portion of a header or footer. That region may contain a sequence of `<text:p>` elements.

The `<style:region-right>` element is usable within the following elements:

`<style:footer>` 16.12, `<style:footer-first>` 16.13, `<style:footer-left>` 16.15, `<style:header>` 16.10, `<style:header-first>` 16.11 and `<style:header-left>` 16.14.

The `<style:region-right>` element has no attributes.

The `<style:region-right>` element has the following child element: `<text:p>` 5.1.3.

## 16.19 `<presentation:notes>`

The `<presentation:notes>` element defines a notes page. A notes page contains a preview of a drawing page and additional graphic shapes.

The `<presentation:notes>` element is usable within the following elements:

`<draw:page>` 10.2.4 and `<style:master-page>` 16.9.

The `<presentation:notes>` element has the following attributes: `draw:style-name` 19.219, `presentation:use-date-time-name` 19.428, `presentation:use-footer-name` 19.429, `presentation:use-header-name` 19.430 and `style:page-layout-name` 19.508.

The `<presentation:notes>` element has the following child elements: `<dr3d:scene>` 10.5.2, `<draw:a>` 10.4.12, `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:connector>` 10.3.10, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:g>` 10.3.15, `<draw:line>` 10.3.3, `<draw:measure>` 10.3.12, `<draw:page-thumbnail>` 10.3.14, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2, `<draw:regular-polygon>` 10.3.6 and `<office:forms>` 13.2.

## 16.20 `<table:table-template>`

A `<table:table-template>` element represents a set of references to table cell styles that specify the formatting to be used when creating a table.

If more than one table cell style reference is applicable to a cell, the first style that is applicable is chosen in the following order:

- First Column `<table:first-column>` 16.21.4
- Last Column `<table:last-column>` 16.21.5
- First Row `<table:first-row>` 16.21.2
- Last Row `<table:last-row>` 16.21.3
- Even rows `<table:even-rows>` 16.21.7, odd rows `<table:odd-rows>` 16.21.8 (Banded Rows)
- Even columns `<table:even-columns>` 16.21.9, odd columns `<table:odd-columns>` 16.21.10 (Banded Columns)

- Body `<table:body>` 16.21.6

The `<table:table-template>` element is usable within the following element:  
`<office:styles>` 3.15.2.

The `<table:table-template>` element has the following attributes: `table:first-row-end-column` 19.644, `table:first-row-start-column` 19.645, `table:last-row-end-column` 19.663, `table:last-row-start-column` 19.665 and `table:name` 19.677.

The `<table:table-template>` element has the following child elements:

`<table:background>` 16.22, `<table:body>` 16.21.6, `<table:even-columns>` 16.21.9, `<table:even-rows>` 16.21.7, `<table:first-column>` 16.21.4, `<table:first-row>` 16.21.2, `<table:last-column>` 16.21.5, `<table:last-row>` 16.21.3, `<table:odd-columns>` 16.21.10 and `<table:odd-rows>` 16.21.8.

## 16.21 Cell Styles

### 16.21.1 General

The cell styles applied in a table are based on the position of cells in a table.

### 16.21.2 `<table:first-row>`

The `<table:first-row>` element specifies a cell style that shall be applied to the first row of a table.

The `<table:first-row>` element is usable within the following element: `<table:table-template>` 16.20.

The `<table:first-row>` element has the following attributes: `table:paragraph-style-name` 19.692 and `table:style-name` 19.730.

The `<table:first-row>` element has no child elements.

### 16.21.3 `<table:last-row>`

The `<table:last-row>` element specifies a cell style that shall be applied to the last row of a table.

The `<table:last-row>` element is usable within the following element: `<table:table-template>` 16.20.

The `<table:last-row>` element has the following attributes: `table:paragraph-style-name` 19.692 and `table:style-name` 19.730.

The `<table:last-row>` element has no child elements.

### 16.21.4 `<table:first-column>`

The `<table:first-column>` element specifies a cell style that shall be applied to the first column in a table.

The `<table:first-column>` element is usable within the following element:  
`<table:table-template>` 16.20.

The `<table:first-column>` element has the following attributes: `table:paragraph-style-name` 19.692 and `table:style-name` 19.730.

The `<table:first-column>` element has no child elements.

### 16.21.5 <table:last-column>

The `<table:last-column>` element specifies a cell style that shall be applied to the last column in a table.

The `<table:last-column>` element is usable within the following element:

`<table:table-template> 16.20.`

The `<table:last-column>` element has the following attributes: `table:paragraph-style-name 19.692` and `table:style-name 19.730`.

The `<table:last-column>` element has no child elements.

### 16.21.6 <table:body>

The `<table:body>` element specifies styles for cells that are not governed by other specified styles.

The `<table:body>` element is usable within the following element: `<table:table-template> 16.20.`

The `<table:body>` element has the following attributes: `table:paragraph-style-name 19.692` and `table:style-name 19.730`.

The `<table:body>` element has no child elements.

### 16.21.7 <table:even-rows>

The `<table:even-rows>` element specifies styles to be applied to even rows in a table.

The `<table:even-rows>` element is usable within the following element: `<table:table-template> 16.20.`

The `<table:even-rows>` element has the following attributes: `table:paragraph-style-name 19.692` and `table:style-name 19.730`.

The `<table:even-rows>` element has no child elements.

### 16.21.8 <table:odd-rows>

The `<table:odd-rows>` element specifies styles to be applied to odd rows in a table.

The `<table:odd-rows>` element is usable within the following element: `<table:table-template> 16.20.`

The `<table:odd-rows>` element has the following attributes: `table:paragraph-style-name 19.692` and `table:style-name 19.730`.

The `<table:odd-rows>` element has no child elements.

### 16.21.9 <table:even-columns>

The `<table:even-columns>` element specifies styles to be applied to even columns in a table.

The `<table:even-columns>` element is usable within the following element: `<table:table-template> 16.20.`

The `<table:even-columns>` element has the following attributes: `table:paragraph-style-name 19.692` and `table:style-name 19.730`.

The `<table:even-columns>` element has no child elements.

### 16.21.10 <table:odd-columns>

The `<table:odd-columns>` element specifies styles to be applied to odd columns in a table.

The `<table:odd-columns>` element is usable within the following element:

`<table:table-template>` 16.20.

The `<table:odd-columns>` element has the following attributes: `table:paragraph-style-name` 19.692 and `table:style-name` 19.730.

The `<table:odd-columns>` element has no child elements.

### 16.22 <table:background>

The `<table:background>` element specifies a table style that provides a background to a table that is visible if all or part of the table is transparent.

The `<table:background>` element is usable within the following element:

`<table:table-template>` 16.20.

The `<table:background>` element has the following attribute: `table:style-name` 19.730.

The `<table:background>` element has no child elements.

### 16.23 <style:font-face>

The `<style:font-face>` element represents a font face declaration which documents the properties of a font used in a document.

OpenDocument font face declarations directly correspond to the `@font-face` font description of [CSS2] (see §15.3.1) and the `<font-face>` element of [SVG] (see §20.8.3).

OpenDocument font face declarations may have an unique name. This name can be used inside styles (as an attribute of `<style:text-properties>` element) as value of the `style:font-name` attribute to select a font face declaration. If a font face declaration is referenced by name, the font-matching algorithms for selecting a font declaration based on the font-family, font-style, font-variant, font-weight and font-size descriptors are not used but the referenced font face declaration is used directly. (See §15.5 [CSS2])

Consumers should implement the CSS2 font-matching algorithm with the OpenDocument font face extensions. They may implement variations of the CSS2 font-matching algorithm. They may implement a font-matching based only on the font face declarations, that is, a font-matching that is not applied to every character independently but only once for each font face declaration. (See §15.5 [CSS2])

Font face declarations support the font descriptor attributes and elements described in §20.8.3 of [SVG].

The `<style:font-face>` element is usable within the following element: `<office:font-face-decls>` 3.14.

The `<style:font-face>` element has the following attributes: `style:font-adornments` 19.482, `style:font-charset` 19.483, `style:font-family-generic` 19.484, `style:font-pitch` 19.485, `style:name` 19.502, `svg:accent-height` 19.523, `svg:alphabetic` 19.524, `svg:ascent` 19.525, `svg:bbox` 19.526, `svg:cap-height` 19.527, `svg:descent` 19.531, `svg:font-family` 19.532, `svg:font-size` 19.533, `svg:font-stretch` 19.534, `svg:font-style` 19.535, `svg:font-variant` 19.536, `svg:font-weight` 19.537, `svg:hanging` 19.542, `svg:ideographic` 19.544, `svg:mathematical` 19.545, `svg:overline-position` 19.549, `svg:overline-thickness` 19.550, `svg:panose-1` 19.551, `svg:slope` 19.556, `svg:stemh` 19.558, `svg:stemv` 19.559, `svg:strikethrough-position` 19.562, `svg:strikethrough-`

thickness 19.563, `svg:underline-position` 19.566, `svg:underline-thickness` 19.567, `svg:unicode-range` 19.568, `svg:units-per-em` 19.569, `svg:v-alphabetic` 19.570, `svg:v-hanging` 19.571, `svg:v-ideographic` 19.572, `svg:v-mathematical` 19.573, `svg:widths` 19.576 and `svg:x-height` 19.580.

The `<style:font-face>` element has the following child elements: `<svg:definition-src>` 16.27 and `<svg:font-face-src>` 16.24.

## 16.24 `<svg:font-face-src>`

See §20.8.3 of [SVG].

The `<svg:font-face-src>` element is usable within the following element: `<style:font-face>` 16.23.

The `<svg:font-face-src>` element has no attributes.

The `<svg:font-face-src>` element has the following child elements: `<svg:font-face-name>` 16.25 and `<svg:font-face-uri>` 16.26.

## 16.25 `<svg:font-face-name>`

See §20.8.3 of [SVG].

The `<svg:font-face-name>` element is usable within the following element: `<svg:font-face-src>` 16.24.

The `<svg:font-face-name>` element has the following attribute: `svg:name` 19.546.

The `<svg:font-face-name>` element has no child elements.

## 16.26 `<svg:font-face-uri>`

See § 20.8.3 of [SVG].

The `<svg:font-face-uri>` element is usable within the following element: `<svg:font-face-src>` 16.24.

The `<svg:font-face-uri>` element has the following attributes: `xlink:actuate` 19.915, `xlink:href` 19.916 and `xlink:type` 19.919.

The `<svg:font-face-uri>` element has the following child element: `<svg:font-face-format>` 16.28.

## 16.27 `<svg:definition-src>`

See § 20.8.3 of [SVG].

The `<svg:definition-src>` element is usable within the following element: `<style:font-face>` 16.23.

The `<svg:definition-src>` element has the following attributes: `xlink:actuate` 19.915, `xlink:href` 19.916 and `xlink:type` 19.919.

The `<svg:definition-src>` element has no child elements.

## 16.28 `<svg:font-face-format>`

See §20.8.3 of [SVG].

The `<svg:font-face-format>` element is usable within the following element: `<svg:font-face-uri>` 16.26.

The `<svg:font-face-format>` element has the following attribute: `svg:string` 19.564.

The `<svg:font-face-format>` element has no child elements.

## 16.29 Data Styles

### 16.29.1 General

Data styles specify the styles to be applied to defined data types for display.

Data styles shall not specify two `<number:text>` elements in sequence as part of a data style.

Data styles shall specify at most one `<number:fill-character>` element.

### 16.29.2 `<number:number-style>`

The `<number:number-style>` element is a container for elements that define a style for decimal numbers.

**Note:** Tab alignment should be supported in spreadsheets for interoperability (display) purposes.

The `<number:number-style>` element is usable within the following elements: `<office:automatic-styles>` 3.15.3 and `<office:styles>` 3.15.2.

The `<number:number-style>` element has the following attributes: `number:country` 19.342, `number:language` 19.351, `number:rfc-language-tag` 19.360, `number:script` 19.361, `number:title` 19.364, `number:transliteration-country` 19.365, `number:transliteration-format` 19.366, `number:transliteration-language` 19.367, `number:transliteration-style` 19.368, `style:display-name` 19.476, `style:name` 19.502 and `style:volatile` 19.521.

The `<number:number-style>` element has the following child elements: `<number:fill-character>` 16.29.5, `<number:fraction>` 16.29.7, `<number:number>` 16.29.3, `<number:scientific-number>` 16.29.6, `<number:text>` 16.29.27, `<style:map>` 16.3 and `<style:text-properties>` 16.29.29.

### 16.29.3 `<number:number>`

The `<number:number>` element specifies the display formatting properties for a decimal number.

The `<number:number>` element is usable within the following elements: `<number:currency-style>` 16.29.8, `<number:number-style>` 16.29.2 and `<number:percentage-style>` 16.29.10.

The `<number:number>` element has the following attributes: `number:decimal-places` 19.343, `number:decimal-replacement` 19.344, `number:display-factor` 19.346, `number:grouping` 19.350, `number:min-decimal-places` 19.356 and `number:min-integer-digits` 19.355.

The `<number:number>` element has the following child element: `<number:embedded-text>` 16.29.4.

### 16.29.4 `<number:embedded-text>`

The `<number:embedded-text>` element specifies text that is displayed at one specific position within a number.

The `<number:embedded-text>` element is usable within the following element:  
`<number:number>` 16.29.3.

The `<number:embedded-text>` element has the following attribute: `number:position 19.358`.

The `<number:embedded-text>` element has no child elements.

The `<number:embedded-text>` element has character data content.

### 16.29.5 `<number:fill-character>`

The `<number:fill-character>` element specifies a Unicode character that is displayed repeatedly at the position where the element occurs. The character specified is repeated as many times as possible, but the total resulting string shall not exceed the given cell content area.

If fill characters do not fill all the available space in a cell, the distribution of the remaining space is implementation-dependent.

The `<number:fill-character>` element is usable within the following elements:  
`<number:currency-style>` 16.29.8, `<number:date-style>` 16.29.11,  
`<number:number-style>` 16.29.2, `<number:percentage-style>` 16.29.10,  
`<number:text-style>` 16.29.26 and `<number:time-style>` 16.29.19.

The `<number:fill-character>` element has no attributes.

The `<number:fill-character>` element has no child elements.

The `<number:fill-character>` element has character data content.

### 16.29.6 `<number:scientific-number>`

The `<number:scientific-number>` element specifies the display formatting properties for a number style that should be displayed in scientific format.

The `<number:scientific-number>` element is usable within the following element:  
`<number:number-style>` 16.29.2.

The `<number:scientific-number>` element has the following attributes:  
`number:decimal-places 19.343`, `number:exponent-interval 19.347`,  
`number:forced-exponent-sign 19.349`, `number:grouping 19.350`, `number:min-decimal-places 19.356`, `number:min-exponent-digits 19.354` and `number:min-integer-digits 19.355`.

The `<number:scientific-number>` element has no child elements.

### 16.29.7 `<number:fraction>`

The `<number:fraction>` element specifies the display formatting properties for a number style that should be displayed as a fraction.

The `<number:fraction>` element is usable within the following element:  
`<number:number-style>` 16.29.2.

The `<number:fraction>` element has the following attributes: `number:denominator-value 19.345`, `number:grouping 19.350`, `number:max-denominator-value 19.352`,  
`number:min-denominator-digits 19.353`, `number:min-integer-digits 19.355` and `number:min-numerator-digits 19.357`.

The `<number:fraction>` element has no child elements.

## 16.29.8 <number:currency-style>

The `<number:currency-style>` element specifies the style for currency values.

The `<number:currency-style>` element is usable within the following elements: `<office:automatic-styles>` 3.15.3 and `<office:styles>` 3.15.2.

The `<number:currency-style>` element has the following attributes: `number:automatic-order` 19.340, `number:country` 19.342, `number:language` 19.351, `number:rfc-language-tag` 19.360, `number:script` 19.361, `number:title` 19.364, `number:transliteration-country` 19.365, `number:transliteration-format` 19.366, `number:transliteration-language` 19.367, `number:transliteration-style` 19.368, `style:display-name` 19.476, `style:name` 19.502 and `style:volatile` 19.521.

The `<number:currency-style>` element has the following child elements: `<number:currency-symbol>` 16.29.9, `<number:fill-character>` 16.29.5, `<number:number>` 16.29.3, `<number:text>` 16.29.27, `<style:map>` 16.3 and `<style:text-properties>` 16.29.29.

## 16.29.9 <number:currency-symbol>

The `<number:currency-symbol>` element specifies whether a currency symbol is displayed in a currency style.

The content of this element is the text that is displayed as the currency symbol. If the element is empty or contains white space characters only, the default currency symbol for the currency style or the language and country of the currency style is displayed.

The `<number:currency-symbol>` element is usable within the following element: `<number:currency-style>` 16.29.8.

The `<number:currency-symbol>` element has the following attributes: `number:country` 19.342, `number:language` 19.351, `number:rfc-language-tag` 19.360 and `number:script` 19.361.

The `<number:currency-symbol>` element has no child elements.

The `<number:currency-symbol>` element has character data content.

## 16.29.10 <number:percentage-style>

The `<number:percentage-style>` element specifies the style for percentage values.

A `<number:percentage-style>` element should have `<number:text>` child element whose character data contains a "%" (U+0025, PERCENT SIGN) character.

The `<number:percentage-style>` element is usable within the following elements: `<office:automatic-styles>` 3.15.3 and `<office:styles>` 3.15.2.

The `<number:percentage-style>` element has the following attributes: `number:country` 19.342, `number:language` 19.351, `number:rfc-language-tag` 19.360, `number:script` 19.361, `number:title` 19.364, `number:transliteration-country` 19.365, `number:transliteration-format` 19.366, `number:transliteration-language` 19.367, `number:transliteration-style` 19.368, `style:display-name` 19.476, `style:name` 19.502 and `style:volatile` 19.521.

The `<number:percentage-style>` element has the following child elements: `<number:fill-character>` 16.29.5, `<number:number>` 16.29.3, `<number:text>` 16.29.27, `<style:map>` 16.3 and `<style:text-properties>` 16.29.29.

### 16.29.11 <number:date-style>

The `<number:date-style>` element specifies a style for date values.

The `<number:date-style>` element is usable within the following elements: `<office:automatic-styles>` 3.15.3 and `<office:styles>` 3.15.2.

The `<number:date-style>` element has the following attributes: `number:automatic-order` 19.340, `number:country` 19.342, `number:format-source` 19.348, `number:language` 19.351, `number:rfc-language-tag` 19.360, `number:script` 19.361, `number:title` 19.364, `number:transliteration-country` 19.365, `number:transliteration-format` 19.366, `number:transliteration-language` 19.367, `number:transliteration-style` 19.368, `style:display-name` 19.476, `style:name` 19.502 and `style:volatile` 19.521.

The `<number:date-style>` element has the following child elements: `<number:am-pm>` 16.29.23, `<number:day>` 16.29.12, `<number:day-of-week>` 16.29.16, `<number:era>` 16.29.15, `<number:fill-character>` 16.29.5, `<number:hours>` 16.29.20, `<number:minutes>` 16.29.21, `<number:month>` 16.29.13, `<number:quarter>` 16.29.18, `<number:seconds>` 16.29.22, `<number:text>` 16.29.27, `<number:week-of-year>` 16.29.17, `<number:year>` 16.29.14, `<style:map>` 16.3 and `<style:text-properties>` 16.29.29.

### 16.29.12 <number:day>

The `<number:day>` element specifies a day of a month in a date.

The `<number:day>` element is usable within the following element: `<number:date-style>` 16.29.11.

The `<number:day>` element has the following attributes: `number:calendar` 19.341 and `number:style` 19.362.

The `<number:day>` element has no child elements.

### 16.29.13 <number:month>

The `<number:month>` element specifies a month in a date.

The `<number:month>` element is usable within the following element: `<number:date-style>` 16.29.11.

The `<number:month>` element has the following attributes: `number:calendar` 19.341, `number:possessive-form` 19.359, `number:style` 19.362 and `number:textual` 19.363.

The `<number:month>` element has no child elements.

### 16.29.14 <number:year>

The `<number:year>` element specifies a year in a date.

The `<number:year>` element is usable within the following element: `<number:date-style>` 16.29.11.

The `<number:year>` element has the following attributes: `number:calendar` 19.341 and `number:style` 19.362.

The `<number:year>` element has no child elements.

### 16.29.15 <number:era>

The <number:era> element specifies an era in which a year is counted.

The <number:era> element is usable within the following element: <number:date-style> 16.29.11.

The <number:era> element has the following attributes: number:calendar 19.341 and number:style 19.362.

The <number:era> element has no child elements.

### 16.29.16 <number:day-of-week>

The <number:day-of-week> element specifies a day of a week in a date.

The <number:day-of-week> element is usable within the following element: <number:date-style> 16.29.11.

The <number:day-of-week> element has the following attributes: number:calendar 19.341 and number:style 19.362.

The <number:day-of-week> element has no child elements.

### 16.29.17 <number:week-of-year>

The <number:week-of-year> element specifies a week of a year in a date.

The <number:week-of-year> element is usable within the following element: <number:date-style> 16.29.11.

The <number:week-of-year> element has the following attribute: number:calendar 19.341.

The <number:week-of-year> element has no child elements.

### 16.29.18 <number:quarter>

The <number:quarter> element specifies a quarter of a year in a date.

The <number:quarter> element is usable within the following element: <number:date-style> 16.29.11.

The <number:quarter> element has the following attributes: number:calendar 19.341 and number:style 19.362.

The <number:quarter> element has no child elements.

### 16.29.19 <number:time-style>

The <number:time-style> element specifies a style for time values.

This element can contain *one* instance of any of the following elements: <number:hours>, <number:minutes>, <number:seconds> and <number:am-pm>.

The <number:time-style> element can also contain <number:text> elements, which display additional text.

The <number:time-style> element is usable within the following elements: <office:automatic-styles> 3.15.3 and <office:styles> 3.15.2.

The <number:time-style> element has the following attributes: number:country 19.342, number:format-source 19.348, number:language 19.351, number:rfc-language-tag 19.360, number:script 19.361, number:title 19.364,

number:transliteration-country 19.365, number:transliteration-format 19.366, number:transliteration-language 19.367, number:transliteration-style 19.368, number:truncate-on-overflow 19.369, style:display-name 19.476, style:name 19.502 and style:volatile 19.521.

The `<number:time-style>` element has the following child elements: `<number:am-pm>` 16.29.23, `<number:fill-character>` 16.29.5, `<number:hours>` 16.29.20, `<number:minutes>` 16.29.21, `<number:seconds>` 16.29.22, `<number:text>` 16.29.27, `<style:map>` 16.3 and `<style:text-properties>` 16.29.29.

## 16.29.20 `<number:hours>`

The `<number:hours>` element specifies whether hours are displayed as part of a date or time.

The `<number:hours>` element is usable within the following elements: `<number:date-style>` 16.29.11 and `<number:time-style>` 16.29.19.

The `<number:hours>` element has the following attribute: `number:style` 19.362.

The `<number:hours>` element has no child elements.

## 16.29.21 `<number:minutes>`

The `<number:minutes>` element specifies whether minutes are displayed as part of a date or time.

The `<number:minutes>` element is usable within the following elements: `<number:date-style>` 16.29.11 and `<number:time-style>` 16.29.19.

The `<number:minutes>` element has the following attribute: `number:style` 19.362.

The `<number:minutes>` element has no child elements.

## 16.29.22 `<number:seconds>`

The `<number:seconds>` element specifies whether seconds are displayed as part of a date or time.

The `<number:seconds>` element is usable within the following elements: `<number:date-style>` 16.29.11 and `<number:time-style>` 16.29.19.

The `<number:seconds>` element has the following attributes: `number:decimal-places` 19.343 and `number:style` 19.362.

The `<number:seconds>` element has no child elements.

## 16.29.23 `<number:am-pm>`

The `<number:am-pm>` element specifies whether AM/PM is included as part of a date or time.

If a `<number:am-pm>` element is contained in a date or time style, hours are displayed using values from 1 to 12 only.

The `<number:am-pm>` element is usable within the following elements: `<number:date-style>` 16.29.11 and `<number:time-style>` 16.29.19.

The `<number:am-pm>` element has no attributes.

The `<number:am-pm>` element has no child elements.

## 16.29.24 <number:boolean-style>

The `<number:boolean-style>` element defines a style for Boolean values.

The `<number:boolean-style>` element is usable within the following elements: `<office:automatic-styles>` 3.15.3 and `<office:styles>` 3.15.2.

The `<number:boolean-style>` element has the following attributes: `number:country` 19.342, `number:language` 19.351, `number:rfc-language-tag` 19.360, `number:script` 19.361, `number:title` 19.364, `number:transliteration-country` 19.365, `number:transliteration-format` 19.366, `number:transliteration-language` 19.367, `number:transliteration-style` 19.368, `style:display-name` 19.476, `style:name` 19.502 and `style:volatile` 19.521.

The `<number:boolean-style>` element has the following child elements: `<number:boolean>` 16.29.25, `<number:text>` 16.29.27, `<style:map>` 16.3 and `<style:text-properties>` 16.29.29.

## 16.29.25 <number:boolean>

The `<number:boolean>` element marks the position of the Boolean value of a Boolean style.

The `<number:boolean>` element is usable within the following element: `<number:boolean-style>` 16.29.24.

The `<number:boolean>` element has no attributes.

The `<number:boolean>` element has no child elements.

## 16.29.26 <number:text-style>

The `<number:text-style>` element defines a style for displaying text.

It can also contain `<number:text>` elements, which display additional text.

The `<number:text-style>` element is usable within the following elements: `<office:automatic-styles>` 3.15.3 and `<office:styles>` 3.15.2.

The `<number:text-style>` element has the following attributes: `number:country` 19.342, `number:language` 19.351, `number:rfc-language-tag` 19.360, `number:script` 19.361, `number:title` 19.364, `number:transliteration-country` 19.365, `number:transliteration-format` 19.366, `number:transliteration-language` 19.367, `number:transliteration-style` 19.368, `style:display-name` 19.476, `style:name` 19.502 and `style:volatile` 19.521.

The `<number:text-style>` element has the following child elements: `<number:fill-character>` 16.29.5, `<number:text>` 16.29.27, `<number:text-content>` 16.29.28, `<style:map>` 16.3 and `<style:text-properties>` 16.29.29.

## 16.29.27 <number:text>

The `<number:text>` element contains any fixed text for a data style.

The `<number:text>` element is usable within the following elements: `<number:boolean-style>` 16.29.24, `<number:currency-style>` 16.29.8, `<number:date-style>` 16.29.11, `<number:num-list-format>` 16.29.30, `<number:number-style>` 16.29.2, `<number:percentage-style>` 16.29.10, `<number:text-style>` 16.29.26 and `<number:time-style>` 16.29.19.

The `<number:text>` element has no attributes.

The `<number:text>` element has no child elements.

The `<number:text>` element has character data content.

### 16.29.28 `<number:text-content>`

The `<number:text-content>` element marks the position of variable text content of a text style.

The `<number:text-content>` element is usable within the following element:

`<number:text-style>` 16.29.26.

The `<number:text-content>` element has no attributes.

The `<number:text-content>` element has no child elements.

### 16.29.29 `<style:text-properties>`

The `<style:text-properties>` element specifies formatting properties for text.

The `<style:text-properties>` element is usable within the following elements:

`<number:boolean-style>` 16.29.24, `<number:currency-style>` 16.29.8, `<number:date-style>` 16.29.11, `<number:number-style>` 16.29.2, `<number:percentage-style>` 16.29.10, `<number:text-style>` 16.29.26, `<number:time-style>` 16.29.19, `<style:default-style>` 16.4, `<style:style>` 16.2, `<text:list-level-style-bullet>` 16.33, `<text:list-level-style-number>` 16.34 and `<text:outline-level-style>` 16.37.

The `<style:text-properties>` element has the following attributes: `fo:background-color` 20.182, `fo:color` 20.187, `fo:country` 20.188, `fo:font-family` 20.189, `fo:font-size` 20.190, `fo:font-style` 20.191, `fo:font-variant` 20.192, `fo:font-weight` 20.193, `fo:hyphenate` 20.195, `fo:hyphenation-push-char-count` 20.198, `fo:hyphenation-remain-char-count` 20.199, `fo:language` 20.202, `fo:letter-spacing` 20.203, `fo:script` 20.222, `fo:text-shadow` 20.226, `fo:text-transform` 20.227, `style:country-asian` 20.256, `style:country-complex` 20.257, `style:font-charset` 20.268, `style:font-charset-asian` 20.269, `style:font-charset-complex` 20.270, `style:font-family-asian` 20.271, `style:font-family-complex` 20.272, `style:font-family-generic` 20.273, `style:font-family-generic-asian` 20.274, `style:font-family-generic-complex` 20.275, `style:font-name` 20.277, `style:font-name-asian` 20.278, `style:font-name-complex` 20.279, `style:font-pitch` 20.280, `style:font-pitch-asian` 20.281, `style:font-pitch-complex` 20.282, `style:font-relief` 20.283, `style:font-size-asian` 20.284, `style:font-size-complex` 20.285, `style:font-size-rel` 20.286, `style:font-size-rel-asian` 20.287, `style:font-size-rel-complex` 20.288, `style:font-style-asian` 20.289, `style:font-style-complex` 20.290, `style:font-style-name` 20.291, `style:font-style-name-asian` 20.292, `style:font-style-name-complex` 20.293, `style:font-weight-asian` 20.294, `style:font-weight-complex` 20.295, `style:language-asian` 20.302, `style:language-complex` 20.303, `style:letter-kerning` 20.316, `style:rfc-language-tag` 20.343, `style:rfc-language-tag-asian` 20.344, `style:rfc-language-tag-complex` 20.345, `style:script-asian` 20.356, `style:script-complex` 20.357, `style:script-type` 20.358, `style:text-blinking` 20.366, `style:text-combine` 20.367, `style:text-combine-end-char` 20.369, `style:text-combine-start-char` 20.368, `style:text-emphasize` 20.370, `style:text-line-through-color` 20.371, `style:text-line-through-mode` 20.372, `style:text-line-through-style` 20.373, `style:text-line-through-text` 20.374, `style:text-line-through-text-style` 20.375, `style:text-line-through-type` 20.376, `style:text-line-through-width` 20.377, `style:text-outline` 20.378, `style:text-overline-color` 20.379, `style:text-overline-`

```
mode 20.380, style:text-overline-style 20.381, style:text-overline-type 20.382, style:text-overline-width 20.383, style:text-position 20.384, style:text-rotation-angle 20.385, style:text-rotation-scale 20.386, style:text-scale 20.387, style:text-underline-color 20.388, style:text-underline-mode 20.389, style:text-underline-style 20.390, style:text-underline-type 20.391, style:text-underline-width 20.392, style:use-window-font-color 20.395, text:condition 20.426, text:display 20.427.
```

The `<style:text-properties>` element has no child elements.

### 16.29.30 `<number:num-list-format>`

The `<number:num-list-format>` element defines a format for the displayed number string of a list level.

The `<number:num-list-format>` element is usable within the following elements: `<office:automatic-styles>` 3.15.3 and `<office:styles>` 3.15.2.

The `<number:num-list-format>` element has the following attribute: `style:name 19.502`.

The `<number:num-list-format>` element has the following child elements: `<number:num-list-label>` 16.29.31 and `<number:text>` 16.29.27.

### 16.29.31 `<number:num-list-label>`

For display, at every list item these placeholders are replaced by the item's list number corresponding to the list level specified by the `text:level 19.834` attribute, formatted according to the `style:num-format 19.504` attribute of the placeholder's list level.

Any placeholder specifying a list level higher than the list level containing the `style:num-list-format-name 19.922` attribute shall be ignored.

The `<number:num-list-label>` element is usable within the following element: `<number:num-list-format>` 16.29.30.

The `<number:num-list-label>` element has the following attribute: `text:level 19.834`.

The `<number:num-list-label>` element has no child elements.

## 16.30 Text Content Style Families

### 16.30.1 Text Styles

Text styles are represented by `<style:style>` elements whose `style:family` attribute has the value `text`.

### 16.30.2 Paragraph Styles

Paragraph styles are represented by `<style:style>` elements whose `style:family` attribute has the value `paragraph`.

In addition to paragraph properties, paragraph styles may define text properties. These are applied to the character content of the paragraph unless they are overwritten by a text style that is specified by any of the descendant elements of the paragraph element.

### 16.30.3 Section Styles

Section styles are represented by `<style:style>` elements whose `style:family` attribute has the value `section`.

#### 16.30.4 Ruby Styles

Ruby styles are represented by `<style:style>` elements whose `style:family` attribute has the value `ruby`.

### 16.31 Enhanced Text Styles

#### 16.31.1 `<text:linenumbering-configuration>`

The `<text:linenumbering-configuration>` element specifies line numbering within the `<office:styles>` element.

The `<text:linenumbering-configuration>` element is usable within the following element: `<office:styles> 3.15.2`.

The `<text:linenumbering-configuration>` element has the following attributes: `style:num-format 19.504`, `style:num-letter-sync 19.505`, `text:count-empty-lines 19.788`, `text:count-in-text-boxes 19.789`, `text:increment 19.814`, `text:number-lines 19.845`, `text:number-position 19.846`, `text:offset 19.848`, `text:restart-on-page 19.864` and `text:style-name 19.880`.

The `<text:linenumbering-configuration>` element has the following child element: `<text:linenumbering-separator> 16.31.2`.

#### 16.31.2 `<text:linenumbering-separator>`

The `<text:linenumbering-separator>` element contains the text that is displayed as a separator. A separator is text that is displayed instead of a line number for lines where no number is displayed.

The `<text:linenumbering-separator>` element is usable within the following element: `<text:linenumbering-configuration> 16.31.1`.

The `<text:linenumbering-separator>` element has the following attribute: `text:increment 19.814`.

The `<text:linenumbering-separator>` element has no child elements.

The `<text:linenumbering-separator>` element has character data content.

#### 16.31.3 `<text:notes-configuration>`

A `<text:notes-configuration>` element specifies values for each note class used in a document. If there is no note configuration element, an implementation-defined default note configuration is used.

The `<text:notes-configuration>` element is usable within the following elements: `<office:styles> 3.15.2` and `<style:section-properties> 17.11`.

The `<text:notes-configuration>` element has the following attributes: `style:num-format 19.504`, `style:num-letter-sync 19.505`, `style:num-prefix 19.506`, `style:num-suffix 19.507`, `text:citation-body-style-name 19.773`, `text:citation-style-name 19.774`, `text:default-style-name 19.800`, `text:footnotes-position 19.810`, `text:master-page-name 19.839`, `text:note-class 19.843`, `text:start-numbering-at 19.875` and `text:start-value 19.874`.

The `<text:notes-configuration>` element has the following child elements: `<text:note-continuation-notice-backward> 16.31.5` and `<text:note-continuation-notice-forward> 16.31.4`.

#### 16.31.4 <text:note-continuation-notice-forward>

The <text:note-continuation-notice-forward> elements specifies the text that is displayed at the end of a footnote that is continued on the next page

The <text:note-continuation-notice-forward> element is usable within the following element: <text:notes-configuration> 16.31.3.

The <text:note-continuation-notice-forward> element has no attributes.

The <text:note-continuation-notice-forward> element has no child elements.

The <text:note-continuation-notice-forward> element has character data content.

#### 16.31.5 <text:note-continuation-notice-backward>

The <text:note-continuation-notice-backward> elements specifies the text that is displayed before continued text.

The <text:note-continuation-notice-backward> element is usable within the following element: <text:notes-configuration> 16.31.3.

The <text:note-continuation-notice-backward> element has no attributes.

The <text:note-continuation-notice-backward> element has no child elements.

The <text:note-continuation-notice-backward> element has character data content.

#### 16.31.6 <text:bibliography-configuration>

The <text:bibliography-configuration> element specifies how bibliography entries are displayed in-line, and how they are displayed in a bibliography index.

The <text:bibliography-configuration> element is usable within the following element: <office:styles> 3.15.2.

The <text:bibliography-configuration> element has the following attributes: fo:country 19.234, fo:language 19.236, fo:script 19.242, style:rfc-language-tag 19.516, text:numbered-entries 19.847, text:prefix 19.854, text:sort-algorithm 19.871, text:sort-by-position 19.873 and text:suffix 19.882.

The <text:bibliography-configuration> element has the following child element: <text:sort-key> 16.31.7.

#### 16.31.7 <text:sort-key>

The <text:sort-key> element specifies a single sort key if bibliography entries are not displayed in document order.

The <text:sort-key> element is usable within the following element: <text:bibliography-configuration> 16.31.6.

The <text:sort-key> element has the following attributes: text:key 19.826 and text:sort-ascending 19.872.

The <text:sort-key> element has no child elements.

### 16.32 <text:list-style>

The <text:list-style> element contains a set of style elements for each list level, which are called *list level styles*. There are three different list level style elements, depending on whether a list level is to have a list label containing the list numbering, a bullet, or an image.

If a list style is applied to a list but does not contain a list level specification for a specific level, the list level style of the next lower level is used. If no specific list level style has been defined, an implementation-dependent default style is used.

Including a list style element in a graphic style has the same semantics as adding a `style:list-style-name` attribute ( 19.500) to the style that references a list style that is declared outside a graphic style.

**Note:** The inclusion of a list style element is needed in cases where a common graphic style should be associated with an automatic list style.

List styles contained in graphic styles can be referenced by other graphic styles using the `style:list-style-name` attribute. Their names are not displayed in the user interface, even if the graphic style that contains it is a common style.

The `<text:list-style>` element is usable within the following elements:  
`<office:automatic-styles>` 3.15.3, `<office:styles>` 3.15.2 and  
`<style:graphic-properties>` 17.21.

The `<text:list-style>` element has the following attributes: `style:display-name` 19.476, `style:name` 19.502 and `text:consecutive-numbering` 19.784.

The `<text:list-style>` element has the following child elements: `<text:list-level-style-bullet>` 16.33, `<text:list-level-style-image>` 16.35 and `<text:list-level-style-number>` 16.34.

### 16.33 `<text:list-level-style-bullet>`

A `<text:list-level-style-bullet>` element specifies a list style where list items are preceded by bullets.

The `<text:list-level-style-bullet>` element is usable within the following element:  
`<text:list-style>` 16.32.

The `<text:list-level-style-bullet>` element has the following attributes:  
`style:num-prefix` 19.506, `style:num-suffix` 19.507, `text:bullet-char` 19.765, `text:bullet-relative-size` 19.766, `text:level` 19.834 and `text:style-name` 19.880.

The `<text:list-level-style-bullet>` element has the following child elements:  
`<style:list-level-properties>` 17.19 and `<style:text-properties>` 16.29.29.

### 16.34 `<text:list-level-style-number>`

A `<text:list-level-style-number>` specifies a list style where list items are preceded by numbers.

The `<text:list-level-style-number>` element is usable within the following element:  
`<text:list-style>` 16.32.

The `<text:list-level-style-number>` element has the following attributes:  
`style:num-format` 19.504, `style:num-letter-sync` 19.505, `style:num-list-format-name` 19.922, `style:num-prefix` 19.506, `style:num-suffix` 19.507, `text:display-levels` 19.803, `text:level` 19.834, `text:start-value` 19.874 and `text:style-name` 19.880.

The `<text:list-level-style-number>` element has the following child elements:  
`<style:list-level-properties>` 17.19 and `<style:text-properties>` 16.29.29.

## 16.35 <text:list-level-style-image>

A `<text:list-level-style-image>` specifies a list style where list items are preceded by images. This element can be the target of a link [XLink] and can only be contained in list style elements.

The `<text:list-level-style-image>` element is usable within the following element: `<text:list-style>` 16.32.

The `<text:list-level-style-image>` element has the following attributes: `text:level 19.834, xlink:actuate 19.915, xlink:href 19.916, xlink:show 19.917` and `xlink:type 19.919`.

The `<text:list-level-style-image>` element has the following child elements: `<office:binary-data>` 10.4.5 and `<style:list-level-properties>` 17.19.

## 16.36 <text:outline-style>

The `<text:outline-style>` element contains the elements that specify a style for each outline level.

An outline style is a list style that is applied to all headings within a text document where the heading's paragraph style does not define a list style to use itself.

The `<text:outline-style>` element is usable within the following element: `<office:styles>` 3.15.2.

The `<text:outline-style>` element has the following attribute: `style:name 19.502`.

The `<text:outline-style>` element has the following child element: `<text:outline-level-style>` 16.37.

## 16.37 <text:outline-level-style>

The `<text:outline-level-style>` element specifies a style for the specified level.

The `<text:outline-level-style>` element is usable within the following element: `<text:outline-style>` 16.36.

The `<text:outline-level-style>` element has the following attributes: `style:num-format 19.504, style:num-letter-sync 19.505, style:num-list-format-name 19.922, style:num-prefix 19.506, style:num-suffix 19.507, text:display-levels 19.803, text:level 19.834, text:start-value 19.874` and `text:style-name 19.880`.

The `<text:outline-level-style>` element has the following child elements: `<style:list-level-properties>` 17.19 and `<style:text-properties>` 16.29.29.

## 16.38 Table Styles

### 16.38.1 Table Styles

Table styles are `<style:style>` elements that have the family `table`.

### 16.38.2 Table Column Styles

Table column styles are `<style:style>` elements that have the family `table-column`.

### 16.38.3 Table Row Styles

Table row styles are `<style:style>` elements that have the family `table-row`.

## 16.38.4 Table Cell Styles

Table cell styles are `<style:style>` elements that have the family `table-cell`.

In addition to table cell properties, table cell styles may define paragraph and text properties. These are applied to paragraphs contained in a table cell unless they are overwritten by paragraph styles that are specified by the paragraph elements themselves.

## 16.39 Graphic Styles

Graphic styles are `<style:style>` elements that have the family `graphic`.

In addition to graphic properties, graphic styles may define paragraph and text properties. These are applied to paragraphs contained in drawing objects unless they are overwritten by paragraph styles that are specified by the paragraph elements themselves.

## 16.40 Presentation Styles

Presentation styles are `<style:style>` elements that have the family `presentation`.

In addition to graphic properties, presentation styles may define paragraph and text properties. These are applied to paragraphs contained in drawing objects unless they are overwritten by paragraph styles that are specified by the paragraph elements themselves.

## 16.41 Drawing Page Style

A drawing page style is a `<style:style>` element with family `drawing-page`.

Drawing page styles can be used to change the background of a page. If a background is set with the help of a drawing page style, it overrides the background of the master page that is assigned to the page, but not the shapes that are on the master page.

Within presentation documents, the drawing page style may contain presentation formatting properties.

## 16.42 Enhanced Graphic Style Elements

### 16.42.1 `<draw:gradient>`

The `<draw:gradient>` element defines a gradient for filling a drawing object.

The `<draw:gradient>` element is usable within the following element: `<office:styles>` 3.15.2.

The `<draw:gradient>` element has the following attributes: `draw:angle 19.112`, `draw:border 19.114`, `draw:cx 19.129`, `draw:cy 19.130`, `draw:display-name 19.133`, `draw:end-color 19.141`, `draw:end-intensity 19.143`, `draw:name 19.197`, `draw:start-color 19.214`, `draw:start-intensity 19.216` and `draw:style 19.218`.

The `<draw:gradient>` element has no child elements.

### 16.42.2 `<svg:linearGradient>`

The `<svg:linearGradient>` element as specified in §13.2 of [SVG] can be used in documents in OpenDocument format subject to the following rules:

- The gradient shall have a name. It is specified by the `draw:name` attribute.
- Only the `svg:gradientTransform`, `svg:x1`, `svg:y1`, `svg:x2`, `svg:y2` and `svg:spreadMethod` attributes are evaluated.
- The gradient shall be calculated as having a `svg:gradientUnits` attribute value of `objectBoundingBox`.

- The only child element that is evaluated is `<svg:stop>`.
- For `<svg:stop>`, only the `svg:offset`, `svg:stop-color` and `svg:stop-opacity` attributes are evaluated.

The `<svg:linearGradient>` element is usable within the following element:  
`<office:styles> 3.15.2.`

The `<svg:linearGradient>` element has the following attributes: `draw:display-name 19.133`, `draw:name 19.197`, `svg:gradientTransform 19.540`, `svg:gradientUnits 19.541`, `svg:spreadMethod 19.557`, `svg:x1 19.578`, `svg:x2 19.579`, `svg:y1 19.582` and `svg:y2 19.583`.

The `<svg:linearGradient>` element has the following child element: `<svg:stop>`  
`16.42.4.`

### 16.42.3 `<svg:radialGradient>`

The `<svg:radialGradient>` element as specified in §13.2 of [SVG] can be used in documents in OpenDocument format subject to the following rules:

- The gradient shall have a name. It is specified by the `draw:name` attribute.
- For `<svg:radialGradient>`, only the `svg:gradientTransform`, `svg:cx`, `svg:cy`, `svg:r`, `svg:fx`, `svg:fy` and `svg:spreadMethod` attributes are evaluated.
- The gradient shall be calculated as having a `svg:gradientUnits` attribute value of `objectBoundingBox`.
- The only child element that is evaluated is `<svg:stop>`.
- For `<svg:stop>`, only the `svg:offset`, `svg:stop-color` and `svg:stop-opacity` attributes are evaluated.

The `<svg:radialGradient>` element is usable within the following element:  
`<office:styles> 3.15.2.`

The `<svg:radialGradient>` element has the following attributes: `draw:display-name 19.133`, `draw:name 19.197`, `svg:cx 19.528`, `svg:cy 19.529`, `svg:fx 19.538`, `svg:fy 19.539`, `svg:gradientTransform 19.540`, `svg:gradientUnits 19.541`, `svg:r 19.553` and `svg:spreadMethod 19.557`.

The `<svg:radialGradient>` element has the following child element: `<svg:stop>`  
`16.42.4.`

### 16.42.4 `<svg:stop>`

See §13.2.4 of [SVG].

The `<svg:stop>` element is usable within the following elements:  
`<svg:linearGradient> 16.42.2` and `<svg:radialGradient> 16.42.3.`

The `<svg:stop>` element has the following attributes: `svg:offset 19.547`, `svg:stop-color 19.560` and `svg:stop-opacity 19.561`.

The `<svg:stop>` element has no child elements.

### 16.42.5 `<draw:hatch>`

The `<draw:hatch>` element defines a hatch for filling graphic objects. A hatch is a simple pattern of straight lines that is repeated in the fill area.

The `<draw:hatch>` element is usable within the following element: `<office:styles> 3.15.2.`

The `<draw:hatch>` element has the following attributes: `draw:color 19.122`, `draw:display-name 19.133`, `draw:distance 19.134`, `draw:name 19.197`, `draw:rotation 19.209` and `draw:style 19.218`.

The `<draw:hatch>` element has no child elements.

#### 16.42.6 `<draw:fill-image>`

The `<draw:fill-image>` element either specifies a link to a bitmap resource or contains an image in the `<office:binary-data>` child element. Producers shall not write the `<office:binary-data>` child element if the document is represented as a package.

Fill images are not available as automatic styles.

The `<draw:fill-image>` element is usable within the following element: `<office:styles> 3.15.2`.

The `<draw:fill-image>` element has the following attributes: `draw:display-name 19.133`, `draw:name 19.197`, `svg:height 19.543`, `svg:width 19.575`, `xlink:actuate 19.915`, `xlink:href 19.916`, `xlink:show 19.917` and `xlink:type 19.919`.

The `<draw:fill-image>` element has the following child element: `<office:binary-data> 10.4.5`.

#### 16.42.7 `<draw:opacity>`

The `<draw:opacity>` element specifies an opacity gradient for a graphic object. An opacity gradient interpolates opacity. Opacity gradients are not available as automatic styles.

The `<draw:opacity>` element is usable within the following element: `<office:styles> 3.15.2`.

The `<draw:opacity>` element has the following attributes: `draw:angle 19.112`, `draw:border 19.114`, `draw:cx 19.129`, `draw:cy 19.130`, `draw:display-name 19.133`, `draw:end 19.139`, `draw:name 19.197`, `draw:start 19.212` and `draw:style 19.218`.

The `<draw:opacity>` element has no child elements.

#### 16.42.8 `<draw:marker>`

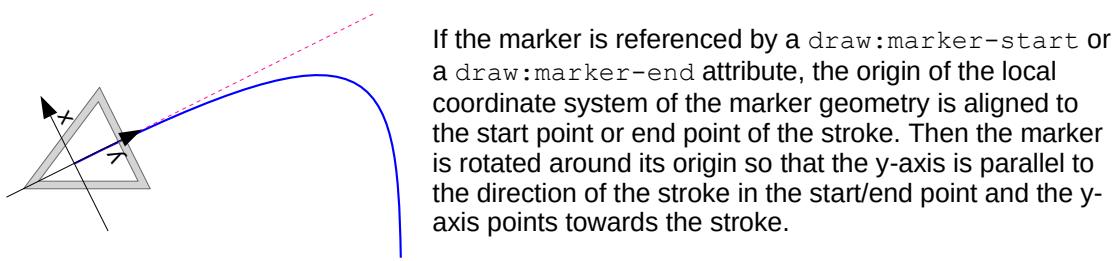
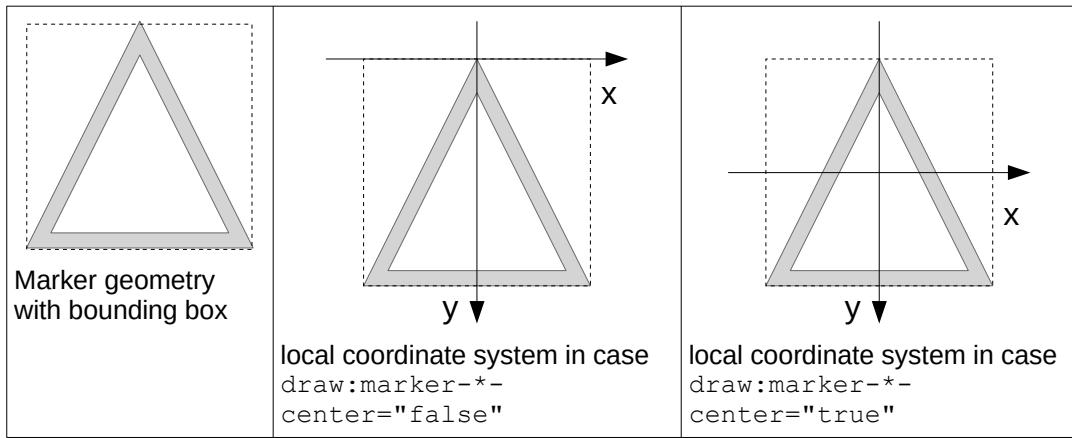
The `<draw:marker>` element represents a marker, which is used to draw polygons at the start or end point of a stroke depending on whether it is referenced by a `draw:marker-start` or `draw:marker-end` attribute.

Marker geometry is defined by a `svg:d` attribute and in conjunction with a `svg:viewBox` attribute.

When the marker is painted, its geometry is first mapped to the stroke start or end point as follows:

If a marker is referenced by a `draw:marker-start` attribute, the marker geometry is scaled so that its width equals the width specified by `draw:marker-start-width` attribute. If the marker is referenced by a `draw:marker-end` attribute, the marker geometry is scaled so that its width equals the width specified by `draw:marker-end-width` attribute. The aspect ratio of the geometry is kept in both cases.

For the marker a local coordinate system is assumed, where the x-axis is horizontal from left to right and the y-axis is vertical from top to bottom. In case of `draw:marker-* center="true"` the origin of this local coordinate system is the center of the bounding box of the marker geometry. In case of `draw:marker-* center="false"` the origin is the center of the top side of the bounding box.



If the marker is referenced by a `draw:marker-start` or a `draw:marker-end` attribute, the origin of the local coordinate system of the marker geometry is aligned to the start point or end point of the stroke. Then the marker is rotated around its origin so that the y-axis is parallel to the direction of the stroke in the start/end point and the y-axis points towards the stroke.

Markers are not available as automatic styles.

The `<draw:marker>` element is usable within the following element: `<office:styles> 3.15.2`.

The `<draw:marker>` element has the following attributes: `draw:display-name 19.133`, `draw:name 19.197`, `svg:d 19.530` and `svg:viewBox 19.574`.

The `<draw:marker>` element has no child elements.

### 16.42.9 `<draw:stroke-dash>`

The `<draw:stroke-dash>` element represents a dash style that can be used to render strokes of shapes. Dash styles are not available as automatic styles.

The `<draw:stroke-dash>` element is usable within the following element: `<office:styles> 3.15.2`.

The `<draw:stroke-dash>` element has the following attributes: `draw:display-name 19.133`, `draw:distance 19.134`, `draw:dots1 19.135`, `draw:dots1-length 19.136`, `draw:dots2 19.137`, `draw:dots2-length 19.138`, `draw:name 19.197` and `draw:style 19.218`.

The `<draw:stroke-dash>` element has no child elements.

## 16.43 `<style:presentation-page-layout>`

The `<style:presentation-page-layout>` element is a container for `<presentation:placeholder>` elements. These placeholders are used as templates for creating new presentation objects and to mark the size and position of an object if the presentation page layout of a drawing page is changed.

The `<style:presentation-page-layout>` element is usable within the following element: `<office:styles> 3.15.2`.

The `<style:presentation-page-layout>` element has the following attributes: `style:display-name 19.476` and `style:name 19.502`.

The `<style:presentation-page-layout>` element has the following child element:  
`<presentation:placeholder> 16.44.`

## 16.44 `<presentation:placeholder>`

The `<presentation:placeholder>` element specifies a placeholder for presentation objects.

The `<presentation:placeholder>` element is usable within the following element:  
`<style:presentation-page-layout> 16.43.`

The `<presentation:placeholder>` element has the following attributes:  
`presentation:object 19.407, svg:height 19.543, svg:width 19.575, svg:x 19.577`  
and `svg:y 19.581.`

The `<presentation:placeholder>` element has no child elements.

## 16.45 Chart Styles

Chart styles are `<style:style>` elements that have the family `chart`.

There are two style inheritance models at work with chart styles. The first model, is standard style inheritance as described in `<style:style>`. 16.2 The second model is inheritance between elements representing parts of a chart.

If an attribute is not specified with a chart style on a `<chart:data-point>` element, the specification of that attribute on its parent `<chart:series>` element is used. If the attribute is not specified on the parent `<chart:series>` element, then the specification of that attribute on its ancestor `<chart:plot-area>` element is used.

Chart styles may define paragraph and text properties. These are applied to paragraphs contained in chart elements unless they are over-ridden by paragraph styles that are specified by the paragraph elements themselves.

# 17 Formatting Properties Elements

## 17.1 General

The elements described in this chapter, which are child elements of style elements defined in chapter 16, together with their attributes, define formatting properties.

## 17.2 <style:page-layout-properties>

The `<style:page-layout-properties>` element acts as a container for attributes and elements that define a page layout.

The `<style:page-layout-properties>` element is usable within the following elements: `<style:default-page-layout>` 16.8 and `<style:page-layout>` 16.5.

The `<style:page-layout-properties>` element has the following attributes: `fo:background-color` 20.182, `fo:border` 20.183.2, `fo:border-bottom` 20.183.3, `fo:border-left` 20.183.4, `fo:border-right` 20.183.5, `fo:border-top` 20.183.6, `fo:margin` 20.205, `fo:margin-bottom` 20.206, `fo:margin-left` 20.207, `fo:margin-right` 20.208, `fo:margin-top` 20.209, `fo:padding` 20.217, `fo:padding-bottom` 20.218, `fo:padding-left` 20.219, `fo:padding-right` 20.220, `fo:padding-top` 20.221, `fo:page-height` 20.215, `fo:page-width` 20.216, `style:border-line-width` 20.248, `style:border-line-width-bottom` 20.249, `style:border-line-width-left` 20.250, `style:border-line-width-right` 20.251, `style:border-line-width-top` 20.252, `style:first-page-number` 20.266, `style:footnote-max-height` 20.296, `style:layout-grid-base-height` 20.304, `style:layout-grid-base-width` 20.305, `style:layout-grid-color` 20.306, `style:layout-grid-display` 20.307, `style:layout-grid-lines` 20.308, `style:layout-grid-mode` 20.309, `style:layout-grid-print` 20.310, `style:layout-grid-ruby-below` 20.311, `style:layout-grid-ruby-height` 20.312, `style:layout-grid-snap-to` 20.313, `style:layout-grid-standard-mode` 20.314, `style:margin-gutter` 20.439, `style:num-format` 20.322, `style:num-letter-sync` 20.323, `style:num-prefix` 20.324, `style:num-suffix` 20.325, `style:paper-tray-name` 20.329, `style:print` 20.330, `style:print-orientation` 20.333, `style:print-page-order` 20.332, `style:register-truth-ref-style-name` 20.337, `style:scale-to` 20.352, `style:scale-to-X` 20.354, `style:scale-to-Y` 20.355, `style:scale-to-pages` 20.353, `style:shadow` 20.359, `style:table-centering` 20.363 and `style:writing-mode` 20.404.

The `<style:page-layout-properties>` element has the following child elements: `<style:background-image>` 17.3, `<style:columns>` 17.12 and `<style:footnote-sep>` 17.4.

## 17.3 <style:background-image>

The `<style:background-image>` element specifies an image as a background.

If the `<style:background-image>` element is empty and if there is no color specified by an `fo:background-color` element in the same formatting properties element, the background color is set to transparent.

**Note:** The `fo:background-color` property also sets a background attribute.

The `<style:background-image>` element is usable within the following elements: `<style:graphic-properties>` 17.21, `<style:header-footer-properties>` 17.5, `<style:page-layout-properties>` 17.2, `<style:paragraph-properties>` 17.6, `<style:section-properties>` 17.11, `<style:table-cell-properties>` 17.18, `<style:table-properties>` 17.15 and `<style:table-row-properties>` 17.17.

The `<style:background-image>` element has the following attributes: `draw:opacity` 19.202, `style:filter-name` 19.481, `style:position` 19.512, `style:repeat` 19.515, `xlink:actuate` 19.915, `xlink:href` 19.916, `xlink:show` 19.917 and `xlink:type` 19.919.

The `<style:background-image>` element has the following child element: `<office:binary-data>` 10.4.5.

## 17.4 `<style:footnote-sep>`

The `<style:footnote-sep>` element specifies the line that separates a footnote area from a body-text area on a page.

The `<style:footnote-sep>` element is usable within the following element: `<style:page-layout-properties>` 17.2.

The `<style:footnote-sep>` element has the following attributes: `style:adjustment` 19.465, `style:color` 19.471, `style:distance-after-sep` 19.479, `style:distance-before-sep` 19.478, `style:line-style` 19.497, `style:rel-width` 19.514 and `style:width` 19.522.

The `<style:footnote-sep>` element has no child elements.

## 17.5 `<style:header-footer-properties>`

The `<style:header-footer-properties>` element specifies formatting properties for both headers and footers.

The `<style:header-footer-properties>` element is usable within the following elements: `<style:footer-style>` 16.7 and `<style:header-style>` 16.6.

The `<style:header-footer-properties>` element has the following attributes: `fo:background-color` 20.182, `fo:border` 20.183.2, `fo:border-bottom` 20.183.3, `fo:border-left` 20.183.4, `fo:border-right` 20.183.5, `fo:border-top` 20.183.6, `fo:margin` 20.205, `fo:margin-bottom` 20.206, `fo:margin-left` 20.207, `fo:margin-right` 20.208, `fo:margin-top` 20.209, `fo:min-height` 20.212, `fo:padding` 20.217, `fo:padding-bottom` 20.218, `fo:padding-left` 20.219, `fo:padding-right` 20.220, `fo:padding-top` 20.221, `style:border-line-width` 20.248, `style:border-line-width-bottom` 20.249, `style:border-line-width-left` 20.250, `style:border-line-width-right` 20.251, `style:border-line-width-top` 20.252, `style:dynamic-spacing` 20.264, `style:shadow` 20.359 and `svg:height` 20.407.

The `<style:header-footer-properties>` element has the following child elements: `<style:background-fill>` 17.26 and `<style:background-image>` 17.3.

## 17.6 `<style:paragraph-properties>`

The `<style:paragraph-properties>` element specifies formatting properties for paragraphs.

The `<style:paragraph-properties>` element is usable within the following elements: `<style:default-style>` 16.4 and `<style:style>` 16.2.

The `<style:paragraph-properties>` element has the following attributes: `fo:background-color` 20.182, `fo:border` 20.183.2, `fo:border-bottom` 20.183.3,

```
fo:border-left 20.183.4, fo:border-right 20.183.5, fo:border-top 20.183.6,  
fo:break-after 20.184, fo:break-before 20.185, fo:hyphenation-keep 20.196,  
fo:hyphenation-ladder-count 20.197, fo:keep-together 20.200, fo:keep-  
with-next 20.201, fo:line-height 20.204, fo:margin 20.205, fo:margin-bottom  
20.206, fo:margin-left 20.207, fo:margin-right 20.208, fo:margin-top 20.209,  
fo:orphans 20.214, fo:padding 20.217, fo:padding-bottom 20.218, fo:padding-  
left 20.219, fo:padding-right 20.220, fo:padding-top 20.221, fo:text-align  
20.223, fo:text-align-last 20.224, fo:text-indent 20.225, fo:widows 20.228,  
style:auto-text-indent 20.246, style:background-transparency 20.247,  
style:border-line-width 20.248, style:border-line-width-bottom 20.249,  
style:border-line-width-left 20.250, style:border-line-width-right  
20.251, style:border-line-width-top 20.252, style:contextual-spacing  
20.255, style:font-independent-line-spacing 20.276, style:join-border  
20.300, style:justify-single-word 20.301, style:line-break 20.315,  
style:line-height-at-least 20.317, style:line-spacing 20.318, style:page-  
number 20.328, style:punctuation-wrap 20.335, style:register-true 20.336,  
style:shadow 20.359, style:snap-to-layout-grid 20.361, style:tab-stop-  
distance 20.362, style:text-autospace 20.365, style:vertical-align 20.396,  
style:writing-mode 20.404, style:writing-mode-automatic 20.405,  
text:line-number 20.430, text:number-lines 20.434 and text:relative-tab-  
stop-position 20.440.
```

The `<style:paragraph-properties>` element has the following child elements:

`<style:background-fill>` 17.26, `<style:background-image>` 17.3,

`<style:drop-cap>` 17.9 and `<style:tab-stops>` 17.7.

## 17.7 `<style:tab-stops>`

The `<style:tab-stops>` element is a container for `<style:tab-stop>` elements.

If a style contains a `<style:tab-stops>` element, it overrides the entire `<style:tab-  
stops>` element of the parent style such that no `<style:tab-stop>` children are inherited;  
otherwise, the style inherits the entire `<style:tab-stops>` element as specified in section  
16.2 `<style:style>`.

The `<style:tab-stops>` element is usable within the following element:

`<style:paragraph-properties>` 17.6.

The `<style:tab-stops>` element has no attributes.

The `<style:tab-stops>` element has the following child element: `<style:tab-stop>`  
17.8.

## 17.8 `<style:tab-stop>`

The `<style:tab-stop>` element defines a tab stop.

The `<style:tab-stop>` element is usable within the following element: `<style:tab-  
stops>` 17.7.

The `<style:tab-stop>` element has the following attributes: `style:char` 19.469,  
`style:leader-color` 19.488, `style:leader-style` 19.489, `style:leader-text`  
19.490, `style:leader-text-style` 19.491, `style:leader-type` 19.492,  
`style:leader-width` 19.493, `style:position` 19.512, `style:type` 19.519.

The `<style:tab-stop>` element has no child elements.

## 17.9 <style:drop-cap>

The `<style:drop-cap>` element specifies if the one or more characters at the start of a paragraph are to be displayed in a larger font and span multiple lines of text.

The `<style:drop-cap>` element is usable within the following element:

`<style:paragraph-properties>` 17.6.

The `<style:drop-cap>` element has the following attributes: `style:distance` 19.477, `style:length` 19.496, `style:lines` 19.498 and `style:style-name` 19.518.

The `<style:drop-cap>` element has no child elements.

## 17.10 <style:ruby-properties>

The `<style:ruby-properties>` element specifies formatting properties for ruby elements.

The `<style:ruby-properties>` element is usable within the following elements:

`<style:default-style>` 16.4 and `<style:style>` 16.2.

The `<style:ruby-properties>` element has the following attributes: `style:ruby-align` 20.349 and `style:ruby-position` 20.350.

The `<style:ruby-properties>` element has no child elements.

## 17.11 <style:section-properties>

The `<style:section-properties>` element specifies the formatting properties used by section styles.

The `<style:section-properties>` element is usable within the following elements:

`<style:default-style>` 16.4 and `<style:style>` 16.2.

The `<style:section-properties>` element has the following attributes:

`fo:background-color` 20.182, `fo:margin-left` 20.207, `fo:margin-right` 20.208, `style:editable` 20.265, `style:protect` 20.334, `style:writing-mode` 20.404 and `text:dont-balance-text-columns` 20.428.

The `<style:section-properties>` element has the following child elements:

`<style:background-fill>` 17.26, `<style:background-image>` 17.3,

`<style:columns>` 17.12 and `<text:notes-configuration>` 16.31.3.

## 17.12 <style:columns>

The `<style:columns>` element specifies the number and formatting of columns on a page or in a section or graphic. If it does not contain any child formatting properties elements, all columns are assigned the same width.

The `<style:columns>` element is usable within the following elements:

`<style:graphic-properties>` 17.21, `<style:page-layout-properties>` 17.2 and `<style:section-properties>` 17.11.

The `<style:columns>` element has the following attributes: `fo:column-count` 19.232 and `fo:column-gap` 19.233.

The `<style:columns>` element has the following child elements: `<style:column>` 17.13 and `<style:column-sep>` 17.14.

## 17.13 <style:column>

The `<style:column>` element specifies formatting properties for a column.

The `<style:column>` element is usable within the following element: `<style:columns>` 17.12.

The `<style:column>` element has the following attributes: `fo:end-indent` 19.235, `fo:space-after` 19.243, `fo:space-before` 19.244, `fo:start-indent` 19.245 and `style:rel-width` 19.514.

The `<style:column>` element has no child elements.

## 17.14 `<style:column-sep>`

The `<style:column-sep>` element specifies a separator line between columns.

The `<style:column-sep>` element is usable within the following element: `<style:columns>` 17.12.

The `<style:column-sep>` element has the following attributes: `style:color` 19.471, `style:height` 19.486, `style:style` 19.517, `style:vertical-align` 19.520 and `style:width` 19.522.

The `<style:column-sep>` element has no child elements.

## 17.15 `<style:table-properties>`

The `<style:table-properties>` element specifies formatting properties for tables.

The `<style:table-properties>` element is usable within the following elements: `<style:default-style>` 16.4 and `<style:style>` 16.2.

The `<style:table-properties>` element has the following attributes: `fo:background-color` 20.182, `fo:break-after` 20.184, `fo:break-before` 20.185, `fo:keep-with-next` 20.201, `fo:margin` 20.205, `fo:margin-bottom` 20.206, `fo:margin-left` 20.207, `fo:margin-right` 20.208, `fo:margin-top` 20.209, `style:may-break-between-rows` 20.319, `style:page-number` 20.328, `style:rel-width` 20.340, `style:shadow` 20.359, `style:width` 20.399, `style:writing-mode` 20.404, `table:align` 20.414, `table:border-model` 20.415, `table:display` 20.416 and `table:tab-color` 20.442.

The `<style:table-properties>` element has the following child elements: `<style:background-fill>` 17.26 and `<style:background-image>` 17.3.

## 17.16 `<style:table-column-properties>`

The `<style:table-column-properties>` element specifies formatting properties for table columns.

The `<style:table-column-properties>` element is usable within the following elements: `<style:default-style>` 16.4 and `<style:style>` 16.2.

The `<style:table-column-properties>` element has the following attributes: `fo:break-after` 20.184, `fo:break-before` 20.185, `style:column-width` 20.254, `style:rel-column-width` 20.338 and `style:use-optimal-column-width` 20.393.

The `<style:table-column-properties>` element has no child elements.

## 17.17 `<style:table-row-properties>`

The `<style:table-row-properties>` element specifies formatting properties for table rows.

The `<style:table-row-properties>` element is usable within the following elements: `<style:default-style>` 16.4 and `<style:style>` 16.2.

The `<style:table-row-properties>` element has the following attributes: `fo:background-color` 20.182, `fo:break-after` 20.184, `fo:break-before` 20.185, `fo:keep-together` 20.200, `style:min-row-height` 20.320, `style:row-height` 20.348 and `style:use-optimal-row-height` 20.394.

The `<style:table-row-properties>` element has the following child elements: `<style:background-fill>` 17.26 and `<style:background-image>` 17.3.

## 17.18 `<style:table-cell-properties>`

The `<style:table-cell-properties>` element specifies formatting properties for cells.

The `<style:table-cell-properties>` element is usable within the following elements: `<style:default-style>` 16.4 and `<style:style>` 16.2.

The `<style:table-cell-properties>` element has the following attributes: `fo:background-color` 20.182, `fo:border` 20.183.2, `fo:border-bottom` 20.183.3, `fo:border-left` 20.183.4, `fo:border-right` 20.183.5, `fo:border-top` 20.183.6, `fo:padding` 20.217, `fo:padding-bottom` 20.218, `fo:padding-left` 20.219, `fo:padding-right` 20.220, `fo:padding-top` 20.221, `fo:wrap-option` 20.230, `style:border-line-width` 20.248, `style:border-line-width-bottom` 20.249, `style:border-line-width-left` 20.250, `style:border-line-width-right` 20.251, `style:border-line-width-top` 20.252, `style:cell-protect` 20.253, `style:decimal-places` 20.258, `style:diagonal-bl-tr` 20.259, `style:diagonal-bl-tr-widths` 20.260, `style:diagonal-tl-br` 20.261, `style:diagonal-tl-br-widths` 20.262, `style:direction` 20.263, `style:glyph-orientation-vertical` 20.297, `style:print-content` 20.331, `style:repeat-content` 20.342, `style:rotation-align` 20.346, `style:rotation-angle` 20.347, `style:shadow` 20.359, `style:shrink-to-fit` 20.360, `style:text-align-source` 20.364, `style:vertical-align` 20.396 and `style:writing-mode` 20.404.

The `<style:table-cell-properties>` element has the following child elements: `<style:background-fill>` 17.26 and `<style:background-image>` 17.3.

## 17.19 `<style:list-level-properties>`

The `<style:list-level-properties>` element specifies formatting properties for a list level.

The `<style:list-level-properties>` element is usable within the following elements: `<text:list-level-style-bullet>` 16.33, `<text:list-level-style-image>` 16.35, `<text:list-level-style-number>` 16.34 and `<text:outline-level-style>` 16.37.

The `<style:list-level-properties>` element has the following attributes: `fo:height` 20.194, `fo:text-align` 20.223, `fo:width` 20.229, `style:font-name` 20.277, `style:vertical-pos` 20.397, `style:vertical-rel` 20.398, `svg:y` 20.412, `text:list-level-position-and-space-mode` 20.431, `text:min-label-distance` 20.432, `text:min-label-width` 20.433 and `text:space-before` 20.435.

The `<style:list-level-properties>` element has the following child element: `<style:list-level-label-alignment>` 17.20.

## 17.20 `<style:list-level-label-alignment>`

The `<style:list-level-label-alignment>` element specifies the position and spacing of a list label and its list item.

The element will be evaluated only if the `text:list-level-position-and-space-mode` attribute has the value `label-alignment`.

The `fo:text-indent` and `fo:margin-left` attributes specify the indent and margin for the text lines of a list item. The text lines of a list item contain the list level properties specific content and the text of the paragraph inside the list item. The value `fo:text-indent + fo:margin-left` specifies the indent of the first text line of the list item. This position is also the alignment position for the list label, because the value of `text:min-label-width` is treated as 0. 19.246, 20.431

The `fo:text-indent` and `fo:margin-left` attributes are evaluated only for paragraphs inside list items whose paragraph styles do not specify them. If one of the two properties, or both, are specified by the paragraph style, the text indent and/or left margin are taken from the paragraph style. In this case the paragraph style's properties are used to determine the indent and margin for the text lines and thus also the alignment position.

The `<style:list-level-label-alignment>` element is usable within the following element: `<style:list-level-properties>` 17.19.

The `<style:list-level-label-alignment>` element has the following attributes: `fo:margin-left 19.237`, `fo:text-indent 19.246`, `text:label-followed-by 19.833` and `text:list-tab-stop-position 19.836`.

The `<style:list-level-label-alignment>` element has no child elements.

## 17.21 `<style:graphic-properties>`

The `<style:graphic-properties>` element specifies formatting properties for chart, draw, graphic, and frame elements.

The `<style:graphic-properties>` element is usable within the following elements: `<style:default-style>` 16.4 and `<style:style>` 16.2.

The `<style:graphic-properties>` element has the following attributes: `dr3d:ambient-color 20.74`, `dr3d:back-scale 20.75`, `dr3d:backface-culling 20.76`, `dr3d:close-back 20.77`, `dr3d:close-front 20.78`, `dr3d:depth 20.79`, `dr3d:diffuse-color 20.80`, `dr3d:edge-rounding 20.81`, `dr3d:edge-rounding-mode 20.82`, `dr3d:emissive-color 20.83`, `dr3d:end-angle 20.84`, `dr3d:horizontal-segments 20.85`, `dr3d:lighting-mode 20.86`, `dr3d:normals-direction 20.87`, `dr3d:normals-kind 20.88`, `dr3d:shadow 20.89`, `dr3d:shininess 20.90`, `dr3d:specular-color 20.91`, `dr3d:texture-filter 20.92`, `dr3d:texture-generation-mode-x 20.95`, `dr3d:texture-generation-mode-y 20.96`, `dr3d:texture-kind 20.93`, `dr3d:texture-mode 20.94`, `dr3d:vertical-segments 20.97`, `draw:allow-overlap 20.438`, `draw:auto-grow-height 20.98`, `draw:auto-grow-width 20.99`, `draw:blue 20.101`, `draw:caption-angle 20.102`, `draw:caption-angle-type 20.103`, `draw:caption-escape 20.104`, `draw:caption-escape-direction 20.105`, `draw:caption-fit-line-length 20.106`, `draw:caption-gap 20.107`, `draw:caption-line-length 20.108`, `draw:caption-type 20.109`, `draw:color-inversion 20.110`, `draw:color-mode 20.111`, `draw:contrast 20.112`, `draw:decimal-places 20.113`, `draw:decorative 20.441`, `draw:draw-aspect 20.114`, `draw:end-guide 20.115`, `draw:end-line-spacing-horizontal 20.116`, `draw:end-line-spacing-vertical 20.117`, `draw:fill 20.118`, `draw:fill-color 20.119`, `draw:fill-gradient-name 20.120`, `draw:fill-hatch-name 20.121`, `draw:fill-hatch-solid 20.122`, `draw:fill-image-height 20.123`, `draw:fill-image-name 20.124`, `draw:fill-image-ref-point 20.125`, `draw:fill-image-ref-point-x 20.126`, `draw:fill-image-ref-point-y 20.127`, `draw:fill-image-width 20.128`, `draw:fit-to-contour 20.129`, `draw:fit-to-size 20.130`, `draw:frame-display-border 20.131`, `draw:frame-display-scrollbar 20.133`, `draw:frame-margin-horizontal 20.132`, `draw:frame-margin-vertical 20.134`, `draw:gamma 20.135`, `draw:gradient-step-count 20.137`, `draw:green 20.136`, `draw:guide-distance 20.138`, `draw:guide-overhang 20.139`, `draw:image-opacity 20.140`, `draw:line-distance 20.141`, `draw:luminance 20.142`, `draw:marker-end 20.143`

```

draw:marker-end-center 20.144, draw:marker-end-width 20.145, draw:marker-
start 20.146, draw:marker-start-center 20.147, draw:marker-start-width
20.148, draw:measure-align 20.149, draw:measure-vertical-align 20.150,
draw:ole-draw-aspect 20.151, draw:opacity 20.152, draw:opacity-name 20.153,
draw:parallel 20.154, draw:placing 20.155, draw:red 20.156, draw:secondary-
fill-color 20.157, draw:shadow 20.158, draw:shadow-color 20.159,
draw:shadow-offset-x 20.160, draw:shadow-offset-y 20.161, draw:shadow-
opacity 20.162, draw:show-unit 20.163, draw:start-guide 20.164, draw:start-
line-spacing-horizontal 20.165, draw:start-line-spacing-vertical 20.166,
draw:stroke 20.167, draw:stroke-dash 20.168, draw:stroke-dash-names 20.169,
draw:stroke-linejoin 20.170, draw:symbol-color 20.172, draw:textarea-
horizontal-align 20.173, draw:textarea-vertical-align 20.174, draw:tile-
repeat-offset 20.175, draw:unit 20.180, draw:visible-area-height 20.176,
draw:visible-area-left 20.177, draw:visible-area-top 20.178,
draw:visible-area-width 20.179, draw:wrap-influence-on-position 20.181,
fo:background-color 20.182, fo:border 20.183.2, fo:border-bottom 20.183.3,
fo:border-left 20.183.4, fo:border-right 20.183.5, fo:border-top 20.183.6,
fo:clip 20.186, fo:margin 20.205, fo:margin-bottom 20.206, fo:margin-left
20.207, fo:margin-right 20.208, fo:margin-top 20.209, fo:max-height 20.210,
fo:max-width 20.211, fo:min-height 20.212, fo:min-width 20.213, fo:padding
20.217, fo:padding-bottom 20.218, fo:padding-left 20.219, fo:padding-right
20.220, fo:padding-top 20.221, fo:wrap-option 20.230, style:background-
transparency 20.247, style:border-line-width 20.248, style:border-line-
width-bottom 20.249, style:border-line-width-left 20.250, style:border-
line-width-right 20.251, style:border-line-width-top 20.252,
style:editable 20.265, style:flow-with-text 20.267, style:horizontal-pos
20.298, style:horizontal-rel 20.299, style:mirror 20.321, style:number-
wrapped-paragraphs 20.326, style:overflow-behavior 20.327, style:print-
content 20.331, style:protect 20.334, style:rel-height 20.339, style:rel-
width 20.340, style:repeat 20.341, style:run-through 20.351, style:shadow
20.359, style:shrink-to-fit 20.360, style:vertical-pos 20.397,
style:vertical-rel 20.398, style:wrap 20.400, style:wrap-contour 20.401,
style:wrap-contour-mode 20.402, style:wrap-dynamic-threshold 20.403,
style:writing-mode 20.404, svg:fill-rule 20.406, svg:height 20.407,
svg:stroke-color 20.408, svg:stroke-linecap 20.171, svg:stroke-opacity
20.409, svg:stroke-width 20.410, svg:width 20.413, svg:x 20.411, svg:y 20.412,
text:anchor-page-number 20.417, text:anchor-type 20.418, text:animation
20.419, text:animation-delay 20.420, text:animation-direction 20.421,
text:animation-repeat 20.422, text:animation-start-inside 20.423,
text:animation-steps 20.424 and text:animation-stop-inside 20.425.

```

The `<style:graphic-properties>` element has the following child elements:  
`<style:background-image>` 17.3, `<style:columns>` 17.12 and `<text:list-
style>` 16.32.

## 17.22 `<style:chart-properties>`

The `<style:chart-properties>` element specifies formatting properties for charts.

The `<style:chart-properties>` element is usable within the following elements:  
`<style:default-style>` 16.4 and `<style:style>` 16.2.

The `<style:chart-properties>` element has the following attributes: `chart:angle-
offset 20.2`, `chart:auto-position 20.3`, `chart:auto-size 20.4`, `chart:axis-
label-position 20.5`, `chart:axis-position 20.6`, `chart:connect-bars 20.7`,
`chart:data-label-number 20.8`, `chart:data-label-series 20.9`, `chart:data-
label-symbol 20.10`, `chart:data-label-text 20.11`, `chart:deep 20.12`,

```

chart:display-label 20.13, chart:error-category 20.14, chart:error-lower-indicator 20.15, chart:error-lower-limit 20.16, chart:error-lower-range 20.17, chart:error-margin 20.18, chart:error-percentage 20.19, chart:error-upper-indicator 20.20, chart:error-upper-limit 20.21, chart:error-upper-range 20.22, chart:gap-width 20.23, chart:group-bars-per-axis 20.24, chart:hole-size 20.25, chart:include-hidden-cells 20.26, chart:interpolation 20.27, chart:interval-major 20.28, chart:interval-minor-divisor 20.29, chart:japanese-candle-stick 20.30, chart:label-arrangement 20.31, chart:label-position 20.32, chart:label-position-negative 20.33, chart:lines 20.34, chart:link-data-style-to-source 20.35, chart:logarithmic 20.36, chart:major-origin 20.436, chart:maximum 20.37, chart:mean-value 20.38, chart:minimum 20.39, chart:minor-logarithmic 20.437, chart:origin 20.40, chart:overlap 20.41, chart:percentage 20.43, chart:pie-offset 20.42, chart:regression-force-intercept 20.44, chart:regression-intercept-value 20.45, chart:regression-max-degree 20.46, chart:regression-moving-type 20.47, chart:regression-name 20.48, chart:regression-period 20.49, chart:regression-type 20.50, chart:reverse-direction 20.51, chart:right-angled-axes 20.52, chart:scale-text 20.53, chart:series-source 20.54, chart:solid-type 20.56, chart:sort-by-x-values 20.55, chart:spline-order 20.57, chart:spline-resolution 20.58, chart:stacked 20.59, chart:symbol-height 20.60, chart:symbol-name 20.61, chart:symbol-type 20.62, chart:symbol-width 20.63, chart:text-overlap 20.64, chart:three-dimensional 20.65, chart:tick-mark-position 20.70, chart:tick-marks-major-inner 20.66, chart:tick-marks-major-outer 20.67, chart:tick-marks-minor-inner 20.68, chart:tick-marks-minor-outer 20.69, chart:treat-empty-cells 20.71, chart:vertical 20.72, chart:visible 20.73, style:direction 20.263, style:rotation-angle 20.347 and text:line-break 20.429.

```

The `<style:chart-properties>` element has the following child elements: `<chart:label-separator>` 17.24 and `<chart:symbol-image>` 17.23.

## 17.23 <chart:symbol-image>

The `<chart:symbol-image>` element specifies an image to be used for a data point in a chart.

This element should be used only together with a `chart:symbol-type` attribute and only when that attribute has the value `image`.

The `<chart:symbol-image>` element is usable within the following element: `<style:chart-properties>` 17.22.

The `<chart:symbol-image>` element has the following attribute: `xlink:href` 19.916.

The `<chart:symbol-image>` element has no child elements.

## 17.24 <chart:label-separator>

The `<chart:label-separator>` element specifies text that is used for separating different parts of a textual data label. These parts can be the value as number, the value as a percentage (20.8), and the label text (20.11).

If the `<chart:label-separator>` element is omitted, the default value is a space character “ ” (U+0020, SPACE).

This element should be used within chart styles that are applied to `<chart:label>`, `<chart:point>`, `<chart:series>` or `<chart:plot-area>` elements.

If a separator is not specified by this element in a chart style applied to a `<chart:label>` element, the separator specified with a chart style applied to a parent element is used.

The `<chart:label-separator>` element is usable within the following element: `<style:chart-properties>` 17.22.

The `<chart:label-separator>` element has no attributes.

The `<chart:label-separator>` element has the following child element: `<text:p>` 5.1.3.

## 17.25 `<style:drawing-page-properties>`

The `<style:drawing-page-properties>` element specifies formatting that can be applied to a `<draw:page>` element, a `<presentation:notes>` element, a `<style:handout-master>` element, or a `<style:master-page>` element.

The `<style:drawing-page-properties>` element is usable within the following elements: `<style:default-style>` 16.4 and `<style:style>` 16.2.

The `<style:drawing-page-properties>` element has the following attributes: `draw:background-size` 20.100, `draw:fill` 20.118, `draw:fill-color` 20.119, `draw:fill-gradient-name` 20.120, `draw:fill-hatch-name` 20.121, `draw:fill-hatch-solid` 20.122, `draw:fill-image-height` 20.123, `draw:fill-image-name` 20.124, `draw:fill-image-ref-point` 20.125, `draw:fill-image-ref-point-x` 20.126, `draw:fill-image-ref-point-y` 20.127, `draw:fill-image-width` 20.128, `draw:gradient-step-count` 20.137, `draw:opacity` 20.152, `draw:opacity-name` 20.153, `draw:secondary-fill-color` 20.157, `draw:tile-repeat-offset` 20.175, `presentation:background-objects-visible` 20.231, `presentation:background-visible` 20.232, `presentation:display-date-time` 20.233, `presentation:display-footer` 20.234, `presentation:display-header` 20.235, `presentation:display-page-number` 20.236, `presentation:duration` 20.237, `presentation:transition-speed` 20.238, `presentation:transition-style` 20.239, `presentation:transition-type` 20.240, `presentation:visibility` 20.241, `smil:direction` 20.242, `smil:fadeColor` 20.243, `smil:subtype` 20.244, `smil:type` 20.245, `style:repeat` 20.341 and `svg:fill-rule` 20.406.

The `<style:drawing-page-properties>` element has the following child element: `<presentation:sound>` 10.8.2.

## 17.26 `<style:background-fill>`

The `<style:background-fill>` element serves as a container for fill attributes to specify a background. It has a mandatory attribute `draw:fill` 20.118 to specify the kind of fill. The other attributes specify details to the fill.

The `<style:background-fill>` element takes precedence over element `<style:background-image>` 17.3, attribute `fo:background-color` 20.182 and attribute `style:background-transparency` 20.247. Consumers not able to use the `<style:background-fill>` element may ignore it and use a given `<style:background-image>` element or `fo:background-color` and `fo:background-transparency` attributes instead.

The background of an object is transparent if none of the styles applied following the rules specified in a `<style:style>` 16.2 element defines a background formatting.

The `<style:background-fill>` element is usable within the following elements: `<style:header-footer-properties>` 17.5, `<style:paragraph-properties>` 17.6,

```
<style:section-properties> 17.11, <style:table-cell-properties> 17.18,  
<style:table-properties> 17.15 and <style:table-row-properties> 17.17.  
The <style:background-fill> element has the following attributes: draw:fill Error:  
Reference source not found, draw:fill-color Error: Reference source not found,  
draw:fill-gradient-name Error: Reference source not found, draw:fill-hatch-  
name Error: Reference source not found, draw:fill-hatch-solid Error: Reference  
source not found, draw:fill-image-height Error: Reference source not found,  
draw:fill-image-name Error: Reference source not found, draw:fill-image-ref-  
point Error: Reference source not found, draw:fill-image-ref-point-x Error:  
Reference source not found, draw:fill-image-ref-point-y Error: Reference source  
not found, draw:fill-image-width Error: Reference source not found,  
draw:gradient-step-count Error: Reference source not found, draw:opacity 19.202,  
draw:opacity-name Error: Reference source not found, draw:secondary-fill-color  
Error: Reference source not found, draw:tile-repeat-offset Error: Reference source  
not found, style:repeat 19.515 and svg:fill-rule Error: Reference source not found.  
The <style:background-fill> element has no child elements.
```

---

# 18 Datatypes

## 18.1 Introduction

The values of attributes and elements have datatypes. This specification defines datatypes for values of elements and attributes of OpenDocument documents.

**Note:** This specification and the schema make reference to other specifications for some datatype definitions. Some datatypes introduced here may be specializations of other datatypes defined here or in other specifications.

## 18.2 W3C Schema Datatypes

The following [XMLSchema-2] datatypes are used in this specification:

- string
- date
- time
- dateTime
- decimal
- duration
- int
- integer
- nonNegativeInteger
- positiveInteger
- double
- long
- short
- base64Binary
- ID
- IDREF
- IDREFS
- NCName

## 18.3 Other Datatypes

### 18.3.1 angle

An angle, as defined in §4.1 of [SVG], is a double value that may be followed immediately by one of the following angle unit identifiers: `deg` (degrees), `grad` (gradiants) or `rad` (radians). If no unit identifier is specified, the value is assumed to be in degrees.

**Note:** OpenDocument v1.1 did not support angle specifications that contain an angle unit identifier. Angle unit identifiers should be omitted for compatibility with OpenDocument v1.1.

### 18.3.2 anyIRI

An IRI-reference as defined in [RFC3987], expressed in an [XMLSchema-2] anyURI.

**Note:** The procedure for resolution of anyIRI values that are not IRI values is undefined.

### 18.3.3 boolean

A Boolean value may have either of the values `true` or `false`.

### 18.3.4 cellAddress

A cell address as specified in 9.2.1.

### 18.3.5 cellRangeAddress

A cell range address as specified in 9.2.3.

### 18.3.6 cellRangeAddressList

A cell range address list as specified in 9.2.5.

### 18.3.7 character

A character value is a string with only one character.

### 18.3.8 clipShape

A clip shape as defined in §7.20.1 of [XSL]. Offsets shall not be specified using the units `em` and `px`.

### 18.3.9 color

A RGB color in conformance with §5.9.9 of [XSL], that is a RGB color in notation “#rrggbb”, where rr, gg and bb are pairs of hexadecimal digits.

### 18.3.10 coordinate

A coordinate represents a length in a coordinate system. It specifies the distance from the origin of the coordinate system along the relevant axis.

### 18.3.11 countryCode

A country code is a value that conforms to ISO 3166 ([ISO3166-1], [ISO3166-2], [ISO3166-3], or their successors, and further parts).

See §2.2.4 Region Subtag of [RFC5646], or its successors.

### 18.3.12 CURIE

A CURIE, as described in §7 of [RDFa].

### 18.3.13 CURIEs

A list of CURIEs, as described in §7 of [RDFa], separated by white spaces.

### 18.3.14 dateOrDateTime

A dateOrDateTime value is either an [XMLSchema-2] date value or an [XMLSchema-2] dateTime value.

### 18.3.15 distance

The length between two points.

### 18.3.16

### 18.3.17 language

A language is a natural language identifier as defined by [RFC5646], or its successor. The lexical space is the same as the [XMLSchema-2] language data type.

**Note 1:** The language datatype is the same as the [XMLSchema-2] language datatype, except that its value range is not restricted to values of [RFC3066], but follows the syntax of the `xml:lang` attribute. See §2.12 of [XML1.0].

### 18.3.18 languageCode

A language code is a value that conforms to ISO 639 ([ISO639-1], [ISO639-2], [ISO639-3] or their successors, and further parts).

See §2.2.1 Primary Language Subtag of [RFC5646], or its successors.

### 18.3.19 length

A (positive or negative) length, consisting of magnitude and unit, in conformance with the Units of Measure defined in §5.9.13 of [XSL].

### 18.3.20 namespacedToken

A namespaced token is an [XMLSchema-2] QName that matches the definition of `PrefixedName` in §4 of [XML-Names].

### 18.3.21 nonNegativeLength

A measurement with a value that shall be zero or positive.

### 18.3.22 nonNegativePixelLength

A non-negative measurement that shall have the unit `px` (pixels).

### 18.3.23 pathData

Path data as described in §8.3 of [SVG]. Attribute values of this data type shall match the BNF grammar for SVG path data in §8.3.9 of [SVG].

### 18.3.24 percent

(Positive or negative) percentage values in conformance with §5.9.11 of [XSL].

### 18.3.25 point3D

A 3D point. The syntax of a 3D point is the same as that for `vector3D` (18.3.41), except that each coordinate is shall be followed by a unit.

### 18.3.26 points

A sequence of points. Each point consists of two integer coordinates separated by a comma `,` (U+002C, COMMA). The points are separated by white space.

### 18.3.27 positiveLength

A measurement with a value that shall be positive.

### 18.3.28 relativeLength

A relative length is a positive integer, followed by a "\*" (U+002A, ASTERISK) character.

### 18.3.29 safeCURIE

A safe CURIE, as described in §7 of [RDFa].

### 18.3.30 scriptCode

A script code conforming to [ISO15924], or its successors. See §2.2.3 Script Subtag of [RFC5646], or its successors.

### 18.3.31 signedZeroToHundredPercent

Percentage values in the range [-100%,100%] in conformance with §5.9.11 of [XSL].

### 18.3.32 styleName

A NCName as specified in [XMLSchema-2][XMLSchema-2] that is the name of a style.

### 18.3.33 styleNameRef

A NCName as specified in [XMLSchema-2] that is the name of a referenced style, or an empty value.

### 18.3.34 styleNameRefs

A white-space-separated list of NCNames as specified in [XMLSchema-2] that are the names of a style.

### 18.3.35 targetFrameName

The name of a target frame in conformance with §6.16 of [HTML4].

### 18.3.36 textEncoding

A character encoding in the notation described in the §4.3.3 of [XML1.0], or the value `x-symbol`. The value `x-symbol` means that the character encoding is not enumerated by §4.3.3 of [XML1.0].

### 18.3.37 timeOrDateTime

A timeOrDateTime value is either an [XMLSchema-2] time or an [XMLSchema-2] dateTime value.

### 18.3.38 URIorSafeCURIE

An IRI or a safe CURIE, as described in §7 of [RDFa].

### 18.3.39 valueType

A list of value types.

### 18.3.40 variableName

A string specifying the name of a variable.

### **18.3.41 vector3D**

A 3-element vector that is represented by floating-point x, y, z coordinates. The format of a vector3D value is white space separated tokens in the order x, y, and z, and that expression starts with "(" (LEFT PARENTHESIS U+0028) and ends with ")" (RIGHT PARENTHESIS U+0029). If this value represents a normal, then it should be normalized.

### **18.3.42 zeroToHundredPercent**

Percentage values in the range [0%,100%] in conformance with §5.9.11 of [XSL].

### **18.3.43 error**

The string value “error.”

### **18.3.44 nonNegativePercent**

Non-negative percentage values.

---

# 19 General Attributes

## 19.1 General

Attributes for elements defined by OpenDocument are divided between those used by structural elements (19) versus those used by `<style:*-properties>` elements (20).

Attributes have *default values* defined by this standard. When any element appears in a document instance without such an attribute, consumers should behave as if the attribute is present with the defined default value.

## 19.2 anim:color-interpolation

The `anim:color-interpolation` attribute specifies the color space that is used for color interpolation.

The defined values for the `anim:color-interpolation` attribute are:

- `hsl`: HSL (hue, saturation, lightness) color model.
- `rgb`: RGB (red, green, blue) color model.

The default value for this attribute is `rgb`.

The `anim:color-interpolation` attribute is usable with the following element:  
`<anim:animateColor>` 15.2.6.

The values of the `anim:color-interpolation` attribute are `rgb` or `hsl`.

## 19.3 anim:color-interpolation-direction

The `anim:color-interpolation-direction` attribute specifies the direction that is used for color interpolation. This is only valid for the HSL color space.

The defined values for the `anim:color-interpolation-direction` attribute are:

- `clockwise`: color interpolation proceeds in a clockwise direction.
- `counterclockwise`: color interpolation proceeds in a counterclockwise direction.

The default value for this attribute is `clockwise`.

The `anim:color-interpolation-direction` attribute is usable with the following element: `<anim:animateColor>` 15.2.6.

The values of the `anim:color-interpolation-direction` attribute are `clockwise` or `counter-clockwise`.

## 19.4 anim:audio-level

The `anim:audio-level` attribute specifies the audio volume during a playback.

The defined value range for the `anim:audio-level` attribute is a rational number between 0 (inaudible) and 1 (system volume), inclusive.

The `anim:audio-level` attribute is usable with the following element: `<anim:audio>` 15.5.

The `anim:audio-level` attribute has the data type `double` 18.2.

## 19.5 anim:command

The `anim:command` attribute specifies an animation command.

The defined values for the `anim:command` attribute are:

- `custom`: the command is user-defined.
- `verb`: the command targets an embedded [OLE] object. 10.4.6.3. The parameter `verb` is the verb number executed on the embedded object.
- `play`: the command targets a media shape and starts its playback. The parameter `media-time` defines the playback start time in seconds. If this parameter is not set, playback starts at the last position.
- `toggle-pause`: the command targets a media shape and toggles its playback state from `play` to `paused` or from `paused` to `play`.
- `stop`: the command targets a media shape and stops its playback.
- `stop-audio`: the command has no target and stops all running audio playback.

If the value of the `anim:command` attribute is `custom`, then there shall be a `<anim:param>` 15.6.2 child element whose `anim:name` 19.10 attribute and `anim:value` 19.12 attribute values are user-defined.

If the value of the `anim:command` attribute is `verb`, there shall be a `<anim:param>` child element whose `anim:name` attribute value is `verb` and whose `anim:value` attribute value contains the verb number.

If the value of the `anim:command` attribute is `play`, then there shall be a `<anim:param>` child element whose `anim:name` attribute value is `play` and whose `anim:value` attribute value contains a value expressed in seconds.

The `anim:command` attribute is usable with the following element: `<anim:command>` 15.6.1.  
The `anim:command` attribute has the data type `string` 18.2.

## 19.6 anim:formula

The `anim:formula` attribute specifies a formula used in an animation function.

The following is the minimum supported grammar:

```
formula ::= additive_expression

identifier ::= '$' | 'pi' | 'e' | 'x' | 'y' | 'width' | 'height'
function ::= 'abs' | 'sqrt' | 'sin' | 'cos' | 'tan' | 'atan' | 'acos' | 'asin' | 'exp' | 'log'
binary_function ::= 'min' | 'max'
basic_expression ::= number
  | identifier
  | unary_function S* '(' S* additive_expression S* ')'
  | binary_function S* '(' S* additive_expression S* ',' S* additive_expression S* ')'
  | '(' S* additive_expression S* ')'

unary_expression ::= '-' S* basic_expression | basic_expression
multiplicative_expression ::= unary_expression ( ( S* '*' S*
  unary_expression )
  | ( S* '/' S* unary_expression ) )*
additive_expression ::= multiplicative_expression ( ( S* '+' S*
  multiplicative_expression )
  | ( S* '-' S* multiplicative_expression ) )*
number ::= sign? integer | sign? Floating-point-constant
floating-point-constant ::= fractional-constant exponent? | integer
exponent
```

```

fractional-constant ::= integer? '.' integer | integer '.'
exponent ::= ( 'e' | 'E' ) sign? Integer
sign ::= '+' | '-'
integer ::= [0-9] +
S ::= (#x20 | #x9)

```

The defined identifiers for the `anim:formula` attribute are:

- `$$` (U+0024, DOLLAR SIGN): The identifier `$$` is replaced by a value between 0 and 1 (inclusive) that represents a proportional offset into an animation element's duration.
- `pi`: The Pi constant.

When the `anim:formula` attribute appears on a descendant element of `<office:presentation>`, it may use the following additional identifiers:

- `e`: the Euler constant.
- `x`: the animated elements left edge in screen space where 0 is the left edge of the screen and 1 is the right edge.
- `y`: the animated elements top edge in screen space, where 0 is the top edge of the screen and 1 is the bottom edge.
- `width`: the animated elements width in screen space, where 0 is no width and 1 is the screen's width.
- `height`: the animated elements height in screen space, where 0 is no height and 1 is the screen's height.

If an `anim:formula` attribute is given, it overrides the `smil:values`, `smil:to`, `smil:from` and `smil:by` attributes as specified in 15.3.2.

The `anim:formula` attribute is usable with the following elements: `<anim:animate>` 15.2.2, `<anim:animateColor>` 15.2.6, `<anim:animateMotion>` 15.2.5, `<anim:animateTransform>` 15.2.3 and `<anim:transitionFilter>` 15.2.7.

The `anim:formula` attribute has the data type `string` 18.2.

## 19.7 anim:id (deprecated)

The `anim:id` attribute specifies an identifier to an animation element.

OpenDocument consumers shall ignore an `anim:id` attribute if it occurs on an element with an `xml:id` attribute value.

OpenDocument producers may write `anim:id` attributes for elements in addition to an `xml:id` attribute.

The value of an `anim:id` attribute shall equal the value of an `xml:id` attribute on the same element.

The `anim:id` attribute is deprecated in favor of `xml:id`. 19.920

The `anim:id` attribute is usable with the following elements: `<anim:audio>` 15.5, `<anim:command>` 15.6.1, `<anim:iterate>` 15.4.4, `<anim:par>` 15.4.2 and `<anim:seq>` 15.4.3.

The `anim:id` attribute has the data type `NCName` 18.2.

## 19.8 anim:iterate-interval

The `anim:iterate-interval` attribute specifies the time interval between effects that are children of `<anim:iterate>` elements.

The `anim:iterate-interval` attribute is usable with the following element:

`<anim:iterate>` 15.4.4.

The `anim:iterate-interval` attribute has the data type `duration` 18.2.

## 19.9 `anim:iterate-type`

The `anim:iterate-type` attribute of a `<anim:iterate>` element specifies the iteration of child elements.

The defined values for the `anim:iterate-type` attribute are:

- `by-letter`: the target shape or paragraph is iterated by letters.
- `by-paragraph`: the target shape is iterated by paragraphs.
- `by-word`: the target shape or paragraph is iterated by words.

The `anim:iterate-type` attribute is usable with the following element: `<anim:iterate>` 15.4.4.

The `anim:iterate-type` attribute has the data type `string` 18.2.

## 19.10 `anim:name`

The `anim:name` attribute specifies the name of a parameter for an `<anim:command>` 15.6.1 element.

The `anim:name` attribute is usable with the following element: `<anim:param>` 15.6.2.

The `anim:name` attribute has the data type `string` 18.2.

## 19.11 `anim:sub-item`

The `anim:sub-item` attribute specifies components of a target element for application of animation effects.

The defined values for the `anim:sub-item` attribute are:

- `background`: animates only a shape's background and not its text.
- `text`: animates only the text of a shape.
- `whole`: animates both a shape and its text.

The `anim:sub-item` attribute is usable with the following elements: `<anim:animate>` 15.2.2, `<anim:animateColor>` 15.2.6, `<anim:animateMotion>` 15.2.5, `<anim:animateTransform>` 15.2.3, `<anim:command>` 15.6.1, `<anim:iterate>` 15.4.4, `<anim:set>` 15.2.4 and `<anim:transitionFilter>` 15.2.7.

The `anim:sub-item` attribute has the data type `string` 18.2.

## 19.12 `anim:value`

The `anim:value` attribute specifies the value of a parameter for an `<anim:command>` element.

The `anim:value` attribute is usable with the following element: `<anim:param>` 15.6.2.

The `anim:value` attribute has the data type `string` 18.2.

## 19.13 chart:attached-axis

The `chart:attached-axis` attribute specifies an axis to be used with a series. The value shall be the name of an axis as defined with the `chart:name` attribute on a `<chart:axis>` element.

The `chart:attached-axis` attribute is usable with the following element:  
`<chart:series> 11.12.`

The `chart:attached-axis` attribute has the data type `string 18.2.`

## 19.14 chart:automatic-content

The `chart:automatic-content` attribute specifies if an equation to be displayed should be calculated automatically or if text given within a `<text:p>` element should be used instead.

If no `<text:p>` element is provided, the `chart:automatic-content` attribute is assumed to be `true`.

The defined values for the `chart:automatic-content` attribute are:

- `false`: the text within a `<text:p>` element is displayed.
- `true`: an equation is calculated and displayed automatically.

The default value for this attribute is `true`.

The `chart:automatic-content` attribute is usable with the following element:  
`<chart:equation> 11.19.`

The `chart:automatic-content` attribute has the data type `boolean 18.3.3.`

## 19.15 chart:class

### 19.15.1 `<chart:chart>`, `<chart:series>`

The `chart:class` attribute specifies a chart type. The value is a namespaced token. The values for the predefined chart types are defined within the chart namespace (`urn:oasis:names:tc:opendocument:xmlns:chart:1.0`).

The term *line attributes* is used in defining `chart:class` values to refer to the attributes of the `<style:graphic-properties> 17.21` element within a chart `style:draw:stroke` `19.218`, `draw:stroke-dash 20.168` and `draw:stroke-linejoin 20.170`, `svg:stroke-color 20.408`, `svg:stroke-width 20.410`, and `svg:stroke-opacity 20.409`.

The term *fill attributes* is used in defining `chart:class` values to refer to the attributes of the `<style:graphic-properties> 17.21` element within a chart `style:draw:fill` `20.118`, `draw:fill-color 20.119`, `draw:fill-gradient-name 20.120`, `draw:gradient-step-count 20.137`, `draw:fill-hatch-name 20.121`, `draw:fill-hatch-solid 20.122`, `draw:fill-image-height 19.486`, `draw:fill-image-name 20.124`, `draw:fill-image-width 20.128`, `draw:fill-image-ref-point 20.125`, `draw:fill-image-ref-point-x 20.126`, `draw:fill-image-ref-point-y 20.127`, `draw:opacity 20.152`, `draw:opacity-name 20.153`, `draw:tile-repeat-offset 20.175` and `style:repeat 19.515`.

The defined values for the `chart:class` attribute are:

- `chart:area` –The values given by a `chart:values-cell-range-address 19.28` attribute on a `<chart:series>` element are interpreted as y-coordinates within a Cartesian coordinate system with horizontal x-axis and vertical y-axis. The accessory x-coordinates are generated from the positions in the y-value sequence starting with 1.0. The first y-value gets an x-value 1.0. The second y-value is associated with an x-value of

2.0 and so forth. The `<chart:categories> 11.10` element can be used to define labels for the x-axis.

Where `chart:three-dimensional="true" 20.65` a left-handed three-dimensional Cartesian coordinate system is used with horizontal x-axis and vertical y-axis (prior to any 3D scene transformations).

Where `chart:deep="true" 20.12` the z-coordinates are generated from the order of the series elements starting with 1.0. The first series has an associated z-coordinate of 1.0. The second series has a z-coordinate 2.0 and so forth. The series names are used as labels at the z-axis.

The area in the xy-pane between the data points and the x-axis is filled (chart:origin 20.40).

Where `chart:stacked="true" 20.59` the area between a series and the previous series is filled. The filling is done as indicated by the fill attributes of the series. A border is drawn as specified by the line attributes of the series.

- `chart:bar` – The values given by a `chart:values-cell-range-address 19.28` attribute on a `<chart:series>` element are interpreted as y-coordinates within a Cartesian coordinate system with horizontal x-axis and vertical y-axis. The accessory x-coordinates are generated from the positions in the y-value sequence starting with 1.0. The first y-value gets an x-value 1.0. The second y-value is associated with an x-value of 2.0 and so forth. The `<chart:categories> 11.10` element can be used to define labels for the x-axis. Where `chart:three-dimensional="true" 20.65` a left-handed three-dimensional Cartesian coordinate system is used with horizontal x-axis and vertical y-axis (prior to any 3D scene transformations).

Where `chart:deep="true" 20.12` the z-coordinates are generated from the order of the series elements starting with 1.0. The first series has an associated z-coordinate of 1.0. The second series has a z-coordinate 2.0 and so forth. The series names are used as labels at the z-axis.

Bars are displayed for each data point starting at the x-axis and ending at the indicated y-coordinate (see `chart:origin 20.40` for more details). In case of `chart:stacked="true" 20.59` the bar starts at the previous series instead of the x-axis. The width and exact position of the bars along the x-axis is influenced by the attributes `chart:gap-width 20.23`, `chart:group-bars-per-axis 20.24`, `chart:overlap 20.41`, `chart:percentage 20.43` and `chart:stacked 20.59`. In case of `chart:three-dimensional="false" 20.65` the bars are simple rectangles.

Where `chart:three-dimensional="true" 20.65` the shape of the bars are defined by a `chart:solid-type 20.56` attribute. A `chart:vertical 20.72` attribute indicates whether the positions of x- and y-axis should be swapped, resulting in a horizontal y-axis and vertical x-axis. The bars are filled as indicated by the fill attributes of the series and points. A border is drawn as specified by the line attributes.

- `chart:bubble` – Circles(bubbles) with different sizes are displayed in a Cartesian coordinate system with horizontal x-axis and vertical y-axis. The values given by a `chart:values-cell-range-address 19.28` attribute on a `<chart:series>` element are interpreted as relative bubble area. The values for the y-coordinates are given by the first `<chart:domain> 11.13` element. The values for the x-coordinates are given by the second `<chart:domain>` element. The bubbles are filled as indicated by the fill attributes of the series and points. A border is drawn as specified by the line attributes.
- `chart:circle` – The values given by a `chart:values-cell-range-address 19.28` attribute on a `<chart:series>` element are displayed as pie chart. For a value sequence a circle is divided into sectors, the angle of a sector represents the relative weight of a value. Only one series can be displayed with this chart type. By default, the segments are arranged counter-clockwise starting at 12 o'clock. To arrange the segments in clockwise order the attribute `chart:reverse-direction 20.51` at the y-axis is set to true. The `chart:angle-offset 20.2` attribute is used to rotate the pie. The

<chart:categories> 11.10 element at the x-axis can be used to define categories for the data points. The segments are filled as indicated by the fill attributes of the series and points. A border is drawn as specified by the line attributes.

- `chart:filled-radar` – The values given by a `chart:values-cell-range-address` 19.28 attribute on a `<chart:series>` element are interpreted as radius-coordinates within a counter-clockwise polar coordinate system. The angle axis starts at 12 o'clock. The axis element with `chart:dimension="y"` 19.18 is interpreted as radius axis. The axis element with `chart:dimension="x"` is interpreted as angle axis. The `<chart:categories>` 11.10 element can be used to define labels for the x-axis. The x-coordinates are generated from the positions in the value sequence starting with 1.0. First value gets an x-value 1.0. The second value is associated with an x-value of 2.0 and so forth.

The area surrounded by the data points is filled as indicated by the fill attributes of the series. In case of `chart:stacked="true"` 20.59 the area between a series and the previous series is filled instead. A border is drawn as specified by the line attributes.

- `chart:gantt` – The values given by a `chart:values-cell-range-address` 19.28 attribute of two `<chart:series>` (marked as being of type `chart:gantt`) are interpreted together to form the start and end values (in this order) in a Gantt chart. Start and end values are displayed at the y-axis in a Cartesian coordinate system with horizontal x-axis and vertical y-axis. The accessory x-coordinates are generated from the positions in the y-value sequence starting with 1.0. The first y-value gets an x-value 1.0. The second y-value is associated with an x-value of 2.0 and so forth. The `<chart:categories>` 11.10 element can be used to define labels for the x-axis.

For each x-coordinate, a bar is displayed, starting at the start value and ending at the end value. The fill and line attributes of the series and points specify how the bars are to be displayed. If the end value is greater than or equal to the start value the style of the first series, shall be used. If the end value is smaller than the start value, the styles of the second series shall be used.

- `chart:line` – The values given by a `chart:values-cell-range-address` 19.28 attribute on a `<chart:series>` element are interpreted as y-coordinates within a Cartesian coordinate system with horizontal x-axis and vertical y-axis.

The accessory x-coordinates are generated from the positions in the y-value sequence starting with 1.0. The first y-value gets an x-value 1.0. The second y-value is associated with an x-value of 2.0 and so forth. The `<chart:categories>` 11.10 element can be used to define labels for the x-axis.

Where `chart:three-dimensional="true"` 20.65 a left-handed three-dimensional Cartesian coordinate system is used with horizontal x-axis and vertical y-axis (prior to any 3D scene transformations).

Where `chart:deep="true"` 20.12 the z-coordinates are generated from the order of the series elements starting with 1.0. The first series has an associated z-coordinate of 1.0. The second series has a z-coordinate 2.0 and so forth. The series names are used as labels at the z-axis.

Where `chart:three-dimensional="false"` 20.65 the data points are connected with a line as specified by the line attributes of the series. By default a solid line is drawn. If symbols should be used for the data points this shall be specified with the `chart:symbol-type` 20.62 attribute. The fill attributes of the series and data points specify how the symbols are filled. By default, the symbols are filled solid with the same color as the connecting line. In case no line color is given but a fill color, the line uses the fill color.

Where `chart:three-dimensional="true"` 20.65 no symbols are shown. The data points are connected with a line that is extruded in z-direction. The extruded line is filled

as specified by the fill attributes of the series. A border is drawn as specified by the line attributes.

The attribute `chart:interpolation` 20.27 within the chart style at the plot-area specifies which type of interpolation is used while connecting the points with lines.

- `chart:radar` – The values given by a `chart:values-cell-range-address` 19.28 attribute on a `<chart:series>` element are interpreted as radius-coordinates within a counter-clockwise polar coordinate system. The angle axis starts at 12 o'clock. The axis element with `chart:dimension="y"` 19.18 is interpreted as radius axis. The axis element with `chart:dimension="x"` is interpreted as angle axis. The `<chart:categories>` 11.10 element can be used to define labels for the x-axis. The x-coordinates are generated from the positions in the value sequence starting with 1.0. First value gets an x-value 1.0. The second value is associated with an x-value of 2.0 and so forth.

The data points are connected with a line as specified by the line attributes of the series. By default a solid line is drawn. If symbols should be used for the data points this shall be specified with the `chart:symbol-type` 20.62 attribute. The fill attributes of the series and data points specify how the symbols are filled. By default, the symbols are filled solid with the same color as the connecting line. In case no line color is given but a fill color, the line uses the fill color.

- `chart:ring` – The values given by a `chart:values-cell-range-address` 19.28 attribute on a `<chart:series>` element are displayed as one ring in a doughnut chart. For a given value sequence, the ring is divided into sectors, the angle of a sector represents the relative weight of a value.

By default, segments are arranged counter-clockwise starting at 12 o'clock. To arrange the segments in clockwise order the attribute `chart:reverse-direction` 20.51 at the y-axis is set to "true". The `chart:angle-offset` 20.2 attribute is used to rotate the doughnut.

By default, the first series of class ring forms the inner ring and the last series forms the outermost ring. To arrange the rings in opposite order the attribute `chart:reverse-direction` 20.51 is set to "true" at the x-axis. The attribute `chart:hole-size` 20.25 is used to influence the inner hole size.

The `<chart:categories>` 11.10 element at the x-axis can be used to define categories for the data points. The segments are filled as specified by the fill attributes of the series and points. A border is drawn around a segment as specified by the line attributes.

- `chart:scatter` – The values given by a `chart:values-cell-range-address` 19.28 attribute on a `<chart:series>` element are interpreted as y-coordinates within a Cartesian coordinate system with horizontal x-axis and vertical y-axis. The values for the x-coordinates are given by the first `<chart:domain>` 11.13 element.

Where `chart:three-dimensional="true"` 20.65 a left-handed three-dimensional Cartesian coordinate system is used with horizontal x-axis and vertical y-axis (prior to any 3D scene transformations).

Where `chart:deep="true"` 20.12 the z-coordinates are generated from the order of the series elements starting with 1.0. The first series has an associated z-coordinate of 1.0. The second series has a z-coordinate 2.0 and so forth. The series names are used as labels at the z-axis.

Where `chart:three-dimensional="false"` the data points are connected with a line as specified by the line attributes of the series. By default a solid line is drawn. If symbols should be used for the data points this shall be specified with the `chart:symbol-type` 20.62 attribute.

The fill attributes of the series and data points specify how the symbols are filled. By default the symbols are filled solid with the same color as the connecting line. In case no line color is given but a fill color, the line uses the fill color.

Where `chart:three-dimensional="true"` 20.65 no symbols are shown. The data points are connected with a line that is extruded in z-direction. The extruded line is filled as specified by the fill attributes of the series. A border is drawn as specified by the line attributes.

The attribute `chart:interpolation` 20.27 within the chart style at the plot-area specifies which type of interpolation is used while connecting the points with lines.

- `chart:stock` – The values given by a `chart:values-cell-range-address` 19.28 attribute on up to four `<chart:series>` (marked as being of type `chart:stock`) are interpreted together to form the opening, minimum, maximum and closing (in this order) values of a stock symbol displayed at the y scale in a Cartesian coordinate system with horizontal x-axis and vertical y-axis. If only three series are given, the values are interpreted as minimum, maximum and closing values (in this order). The accessory x-coordinates are generated from the positions in the y-value sequence starting with 1.0. The first y-value gets an x-value 1.0. The second y-value is associated with an x-value of 2.0 and so forth. The `<chart:categories>` 11.10 element can be used to define labels for the x-axis.

For each x-coordinate, the minimum and maximum y-values are connected with a line. The line style is specified with element `<chart:stock-range-line>` 11.22. The `chart:japanese-candle-stick` 20.30 attribute specifies the display of opening and closing values in a stock chart. In case of `chart:japanese-candle-stick="false"` opening and closing values are displayed as left and right tick-lines in the same style as the line connecting minimum and maximum value. This is the default. Only in case of `chart:japanese-candle-stick="true"` a bar is drawn from opening to closing value. The styles to use for those bars are defined with elements `<chart:stock-gain-marker>` 11.20 and `<chart:stock-loss-marker>` 11.21.

- `chart:surface` – Depending on the size of the range specified by `chart:values-cell-range-address` 19.28 attribute of the first `<chart:series>` element one of two types of surfaces is rendered:
  - If the `chart:values-cell-range-address` 19.28 attribute of the first `<chart:series>` element specifies a range with more than one row and more than one column, the first version of a surface chart is rendered:

The value in the  $i$ th column and  $j$ th row of the cell range given by the `chart:values-cell-range-address` 19.28 attribute of the `<chart:series>` element (marked as being of type `chart:surface`) is the altitude corresponding to the x-value  $i$  and the y-value  $j$  unless `<chart:domain>` 11.13 elements are contained in the `<chart:series>` element.

If `chart:three-dimensional="true"` 20.65 a surface chart is displayed in a right-handed three-dimensional Cartesian coordinate system with horizontal x-axis (from left to right), a perspective y-axis running from the front to the right rear and a vertical z-axis (prior to any 3D scene transformation). The altitude values are displayed on the z-axis in this case. The surface is displayed as specified by the line and fill attributes of the first series. Furthermore the `<chart:data-point>` 11.14 elements of the series can be used to specify differing fill and line styles for the different altitude intervals. The surface within the first major interval (starting at lower values) is displayed with the line and fill attributes of the first data-point, within the second major interval the surface is displayed with the line and fill attributes of the second data-point and so on.

Where `chart:three-dimensional="false"` 20.65 a contour plot is displayed in a two-dimensional Cartesian coordinate system with horizontal x-axis and vertical y-axis. An axis element with `chart:dimension="z"` 19.18 is used to define the range and segmentation of the displayed altitude values. Contour lines are drawn along lines of equal altitude specified by the major interval of the z-axis. The area between the contour lines can be filled. The line and fill style is specified by the line and fill attributes of the series. Furthermore the `<chart:data-point>` 11.14 elements of

the series can be used to specify differing line and fill attributes for the different altitude intervals. The contour line and area belonging to the first major interval is displayed with the line and fill attributes of the first data-point, the contour line and area belonging to the second interval is displayed as specified by the line and fill attributes of the second data-point and so on.

In both cases two `<chart:domain>` 11.13 elements may be used to specify the y-values and x-values to be used (19.597.5). In this case the *j*th value of the `table:cell-range-address` 19.597 attribute of the first `<chart:domain>` element specifies the y-values used for the values in the *j*th row of the cell range given by the `chart:values-cell-range-address` 19.28 attribute of the `<chart:series>` element and the *i*th value of the `table:cell-range-address` attribute of the second `<chart:domain>` element specifies the x-values used for the values in the *i*th column.

The attribute `chart:deep` 20.12 shall be `false`. Multiple `<chart:series>` elements of type `chart:surface` will result in multiple surfaces being rendered.

- If the `chart:values-cell-range-address` 19.28 attribute of the first `<chart:series>` element specifies a range with a single row or with a single column the second version of a surface chart is rendered:

The values given by a `chart:values-cell-range-address` 19.28 attribute on multiple `<chart:series>` elements (marked as being of type `chart:surface`) are interpreted as altitudes at specific grid locations.

Where `chart:three-dimensional="true"` 20.65 a surface chart is displayed in a left-handed three-dimensional Cartesian coordinate system with horizontal x-axis and vertical y-axis (prior to any 3D scene transformation). The altitude values are displayed on the y-axis in this case. The attribute `chart:deep` 20.12 shall be `true`. The accessory x-coordinates are generated from the positions in the altitude -value sequence starting with 1.0. The first altitude value in each series gets an x-value 1.0. The second altitude value is associated with an x-value of 2.0 and so forth. The `<chart:categories>` 11.10 element can be used to define labels for the x-axis. The z-coordinates are generated from the order of the series elements starting with 1.0. The first series marked as being of type surface gets an associated z-coordinate of 1.0. The second series marked as being of type surface gets a z-coordinate 2.0 and so forth. The series names are used as labels at the z-axis. The surface is displayed as specified by the line and fill attributes of the first series. Furthermore the `<chart:point>` 11.14 elements of the first series can be used to specify differing fill and line styles for the different altitude intervals. The surface within the first major interval (starting at lower values) is displayed with the line and fill attributes of the first data-point, within the second major interval the surface is displayed with the line and fill attributes of the second data-point and so on.

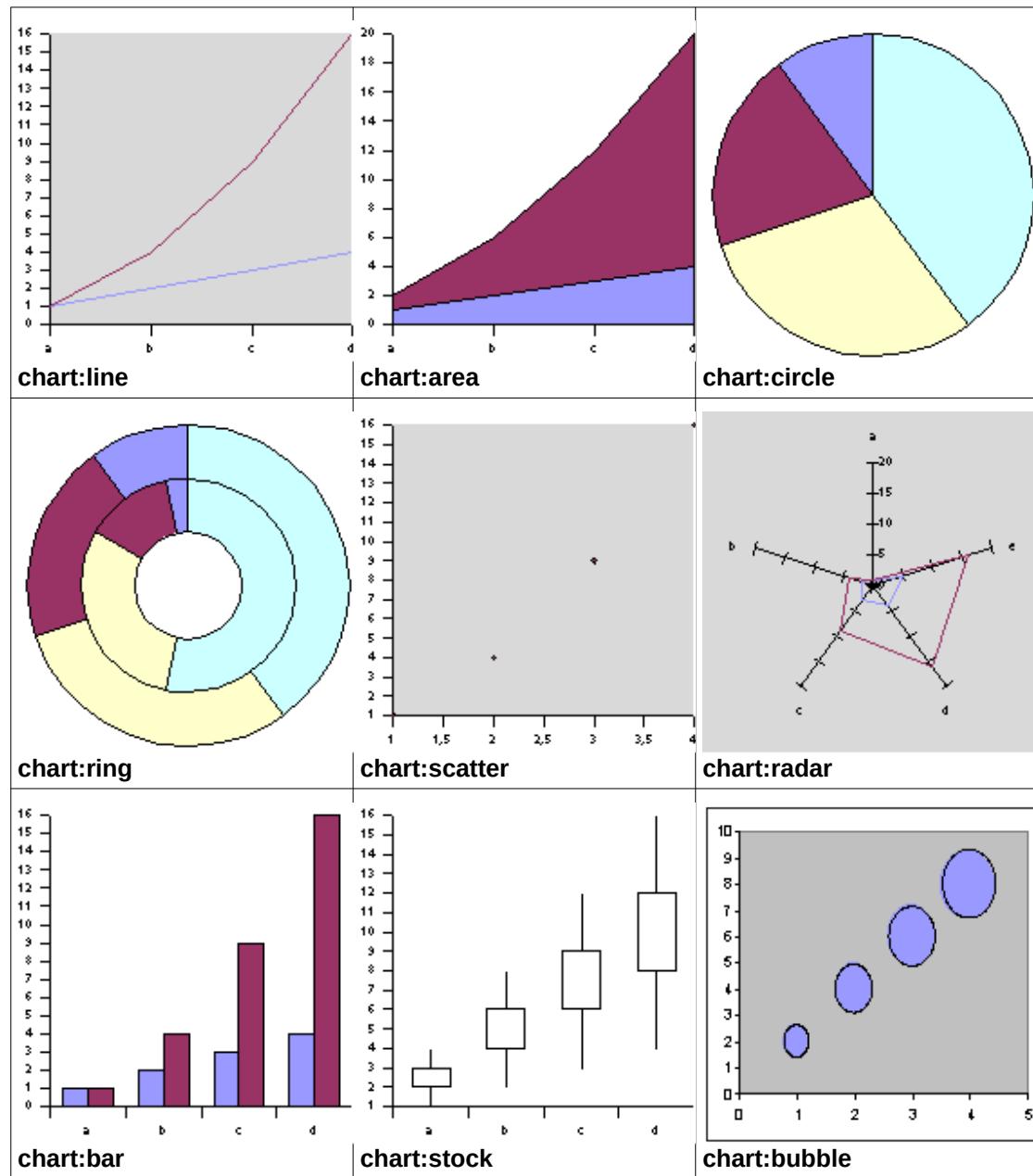
Where `chart:three-dimensional="false"` 20.65 a contour plot is displayed in a 2 dimensional Cartesian coordinate system with horizontal x-axis and vertical y-axis. The x-coordinates are generated from the positions in the altitude -value sequence starting with 1.0. The `<chart:categories>` 11.10 element can be used to define labels for the x-axis. The y-coordinates are generated from the order of the series elements starting with 1.0. The series names are used as labels at the y-axis. A third axis element with `chart:dimension="z"` 19.18 is used to define the range and segmentation of the displayed altitude values. Contour lines are drawn along lines of equal altitude specified by the major interval of the z-axis. The area between the contour lines can be filled. The line and fill style is specified by the line and fill attributes of the first series. Furthermore the `<chart:point>` 11.14 elements of the first series can be used to specify differing line and fill attributes for the different altitude intervals. The contour line and area belonging to the first major interval is displayed with the line and fill attributes of the first data-point, the contour line and area belonging to the second interval is displayed as specified by the line and fill attributes of the second data-point and so on.

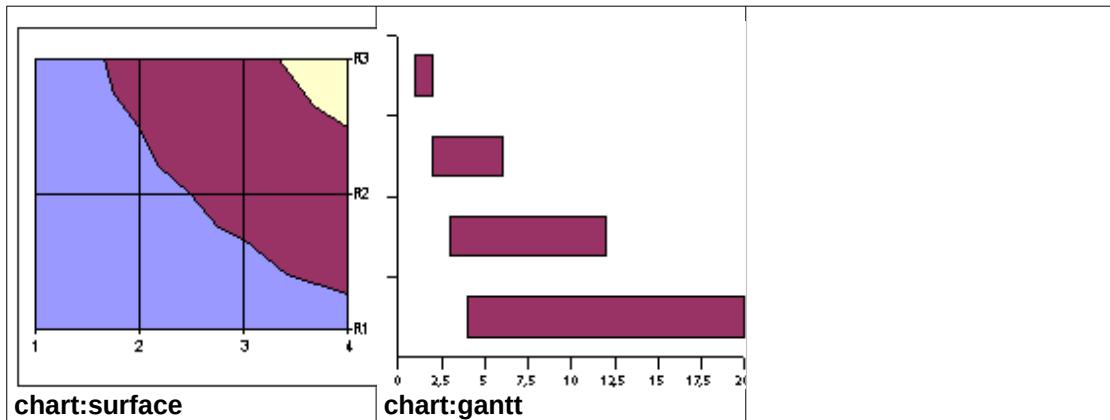
Additional chart types may be supported by using a different namespace.

A `chart:class` 19.15 attribute for a `<chart:series>` element overrides the `chart:class` attribute for the entire chart.

**Note:** The following table shows examples for the pre-defined chart types.

Those charts that use one or two data series use two data series with the values 1;2;3;4 and 1;4;9;16 and the labels a;b;c;d. Those chart types that use more than two data series (stock and bubble) use the data series 1;2;3;4 and multiples thereof. The radar chart uses two data series with five data points.





The `chart:class` attribute is usable with the following elements: `<chart:chart>` 11.1 and `<chart:series>` 11.12.

The `chart:class` attribute has the data type `namespacedToken` 18.3.20.

## 19.15.2 `<chart:grid>`

The `chart:class` attribute of a `<chart:grid>` element specifies whether major or minor tick marks are used on a grid.

The defined values for the `chart:class` attribute are:

- `major`: major tick marks are extended to grid lines.
- `minor`: minor tick marks are extended to grid lines.

For a `<chart:grid>` 11.11 element the default value for this attribute is `major`.

The `chart:class` attribute is usable with the following element: `<chart:grid>` 11.11.

The values of the `chart:class` attribute are `major` or `minor`.

## 19.16 `chart:column-mapping` (deprecated)

The `chart:column-mapping` attribute is deprecated. It specifies a list of indexes of series. The numbers define a reordering of data that comes from a container document that provides the data for the chart. The numbering begins with 1. A list of ascending numbers beginning with 1 has no effect. To exchange two series, their numbers shall be exchanged in the list.

The `chart:column-mapping` 19.16 and `chart:row-mapping` 19.26 attributes shall not be used simultaneously.

**Note:** Instead of using the deprecated `chart:column-mapping` attribute, `<chart:series>` elements may be reordered and specific cell ranges may be assigned to them using the `chart:values-cell-range-address` 19.28 attribute.

The `chart:column-mapping` attribute is usable with the following element: `<chart:chart>` 11.1.

The `chart:column-mapping` attribute has the data type `string` 18.2.

## 19.17 `chart:data-source-has-labels` (deprecated)

The `chart:data-source-has-labels` attribute specifies whether a chart's source range contain labels.

The defined values for the `chart:data-source-has-labels` attribute are:

- **both**: first row and column of a chart's source range both contain labels.
- **column**: first column of a chart's source range contains labels.
- **none**: neither the first row nor the first column of a chart's source range contains labels.
- **row**: first row of a chart's source range contains labels.

The default value for this attribute is `none`.

The `chart:data-source-has-labels` attribute is usable with the following element:  
`<chart:plot-area>` 11.5.

The values of the `chart:data-source-has-labels` attribute are `none`, `row`, `column` or `both`.

## 19.18 chart:dimension

The `chart:dimension` attribute specifies a dimension in a coordinate system.

For charts with less than three axes the `chart:dimension` attribute may appear with values for the x-axis and y-axis only.

The defined values for the `chart:dimension` attribute are:

- `x`: dimension represented by the x-axis of a chart.
- `y`: dimension represented by the y-axis of a chart.
- `z`: dimension represented by the z-axis of a chart.

The `chart:dimension` attribute is usable with the following elements: `<chart:axis>` 11.9 and `<chart:error-indicator>` 11.17.

The values of the `chart:dimension` attribute are `x`, `y` or `z`.

## 19.19 chart:display-equation

The `chart:display-equation` attribute specifies whether the equation itself should be displayed or not. It is only evaluated if the value of the `chart:automatic-content` 19.14 attribute is `true`.

The defined values for the `chart:display-equation` attribute are:

- `false`: the equation is not displayed.
- `true`: the equation is displayed.

The default value for this attribute is `true`.

The `chart:display-equation` attribute is usable with the following element:  
`<chart:equation>` 11.19.

The `chart:display-equation` attribute has the data type `boolean` 18.3.3.

## 19.20 chart:display-r-square

The `chart:display-r-square` attribute specifies whether an R-square value should be displayed or not. It is only evaluated if the value of the `chart:automatic-content` 19.14 attribute is `true`.

The defined values for the `chart:display-r-square` attribute are:

- `false`: R-square value is not displayed.
- `true`: R-square value is displayed.

The default value for this attribute is `false`.

The `chart:display-r-square` attribute is usable with the following element:  
`<chart:equation>` 11.19.

The `chart:display-r-square` attribute has the data type `boolean` 18.3.3.

## 19.21 chart:label-cell-address

The `chart:label-cell-address` attribute specifies a cell range list that contains the name for a series.

The `chart:label-cell-address` attribute is usable with the following element:  
`<chart:series>` 11.12.

The `chart:label-cell-address` attribute has the data type `cellRangeAddressList` 18.3.6.

## 19.22 chart:legend-align

The `chart:legend-align` attribute specifies the alignment of a legend with the plot area.

The defined values for the `chart:legend-align` attribute are:

- `center`: legend is aligned at the center of the plot-area.
- `end`: legend is aligned at the end of the plot-area (which is right or bottom).
- `start`: legend is aligned at the beginning of the plot-area (which is left or top).

**Note:** This attribute can be used when the `chart:legend-position` attribute has one of the following values: `start`, `end`, `top`, `bottom`.

The `chart:legend-align` attribute is usable with the following element:  
`<chart:legend>` 11.4.

The values of the `chart:legend-align` attribute are `start`, `center` or `end`.

## 19.23 chart:legend-position

The `chart:legend-position` attribute specifies the placement of a legend.

The defined values for the `chart:legend-position` attribute, to specify the location of a legend in one of the corners of a chart outside the plot area, are:

- `bottom-end`: place legend in the bottom right corner.
- `bottom-start`: place the legend in the bottom left corner.
- `top-end`: place legend in the top right corner.
- `top-start`: place legend in the top left corner.

The defined values for the `chart:legend-position` attribute, to specify the location of a legend next to the plot area, are:

- `bottom`: place legend below the plot area.
- `end`: place legend on the right side of the plot area.
- `start`: place legend on the left side of the plot area.
- `top`: place legend above the plot area.

The legend position can also be given in absolute coordinates with `svg:x` and `svg:y` attributes, as with any drawing object. If both absolute coordinates and the `legend-position` attribute are given, the `chart:legend-position` attribute shall be used.

The `chart:legend-position` attribute is usable with the following element:  
`<chart:legend>` 11.4.

## 19.24 chart:name

The `chart:name` attribute specifies a name for an axis.

**Note:** The value of a `chart:name` attribute can be used to reference an axis.

The `chart:name` attribute is usable with the following element: `<chart:axis>` 11.9.

The `chart:name` attribute has the data type `string` 18.2.

## 19.25 chart:repeated

The `chart:repeated` attribute specifies how many consecutive data points have the same style.

The `chart:repeated` attribute is usable with the following element: `<chart:datapoint>` 11.14.

The `chart:repeated` attribute has the data type `positiveInteger` 18.2.

## 19.26 chart:row-mapping (deprecated)

The `chart:row-mapping` attribute specifies a list of indexes of series. The numbers define a reordering of data that comes from a container document that provides the data for the chart. The numbering begins with 1. A list of ascending numbers beginning with 1 has no effect. To exchange two series, their numbers shall be exchanged in the list.

The `chart:row-mapping` and `chart:column-mapping` 19.16 attributes shall not be used simultaneously.

**Note:** Instead of using the deprecated `chart:row-mapping` attribute, `<chart:series>` 11.12 elements may be reordered and specific cell ranges may be assigned to them using the `chart:values-cell-range-address` 19.28 attribute.

The `chart:row-mapping` attribute is usable with the following element: `<chart:chart>` 11.1.

The `chart:row-mapping` attribute has the data type `string` 18.2.

## 19.27 chart:style-name

The `chart:style-name` attribute references a chart style by its name. 19.502

The `chart:style-name` attribute is usable with the following elements: `<chart:axis>` 11.9, `<chart:chart>` 11.1, `<chart:data-label>` 11.15, `<chart:datapoint>` 11.14, `<chart:equation>` 11.19, `<chart:error-indicator>` 11.17, `<chart:floor>` 11.8, `<chart:footer>` 11.2.3, `<chart:grid>` 11.11, `<chart:legend>` 11.4, `<chart:mean-value>` 11.16, `<chart:plot-area>` 11.5, `<chart:regression-curve>` 11.18, `<chart:series>` 11.12, `<chart:stock-gain-marker>` 11.20, `<chart:stock-loss-marker>` 11.21, `<chart:stock-range-line>` 11.22, `<chart:subtitle>` 11.2.2, `<chart:title>` 11.2.1 and `<chart:wall>` 11.7.

The `chart:style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.28 chart:values-cell-range-address

The `chart:values-cell-range-address` attribute specifies a cell range that contains the values for a data series.

The `chart:values-cell-range-address` attribute is usable with the following element:  
`<chart:series> 11.12.`

The `chart:values-cell-range-address` attribute has the data type `cellRangeAddressList` 18.3.6.

## 19.29 config:name

The `config:name` attribute specifies a name of an application setting, an application setting sequence, or an application setting container.

For `<config:config-item-set>` 3.10.2 elements that are children of `<office:settings>` 3.10 elements, the name shall begin with a namespace prefix followed by a ":" (U+003A, COLON). The XML namespace bound to the prefix identifies the implementation that defines the name of a setting.

The `config:name` attribute is usable with the following elements: `<config:config-item> 3.10.3`, `<config:config-item-map-entry> 3.10.5`, `<config:config-item-map-indexed> 3.10.4`, `<config:config-item-map-named> 3.10.6` and `<config:config-item-set> 3.10.2`.

The `config:name` attribute has the data type `string` 18.2.

## 19.30 config:type

The `config:type` attribute specifies a data type for a setting.

The defined values for the `config:type` attribute are:

- `base64Binary: 18.2`
- `boolean: 18.3.3`
- `datetime: 18.2`
- `double: 18.2`
- `int: 18.2`
- `long: 18.2`
- `short: 18.2`
- `string: 18.2`

The `config:type` attribute is usable with the following element: `<config:config-item> 3.10.3`.

The values of the `config:type` attribute are `boolean`, `short`, `int`, `long`, `double`, `string`, `datetime` or `base64Binary`.

## 19.31 db:additional-column-statement

The `db:additional-column-statement` attribute specifies the auto-increment keyword of an SQL create statement and any start value.

**Note:** The auto-increment keyword, optional start value and the format of the string specifying them are database specific.

The `db:additional-column-statement` attribute is usable with the following element:

`<db:increment>` 12.10.

The `db:additional-column-statement` attribute has the data type `string` 18.2.

## 19.32 db:append-table-alias-name

The `db:append-table-alias-name` attribute specifies that a table alias name should be appended when creating an SQL query.

The defined values for the `db:append-table-alias-name` attribute are:

- `false`: table alias name is not appended to a query when creating an SQL query.
- `true`: table alias name is appended to a query when creating an SQL query.

The default value for this attribute is `true`.

The `db:append-table-alias-name` attribute is usable with the following element:

`<db:application-connection-settings>` 12.15.

The `db:append-table-alias-name` attribute has the data type `boolean` 18.3.3.

## 19.33 db:apply-command

The `db:apply-command` attribute specifies whether an SQL clause specified by a `db:command` attribute should be appended to an underlying statement.

The defined values for the `db:apply-command` attribute are:

- `false`: an SQL clause specified by a `db:command` 19.38 will not be appended to an underlying statement.
- `true`: an SQL clause specified by a `db:command` attribute will be appended to an underlying statement.

The default value for this attribute is `false`.

The `db:apply-command` attribute is usable with the following elements: `<db:filter-statement>` 12.30 and `<db:order-statement>` 12.29.

The `db:apply-command` attribute has the data type `boolean` 18.3.3.

## 19.34 db:as-template

The `db:as-template` attribute specifies that a document is used as a template for a table structure to be created inside a database.

The defined values for the `db:as-template` attribute are:

- `false`: document is not used as a template.
- `true`: document is used as a template.

The `db:as-template` attribute is usable with the following element: `<db:component>` 12.25.5.

The `db:as-template` attribute has the data type `boolean` 18.3.3.

## 19.35 db:base-dn

The `db:base-dn` attribute specifies a base DN for data that is retrieved using LDAP. [RFC4514]

The `db:base-dn` attribute is usable with the following element: `<db:driver-settings>` 12.9.

The `db:base-dn` attribute has the data type `string` 18.2.

## 19.36 db:boolean-comparison-mode

The `db:boolean-comparison-mode` attribute specifies a comparison mode used for Boolean values in SQL SELECT statements.

The defined values for the `db:boolean-comparison-mode` attribute are:

- `equal-boolean`: “`= TRUE`” will be used for true and “`= FALSE`” for false.
- `equal-integer`: “`= 1`” will be used for true and “`= 0`” for false.
- `equal-use-only-zero`: “`NOT ($value = 0 or $value IS NULL)`” will be used for true and “`= 0`” for false.
- `is-boolean`: “`IS TRUE`” will be used for true and “`IS FALSE`” for false.

The default value for this attribute is `equal-integer`.

The `db:boolean-comparison-mode` attribute is usable with the following element: `<db:application-connection-settings>` 12.15.

The values of the `db:boolean-comparison-mode` attribute are `equal-integer`, `is-boolean`, `equal-boolean` or `equal-use-only-zero`.

## 19.37 db:catalog-name

The `db:catalog-name` attribute specifies the catalog part of a fully-qualified SQL table name or index name.

The `db:catalog-name` attribute is usable with the following elements: `<db:index>` 12.46, `<db:table-definition>` 12.38, `<db:table-representation>` 12.33 and `<db:update-table>` 12.31.

The `db:catalog-name` attribute has the data type `string` 18.2.

## 19.38 db:command

The `db:command` attribute specifies an SQL select, order or filter clause that is used as part of a SELECT statement.

The `db:command` attribute is usable with the following elements: `<db:filter-statement>` 12.30, `<db:order-statement>` 12.29 and `<db:query>` 12.28.

The `db:command` attribute has the data type `string` 18.2.

## 19.39 db:data-source-setting-is-list

The `db:data-source-setting-is-list` attribute specifies whether a `<db:data-source-setting>` element contains a list of `<db:data-source-setting-value>` elements.

The defined values for the `db:data-source-setting-is-list` attribute are:

- `false`: `<db:data-source-settings>` 12.22 element should not contain a list of `<db:data-source-setting-value>` 12.24 elements.
- `true`: `<db:data-source-settings>` element should contain a list of `<db:data-source-setting-value>` elements.

The `db:data-source-setting-is-list` attribute is usable with the following element:

`<db:data-source-setting> 12.23.`

The `db:data-source-setting-is-list` attribute has the data type `boolean` 18.3.3.

## 19.40 db:data-source-setting-name

The `db:data-source-setting-name` attribute specifies a name for a `<db:data-source-setting>` element.

The `db:data-source-setting-name` attribute is usable with the following element:  
`<db:data-source-setting> 12.23.`

The `db:data-source-setting-name` attribute has the data type `string` 18.2.

## 19.41 db:data-source-setting-type

The `db:data-source-setting-type` attribute specifies a data type for content of a `<db:data-source-setting-value>` 12.24 element.

The defined values for the `db:data-source-setting-type` attribute are:

- `boolean`: 18.3.3
- `double`: 18.2
- `int`: 18.2
- `long`: defined by [SQL].
- `string`: 18.2

The `db:data-source-setting-type` attribute is usable with the following element:

`<db:data-source-setting> 12.23.`

The values of the `db:data-source-setting-type` attribute are `boolean`, `short`, `int`, `long`, `double` or `string`.

## 19.42 db:data-type

The `db:data-type` attribute specifies an SQL data type for a column.

The values accepted by `db:data-type` are defined by [SQL].

The `db:data-type` attribute is usable with the following element: `<db:column-definition> 12.40.`

The values of the `db:data-type` attribute are `bit`, `boolean`, `tinyint`, `smallint`, `integer`, `bigint`, `float`, `real`, `double`, `numeric`, `decimal`, `char`, `varchar`, `longvarchar`, `date`, `time`, `timestamp`, `binary`, `varbinary`, `longvarbinary`, `sqlnull`, `other`, `object`, `distinct`, `struct`, `array`, `blob`, `clob` or `ref`.

## 19.43 db:database-name

The `db:database-name` attribute specifies the database for a connection, if the database type supports multiple logical databases per server.

The `db:database-name` attribute is usable with the following element: `<db:server-database> 12.6.`

The `db:database-name` attribute has the data type `string` 18.2.

## 19.44 db:decimal

The `db:decimal` attribute specifies a decimal separator for numeric fields.

The default value for this attribute is “.” (U+002E, FULL STOP).

The `db:decimal` attribute is usable with the following element: `<db:delimiter>` 12.11.

The `db:decimal` attribute has the data type `string` 18.2.

## 19.45 db:default-cell-style-name

The `db:default-cell-style-name` attribute specifies a default cell style. This style is applied to all cells in a column.

The `db:default-cell-style-name` attribute is usable with the following element: `<db:column>` 12.35.

The `db:default-cell-style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.46 db:default-row-style-name

The `db:default-row-style-name` attribute specifies a default row style. This style is applied to all rows in a table.

The `db:default-row-style-name` attribute is usable with the following elements: `<db:query>` 12.28 and `<db:table-representation>` 12.33.

The `db:default-row-style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.47 db:delete-rule

The `db:delete-rule` attribute specifies a rule that is applied for deleting records from a table. It is used only with foreign keys.

The defined values for the `db:delete-rule` attribute are:

- `cascade`: when the primary key is deleted, rows that imported that key are deleted.
- `no-action`: indicates that if the primary key has been imported, it cannot be deleted.
- `restrict`: a primary key shall not be deleted if it has been imported by another table as a foreign key.
- `set-default`: when the primary key is deleted, the foreign key (imported key) is set to the default value.
- `set-null`: when the primary key is deleted, the foreign key (imported key) is changed to `NULL`.

The default value for this attribute is `no-action`.

The `db:delete-rule` attribute is usable with the following element: `<db:key>` 12.42.

The values of the `db:delete-rule` attribute are `cascade`, `restrict`, `set-null`, `no-action` or `set-default`.

## 19.48 db:description

The `db:description` attribute specifies a description of a database object.

The `db:description` attribute is usable with the following elements: `<db:column>` 12.35, `<db:component>` 12.25.5, `<db:component-collection>` 12.25.4, `<db:query>` 12.28, `<db:query-collection>` 12.27 and `<db:table-representation>` 12.33.

The `db:description` attribute has the data type [string 18.2](#).

## 19.49 db:enable-sql92-check

The `db:enable-sql92-check` attribute specifies whether the names of tables, views, columns, and queries may contain characters defined by the [SQL] feature F392.

The defined values for the `db:enable-sql92-check` attribute are:

- `false`: consumers should permit characters defined by the [SQL] feature F392 for new or changed names of tables, views, columns, and queries.
- `true`: consumers should not permit characters defined by the [SQL] feature F392 for new or changed names of tables, views, columns, and queries.

The default value for this attribute is `false`.

The `db:enable-sql92-check` attribute is usable with the following element:  
`<db:application-connection-settings>` [12.15](#).

The `db:enable-sql92-check` attribute has the data type [boolean 18.3.3](#).

## 19.50 db:encoding

The `db:encoding` attribute specifies a text encoding for string data.

The default value for this attribute is `utf-8`.

The `db:encoding` attribute is usable with the following element: `<db:character-set>` [12.12](#).

The `db:encoding` attribute has the data type [textEncoding 18.3.36](#).

## 19.51 db:escape-processing

The `db:escape-processing` attribute specifies whether escape processing for a query is used or not.

The defined values for the `db:escape-processing` attribute are:

- `false`: consumer should pass a query statement to the database driver without interpreting it itself.
- `true`: consumer may pass a query statement to a database driver after interpreting it itself.

The default value for this attribute is `true`.

The `db:escape-processing` attribute is usable with the following element: `<db:query>` [12.28](#).

The `db:escape-processing` attribute has the data type [boolean 18.3.3](#).

## 19.52 db:extension

The `db:extension` attribute specifies a file name extension for files which are used as tables.

The `db:extension` attribute is usable with the following element: `<db:file-based-database>` [12.5](#).

The `db:extension` attribute has the data type [string 18.2](#).

## 19.53 db:field

The `db:field` attribute specifies a separator for database fields.

The default value for this attribute is ";" (U+003B, SEMICOLON).

The `db:field` attribute is usable with the following element: `<db:delimiter>` 12.11.

The `db:field` attribute has the data type `string` 18.2.

## 19.54 db:hostname

The `db:hostname` attribute specifies the name of a server for a server-based database.

The `db:hostname` attribute is usable with the following element: `<db:server-database>` 12.6.

The `db:hostname` attribute has the data type `string` 18.2.

## 19.55 db:ignore-driver-privileges

The `db:ignore-driver-privileges` attribute specifies whether privileges returned by a database driver should be ignored.

The defined values for the `db:ignore-driver-privileges` attribute are:

- `false`: privileges returned by a database driver should not be ignored.
- `true`: privileges returned by a database driver should be ignored.

The default value for this attribute is `true`.

The `db:ignore-driver-privileges` attribute is usable with the following element: `<db:application-connection-settings>` 12.15.

The `db:ignore-driver-privileges` attribute has the data type `boolean` 18.3.3.

## 19.56 db:is-ascending

The `db:is-ascending` attribute specifies that a column is to be sorted in ascending order.

The defined values for the `db:is-ascending` attribute are:

- `false`: column is not sorted in ascending order.
- `true`: column is sorted in ascending order.

The `db:is-ascending` attribute is usable with the following element: `<db:index-column>` 12.48.

The `db:is-ascending` attribute has the data type `boolean` 18.3.3.

## 19.57 db:is-autoincrement

The `db:is-autoincrement` attribute specifies whether a column is automatically numbered.

The defined values for the `db:is-autoincrement` attribute are:

- `false`: column is not automatically numbered.
- `true`: column is automatically numbered.

The `db:is-autoincrement` attribute is usable with the following element: `<db:column-definition>` 12.40.

The `db:is-autoincrement` attribute has the data type `boolean` 18.3.3.

## 19.58 db:is-clustered

The db:is-clustered attribute specifies that an index is clustered.

The defined values for the db:is-clustered attribute are:

- false: index is not clustered.
- true: index is clustered.

The db:is-clustered attribute is usable with the following element: <db:index> 12.46.

The db:is-clustered attribute has the data type boolean 18.3.3.

## 19.59 db:is-empty-allowed

The db:is-empty-allowed attribute specifies whether a column may contain empty values.

The defined values for the db:is-empty-allowed attribute are:

- false: column shall not contain empty values.
- true: column may contain empty values.

The db:is-empty-allowed attribute is usable with the following element: <db:column-definition> 12.40.

The db:is-empty-allowed attribute has the data type boolean 18.3.3.

## 19.60 db:is-first-row-header-line

The db:is-first-row-header-line attribute specifies whether the first row in a text file, which is interpreted as a table of a file-based database, is used to define the columns of a table.

The defined values for the db:is-first-row-header-line attribute are:

- false: first row of text file does not define columns of a table.
- true: first row of text file does define columns of a table.

The default value for this attribute is true.

The db:is-first-row-header-line attribute is usable with the following elements: <db:driver-settings> 12.9 and <db:table-setting> 12.14.

The db:is-first-row-header-line attribute has the data type boolean 18.3.3.

## 19.61 db:login-timeout

The db:login-timeout attribute indicates a default time-out to be used when establishing a connection for this data source. If the attribute is not present, then a default time-out is specified by the database server.

The db:login-timeout attribute is usable with the following element: <db:login> 12.8.

The db:login-timeout attribute has the data type positiveInteger 18.2.

## 19.62 db:is-nullable

The db:is-nullable attribute specifies the nullability of values in a column.

The defined values for the db:is-nullable attribute are:

- no-nulls: null values should not appear in a column.

- `nullable`: null values may appear in a column.

The `db:is-nullable` attribute is usable with the following element: `<db:column-definition>` 12.40.

The values of the `db:is-nullable` attribute are `no-nulls` or `nullable`.

## 19.63 db:is-password-required

The `db:is-password-required` attribute specifies if a password is always necessary when establishing a connection for a data source.

The defined values for the `db:is-password-required` attribute are:

- `false`: password should not be required for a connection to data source.
- `true`: password should be required for a connection to data source.

The default value for this attribute is `false`.

The `db:is-password-required` attribute is usable with the following element: `<db:login>` 12.8.

The `db:is-password-required` attribute has the data type `boolean` 18.3.3.

## 19.64 db:is-unique

The `db:is-unique` attribute specifies whether an index shall have unique values.

The defined values for the `db:is-unique` attribute are:

- `false`: index should not require unique values.
- `true`: index shall have unique values.

The `db:is-unique` attribute is usable with the following element: `<db:index>` 12.46.

The `db:is-unique` attribute has the data type `boolean` 18.3.3.

## 19.65 db:precision

The `db:precision` attribute specifies a column's number of decimal digits.

The `db:precision` attribute is usable with the following element: `<db:column-definition>` 12.40.

The `db:precision` attribute has the data type `positiveInteger` 18.2.

## 19.66 db:is-table-name-length-limited

The `db:is-table-name-length-limited` attribute specifies whether table names are limited to eight characters or fewer characters.

The defined values for the `db:is-table-name-length-limited` attribute are:

- `false`: table names may exceed eight characters in length.
- `true`: table names shall not be greater than eight characters in length.

The default value for this attribute is `true`.

The `db:is-table-name-length-limited` attribute is usable with the following element: `<db:application-connection-settings>` 12.15.

The `db:is-table-name-length-limited` attribute has the data type `boolean` 18.3.3.

## 19.67 db:local-socket

The `db:local-socket` attribute specifies the local socket on which a server database is running and accepting connections. The server's local socket filename is used for connecting to a server database when it is running locally.

If `db:local-socket` is present but empty, consumers should use the default local socket for the database type in question.

The `db:local-socket` attribute is usable with the following element: `<db:server-database>` 12.6.

The `db:local-socket` attribute has the data type `string` 18.2.

## 19.68 db:max-row-count

The `db:max-row-count` attribute specifies the maximum number of rows that should be returned in a result set.

The `db:max-row-count` attribute is usable with the following element: `<db:application-connection-settings>` 12.15.

The `db:max-row-count` attribute has the data type `integer` 18.2.

## 19.69 db:media-type

The `db:media-type` attribute of a `<db:file-based-database>` element specifies the type of the file(s) that comprise a database. If the database consists of multiple files of different types which all are required for it to be operational, this attribute specifies the type of the file linked to by the `xlink:href` 19.916 attribute on the `<db:file-based-database>` element.

The `db:media-type` attribute is usable with the following element: `<db:file-based-database>` 12.5.

The `db:media-type` attribute has the data type `string` 18.2.

## 19.70 db:name

### 19.70.1 General

The `db:name` attribute specifies a name of a database or component of a database.

### 19.70.2 <db:column>

The `db:name` attribute specifies the name of a column.

The `db:name` attribute is usable with the following element: `<db:column>` 12.35.

The `db:name` attribute has the data type `string` 18.2.

### 19.70.3 <db:column-definition>

The `db:name` attribute specifies the name of a column.

The `db:name` attribute is usable with the following element: `<db:column-definition>` 12.40.

The `db:name` attribute has the data type `string` 18.2.

#### 19.70.4 <db:component>

The `db:name` attribute specifies the name of a database component.

The `db:name` attribute is usable with the following element: `<db:component>` 12.25.5.

The `db:name` attribute has the data type `string` 18.2.

#### 19.70.5 <db:component-collection>

The `db:name` attribute specifies the name of a collection of `<db:component>` 12.25.5 and `<db:component-collection>` elements.

The `db:name` attribute is usable with the following element: `<db:component-collection>` 12.25.4.

The `db:name` attribute has the data type `string` 18.2.

#### 19.70.6 <db:index>

The `db:name` attribute specifies the name of an index.

The `db:name` attribute is usable with the following element: `<db:index>` 12.46.

The `db:name` attribute has the data type `string` 18.2.

#### 19.70.7 <db:index-column>

The `db:name` attribute specifies the name of an index column.

The `db:name` attribute is usable with the following element: `<db:index-column>` 12.48.

The `db:name` attribute has the data type `string` 18.2.

#### 19.70.8 <db:key>

The `db:name` attribute specifies the name of a key. If the key is not a primary key, the `db:name` attribute with a value shall appear on the `<db:key>` element.

The `db:name` attribute is usable with the following element: `<db:key>` 12.42.

The `db:name` attribute has the data type `string` 18.2.

#### 19.70.9 <db:key-column>

The `db:name` attribute specifies the name of a `<db:key-column>` element.

The `db:name` attribute is usable with the following element: `<db:key-column>` 12.44.

The `db:name` attribute has the data type `string` 18.2.

#### 19.70.10 <db:query>

The `db:name` attribute specifies the name of a `<db:query>` element.

The `db:name` attribute is usable with the following element: `<db:query>` 12.28.

The `db:name` attribute has the data type `string` 18.2.

#### 19.70.11 <db:query-collection>

The `db:name` attribute specifies the name of a collection of `<db:query>` 12.28 and `<db:query-collection>` elements.

The `db:name` attribute is usable with the following element: `<db:query-collection>` 12.27.

The `db:name` attribute has the data type `string` 18.2.

### 19.70.12 `<db:table-definition>`

The `db:name` attribute specifies the name of a schema for a table.

The `db:name` attribute is usable with the following element: `<db:table-definition>` 12.38.

The `db:name` attribute has the data type `string` 18.2.

### 19.70.13 `<db:table-representation>`

The `db:name` attribute specifies the name of a table. Fully identifying a table in a database may require values from the `db:catalog-name` and `db:schema-name` attributes.

The `db:name` attribute is usable with the following element: `<db:table-representation>` 12.33.

The `db:name` attribute has the data type `string` 18.2.

### 19.70.14 `<db:update-table>`

The `db:name` attribute specifies the name of a `<db:update-table>` element.

The `db:name` attribute is usable with the following element: `<db:update-table>` 12.31.

The `db:name` attribute has the data type `string` 18.2.

## 19.71 `db:parameter-name-substitution`

The `db:parameter-name-substitution` attribute specifies that parameter names shall be replaced with a "?" (U+003F, QUESTION MARK).

The defined values for the `db:parameter-name-substitution` attribute are:

- `false`: parameters names may be replaced by a "?".
- `true`: parameters names shall be replaced by a "?".

The default value for this attribute is `true`.

The `db:parameter-name-substitution` attribute is usable with the following element: `<db:driver-settings>` 12.9.

The `db:parameter-name-substitution` attribute has the data type `boolean` 18.3.3.

## 19.72 `db:port`

The `db:port` attribute specifies the port on which a server database is accepting connections. If no port is present, consumers should use the default port for the database type in question.

The `db:port` attribute is usable with the following element: `<db:server-database>` 12.6.

The `db:port` attribute has the data type `positiveInteger` 18.2.

## 19.73 `db:referenced-table-name`

The `db:referenced-table-name` attribute specifies the name of a referenced table. It is only used with foreign keys.

The `db:referenced-table-name` attribute is usable with the following element:  
`<db:key> 12.42.`

The `db:referenced-table-name` attribute has the data type `string 18.2.`

## 19.74 db:related-column-name

The `db:related-column-name` attribute specifies the name of a reference column out of the referenced table. It is used only with foreign keys.

The `db:related-column-name` attribute is usable with the following element: `<db:key-column> 12.44.`

The `db:related-column-name` attribute has the data type `string 18.2.`

## 19.75 db:row-retrieving-statement

The `db:row-retrieving-statement` attribute specifies an SQL statement, which is executed to retrieve auto generated values after inserting a new row.

The `db:row-retrieving-statement` attribute is usable with the following element:  
`<db:auto-increment> 12.10.`

The `db:row-retrieving-statement` attribute has the data type `string 18.2.`

## 19.76 db:scale

The `db:scale` attribute specifies a column's number of digits to the right of the decimal point.

The `db:scale` attribute is usable with the following element: `<db:column-definition> 12.40.`

The `db:scale` attribute has the data type `positiveInteger 18.2.`

## 19.77 db:schema-name

The `db:schema-name` attribute specifies the name of a schema.

The `db:schema-name` attribute is usable with the following elements: `<db:table-definition> 12.38`, `<db:table-representation> 12.33` and `<db:update-table> 12.31.`

The `db:schema-name` attribute has the data type `string 18.2.`

## 19.78 db:show-deleted

The `db:show-deleted` attribute specifies that deleted records should be included in result sets.

The defined values for the `db:show-deleted` attribute are:

- `false`: deleted records not included in result sets.
- `true`: deleted records included in result sets.

The default value for this attribute is `false`.

The `db:show-deleted` attribute is usable with the following elements: `<db:driver-settings> 12.9` and `<db:table-setting> 12.14.`

The `db:show-deleted` attribute has the data type `boolean 18.3.3.`

## 19.79 db:string

The `db:string` attribute specifies a separator for strings.

The default value for this attribute is “.” (U+002E, FULL STOP).

The `db:string` attribute is usable with the following element: `<db:delimiter>` 12.11.

The `db:string` attribute has the data type `string` 18.2.

## 19.80 db:style-name

### 19.80.1 General

The `db:style-name` attribute specifies a style name of a style family for the element upon which it appears.

### 19.80.2 <db:column>

The `db:style-name` attribute specifies the name of a style of the family `table-column`.

The `db:style-name` attribute is usable with the following element: `<db:column>` 12.35.

The `db:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.80.3 <db:query>

The `db:style-name` attribute specifies the name of a style of the family `table`.

The `db:style-name` attribute is usable with the following element: `<db:query>` 12.28.

The `db:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.80.4 <db:table-representation>

The `db:style-name` attribute specifies the name of a style of the family `table`.

The `db:style-name` attribute is usable with the following element: `<db:table-representation>` 12.33.

The `db:style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.81 db:suppress-version-columns

The `db:suppress-version-columns` attribute specifies that version columns, columns that are used for versioning row content, should be suppressed.

The defined values for the `db:suppress-version-columns` attribute are:

- `false`: version columns are not suppressed.
- `true`: version columns should be suppressed.

The default value for this attribute is `true`.

The `db:suppress-version-columns` attribute is usable with the following element: `<db:application-connection-settings>` 12.15.

The `db:suppress-version-columns` attribute has the data type `boolean` 18.3.3.

## 19.82 db:system-driver-settings

The `db:system-driver-settings` attribute specifies system-specific options for a connection.

The `db:system-driver-settings` attribute is usable with the following element:

`<db:driver-settings>` 12.9.

The `db:system-driver-settings` attribute has the data type `string` 18.2.

## 19.83 db:thousand

The `db:thousand` attribute specifies a thousand separator for numeric fields.

The default value for this attribute is “,” (U+002C, COMMA).

The `db:thousand` attribute is usable with the following element: `<db:delimiter>` 12.11.

The `db:thousand` attribute has the data type `string` 18.2.

## 19.84 db:title

The `db:title` attribute specifies a human-readable title of an object.

The `db:title` attribute is usable with the following elements: `<db:column>` 12.35, `<db:component>` 12.25.5, `<db:component-collection>` 12.25.4, `<db:query>` 12.28, `<db:query-collection>` 12.27 and `<db:table-representation>` 12.33.

The `db:title` attribute has the data type `string` 18.2.

## 19.85 db:type

### 19.85.1 General

The `db:type` attribute specifies a type for keys, databases, or tables, depending upon the element where it appears.

### 19.85.2 <db:key>

The `db:type` attribute specifies the type of a key.

The defined values for the `db:type` attribute are:

- `foreign`: a foreign key is a tuple of values in a database record that points to a key of a record in another table.
- `primary`: a primary key is a tuple of values that can be used to identify a row in a table.
- `unique`: a unique key specifies that the value of the key is unique but may be the NULL value.

The `db:type` attribute is usable with the following element: `<db:key>` 12.42.

The values of the `db:type` attribute are `primary`, `unique` or `foreign`.

### 19.85.3 <db:server-database>

The `db:type` attribute specifies the type of a database.

The `db:type` attribute is usable with the following element: `<db:server-database>` 12.6.

The `db:type` attribute has the data type `namespacedToken` 18.3.20.

### 19.85.4 <db:table-definition>

The `db:type` attribute specifies the type of a table.

The defined values for the `db:type` attribute are:

- `system-table`: a table used by a database for administration.
- `table`: a table.
- `view`: a table that is a view of another table.
- Implementation-defined values

The `db:type` attribute is usable with the following element: `<db:table-definition>` 12.38.

The `db:type` attribute has the data type `string` 18.2.

## 19.86 db:type-name

The `db:type-name` attribute specifies a database-dependent type name of a database column.

The `db:type-name` attribute is usable with the following element: `<db:column-definition>` 12.40.

The `db:type-name` attribute has the data type `string` 18.2.

## 19.87 db:update-rule

The `db:update-rule` attribute specifies a rule that is applied for updating records in a table. It is only used for foreign keys.

The defined values for the `db:update-rule` attribute are:

- `cascade`: when the primary key is updated, the foreign key (imported key) is changed to the same value.
- `restrict`: a primary key shall not be updated if it has been imported by another table as a foreign key.
- `no-action`: if the primary key has been imported, it cannot be updated.
- `set-default`: when the primary key is updated, the foreign key (imported key) is set to the default value.
- `set-null`: when the primary key is updated, the foreign key (imported key) is changed to `NULL`.

The default value for this attribute is `no-action`.

The `db:update-rule` attribute is usable with the following element: `<db:key>` 12.42.

The values of the `db:update-rule` attribute are `cascade`, `restrict`, `set-null`, `no-action` or `set-default`.

## 19.88 db:use-catalog

The `db:use-catalog` attribute specifies that a catalog name should be used to reference a table in SQL statements.

The defined values for the `db:use-catalog` attribute are:

- `false`: catalog names are not used to reference tables in SQL statements.
- `true`: catalog names are used to reference tables in SQL statements.

The default value for this attribute is `false`.

The `db:use-catalog` attribute is usable with the following element: `<db:application-connection-settings>` 12.15.

The `db:use-catalog` attribute has the data type [boolean 18.3.3](#).

## 19.89 `db:use-system-user`

The `db:use-system-user` attribute specifies whether the name of a user as logged in to the client operating system is used.

If `db:use-system-user` and `db:user-name` [19.90](#) attributes are omitted and the server requires a user name, the user is prompted to enter a name.

The defined values for the `db:use-system-user` attribute are:

- `false`: name of user logged into client operating system is not used for login.
- `true`: name of user logged into client operating system is used for login.

The `db:use-system-user` attribute is usable with the following element: `<db:login>` [12.8](#).

The `db:use-system-user` attribute has the data type [boolean 18.3.3](#).

## 19.90 `db:user-name`

The `db:user-name` attribute specifies a user name for authentication upon creating a connection to a database.

If `db:user-name` and `db:use-system-user` [19.89](#) attributes are omitted and the server requires a user name, the user is prompted to enter a name.

The `db:user-name` attribute is usable with the following element: `<db:login>` [12.8](#).

The `db:user-name` attribute has the data type [string 18.2](#).

## 19.91 `db:visible`

The `db:visible` attribute specifies whether a column appears in a table grid view.

The defined values for the `db:visible` attribute are:

- `false`: column does not appear in a table grid view.
- `true`: column appears in a table grid view.

The default value for this attribute is `true`.

The `db:visible` attribute is usable with the following element: `<db:column>` [12.35](#).

The `db:visible` attribute has the data type [boolean 18.3.3](#).

## 19.92 `dr3d:ambient-color`

The `dr3d:ambient-color` attribute specifies a color for ambient light.

**Note:** Ambient light is light that seems to come from all directions.

The `dr3d:ambient-color` attribute, along with `dr3d:diffuse-color`, `dr3d:emissive-color`, and `dr3d:specular-color` specifies the four colors that define the color of a material.

The `dr3d:ambient-color` attribute is usable with the following elements: `<chart:plot-area>` [11.5](#) and `<dr3d:scene>` [10.5.2](#).

The `dr3d:ambient-color` attribute has the data type [color 18.3.9](#).

## 19.93 dr3d:center

The dr3d:center attribute specifies the center of a sphere in a three-dimensional space.

The dr3d:center attribute is usable with the following element: <dr3d:sphere> 10.5.5.

The dr3d:center attribute has the data type vector3D 18.3.41.

## 19.94 dr3d:diffuse-color

The dr3d:diffuse-color attribute specifies the base color that a light is emitting.

The dr3d:diffuse-color attribute is usable with the following element: <dr3d:light> 10.5.3.

The dr3d:diffuse-color attribute has the data type color 18.3.9.

## 19.95 dr3d:direction

The dr3d:direction attribute specifies the direction in which light is emitted.

The dr3d:direction attribute is usable with the following element: <dr3d:light> 10.5.3.

The dr3d:direction attribute has the data type vector3D 18.3.41.

## 19.96 dr3d:distance

The dr3d:distance attribute specifies the distance between a camera and an object.

The dr3d:distance attribute is usable with the following elements: <chart:plot-area> 11.5 and <dr3d:scene> 10.5.2.

The dr3d:distance attribute has the data type length 18.3.19.

## 19.97 dr3d:enabled

The dr3d:enabled attribute specifies if a light is enabled. If a light is not enabled, it does not emit any light.

The defined values for the dr3d:enabled attribute are:

- false: light is not enabled.
- true: light is enabled.

The dr3d:enabled attribute is usable with the following element: <dr3d:light> 10.5.3.

The dr3d:enabled attribute has the data type boolean 18.3.3.

## 19.98 dr3d:focal-length

The dr3d:focal-length attribute specifies the focal length of a virtual camera in a scene.

The dr3d:focal-length attribute is usable with the following elements: <chart:plot-area> 11.5 and <dr3d:scene> 10.5.2.

The dr3d:focal-length attribute has the data type length 18.3.19.

## 19.99 dr3d:lighting-mode

### 19.99.1 General

The `dr3d:lighting-mode` attribute specifies the use of lighting in three-dimensional scenes.

**Note:** The `dr3d:lighting-mode` attribute is also defined for 3D objects.  
20.86

### 19.99.2 <chart:plot-area>

The `dr3d:lighting-mode` attribute specifies whether lighting is used in a three-dimensional chart.

The defined values for the `dr3d:lighting-mode` attribute are:

- `false`: lighting is not used.
- `true`: lighting is used.

The `dr3d:lighting-mode` attribute is usable with the following element: `<chart:plot-area>` 11.5.

The `dr3d:lighting-mode` attribute has the data type `boolean` 18.3.3.

### 19.99.3 <dr3d:scene>

The `dr3d:lighting-mode` attribute specifies whether lighting is used in a three-dimensional scene.

The defined values for the `dr3d:lighting-mode` attribute are:

- `false`: lighting is not used.
- `true`: lighting is used.

The `dr3d:lighting-mode` attribute is usable with the following element: `<dr3d:scene>` 10.5.2.

The `dr3d:lighting-mode` attribute has the data type `boolean` 18.3.3.

## 19.100 dr3d:max-edge

The `dr3d:max-edge` attribute specifies the maximum x, y and z coordinate values for a cube.

The `dr3d:max-edge` attribute is usable with the following element: `<dr3d:cube>` 10.5.4.

The `dr3d:max-edge` attribute has the data type `vector3D` 18.3.41.

## 19.101 dr3d:min-edge

The `dr3d:min-edge` attribute specifies the minimum x, y and z coordinate values for a cube.

The `dr3d:min-edge` attribute is usable with the following element: `<dr3d:cube>` 10.5.4.

The `dr3d:min-edge` attribute has the data type `vector3D` 18.3.41.

## 19.102 dr3d:projection

The `dr3d:projection` attribute specifies a projection mode.

The defined values for the `dr3d:projection` attribute are:

- **parallel**: distance from the center of projection to the projection plane is infinite.
- **perspective**: distance from the center of projection to the projection plane is finite.

For a `<draw:enhanced-geometry>` 10.6.2 element the default value for this attribute is **parallel**.

The `dr3d:projection` attribute is usable with the following elements: `<chart:plot-area>` 11.5, `<dr3d:scene>` 10.5.2 and `<draw:enhanced-geometry>` 10.6.2.

The values of the `dr3d:projection` attribute are **parallel** or **perspective**.

## 19.103 dr3d:shade-mode

The `dr3d:shade-mode` attribute specifies how lighting is calculated for rendered surfaces.

The defined values for the `dr3d:shade-mode` attribute are:

- **draft**: surfaces are not lit and drawn as wireframe only.
- **flat**: lighting is calculated by one surface normal.
- **gouraud**: lighting is calculated by interpolating the color calculated with the surface normals at each edge.
- **phong**: lighting is calculated by interpolating the surface normals over the surface.

For a `<draw:enhanced-geometry>` 10.6.2 element the default value for this attribute is **flat**.

The `dr3d:shade-mode` attribute is usable with the following elements: `<chart:plot-area>` 11.5, `<dr3d:scene>` 10.5.2 and `<draw:enhanced-geometry>` 10.6.2.

The values of the `dr3d:shade-mode` attribute are **flat**, **phong**, **gouraud** or **draft**.

## 19.104 dr3d:shadow-slant

The `dr3d:shadow-slant` attribute specifies an angle from a three-dimensional scene to a virtual paper on which a shadow is cast.

The `dr3d:shadow-slant` attribute is usable with the following elements: `<chart:plot-area>` 11.5 and `<dr3d:scene>` 10.5.2.

The `dr3d:shadow-slant` attribute has the data type **angle** 18.3.1.

## 19.105 dr3d:size

The `dr3d:size` attribute specifies the size of a sphere in a three-dimensional space.

The `dr3d:size` attribute is usable with the following element: `<dr3d:sphere>` 10.5.5.

The `dr3d:size` attribute has the data type **vector3D** 18.3.41.

## 19.106 dr3d:specular

The `dr3d:specular` attribute specifies whether a light causes a specular reflection on objects. Consumers may evaluate this attribute only for the first light in a scene.

The defined values for the `dr3d:specular` attribute are:

- **false**: light does not cause specular reflection.
- **true**: light does cause specular reflection.

The `dr3d:specular` attribute is usable with the following element: `<dr3d:light>` 10.5.3.

The `dr3d:specular` attribute has the data type [boolean 18.3.3](#).

## 19.107 dr3d:transform

The `dr3d:transform` attribute specifies a list of transform definitions, which are applied in the order provided. The individual transform definitions are separated by white space.

The defined transforms are:

- `matrix(<a> <b> <c> <d> <e> <f> <g> <h> <i> <j> <k> <l>)`: specifies a transformation in the form of a homogeneous transformation matrix:

$$\begin{pmatrix} a & d & g & j \\ b & e & h & k \\ c & f & i & l \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

where the values (`<j>`, `<k>`, `<l>`) in the right column describe the translation.

- `rotateX(<rotate-angle>)`: specifies a rotation by `<rotate-angle>` degrees along the x-axis.
- `rotateY(<rotate-angle>)`: specifies a rotation by `<rotate-angle>` degrees along the y-axis.
- `rotateZ(<rotate-angle>)`: specifies a rotation by `<rotate-angle>` degrees along the z-axis.
- `scale(<sx> <sy> <sz>)`: specifies a scale operation by `sx`, `sy` and `sz`.
- `translate(<tx> <ty> <tz>)`: specifies a translation by `tx`, `ty` and `tz`.

The `dr3d:transform` attribute is usable with the following elements: `<chart:plot-area> 11.5`, `<dr3d:cube> 10.5.4`, `<dr3d:extrude> 10.5.6`, `<dr3d:rotate> 10.5.7`, `<dr3d:scene> 10.5.2` and `<dr3d:sphere> 10.5.5`.

The `dr3d:transform` attribute has the data type [string 18.2](#).

## 19.108 dr3d:vpn

The `dr3d:vpn` attribute specifies a pointer towards projected objects.

With the `dr3d:vrp` [19.109](#) and `dr3d:vup` [19.110](#) attributes, the `dr3d:vpn` attribute specifies a viewing volume.

The `dr3d:vpn` attribute is usable with the following elements: `<chart:plot-area> 11.5` and `<dr3d:scene> 10.5.2`.

The `dr3d:vpn` attribute has the data type [vector3D 18.3.41](#).

## 19.109 dr3d:vrp

The `dr3d:vrp` attribute specifies an origin.

With the `dr3d:vpn` [19.108](#) and `dr3d:vup` [19.110](#) attributes, the `dr3d:vrp` attribute specifies a viewing volume.

The `dr3d:vrp` attribute is usable with the following elements: `<chart:plot-area> 11.5` and `<dr3d:scene> 10.5.2`.

The `dr3d:vrp` attribute has the data type [vector3D 18.3.41](#).

## 19.110 dr3d:vup

The `dr3d:vup` attribute specifies an up vector.

With the `dr3d:vpn` 19.108 and `dr3d:vrp` 19.109 attributes, the `dr3d:vup` attribute specifies a viewing volume.

The `dr3d:vup` attribute is usable with the following elements: `<chart:plot-area>` 11.5 and `<dr3d:scene>` 10.5.2.

The `dr3d:vup` attribute has the data type `vector3D` 18.3.41.

## 19.111 draw:align

The `draw:align` attribute specifies the shape edge where a glue point is positioned.

The defined values for the `draw:align` attribute are:

- `bottom-left`: The position of the glue point is specified relative to the bottom-left corner of the shape's bounding box.
- `bottom-right`: The position of the glue point is specified relative to the bottom-right corner of the shape's bounding box.
- `center`: The position of the glue point is specified relative to the center of the shape's bounding box.
- `right`: The position of the glue point is specified relative to the right of the shape's bounding box.
- `top`: The position of the glue point is specified relative to the top of the shape's bounding box.
- `top-right`: The position of the glue point is specified relative to the top-right corner of the shape's bounding box.
- `top-left`: The position of the glue point is specified relative to the top-left corner of the shape's bounding box.

The `draw:align` attribute is usable with the following element: `<draw:glue-point>` 10.3.16.

The values of the `draw:align` attribute are `top-left`, `top`, `top-right`, `left`, `center`, `right`, `bottom-left` or `bottom-right`.

## 19.112 draw:angle

The `draw:angle` attribute specifies an angle that rotates the axis at which the gradient values are interpolated. This attribute is ignored for radial style gradients.

The `draw:angle` attribute is usable with the following elements: `<draw:gradient>` 16.42.1 and `<draw:opacity>` 16.42.7.

The `draw:angle` attribute has the data type `angle` 18.3.1.

## 19.113 draw:archive

The `draw:archive` attribute is equivalent to the `archive` attribute of an `<applet>` element in HTML. See. §13.4 of [HTML4].

The `draw:archive` attribute is usable with the following element: `<draw:applet>` 10.4.7.

The `draw:archive` attribute has the data type `string` 18.2.

## 19.114 draw:border

The `draw:border` attribute specifies a percentage value that is used to scale a border area which is filled by a solid color.

For linear, square, rectangle, radial and elliptical gradients the start color is used. For axial gradients, the end color is used.

The `draw:border` attribute is usable with the following elements: `<draw:gradient>` 16.42.1 and `<draw:opacity>` 16.42.7.

The `draw:border` attribute has the data type `percent` 18.3.24.

## 19.115 draw:caption-id

The `draw:caption-id` attribute establishes a relationship between a drawing shape and its caption. It takes a value of type IDREF. The value for the `draw:caption-id` attribute is the target ID assigned to the `<draw:text-box>` 10.4.3 element that contains the caption.

**Note:** When a caption is assigned to a drawing shape, an id shall be assigned to the element containing the text used to caption a drawing shape. Removing the caption should result in removing the `draw:caption-id` attribute of the drawing shape that was being captioned.

The `draw:caption-id` attribute is usable with the following elements: `<dr3d:scene>` 10.5.2, `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:connector>` 10.3.10, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:g>` 10.3.15, `<draw:line>` 10.3.3, `<draw:measure>` 10.3.12, `<draw:page-thumbnail>` 10.3.14, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2 and `<draw:regular-polygon>` 10.3.6.

The `draw:caption-id` attribute has the data type `IDREF` 18.2.

## 19.116 draw:caption-point-x

The `draw:caption-point-x` attribute, along with `draw:caption-point-y` specifies the position of a point that is captioned. A set of lines is rendered to that point from the caption area.

The `draw:caption-point-x` attribute is usable with the following elements: `<draw:caption>` 10.3.11 and `<office:annotation>` 14.1.

The `draw:caption-point-x` attribute has the data type `coordinate` 18.3.10.

## 19.117 draw:caption-point-y

The `draw:caption-point-y` attribute, along with `draw:caption-point-x` specifies the position of a point that is captioned. A set of lines is rendered to that point from the caption area.

The `draw:caption-point-y` attribute is usable with the following elements: `<draw:caption>` 10.3.11 and `<office:annotation>` 14.1.

The `draw:caption-point-y` attribute has the data type `coordinate` 18.3.10.

## 19.118 draw:chain-next-name

The `draw:chain-next-name` attribute specifies a name that is used to chain text boxes together for cases where the content of a text box exceeds its capacity. The content flows into the next text box in the chain. The value of this attribute is the name of the next text box in the chain.

The `draw:chain-next-name` attribute is usable with the following element: `<draw:text-box>` 10.4.3.

The `draw:chain-next-name` attribute has the data type `string` 18.2.

## 19.119 `draw:class-id`

The `draw:class-id` attribute specifies the OLE class id of an object.

The `draw:class-id` attribute is usable with the following element: `<draw:object-ole>` 10.4.6.3.

The `draw:class-id` attribute has the data type `string` 18.2.

## 19.120 `draw:class-names`

The `draw:class-names` attribute specifies a white-space-separated list of styles with the family value of `graphic`. The referenced styles are applied in the order they are contained in the list.

If both `draw:style-name` and `draw:class-names` are present, the style referenced by the `draw:style-name` attribute is applied before the styles referenced by the `draw:class-names` attribute.

The `draw:class-names` attribute is usable with the following elements: `<dr3d:cube>` 10.5.4, `<dr3d:extrude>` 10.5.6, `<dr3d:rotate>` 10.5.7, `<dr3d:scene>` 10.5.2, `<dr3d:sphere>` 10.5.5, `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:connector>` 10.3.10, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:g>` 10.3.15, `<draw:line>` 10.3.3, `<draw:measure>` 10.3.12, `<draw:page-thumbnail>` 10.3.14, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2, `<draw:regular-polygon>` 10.3.6 and `<office:annotation>` 14.1.

The `draw:class-names` attribute has the data type `styleNameRefs` 18.3.34.

## 19.121 `draw:code`

The `draw:code` attribute is equivalent to the `code` attribute of an `<applet>` element in HTML. See. §13.4 of [HTML4].

The `draw:code` attribute is usable with the following element: `<draw:applet>` 10.4.7.

The `draw:code` attribute has the data type `string` 18.2.

## 19.122 `draw:color`

### 19.122.1 General

The `draw:color` attribute specifies different colors depending upon the element where it appears.

### 19.122.2 `<draw:hatch>`

The `draw:color` attribute specifies the color of hatch lines.

The `draw:color` attribute is usable with the following element: `<draw:hatch>` 16.42.5.

The `draw:color` attribute has the data type `color` 18.3.9.

### 19.122.3 `<presentation:dim>`

The `draw:color` attribute specifies the color of that is used to fill a shape when the shape is dimmed.

The `draw:color` attribute is usable with the following element: `<presentation:dim>` 10.8.7.

The `draw:color` attribute has the data type `color` 18.3.9.

### 19.123 `draw:concave`

The `draw:concave` attribute specifies whether the basic model of the polygon is a regular, convex polygon or a star-shaped polygon.

The defined values for the `draw:concave` attribute are:

- `false`: The basic model of the polygon is a regular convex polygon, where the number of corners is specified by the `draw:corners` 19.128 attribute. The basic model is scaled, so that the bounding box of the polygon has the width and height as specified in the `svg:width` 19.575 and `svg:height` 19.543 attributes of the `<draw:regular-polygon>` 10.3.6 element.
- `true`: The polygon is star-shaped. The `draw:sharpness` 19.211 attribute shall be specified in addition to the `draw:concave` attribute.

The `draw:concave` attribute is usable with the following element: `<draw:regular-polygon>` 10.3.6.

### 19.124 `draw:concentric-gradient-fill-allowed`

The `draw:concentric-gradient-fill-allowed` attribute specifies the rendering of a shape with a concentric gradient that uses a custom shape path.

The defined values for the `draw:concentric-gradient-fill-allowed` attribute are:

- `false`: the rendering of a shape with a concentric gradient that does not use a custom shape path.
- `true`: the rendering of a shape with a concentric gradient that does use a custom shape path.

The default value for this attribute is `false`.

The `draw:concentric-gradient-fill-allowed` attribute is usable with the following element: `<draw:enhanced-geometry>` 10.6.2.

The `draw:concentric-gradient-fill-allowed` attribute has the data type `boolean` 18.3.3.

### 19.125 `draw:control`

The `draw:control` attribute specifies a control within a form that is linked to the control shape by its ID.

The `draw:control` attribute is usable with the following element: `<draw:control>` 10.3.13.

The `draw:control` attribute has the data type `IDREF` 18.2.

## 19.126 draw:copy-of

The `draw:copy-of` attribute specifies that a frame displays the contents of another frame. This does not effect style and position information. The style and position information of the frame with the `draw:copy-of` attribute is used to render the copied contents.

**Note:** Multiple frames can be set to display the exact same underlying data: for instance for a company logo, that is to appear somewhere on every page, without being part of a header or footer.

The `draw:copy-of` attribute is usable with the following element: `<draw:frame>` 10.4.2.

The `draw:copy-of` attribute has the data type `string` 18.2.

## 19.127 draw:corner-radius

The `draw:corner-radius` attribute specifies the radius of the circle used to round off the corners of a caption `<draw:caption>`, rectangle `<draw:rect>`, or a text-box `<draw:text-box>`.

The `svg:rx` 19.554 and `svg:ry` 19.555 attributes can also be used to round off the corners of a rectangle `<draw:rect>`.

If `svg:rx` and/or `svg:ry` and `draw:corner-radius` attributes are present on an element, the `svg:rx` and `svg:ry` attributes control the rounding applied to the shape defined by the element. If one or both of `svg:rx` and `svg:ry` attributes are present, any `draw:corner-radius` attribute is ignored.

The `draw:corner-radius` attribute is usable with the following elements:

`<draw:caption>` 10.3.11, `<draw:rect>` 10.3.2, `<draw:text-box>` 10.4.3 and `<office:annotation>` 14.1.

The `draw:corner-radius` attribute has the data type `nonNegativeLength` 18.3.21.

## 19.128 draw:corners

The `draw:corners` attribute specifies the number of corners for a regular-polygon shape. If the `draw:concave` 19.123 attribute of the `<draw:regular-polygon>` 10.3.6 element has the value `false`, the value of the `draw:corners` attribute specifies the number of corners of the polygon. If the `draw:concave` attribute has the value `true`, the polygon has twice as many corners as the value of the `draw:corners` attribute.

**Note:** The drawing of polygons is further described at the `draw:sharpness` 19.211 attribute.

The `draw:corners` attribute is usable with the following element: `<draw:regular-polygon>` 10.3.6.

The `draw:corners` attribute has the data type `positiveInteger` 18.2.

## 19.129 draw:cx

The `draw:cx` attribute, along with the `draw:cy` 19.130 attribute, specifies the center of the geometry that is used for a gradient 19.218.2, if the gradient style is `ellipsoid`, `radial`, `rectangular`, or `square`.

The `draw:cx` attribute is usable with the following elements: `<draw:gradient>` 16.42.1 and `<draw:opacity>` 16.42.7.

The `draw:cx` attribute has the data type `percent` 18.3.24.

## 19.130 draw:cy

The `draw:cy` attribute, along with the `draw:cx` 19.129 attribute, specifies the center of the geometry that is used for a gradient, if the gradient 19.218.2 style is ellipsoid, radial, rectangular, or square.

The `draw:cy` attribute is usable with the following elements: `<draw:gradient>` 16.42.1 and `<draw:opacity>` 16.42.7.

The `draw:cy` attribute has the data type `percent` 18.3.24.

## 19.131 draw:data

The `draw:data` attribute specifies rendering-engine-specific data for a custom shape. This attribute is only evaluated if a rendering engine is specified by the `draw:engine` attribute.

The `draw:data` attribute is usable with the following element: `<draw:custom-shape>` 10.6.1.

The `draw:data` attribute has the data type `string` 18.2.

## 19.132 draw:display

The `draw:display` attribute specifies the visibility of drawing objects contained in a layer for display or printing.

The defined values for the `draw:display` attribute are:

- `always`: drawing objects are visible on screen and in printed output.
- `none`: drawing objects are not visible on screen or in printed output.
- `printer`: drawing objects are visible in printed output but not on screen.
- `screen`: drawing objects are visible on screen but not in printed output.

The default value for this attribute is `always`.

The `draw:display` attribute is usable with the following element: `<draw:layer>` 10.2.3.

The values of the `draw:display` attribute are `always`, `screen`, `printer` or `none`.

## 19.133 draw:display-name

### 19.133.1 General

The `draw:display-name` attribute specifies a name as it should appear in the user interface. If this attribute is not present, the display name is the same as the internal name.

### 19.133.2 <draw:fill-image>

The `draw:display-name` attribute specifies the name of a fill image.

The `draw:display-name` attribute is usable with the following element: `<draw:fill-image>` 16.42.6.

The `draw:display-name` attribute has the data type `string` 18.2.

### 19.133.3 <draw:gradient>

The `draw:display-name` attribute specifies the name of a gradient.

The `draw:display-name` attribute is usable with the following element: `<draw:gradient>` 16.42.1.

The `draw:display-name` attribute has the data type `string` 18.2.

#### 19.133.4 `<draw:hatch>`

The `draw:display-name` attribute specifies the name of a hatch style.

The `draw:display-name` attribute is usable with the following element: `<draw:hatch>` 16.42.5.

The `draw:display-name` attribute has the data type `string` 18.2.

#### 19.133.5 `<draw:marker>`

The `draw:display-name` attribute specifies the name of a marker.

The `draw:display-name` attribute is usable with the following element: `<draw:marker>` 16.42.8.

The `draw:display-name` attribute has the data type `string` 18.2.

#### 19.133.6 `<draw:opacity>`

The `draw:display-name` attribute specifies the name of an opacity gradient.

The `draw:display-name` attribute is usable with the following element: `<draw:opacity>` 16.42.7.

The `draw:display-name` attribute has the data type `string` 18.2.

#### 19.133.7 `<draw:stroke-dash>`

The `draw:display-name` attribute specifies the name of a dash style.

The `draw:display-name` attribute is usable with the following element: `<draw:stroke-dash>` 16.42.9.

The `draw:display-name` attribute has the data type `string` 18.2.

#### 19.133.8 `<svg:linearGradient>`

The `draw:display-name` attribute specifies the name of a gradient.

The `draw:display-name` attribute is usable with the following element: `<svg:linearGradient>` 16.42.2.

The `draw:display-name` attribute has the data type `string` 18.2.

#### 19.133.9 `<svg:radialGradient>`

The `draw:display-name` attribute specifies the name of a gradient.

The `draw:display-name` attribute is usable with the following element: `<svg:radialGradient>` 16.42.3.

The `draw:display-name` attribute has the data type `string` 18.2.

### 19.134 `draw:distance`

#### 19.134.1 General

The `draw:distance` attribute specifies distances.

## 19.134.2 `<draw:hatch>`

The `draw:distance` attribute specifies the distance between two hatch lines.

The `draw:distance` attribute is usable with the following element: `<draw:hatch>` 16.42.5.

The `draw:distance` attribute has the data type `length` 18.3.19.

## 19.134.3 `<draw:stroke-dash>`

The `draw:distance` attribute specifies the distance between the dashes of a stroke.

The value of the attribute may be an absolute length or a percentage value. Percentage values are relative to the width of the stroke as defined by the `svg:stroke-width` attribute. 20.410

The `draw:distance` attribute is usable with the following element: `<draw:stroke-dash>` 16.42.9.

The values of the `draw:distance` attribute are a value of type `length` 18.3.19 or a value of type `percent` 18.3.24.

## 19.135 `draw:dots1`

The `draw:dots1` attribute specifies the number of dashes for the first sequence in an alternating sequence of dots.

The `draw:dots1` attribute is usable with the following element: `<draw:stroke-dash>` 16.42.9.

The `draw:dots1` attribute has the data type `integer` 18.2.

## 19.136 `draw:dots1-length`

The `draw:dots1-length` attributes specifies the length of the dashes in the sequence specified by `draw:dots1`.

The value of the attribute may be an absolute length or a percentage value. Percentage values are relative to the width of the stroke as defined by the `svg:stroke-width` attribute. 20.410

The `draw:dots1-length` attribute is usable with the following element: `<draw:stroke-dash>` 16.42.9.

The values of the `draw:dots1-length` attribute are a value of type `length` 18.3.19 or a value of type `percent` 18.3.24.

## 19.137 `draw:dots2`

The `draw:dots2` attribute specifies the number of dashes for the second sequence in an alternating sequence of dots.

The `draw:dots2` attribute is usable with the following element: `<draw:stroke-dash>` 16.42.9.

The `draw:dots2` attribute has the data type `integer` 18.2.

## 19.138 `draw:dots2-length`

The `draw:dots2-length` attributes specifies the length of the dashes in the sequence specified by `draw:dots2`.

The value of the attribute may be an absolute length or a percentage value. Percentage values relative to the width of the stroke as defined by the `svg:stroke-width` attribute. 20.410

The `draw:dots2-length` attribute is usable with the following element: `<draw:stroke-dash>` 16.42.9.

The values of the `draw:dots2-length` attribute are a value of type `length` 18.3.19 or a value of type `percent` 18.3.24.

## 19.139 `draw:end`

The `draw:end` attribute specifies the end value for the `<draw:opacity>` 16.42.7 element, which interpolates between the value of this attribute and `draw:start`. The defined value range for the `draw:end` attribute is 0% to 100%, where 0% is transparent and 100% is opaque.

The `draw:end` attribute is usable with the following element: `<draw:opacity>` 16.42.7.

The `draw:end` attribute has the data type `zeroToHundredPercent` 18.3.42.

## 19.140 `draw:end-angle`

The `draw:end-angle` attribute specifies the end angle of an arc, cut, or section for circles where the `draw:kind` 19.188 attribute value is `arc`, `cut`, or `section`.

The `draw:end-angle` attribute is usable with the following elements: `<draw:circle>` 10.3.8 and `<draw:ellipse>` 10.3.9.

The `draw:end-angle` attribute has the data type `angle` 18.3.1.

## 19.141 `draw:end-color`

The `draw:end-color` attribute specifies the end value for the `<draw:gradient>` element, which interpolates between the value of this attribute and `draw:start-color` 19.214.

The `draw:end-color` attribute is usable with the following element: `<draw:gradient>` 16.42.1.

The `draw:end-color` attribute has the data type `color` 18.3.9.

## 19.142 `draw:end-glue-point`

The `draw:end-glue-point` attribute identifies the glue point in a shape where a connector ends by its number. See 10.3.16. Glue point numbers are defined by the `draw:id` attributes of the glue point elements `<draw:glue-point>`. 19.187.2

If the connector is not connected to a shape, this attribute is ignored.

The `draw:end-glue-point` attribute is usable with the following element: `<draw:connector>` 10.3.10.

The `draw:end-glue-point` attribute has the data type `nonNegativeInteger` 18.2.

## 19.143 `draw:end-intensity`

The `draw:end-intensity` attribute specifies the intensity of the gradient's end color as a percentage value. If it is not specified, the color is used at 100% intensity. The defined value range for the `draw:end-intensity` attribute is 0% to 100%,

The `draw:end-intensity` attribute is usable with the following element: `<draw:gradient>` 16.42.1.

The `draw:end-intensity` attribute has the data type `zeroToHundredPercent` 18.3.42.

## 19.144 `draw:end-shape`

The `draw:end-shape` attribute specifies a drawing shape to which the end of a connector is connected.

The `draw:end-shape` attribute is usable with the following element: `<draw:connector>` 10.3.10.

The `draw:end-shape` attribute has the data type `IDREF` 18.2.

## 19.145 `draw:enhanced-path`

The `draw:enhanced-path` attribute specifies a path that is the outline of a shape.

The syntax of a value for a `draw:enhanced-path` attribute is defined as:

- Commands are expressed as one character.
- Commands are followed by parameters.
- Commands and parameters are separated by white space characters. Parameters may also be separated by comma "," (U+002C, COMMA) characters.
- White space and commas may be eliminated if they are not necessary to identify individual commands or parameters.
- If a command is repeated multiple times, all repeated command characters except the first one may be omitted.
- The decimal delimiter is the "." (U+002E, FULL STOP) character.

A parameter in the value of a `draw:enhanced-path` attribute may be:

- An integer value.
- An integer value preceded by a "\$" (U+0024, DOLLAR SIGN) character. Such a value is an index to a `draw:modifiers` 19.196 attribute on the element where the `draw:enhanced-path` attribute appears. The corresponding modifier value is used as a parameter value.
- A floating-point value.
- A formula name, preceded by a "?" (U+003F, QUESTION MARK) character. The result of the value of the `draw:formula` 19.171 attribute of the `<draw:equation>` 10.2.5 child element of the `<draw:enhanced-geometry>` element on which the `draw:enhanced-path` attribute appears is used as the parameter value.

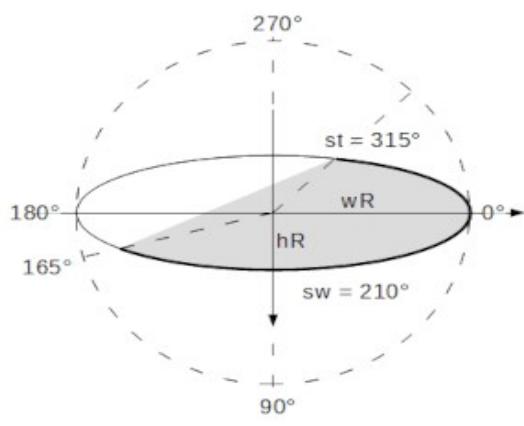
The syntax used in Table 10 for parameters is:

- `()`: grouping of parameters.
- `+`: 1 or more of the given parameter(s) shall be present.

The following commands are defined:

*Table 10 - Enhanced path commands*

<b>Command</b>	<b>Name</b>	<b>Parameters</b>	<b>Description</b>
A	arcto	<code>(x1 y1 x2 y2 x3 y3 x4 y4) +</code>	<code>(x1, y1)</code> and <code>(x2, y2)</code> define the bounding box of an ellipse. A line is then drawn from the current point to the start angle of the arc that is specified by the radial vector of point <code>(x3, y3)</code> and then counter-clockwise to the end-angle that is specified by point <code>(x4, y4)</code> . The intersection of the line and the arc is determined by the <code>draw:intersection</code> attribute of the <code>&lt;draw:arc&gt;</code> element. If the <code>draw:intersection</code> attribute is not present, the intersection is determined by the <code>draw:intersection</code> attribute of the <code>&lt;draw:line&gt;</code> element. If the <code>draw:intersection</code> attribute is not present, the intersection is determined by the <code>draw:intersection</code> attribute of the <code>&lt;draw:arc&gt;</code> element. If the <code>draw:intersection</code> attribute is not present, the intersection is determined by the <code>draw:intersection</code> attribute of the <code>&lt;draw:line&gt;</code> element.

Command	Name	Parameters	Description
			tion of the radial vector of point (x4, y4) with the ellipse becomes the new current point.
B	arc	(x1 y1 x2 y2 x3 y3 x4 y4) +	The same as the "A" command, except that an implied moveto to the starting point is done.
C	curveto	(x1 y1 x2 y2 x y) +	Draws a cubic Bézier curve from the current point to (x,y) using (x1,y1) as the control point at the beginning of the curve and (x2,y2) as the control point at the end of the curve. The point (x, y) becomes the new current point.
F	nofill	(none)	Specifies that the current set of sub-paths will not be filled.
L	lineto	(x y) +	Draws a line from the current point to (x, y). If followed by multiple coordinate pairs, they are all interpreted as lineto. The point (x, y) becomes the new current point in each case.
G	arcangleto	(wR hR st sw) +	<p>Draws an arc of an ellipse. The ellipse is given by the width <math>2*wR</math>, and the height <math>2*hR</math>. The arc is specified by the start angle <math>st</math> in degrees and the swing angle <math>sw</math> in degrees. The ellipse is positioned so that the current point is the start point of the arc. The arc is drawn so that <math>sw</math> is the central angle of the arc. If <math>sw</math> is positive, the arc is drawn clockwise; if <math>sw</math> is negative, the arc is drawn counter-clockwise. The endpoint of the arc becomes the new current point.</p> <p>In case the path is filled, the filling is drawn as an ellipse segment.</p> <p>The swing angle may be greater than <math>360^\circ</math> and may be smaller than <math>-360^\circ</math>. In such cases the fill will overlap.</p> 
M	moveto	(x y) +	Start a new sub-path at the given (x,y) coordinate. If a moveto is followed by multiple pairs of coordinates, they are treated as lineto. The point (x, y) becomes the new current point in

Command	Name	Parameters	Description
			each case.
N	endpath	(none)	Ends the current set of sub-paths. The sub-paths will be filled by using the “even-odd” filling rule. Other following subpaths will be filled independently. <b>The current point becomes undefined.</b>
Q	quadratic-curveto	(x1 y1 x y)+	Draws a quadratic Bézier curve from the current point to (x, y) using (x1, y1) as the control point. (x, y) becomes the new current point at the end of the command.
S	nostroke	(none)	Specifies that the current set of sub-paths will not be stroked.
T	angle-ellipseto	(x y wR hR t0 t1) +	<p>Draws a segment of an ellipse. The ellipse is specified by the center(x, y), the width 2*wR, the height 2*hR.</p> <p>The arc is specified by the start-angle t0 in degrees and end-angle t1 in degrees. The angles may be of any value, but are used modulo 360° with range 0° &lt;= angle &lt; 360° to get start and end point of the arc. The angles specify radial vectors. The start and end point of the arc are the intersection of these vectors with the ellipse. In case the start point equals the end point, only in case  t1 – t0  = 360° a full ellipse is drawn, otherwise an arc of 0° central angle is drawn. The orientation of the angles is the same as in command G, see figure 5.</p> <p>The segment is drawn clockwise.</p> <p><b>The point of the ellipse specified by the end-angle t1 becomes the new current point.</b></p>
U	angle-ellipse	(x y wR hR t0 t1) +	The same as the “T” command, except that no current point is required but an implied moveto to the starting point is done and thus no line to the start point is drawn..
V	clock-wisearc	(x1 y1 x2 y2 x3 y3 x4 y4)+	The same as the “A” command, except that an implied moveto to the starting point is done and the arc is drawn clockwise.
W	clock-wisearcto	(x1 y1 x2 y2 x3 y3 x4 y4) +	The same as the “A” command except, that the arc is drawn clockwise.
X	elliptical-quadrantx	(x y) +	Draws a quarter ellipse, whose initial segment is tangential to the x-axis from the current point to (x, y). For each additional quarter ellipse command, the axis to which the segment is tangential to switches from x to y and from y to x. <b>The point (x, y) becomes the new current point in each case.</b>
Y	elliptical-quadranty	(x y) +	Draws a quarter ellipse, whose initial segment is tangential to the y-axis from the current point to (x, y). For each additional quarter ellipse

Command	Name	Parameters	Description
			command, the axis to which the segment is tangential to switches from y to x and from x to y. The point (x, y) becomes the new current point in each case.
Z	closepath	(none)	Close the current sub-path by drawing a straight line from the current point to current sub-path's initial point. The path's initial point becomes the new current point.

If the value of the `draw:enhanced-path` attribute does not follow the specified syntax, the behavior is undefined. This includes the case, that a command expects a current point, but the current point is undefined.

**Note:** The value of the `draw:enhanced-path` attribute is based upon [SVG] §8.3.

The `draw:enhanced-path` attribute is usable with the following element:  
`<draw:enhanced-geometry>` 10.6.2.

The `draw:enhanced-path` attribute has the data type `string` 18.2.

## 19.146 draw:engine

The `draw:engine` attribute specifies the name of a specific rendering engine that can be used to render a custom shape.

The value of the `draw:engine` attribute shall not influence the geometry of a shape.

The `draw:engine` attribute is usable with the following element: `<draw:custom-shape>` 10.6.1.

The `draw:engine` attribute has the data type `namespacedToken` 18.3.20.

## 19.147 draw:escape-direction

The `draw:escape-direction` attribute specifies the direction in which a connection line leaves a drawing object if a connector connects to a glue point.

The defined values for the `draw:escape-direction` attribute are:

- `auto`: the connection line may leave in all four directions from a drawing object.
- `down`: the connection line leaves down from a drawing object.
- `horizontal`: the connection line may leave to the left or to the right of a drawing object.
- `left`: the connection line leaves to the left of a drawing object.
- `right`: the connection line leaves to the right of a drawing object.
- `up`: the connection line leaves up from a drawing object.
- `vertical`: the connection line may leave up or down from a drawing object.

The `draw:escape-direction` attribute is usable with the following element:  
`<draw:glue-point>` 10.3.16.

The values of the `draw:escape-direction` attribute are `auto`, `left`, `right`, `up`, `down`, `horizontal` or `vertical`.

## 19.148 draw:extrusion

The `draw:extrusion` attribute specifies if an extrusion is displayed.

The defined values for the `draw:extrusion` attribute are:

- `false`: extrusion is not displayed.
- `true`: extrusion is displayed.

The default value for this attribute is `false`.

The `draw:extrusion` attribute is usable with the following element: `<draw:enhanced-geometry>` 10.6.2.

The `draw:extrusion` attribute has the data type `boolean` 18.3.3.

## 19.149 `draw:extrusion-allowed`

The `draw:extrusion-allowed` attribute specifies whether a shape can be rendered as an extrusion object.

The defined values for the `draw:extrusion-allowed` attribute are:

- `false`: shape cannot be rendered as an extrusion object.
- `true`: shape can be rendered as an extrusion object.

The default value for this attribute is `false`.

The `draw:extrusion-allowed` attribute is usable with the following element: `<draw:enhanced-geometry>` 10.6.2.

The `draw:extrusion-allowed` attribute has the data type `boolean` 18.3.3.

## 19.150 `draw:extrusion-brightness`

The `draw:extrusion-brightness` attribute specifies the brightness of a scene.

The default value for this attribute is 33%.

The `draw:extrusion-brightness` attribute is usable with the following element: `<draw:enhanced-geometry>` 10.6.2.

The `draw:extrusion-brightness` attribute has the data type `zeroToHundredPercent` 18.3.42.

## 19.151 `draw:extrusion-color`

The `draw:extrusion-color` attribute specifies if an extrusion color is used. The extrusion color is defined by the `draw:secondary-fill-color` 20.157 attribute specified in a custom shape's graphic style.

The defined values for the `draw:extrusion-color` attribute are:

- `false`: extrusion color is not used.
- `true`: extrusion color is used.

The default value for this attribute is `false`.

The `draw:extrusion-color` attribute is usable with the following element: `<draw:enhanced-geometry>` 10.6.2.

The `draw:extrusion-color` attribute has the data type `boolean` 18.3.3.

## 19.152 `draw:extrusion-depth`

The `draw:extrusion-depth` attribute specifies the depth of an extrusion. It takes two white space separated values. The first value specifies the depth of the extrusion in units, the

second value specifies the fraction of the extrusion that lies before a shape. The second value shall be in the range [0,1]. A value of 0 is the default.

The default value for this attribute is 36pt 0.

The `draw:extrusion-depth` attribute is usable with the following element:  
`<draw:enhanced-geometry>` 10.6.2.

The `draw:extrusion-depth` attribute has two white space separated values. The first value is of type `length` 18.3.19. The second value is of type `double` 18.2.

### 19.153 `draw:extrusion-diffusion`

The `draw:extrusion-diffusion` attribute specifies the amount of diffusion reflected by a shape.

The default value for this attribute is 0%.

The `draw:extrusion-diffusion` attribute is usable with the following element:  
`<draw:enhanced-geometry>` 10.6.2.

The `draw:extrusion-diffusion` attribute has the data type `percent` 18.3.24.

### 19.154 `draw:extrusion-first-light-direction`

The `draw:extrusion-first-light-direction` attribute specifies the direction of the first light.

The default value for this attribute is (5 0 1).

The `draw:extrusion-first-light-direction` attribute is usable with the following element: `<draw:enhanced-geometry>` 10.6.2.

The `draw:extrusion-first-light-direction` attribute has the data type `vector3D` 18.3.41.

### 19.155 `draw:extrusion-first-light-harsh`

The `draw:extrusion-first-light-harsh` attribute specifies if the first light is harsh.

The defined values for the `draw:extrusion-first-light-harsh` attribute are:

- `false`: the light is not harsh. The means by which the light is softened is implementation-defined.
- `true`: the light is harsh. The light is an undiffused point source.

The default value for this attribute is `true`.

The `draw:extrusion-first-light-harsh` attribute is usable with the following element:  
`<draw:enhanced-geometry>` 10.6.2.

The `draw:extrusion-first-light-harsh` attribute has the data type `boolean` 18.3.3.

### 19.156 `draw:extrusion-first-light-level`

The `draw:extrusion-first-light-level` attribute specifies the intensity of the first light.

The default value for this attribute is 66%.

The `draw:extrusion-first-light-level` attribute is usable with the following element:  
`<draw:enhanced-geometry>` 10.6.2.

The `draw:extrusion-first-light-level` attribute has the data type `zeroToHundredPercent` 18.3.42.

## 19.157 `draw:extrusion-light-face`

The `draw:extrusion-light-face` attribute specifies if the front face of an extrusion responds to lighting changes.

The defined values for the `draw:extrusion-light-face` attribute are:

- `false`: front face of extrusion does not respond to lighting changes.
- `true`: front face of extrusion responds to lighting changes.

The default value for this attribute is `true`.

The `draw:extrusion-light-face` attribute is usable with the following element:  
`<draw:enhanced-geometry>` 10.6.2.

The `draw:extrusion-light-face` attribute has the data type `boolean` 18.3.3.

## 19.158 `draw:extrusion-metal`

The `draw:extrusion-metal` attribute specifies the shading of an extruded shape.

The defined values for the `draw:extrusion-metal` attribute are:

- `false`: the specular color for the shading of an extruded shape is white.
- `true`: the specular color for the shading of an extruded shape is gray (red green and blue values of 200) instead of white and 15% is added to the specularity.

The default value for this attribute is `false`.

The `draw:extrusion-metal` attribute is usable with the following element:  
`<draw:enhanced-geometry>` 10.6.2.

The `draw:extrusion-metal` attribute has the data type `boolean` 18.3.3.

## 19.159 `draw:extrusion-number-of-line-segments`

The `draw:extrusion-number-of-line-segments` attribute specifies the number of line segments that should be used to display curved surfaces.

The default value for this attribute is 30.

The `draw:extrusion-number-of-line-segments` attribute is usable with the following element: `<draw:enhanced-geometry>` 10.6.2.

The `draw:extrusion-number-of-line-segments` attribute has the data type `integer` 18.2.

## 19.160 `draw:extrusion-origin`

The `draw:extrusion-origin` attribute specifies the point of origin of a shape within its bounding box in shape-size fractions. The value of the `draw:extrusion-origin` attribute consists of two parameters separated by white space.

The first parameter represents the horizontal origin of a shape.

The defined values for the first parameter of the `draw:extrusion-origin` attribute are:

- `-0.5`: represents the left side of the shape.
- `0`: represents the center of the shape.

- 0.5: represents the right side of the shape.

The second parameter represents the vertical origin of a shape.

The defined values for the second parameter of the `draw:extrusion-origin` attribute are:

- -0.5: represents the top side of the shape.
- 0: represents the center of the shape.
- 0.5: represents the bottom side of the shape.

The default value for this attribute is 0.5 -0.5.

The `draw:extrusion-origin` attribute is usable with the following element:  
`<draw:enhanced-geometry> 10.6.2.`

The `draw:extrusion-origin` attribute has two white space separated values. The first value is a value of type `double` 18.2 in the range [-0.5,0.5]. The second value is a value of type `double` 18.2 in the range [-0.5,0.5].

## 19.161 `draw:extrusion-rotation-angle`

The `draw:extrusion-rotation-angle` attribute specifies rotation about an x-axis and y-axis. The values are separated by a white space. The first angle specifies rotation on the x-axis and the second angle specifies rotation on the y-axis. The z-axis is specified by the `draw:transform` 19.228 attribute on the `<draw:custom-shape>` 10.6.1 element.

The order of the rotation is: z-axis, y-axis and then x-axis.

The default value for this attribute is 0 0.

The `draw:extrusion-rotation-angle` attribute is usable with the following element:  
`<draw:enhanced-geometry> 10.6.2.`

The `draw:extrusion-rotation-angle` attribute has two white space separated values. The first value is of type `angle` 18.3.1. The second value is of type `angle` 18.3.1.

## 19.162 `draw:extrusion-rotation-center`

The `draw:extrusion-rotation-center` attribute specifies the position of a rotation center in shape-size fractions. If the `draw:extrusion-rotation-center` attribute is omitted, the geometric center of the shape is used as the rotation center.

The `draw:extrusion-rotation-center` attribute is usable with the following element:  
`<draw:enhanced-geometry> 10.6.2.`

The `draw:extrusion-rotation-center` attribute has the data type `vector3D` 18.3.41.

## 19.163 `draw:extrusion-second-light-direction`

The `draw:extrusion-second-light-direction` attribute specifies the direction of the second light.

The default value for this attribute is (-5 0 1).

The `draw:extrusion-second-light-direction` attribute is usable with the following element: `<draw:enhanced-geometry> 10.6.2.`

The `draw:extrusion-second-light-direction` attribute has the data type `vector3D` 18.3.41.

## 19.164 draw:extrusion-second-light-harsh

The `draw:extrusion-second-light-harsh` attribute specifies if the second light is harsh.

The defined values for the `draw:extrusion-second-light-harsh` attribute are:

- `false`: the light is not harsh. The means by which the light is softened is implementation-defined.
- `true`: the light is harsh. The light is an undiffused point source.

The default value for this attribute is `true`.

The `draw:extrusion-second-light-harsh` attribute is usable with the following element: `<draw:enhanced-geometry>` 10.6.2.

The `draw:extrusion-second-light-harsh` attribute has the data type `boolean` 18.3.3.

## 19.165 draw:extrusion-second-light-level

The `draw:extrusion-second-light-level` attribute specifies the intensity of the second light.

The default value for this attribute is 66%.

The `draw:extrusion-second-light-level` attribute is usable with the following element: `<draw:enhanced-geometry>` 10.6.2.

The `draw:extrusion-second-light-level` attribute has the data type `zeroToHundredPercent` 18.3.42.

## 19.166 draw:extrusion-shininess

The `draw:extrusion-shininess` attribute specifies the reflectivity of a mirror.

The default value for this attribute is 50%.

The `draw:extrusion-shininess` attribute is usable with the following element: `<draw:enhanced-geometry>` 10.6.2.

The `draw:extrusion-shininess` attribute has the data type `zeroToHundredPercent` 18.3.42.

## 19.167 draw:extrusion-skew

The `draw:extrusion-skew` attribute specifies the skew amount and skew angle of an extrusion. Skew settings are only applied if the `dr3d:projection` 19.102 attribute on the same element has the value `parallel`.

The attribute value is a white space separated number and angle. The first value defines the skew amount in percent, the second value defines the skew angle.

The default value for this attribute is 50 45.

The `draw:extrusion-skew` attribute is usable with the following element: `<draw:enhanced-geometry>` 10.6.2.

The `draw:extrusion-skew` attribute has two white space separated values. The first value is of type `double` 18.2. The second value is of type `angle` 18.3.1.

## 19.168 draw:extrusion-specularity

The `draw:extrusion-specularity` attribute specifies the specularity of an extrusion object.

The default value for this attribute is 0%. The value shall be non-negative.

The `draw:extrusion-specularity` attribute is usable with the following element:  
`<draw:enhanced-geometry>` 10.6.2.

The `draw:extrusion-specularity` attribute has the data type `nonNegativePercent`  
18.3.44.

## 19.169 draw:extrusion-viewpoint

The `draw:extrusion-viewpoint` attribute specifies the viewpoint of an observer as a 3D point.

The default value for this attribute is (3.5cm -3.5cm 25cm).

The `draw:extrusion-viewpoint` attribute is usable with the following element:  
`<draw:enhanced-geometry>` 10.6.2.

The `draw:extrusion-viewpoint` attribute has the data type `point3D` 18.3.25.

## 19.170 draw:filter-name

The `draw:filter-name` attribute specifies the implementation-dependent filter name that has been used to load an image into the document.

The `draw:filter-name` attribute is usable with the following element: `<draw:image>` 10.4.4.

The `draw:filter-name` attribute has the data type `string` 18.2.

## 19.171 draw:formula

The `draw:formula` attribute specifies an equation that should be used to evaluate a value.

The value of a `draw:formula` attribute can have one of the following parameters:

- A "?" (U+003F, QUESTION MARK) is used to mark the beginning of a formula name. The result of the `<draw:equation>` element's `draw:formula` attribute whose `draw:name` 19.197 attribute is the formula name used as the value.
- If "\$" (U+0024, DOLLAR SIGN) precedes an integer value, the value indexes a `draw:modifiers` 19.196 attribute. The corresponding modifier value is used as the parameter value.

```
formula ::= additive_expression
identifier ::= 'pi' | 'left' | 'top' | 'right' | 'bottom' |
'xstretch' |
| 'ystretch' | 'hasstroke' | 'hasfill' | 'width' | 'height' |
'logwidth' |
| 'logheight'
unary_function ::= 'abs' | 'sqrt' | 'sin' | 'cos' | 'tan' | 'atan'
binary_function ::= 'min' | 'max' | 'atan2'
ternary_function ::= 'if'
basic_expression ::= number
| identifier
| function_reference
| modifier_reference
| unary_function S* '(' S* additive_expression S* ')'
| binary_function S* '(' S* additive_expression S* ',' S* ')'
```

```

        additive_expression S* ')'
    | ternary_function S* '(' S* additive_expression S* ',' S*
        additive_expression S* ',' S* additive_expression S* ')'
    | '(' S* additive_expression S* ')'
unary_expression ::= '-' S* basic_expression | basic_expression
multiplicative_expression ::= unary_expression      ( ( S* '*' S*
unary_expression )
    | ( S* '/' S* unary_expression ) )*
additive_expression ::= multiplicative_expression
    ( ( S* '+' S* multiplicative_expression )
    | ( S* '-' S* multiplicative_expression ) )*
number ::= sign? Integer | sign? Floating-point-constant
function_reference ::= "?" name
modifier_reference ::= "$" integer
floating-point-constant ::= fractional-constant exponent? | integer
exponent
fractional-constant ::= integer? '.' integer | integer '.'
exponent ::= ( 'e' | 'E' ) sign? Integer
sign ::= '+' | '-'
name ::= [^#x20#x9]+
integer ::= [0-9]+
S ::= (#x20 | #x9)

```

*Table 11 - Enhanced geometry equation identifiers*

<b>Identifier</b>	<b>Description</b>
bottom	The bottom position of the <code>svg:viewBox</code> 19.574 attribute is used.
hasfill	If the shape has a fill style, a value of 1 is used. If the shape has no fill style, a value of 0 is used.
hasstroke	If the shape has a line style, a value of 1 is used. If the shape has no line style, a value of 0 is used.
height	The height of the <code>svg:viewBox</code> 19.574 attribute is used.
left	The left position of the <code>svg:viewBox</code> 19.574 attribute is used.
logheight	The height in 1/100th mm as specified by the <code>svg:height</code> 19.543 attribute is used.
logwidth	The width in 1/100th mm as specified by the <code>svg:width</code> 19.575 attribute is used.
pi	The number $\pi$ .
right	The right position of the <code>svg:viewBox</code> 19.574 attribute is used.
top	The top position of the <code>svg:viewBox</code> 19.574 attribute is used.
width	The width of the <code>svg:viewBox</code> 19.574 attribute is used.
xstretch	The value of the <code>draw:path-stretchpoint-x</code> 19.204 attribute is used.
ystretch	The value of the <code>draw:path-stretchpoint-y</code> 19.205 attribute is used.

*Table 12 - Functions Used in `draw:formula`*

<b>Function</b>	<b>Description</b>
-----------------	--------------------

abs(n)	returns the absolute value of n
sqrt(n)	returns the positive square root of n
sin(n)	returns the trigonometric sine of n, where n is an angle specified in radians
cos(n)	returns the trigonometric cosine of n, where n is an angle specified in radians
tan(n)	returns the trigonometric tangent of n, where n is an angle specified in radians
atan(n)	returns the arc tangent of n in radians
min(x,y)	returns the smaller of two values
max(x,y)	returns the greater of two values
atan2(y,x)	returns the angle in radians of the vector (x,y) with the x-axis
if(c,x,y)	conditional testing: if c is greater than zero then the result of evaluating x is returned, otherwise the result of evaluating y is returned.

The `draw:formula` attribute is usable with the following element: `<draw:equation>` 10.2.5.

The `draw:formula` attribute has the data type `string` 18.2.

## 19.172 draw:frame-name

The `draw:frame-name` attribute specifies the name of a frame. The name of a frame can be used as a target by hyperlinks.

The `draw:frame-name` attribute is usable with the following element: `<draw:floating-frame>` 10.4.10.

The `draw:frame-name` attribute has the data type `string` 18.2.

## 19.173 draw:glue-point-leaving-directions

The `draw:glue-point-leaving-directions` attribute specifies a comma "," (U+002C, COMMA) separated list of angles. The angles are listed in the same order as the glue points specified in the `draw:glue-points` attribute. 19.175

The `draw:glue-point-leaving-directions` attribute is usable with the following element: `<draw:enhanced-geometry>` 10.6.2.

The `draw:glue-point-leaving-directions` attribute has the data type `string` 18.2.

## 19.174 draw:glue-point-type

The `draw:glue-point-type` attribute specifies a glue point type. If the `draw:glue-points` 19.175 attribute has a value, this attribute is ignored.

The defined values for the `draw:glue-point-type` attribute are:

- `none`: there are no glue points, including default glue points.
- `rectangle`: the middle of each side of the shape bound rectangle specifies an object specific glue point

- **segments**: a connector will connect with each point of the `draw:enhanced-path` `19.145` attribute.

The default value for this attribute is `none`.

The `draw:glue-point-type` attribute is usable with the following element:  
`<draw:enhanced-geometry>` [10.6.2](#).

The values of the `draw:glue-point-type` attribute are `none`, `segments` or `rectangle`.

## 19.175 draw:glue-points

The `draw:glue-points` attribute specifies a list of object-defined glue points. Unlike the user-defined glue points which are defined by the `<draw:glue-point>` sub-element, the position of an object-defined glue point can be defined using equations and modifiers.

The first parameter specifies the horizontal position of the glue point.

The second parameter specifies the vertical position of the glue point.

Each parameter can be a floating-point number, or it can also have one of the following enhancements:

- A “?” (U+003F, QUESTION MARK) is used to mark the beginning of a formula name. The result of the `<draw:equation>` element's `draw:formula` attribute whose `draw:name` attribute is the formula name that is used as the value.
- If “\$” (U+0024, DOLLAR SIGN) precedes an integer value, the value is an index to a `draw:modifiers` attribute. The corresponding modifier value is used as parameter value.

The value of a `draw:glue-points` attribute is formally defined as:

```

gluepoints ::= gluepointsequence?
gluepointsequence ::= gluepoint ( ' ' + gluepointsequence ) *
glue-point ::= position ' ' + position
position ::= formula | modifier | number
formula ::= '?' name
modifier ::= '$' integer
number ::= sign? float | sign? integer
float ::= fractional exponent? | integer exponent
fractional ::= integer? '.' integer | integer '.'
exponent ::= ( 'e' | 'E' ) sign? integer
sign ::= '+' | '-'
integer ::= [0-9] +
name ::= [^ ] +

```

The `draw:glue-points` attribute is usable with the following element: `<draw:enhanced-geometry>` [10.6.2](#).

The `draw:glue-points` attribute has the data type `string` [18.2](#).

## 19.176 draw:handle-mirror-horizontal

The `draw:handle-mirror-horizontal` attribute specifies if the y position of a handle is mirrored.

The defined values for the `draw:handle-mirror-horizontal` attribute are:

- `false`: the y position of a handle is not mirrored.
- `true`: the y position of a handle is mirrored.

The default value for this attribute is `false`.

The `draw:handle-mirror-horizontal` attribute is usable with the following element:  
`<draw:handle>` [10.6.3](#).

The `draw:handle-mirror-horizontal` attribute has the data type [boolean 18.3.3](#).

## 19.177 `draw:handle-mirror-vertical`

The `draw:handle-mirror-vertical` attribute specifies if the x position of a handle is mirrored.

The defined values for the `draw:handle-mirror-vertical` attribute are:

- `false`: the x position of a handle is not mirrored.
- `true`: the x position of a handle is mirrored.

The default value for this attribute is `false`.

The `draw:handle-mirror-vertical` attribute is usable with the following element:  
`<draw:handle>` [10.6.3](#).

The `draw:handle-mirror-vertical` attribute has the data type [boolean 18.3.3](#).

## 19.178 `draw:handle-polar` (deprecated)

The `draw:handle-polar` attribute specifies that a handle is a polar handle. The syntax for this attribute is the same as for the `draw:handle-position` [19.179](#) attribute. The first parameter specifies the horizontal center position, the vertical center position is specified by the second parameter. If this attribute is set, the `draw:handle-range-x-minimum` [19.183](#), `draw:handle-range-x-maximum` [19.182](#), `draw:handle-range-y-minimum` [19.185](#) and `draw:handle-range-y-maximum` [19.184](#) attributes are ignored, and the `draw:handle-radius-range-minimum` [19.181](#) and `draw:handle-radius-range-maximum` [19.180](#) attributes are used.

The `draw:handle-polar` attribute is usable with the following element: `<draw:handle>` [10.6.3](#).

The `draw:handle-polar` attribute has the data type [string 18.2](#).

## 19.179 `draw:handle-position` (deprecated)

The `draw:handle-position` attribute specifies the position of a handle and consists of two values.

Each value can be a floating-point number or it can have one of the following enhancements:

- A "?" (U+003F, QUESTION MARK) is used to mark the beginning of a formula name. The result of the `<draw:equation>` [10.2.5](#) element's `draw:formula` [19.171](#) attribute whose `draw:name` [19.197.8](#) attribute is the formula name that is used as the value.
- If "\$" (U+0024, DOLLAR SIGN) precedes an integer value, the value is an index to a `draw:modifiers` [19.196](#) attribute. The corresponding modifier value is used as parameter value.
- Instead of a number a value can also be one of the following constants:

*Table 13 - Handle position constants*

<b>Constant</b>	<b>Description</b>
<code>bottom</code>	The bottom coordinate of the view box as specified by the <code>svg:viewBox</code> <a href="#">19.574</a> attribute is used.
<code>hasfill</code>	If the shape has a fill style, a value of 1 is used.
<code>hasstroke</code>	If the shape has a line style, a value of 1 is used.

Constant	Description
height	The view box height as specified by the <code>svg:viewBox</code> 19.574 attribute is used.
logheight	The height in 1/100th mm as specified by the <code>svg:height</code> 19.543 attribute is used.
logwidth	The width in 1/100th mm as specified by the <code>svg:width</code> 19.575 attribute is used.
left	The left coordinate of the view box as specified by the <code>svg:viewBox</code> 19.574 attribute is used.
right	The right coordinate of the view box as specified by the <code>svg:viewBox</code> 19.574 attribute is used.
top	The top coordinate of the view box as specified by the <code>svg:viewBox</code> 19.574 attribute is used.
width	The view box width as specified by the <code>svg:viewBox</code> 19.574 attribute is used.
xstretch	The value of the <code>draw:path-stretchpoint-x</code> 19.204 attribute is used.
ystretch	The value of the <code>draw:path-stretchpoint-y</code> 19.205 attribute is used.

If the `draw:handle-polar` 19.178 attribute is not set, the first parameter of the `draw:handle-position` attribute specifies the horizontal handle position, the vertical handle position is described by the second parameter. If the `draw:handle-polar` attribute is set, then a handle is a polar handle and the first parameter of the `draw:handle-position` attribute specifies the angle (as defined in 18.3.1), the handle radius is specified by the second parameter.

The value of a `draw:handle-position` attribute is formally defined as:

```

handleposition ::= position ' ' + position
position ::= formula | modifier | constant | number
formula ::= '?' name
modifier ::= '$' integer
constant ::= 'left' | 'top' | 'right' | 'bottom' |
'xstretch' | 'ystretch' | 'hasstroke' | 'hasfill' |
'width' | 'height' | 'logwidth' | 'logheight'
number ::= sign? float | sign? integer
float ::= fractional exponent? | integer exponent
fractional ::= integer? '.' integer | integer '.'
exponent ::= ( 'e' | 'E' ) sign? integer
sign ::= '+' | '-'
integer ::= [0-9] +
name ::= [^ ] +

```

The `draw:handle-position` attribute is usable with the following element:

`<draw:handle>` 10.6.3.

The `draw:handle-position` attribute has the data type `string` 18.2.

## 19.180 `draw:handle-radius-range-maximum`

The `draw:handle-radius-range-maximum` attribute specifies the maximum radius range for a polar handle. The syntax for the value of the `draw:handle-radius-range-maximum` attribute is the same as the `draw:handle-polar-radius` attribute. 19.928

The `draw:handle-radius-range-maximum` attribute is usable with the following element:

`<draw:handle>` 10.6.3.

The `draw:handle-radius-range-maximum` attribute has the data type `string` 18.2.

## 19.181 `draw:handle-radius-range-minimum`

The `draw:handle-radius-range-minimum` attribute specifies the minimum radius range for a polar handle. The syntax for the value of the `draw:handle-radius-range-minimum` attribute is the same as the `draw:handle-polar-radius` attribute. 19.928

The `draw:handle-radius-range-minimum` attribute is usable with the following element:

`<draw:handle>` 10.6.3.

The `draw:handle-radius-range-minimum` attribute has the data type `string` 18.2.

## 19.182 `draw:handle-range-x-maximum`

The `draw:handle-range-x-maximum` attribute specifies the horizontal maximum value of the range for movement of a handle. The syntax for the value of the `draw:handle-range-x-maximum` attribute is the same as the `draw:handle-position-x` attribute. 19.924

The `draw:handle-range-x-maximum` attribute is usable with the following element:

`<draw:handle>` 10.6.3.

The `draw:handle-range-x-maximum` attribute has the data type `string` 18.2.

## 19.183 `draw:handle-range-x-minimum`

The `draw:handle-range-x-minimum` attribute specifies the horizontal minimum value of the range of movement of a handle. The syntax for the value of the `draw:handle-range-x-minimum` attribute is the same as the `draw:handle-position-x` attribute. 19.924

The `draw:handle-range-x-minimum` attribute is usable with the following element:

`<draw:handle>` 10.6.3.

The `draw:handle-range-x-minimum` attribute has the data type `string` 18.2.

## 19.184 `draw:handle-range-y-maximum`

The `draw:handle-range-y-maximum` attribute specifies the vertical maximum value of the range for movement of a handle. The syntax for the value of the `draw:handle-range-y-maximum` attribute is the same as the `draw:handle-position-y` attribute. 19.925

The `draw:handle-range-y-maximum` attribute is usable with the following element:

`<draw:handle>` 10.6.3.

The `draw:handle-range-y-maximum` attribute has the data type `string` 18.2.

## 19.185 `draw:handle-range-y-minimum`

The `draw:handle-range-y-minimum` attribute specifies the vertical minimum value of the range of movement of a handle. The syntax for the value of the `draw:handle-range-y-minimum` attribute is the same as the `draw:handle-position-y` attribute. 19.925

The `draw:handle-range-y-minimum` attribute is usable with the following element:

`<draw:handle>` 10.6.3.

The `draw:handle-range-y-minimum` attribute has the data type `string` 18.2.

## 19.186 draw:handle-switched

The `draw:handle-switched` attribute specifies if handle directions are swapped if shape height exceeds shape width.

The defined values for the `draw:handle-switched` attribute are:

- `false`: handle directions are not switched when a shape's height exceeds its width.
- `true`: handle directions are switched when a shape's height exceeds its width.

The default value for this attribute is `false`.

The `draw:handle-switched` attribute is usable with the following element:  
`<draw:handle>` 10.6.3.

The `draw:handle-switched` attribute has the data type `boolean` 18.3.3.

## 19.187 draw:id

### 19.187.1 General

The `draw:id` attribute specifies an identifier for an element.

### 19.187.2 <draw:glue-point>

The `draw:id` attribute specifies an identifier for a glue point. See 10.3.16. The identifiers 0 to 3 are reserved for the four standard glue points of drawing objects. The glue points are numbered clockwise, starting at the top left corner edge of the shape.

The `draw:id` attribute is usable with the following element: `<draw:glue-point>` 10.3.16.

The `draw:id` attribute has the data type `nonNegativeInteger` 18.2.

### 19.187.3 <dr3d:cube>, <dr3d:extrude>, <dr3d:rotate>, <dr3d:scene>, <dr3d:sphere>, <draw:caption>, <draw:circle>, <draw:connector>, <draw:control>, <draw:custom-shape>, <draw:ellipse>, <draw:frame>, <draw:g>, <draw:line>, <draw:measure>, <draw:page>, <draw:page-thumbnail>, <draw:path>, <draw:polygon>, <draw:polyline>, <draw:rect>, <draw:regular-polygon> and <office:annotation> Deprecated

The `draw:id` attribute specifies identifiers for draw elements.

OpenDocument consumers shall ignore a `draw:id` attribute if it occurs on a draw element with an `xml:id` attribute value.

OpenDocument producers may write `draw:id` attributes for any draw element in addition to an `xml:id` attribute.

The value of a `draw:id` attribute shall equal the value of an `xml:id` attribute on the same element.

The `draw:id` attribute is deprecated in favor of `xml:id`. 19.920

The `draw:id` attribute is usable with the following elements: `<dr3d:cube>` 10.5.4, `<dr3d:extrude>` 10.5.6, `<dr3d:rotate>` 10.5.7, `<dr3d:scene>` 10.5.2, `<dr3d:sphere>` 10.5.5, `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:connector>` 10.3.10, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:g>` 10.3.15, `<draw:line>` 10.3.3, `<draw:measure>` 10.3.12, `<draw:page>` 10.2.4, `<draw:page-thumbnail>` 10.3.14, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4,

<draw:rect> 10.3.2, <draw:regular-polygon> 10.3.6 and <office:annotation> 14.1.

The `draw:id` attribute has the data type NCName 18.2.

## 19.188 draw:kind

The `draw:kind` attribute specifies the appearance of a circle.

The defined values for the `draw:kind` attribute are:

- `arc`: specifies a circle or ellipse arc, like .
- `cut`: specifies a circle or ellipse with a cut, like .
- `full`: specifies a full circle or ellipse, like .
- `section`: specifies a section of a circle or ellipse, like .

The default value for this attribute is `full`.

The `draw:kind` attribute is usable with the following elements: <draw:circle> 10.3.8 and <draw:ellipse> 10.3.9.

The values of the `draw:kind` attribute are `full`, `section`, `cut` or `arc`.

## 19.189 draw:layer

The `draw:layer` attribute specifies the name of a layer in the layer-set of a document.

**Note:** The effect of this attribute is to assign a shape to a particular layer.

The `draw:layer` attribute is usable with the following elements: <dr3d:cube> 10.5.4, <dr3d:extrude> 10.5.6, <dr3d:rotate> 10.5.7, <dr3d:scene> 10.5.2, <dr3d:sphere> 10.5.5, <draw:caption> 10.3.11, <draw:circle> 10.3.8, <draw:connector> 10.3.10, <draw:control> 10.3.13, <draw:custom-shape> 10.6.1, <draw:ellipse> 10.3.9, <draw:frame> 10.4.2, <draw:line> 10.3.3, <draw:measure> 10.3.12, <draw:page-thumbnail> 10.3.14, <draw:path> 10.3.7, <draw:polygon> 10.3.5, <draw:polyline> 10.3.4, <draw:rect> 10.3.2, <draw:regular-polygon> 10.3.6 and <office:annotation> 14.1.

The `draw:layer` attribute has the data type string 18.2.

## 19.190 draw:line-skew

The `draw:line-skew` attribute specifies a list of offsets for the placements of connector lines if the connector is of type `standard`. The offsets are relative to the default position of the connector, that is, the position of the lines if the `draw:line-skew` attribute is not present. The offsets are applied in the order from the connector's start to its end shape. The first offset is applied to the line that follows the line that is connected to the start shape. The last line to which an offset is applied is the line that precedes the line that is connected to the end shape.

For the application of the offsets the connector lines are interpreted as vectors from their start to their end point. Positive values move a line to the right, while negative values move it to the left.

The `draw:line-skew` attribute is usable with the following element: <draw:connector> 10.3.10.

The values of the `draw:line-skew` attribute are one, two or three white space separated values of type length 18.3.19.

## 19.191 draw:mime-type

The `draw:mime-type` attribute specifies the MIME type of the media type that a plugin processes, or the MIME type of the image given by a `<draw:image>` element. Valid values for this attribute are those defined in accordance with §3.7 of [RFC2616], or registered in accordance with [RFC6838].

**Note:** Additional information on MIME media types can be found at [MIMETYPES].

The `draw:mime-type` attribute is usable with the following elements: `<draw:image>` 10.4.4 and `<draw:plugin>` 10.4.8.

The `draw:mime-type` attribute has the data type `string` 18.2.

## 19.192 draw:mirror-horizontal

The `draw:mirror-horizontal` attribute specifies if the horizontal geometry of a shape is to be mirrored.

The defined values for the `draw:mirror-horizontal` attribute are:

- `false`: horizontal geometry of a shape is not mirrored.
- `true`: horizontal geometry of a shape is mirrored.

The default value for this attribute is `false`.

The `draw:mirror-horizontal` attribute is usable with the following element: `<draw:enhanced-geometry>` 10.6.2.

The `draw:mirror-horizontal` attribute has the data type `boolean` 18.3.3.

## 19.193 draw:mirror-vertical

The `draw:mirror-vertical` attribute specifies if the vertical geometry of a shape is to be mirrored.

The defined values for the `draw:mirror-vertical` attribute are:

- `false`: vertical geometry of a shape is not mirrored.
- `true`: vertical geometry of a shape is mirrored.

The default value for this attribute is `false`.

The `draw:mirror-vertical` attribute is usable with the following element: `<draw:enhanced-geometry>` 10.6.2.

The `draw:mirror-vertical` attribute has the data type `boolean` 18.3.3.

## 19.194 draw:master-page-name

The `draw:master-page-name` attribute specifies the name of a master page assigned to a drawing page.

The `draw:master-page-name` attribute is usable with the following element: `<draw:page>` 10.2.4.

The `draw:master-page-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.195 draw:may-script

The `draw:may-script` attribute specifies whether an applet can be modified by scripts.

The defined values for the `draw:may-script` attribute are:

- `false`: applet cannot be modified by scripts.
- `true`: applet can be modified by scripts.

The default value for this attribute is `false`.

The `draw:may-script` attribute is usable with the following element: `<draw:applet>`  
10.4.7.

The `draw:may-script` attribute has the data type `boolean` 18.3.3.

## 19.196 `draw:modifiers`

The `draw:modifiers` attribute contains list of modifier values.

The value of a `draw:modifiers` attribute is formally defined as:

```
modifiers ::= numbersequence?
numbersequence ::= number (' ' + numbersequence)*
number ::= sign? float | sign? integer
float ::= fractional exponent? | integer exponent
fractional ::= integer? '.' integer | integer '.'
exponent ::= ('e' | 'E') sign? integer
sign ::= '+' | '-'
integer ::= [0-9] +
```

The `draw:modifiers` attribute is usable with the following element: `<draw:enhanced-geometry>` 10.6.2.

The `draw:modifiers` attribute has the data type `string` 18.2.

## 19.197 `draw:name`

### 19.197.1 General

The `draw:name` attribute specifies names that are used for referencing graphical elements.

### 19.197.2 `<draw:caption>`

The `draw:name` attribute specifies a name by which a `<draw:caption>` element can be referenced.

The `draw:name` attribute is usable with the following element: `<draw:caption>` 10.3.11.

The `draw:name` attribute has the data type `string` 18.2.

### 19.197.3 `<draw:circle>`

The `draw:name` attribute specifies a name by which a `<draw:circle>` element can be referenced.

The `draw:name` attribute is usable with the following element: `<draw:circle>` 10.3.8.

The `draw:name` attribute has the data type `string` 18.2.

### 19.197.4 `<draw:connector>`

The `draw:name` attribute specifies a name by which a `<draw:connector>` element can be referenced.

The `draw:name` attribute is usable with the following element: `<draw:connector>` 10.3.10.

The `draw:name` attribute has the data type `string` 18.2.

## 19.197.5 <draw:control>

The `draw:name` attribute specifies a name by which a `<draw:control>` element can be referenced.

The `draw:name` attribute is usable with the following element: `<draw:control>` 10.3.13.

The `draw:name` attribute has the data type `string` 18.2.

## 19.197.6 <draw:custom-shape>

The `draw:name` attribute specifies a name by which a `<draw:custom-shape>` element can be referenced.

The `draw:name` attribute is usable with the following element: `<draw:custom-shape>` 10.6.1.

The `draw:name` attribute has the data type `string` 18.2.

## 19.197.7 <draw:ellipse>

The `draw:name` attribute specifies a name by which a `<draw:ellipse>` element can be referenced.

The `draw:name` attribute is usable with the following element: `<draw:ellipse>` 10.3.9.

The `draw:name` attribute has the data type `string` 18.2.

## 19.197.8 <draw:equation>

The `draw:name` attribute specifies a name by which a `<draw:equation>` element can be referenced.

The `draw:name` attribute is usable with the following element: `<draw:equation>` 10.2.5.

The `draw:name` attribute has the data type `string` 18.2.

## 19.197.9 <draw:fill-image>

The `draw:name` attribute specifies a name by which a `<draw:fill-image>` element can be referenced.

The `draw:name` attribute is usable with the following element: `<draw:fill-image>` 16.42.6.

The `draw:name` attribute has the data type `styleName` 18.3.32.

## 19.197.10 <draw:frame>

The `draw:name` attribute specifies a name by which a `<draw:frame>` element can be referenced.

The `draw:name` attribute is usable with the following element: `<draw:frame>` 10.4.2.

The `draw:name` attribute has the data type `string` 18.2.

## 19.197.11 <draw:g>

The `draw:name` attribute specifies a name by which a `<draw:g>` element can be referenced.

The `draw:name` attribute is usable with the following element: `<draw:g>` 10.3.15.

The `draw:name` attribute has the data type `string` 18.2.

## 19.197.12 <draw:gradient>

The `draw:name` attribute specifies a name by which a `<draw:gradient>` element can be referenced.

The `draw:name` attribute is usable with the following element: `<draw:gradient>` 16.42.1.

The `draw:name` attribute has the data type `styleName` 18.3.32.

## 19.197.13 <draw:hatch>

The `draw:name` attribute specifies a name by which a `<draw:hatch>` element can be referenced.

The `draw:name` attribute is usable with the following element: `<draw:hatch>` 16.42.5.

The `draw:name` attribute has the data type `styleName` 18.3.32.

## 19.197.14 <draw:layer>

The `draw:name` attribute specifies a name by which a `<draw:layer>` element can be referenced.

The `draw:name` attribute is usable with the following element: `<draw:layer>` 10.2.3.

The `draw:name` attribute has the data type `string` 18.2.

## 19.197.15 <draw:line>

The `draw:name` attribute specifies a name by which a `<draw:line>` element can be referenced.

The `draw:name` attribute is usable with the following element: `<draw:line>` 10.3.3.

The `draw:name` attribute has the data type `string` 18.2.

## 19.197.16 <draw:marker>

The `draw:name` attribute specifies a name by which a `<draw:marker>` element can be referenced.

The `draw:name` attribute is usable with the following element: `<draw:marker>` 16.42.8.

The `draw:name` attribute has the data type `styleName` 18.3.32.

## 19.197.17 <draw:measure>

The `draw:name` attribute specifies a name by which a `<draw:measure>` element can be referenced.

The `draw:name` attribute is usable with the following element: `<draw:measure>` 10.3.12.

The `draw:name` attribute has the data type `string` 18.2.

## 19.197.18 <draw:opacity>

The `draw:name` attribute specifies a name by which a `<draw:opacity>` element can be referenced.

The `draw:name` attribute is usable with the following element: `<draw:opacity>` 16.42.7.

The `draw:name` attribute has the data type `styleName` 18.3.32.

### 19.197.19 `<draw:page>`

The `draw:name` attribute specifies a name by which a `<draw:page>` element can be referenced. The name shall be unique within the document instance. If the attribute is not present, a consumer may generate a unique name.

The `draw:name` attribute is usable with the following element: `<draw:page>` 10.2.4.

The `draw:name` attribute has the data type `string` 18.2.

### 19.197.20 `<draw:page-thumbnail>`

The `draw:name` attribute specifies a name by which a `<draw:page-thumbnail>` element can be referenced.

The `draw:name` attribute is usable with the following element: `<draw:page-thumbnail>` 10.3.14.

The `draw:name` attribute has the data type `string` 18.2.

### 19.197.21 `<draw:param>`

The `draw:name` attribute specifies the name of a run-time parameter.

The `draw:name` attribute is usable with the following element: `<draw:param>` 10.4.9.

The `draw:name` attribute has the data type `string` 18.2.

### 19.197.22 `<draw:path>`

The `draw:name` attribute specifies a name by which a `<draw:path>` element can be referenced.

The `draw:name` attribute is usable with the following element: `<draw:path>` 10.3.7.

The `draw:name` attribute has the data type `string` 18.2.

### 19.197.23 `<draw:polygon>`

The `draw:name` attribute specifies a name by which a `<draw:polygon>` element can be referenced.

The `draw:name` attribute is usable with the following element: `<draw:polygon>` 10.3.5.

The `draw:name` attribute has the data type `string` 18.2.

### 19.197.24 `<draw:polyline>`

The `draw:name` attribute specifies a name by which a `<draw:polyline>` element can be referenced.

The `draw:name` attribute is usable with the following element: `<draw:polyline>` 10.3.4.

The `draw:name` attribute has the data type `string` 18.2.

### 19.197.25 `<draw:rect>`

The `draw:name` attribute specifies a name by which a `<draw:rect>` element can be referenced.

The `draw:name` attribute is usable with the following element: `<draw:rect>` 10.3.2.

The `draw:name` attribute has the data type `string` 18.2.

## 19.197.26 <draw:regular-polygon>

The `draw:name` attribute specifies a name by which a `<draw:regular-polygon>` element can be referenced.

The `draw:name` attribute is usable with the following element: `<draw:regular-polygon>` 10.3.6.

The `draw:name` attribute has the data type `string` 18.2.

## 19.197.27 <draw:stroke-dash>

The `draw:name` attribute specifies a name by which a `<draw:stroke-dash>` element can be referenced.

The `draw:name` attribute is usable with the following element: `<draw:stroke-dash>` 16.42.9.

The `draw:name` attribute has the data type `styleName` 18.3.32.

## 19.197.28 <office:annotation> (deprecated)

The `draw:name` attribute specifies a name for an `<office:annotation>` element.

The `draw:name` attribute has been deprecated in favor of the `office:name` attribute.

The `draw:name` attribute is usable with the following element: `<office:annotation>` 14.1.

The `draw:name` attribute has the data type `string` 18.2.

## 19.197.29 <svg:linearGradient>

The `draw:name` attribute specifies a name by which a `<svg:linearGradient>` element can be referenced.

SVG gradients are referenced by the name assigned to them by use of the `draw:fill-gradient-name` 20.120 attribute within a graphic style. SVG gradients cannot be referenced by a `draw:opacity-name` 20.153 attribute. The result of referencing a SVG gradient with a `draw:fill-gradient-name` attribute and an opacity gradient with a `draw:opacity-name` attribute at the same time is unspecified.

The `draw:name` attribute is usable with the following element: `<svg:linearGradient>` 16.42.2.

The `draw:name` attribute has the data type `styleName` 18.3.32.

## 19.197.30 <svg:radialGradient>

The `draw:name` attribute specifies a name by which a `<svg:radialGradient>` element can be referenced.

SVG gradients are referenced by the name assigned to them by use of the `draw:fill-gradient-name` 20.120 attribute within a graphic style. SVG gradients cannot be referenced by a `draw:opacity-name` 20.153 attribute. The result of referencing a SVG gradient with a `draw:fill-gradient-name` attribute and an opacity gradient with a `draw:opacity-name` attribute at the same time is unspecified.

The `draw:name` attribute is usable with the following element: `<svg:radialGradient>` 16.42.3.

The `draw:name` attribute has the data type `styleName` 18.3.32.

### 19.197.31 <dr3d:cube>

The `draw:name` attribute specifies a name by which a `<dr3d:cube>` can be referenced.

The `draw:name` attribute is usable with the following element: `<dr3d:cube>` 10.5.4.

The `draw:name` attribute has the data type `string` 18.2.

### 19.197.32 <dr3d:extrude>

The `draw:name` attribute specifies a name by which a `<dr3d:extrude>` can be referenced.

The `draw:name` attribute is usable with the following element: `<dr3d:extrude>` 10.5.6.

The `draw:name` attribute has the data type `string` 18.2.

### 19.197.33 <dr3d:rotate>

The `draw:name` attribute specifies a name by which a `<dr3d:rotate>` can be referenced.

The `draw:name` attribute is usable with the following element: `<dr3d:rotate>` 10.5.7.

The `draw:name` attribute has the data type `string` 18.2.

### 19.197.34 <dr3d:scene>

The `draw:name` attribute specifies a name by which a `<dr3d:scene>` can be referenced.

The `draw:name` attribute is usable with the following element: `<dr3d:scene>` 10.5.2.

The `draw:name` attribute has the data type `string` 18.2.

### 19.197.35 <dr3d:sphere>

The `draw:name` attribute specifies a name by which a `<dr3d:sphere>` can be referenced.

The `draw:name` attribute is usable with the following element: `<dr3d:sphere>` 10.5.5.

The `draw:name` attribute has the data type `string` 18.2.

## 19.198 draw:nav-order

The `draw:nav-order` attribute defines a navigation sequence for the graphical elements included in a `<draw:page>` element. Its value is a sequence of IDREFs. It shall include all graphic elements in the page that are not contained within a `<draw:g>` element. Any `<draw:g>` elements are included in that list as a single graphical element.

The value of this attribute is the ordering of graphics as set by the document author.

**Note:** The ordering within any `<draw:g>` 10.3.15 element is set by the value of its `draw:z-index` 19.231 attribute.

The `draw:nav-order` attribute is usable with the following element: `<draw:page>` 10.2.4.

The `draw:nav-order` attribute has the data type `IDREFS` 18.2.

## 19.199 draw:nowrap

The `draw:nowrap` attribute specifies that an image map element and its associated area are inactive.

The `draw:nowrap` attribute is usable with the following elements: `<draw:area-circle>` 10.4.13.4, `<draw:area-polygon>` 10.4.13.5 and `<draw:area-rectangle>` 10.4.13.3.

The only value of the `draw:nonref` attribute is `nonref`.

## 19.200 `draw:notify-on-update-of-ranges`

The `draw:notify-on-update-of-ranges` attribute specifies that a `<draw:object>` will be notified if content within a given ranges changes.

The defined values for the `draw:notify-on-update-of-ranges` attribute are:

- a list of cell ranges.
- a table name.

If a table name is specified, the `<draw:object>` element is notified if any of the table's cells change.

The `draw:notify-on-update-of-ranges` attribute is usable with the following element: `<draw:object>` 10.4.6.2.

The values of the `draw:notify-on-update-of-ranges` attribute are a value of type `cellRangeAddressList` 18.3.6 or a value of type `string` 18.2.

## 19.201 `draw:object`

The `draw:object` attribute is equivalent to the `object` attribute of an `<applet>` element in HTML. See. §13.4 of [HTML4].

The `draw:object` attribute is usable with the following element: `<draw:applet>` 10.4.7.

The `draw:object` attribute has the data type `string` 18.2.

## 19.202 `draw:opacity`

The `draw:opacity` attribute specifies the opacity for an image or graphic object. The defined value range for the `draw:opacity` attribute is 0% to 100%, where 0% is fully transparent and 100% is fully opaque.

Use of the `draw:opacity` attribute disables any transparency effect and set the opacity for the fill area of a graphic object.

The `draw:opacity` attribute is usable with the following elements: `<style:background-fill>` 17.26 and `<style:background-image>` 17.3.

The `draw:opacity` attribute has the data type `zeroToHundredPercent` 18.3.42.

## 19.203 `draw:page-number`

The `draw:page-number` attribute specifies the number of a page that is displayed as a thumbnail. For thumbnails on notes pages, the value of this attribute is fixed to the drawing page of a notes page. For thumbnails on handout master pages, the value of this attribute is the order in which the pages are previewed on the handout.

The `draw:page-number` attribute is usable with the following element: `<draw:page-thumbnail>` 10.3.14.

The `draw:page-number` attribute has the data type `positiveInteger` 18.2.

## 19.204 `draw:path-stretchpoint-x`

The `draw:path-stretchpoint-x` attribute specifies that a drawing shape is stretched horizontally.

If specified, and if the x/y aspect ratio of the `svg:viewBox` 19.574 is lower than the x/y aspect ratio of the shape size then the whole path is stretched horizontally at `draw:path-stretchpoint-x`.

Stretching is done by:

- 1) intersecting the path vertically at `draw:path-stretchpoint-x`.
- 2) moving the right path fragments to the right border of the drawing shape, using its original aspect ratio. The left path fragment remains unmodified at the left border of drawing the shape

The `draw:path-stretchpoint-x` attribute is usable with the following element:  
`<draw:enhanced-geometry>` 10.6.2.

The `draw:path-stretchpoint-x` attribute has the data type `double` 18.2.

## 19.205 `draw:path-stretchpoint-y`

The `draw:path-stretchpoint-y` attribute specifies that a drawing shape is stretched vertically. If the x/y aspect ratio of the `svg:viewBox` 19.574 is greater than the x/y aspect ratio of the shape size then the whole path is stretched vertically at `draw:path-stretchpoint-y`.

Stretching is done by:

- 1) intersecting the path horizontally at `draw:path-stretchpoint-y`.
- 2) moving the bottom path fragments to the bottom border of the drawing shape, using its original aspect ratio.

The top path fragment remains unmodified at the top border of drawing the shape

The `draw:path-stretchpoint-y` attribute is usable with the following element:  
`<draw:enhanced-geometry>` 10.6.2.

The `draw:path-stretchpoint-y` attribute has the data type `double` 18.2.

## 19.206 `draw:points`

The `draw:points` attribute stores a sequence of points, which are connected by straight lines. Each point consists of two coordinates. The coordinates are separated by a comma "," (U+002C, COMMA) and the points are separated by white spaces. The coordinates are relative to the coordinate system established by the `svg:viewBox` 19.574 attribute.

The `draw:points` attribute is usable with the following elements: `<draw:area-polygon>` 10.4.13.5, `<draw:contour-polygon>` 10.4.11.2, `<draw:polygon>` 10.3.5 and `<draw:polyline>` 10.3.4.

The `draw:points` attribute has the data type `points` 18.3.26.

## 19.207 `draw:protected`

The `draw:protected` attribute specifies if drawing objects in a layer are protected from modification.

The defined values for the `draw:protected` attribute are:

- `false`: objects in a layer are not protected from modification.
- `true`: objects in a layer are protected from modification.

The default value for this attribute is `false`.

The `draw:protected` attribute is usable with the following element: `<draw:layer>` 10.2.3.

The `draw:protected` attribute has the data type `boolean` 18.3.3.

## 19.208 `draw:recreate-on-edit`

The `draw:recreate-on-edit` attribute specifies if the contour of an image or object should be recreated automatically when the image or object is edited.

The defined values for the `draw:recreate-on-edit` attribute are:

- `false`: the contour of an image or object is not recreated automatically if the image or object is edited.
- `true`: the contour of an image or object is recreated automatically if the image or object is edited.

The `draw:recreate-on-edit` attribute is usable with the following elements:

`<draw:contour-path>` 10.4.11.3 and `<draw:contour-polygon>` 10.4.11.2.

The `draw:recreate-on-edit` attribute has the data type `boolean` 18.3.3.

## 19.209 `draw:rotation`

The `draw:rotation` attribute specifies the rotation angle of a hatch axis. The direction is clockwise, 0 degrees is the vertical axis.

The `draw:rotation` attribute is usable with the following element: `<draw:hatch>` 16.42.5.

The `draw:rotation` attribute has the data type `angle` 18.3.1.

## 19.210 `draw:shape-id`

The `draw:shape-id` attribute identifies the shape to which an effect defined by a presentation shape effect element is applied.

The `draw:shape-id` attribute is usable with the following elements:

`<presentation:dim>` 10.8.7, `<presentation:hide-shape>` 10.8.5,  
`<presentation:hide-text>` 10.8.6, `<presentation:play>` 10.8.8,  
`<presentation:show-shape>` 10.8.3 and `<presentation:show-text>` 10.8.4.

The `draw:shape-id` attribute has the data type `IDREF` 18.2.

## 19.211 `draw:sharpness`

The `draw:sharpness` attribute is only evaluated for a `<draw:regular-polygon>` 10.3.6 element that has the attribute `draw:concave="true"`. In this case, the outline of the drawing object is based on a basic model of a star-shaped polygon, where the corner points are alternately located on two concentric circles, a primary and a secondary circle. The number of points in the number as specified with the `draw:corners` 19.128 attribute are evenly distributed on the primary circle. Let  $M$  be the center of the circle, and let  $P$  and  $Q$  be two adjacent points on the primary circle, and  $R$  the center of the shorter of the two arcs  $PQ$ .

Now find the point  $T$  with  $MT = MR \cdot (1 - s\%)$  on the connecting line of points  $M$  and  $R$ , where  $s\%$  is the value of the `draw:sharpness` attribute. The radius of the secondary circle is such that  $T$  lies on this secondary circle. The points  $P$  and  $T$  and the points  $T$  and  $Q$  are then connected by straight sections. With an odd value of `draw:corners` the first corner point is located at top of the primary circle, with an even value at top of the secondary circle.

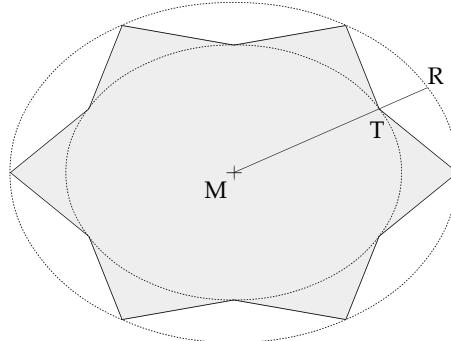
Note: The drawing object has twice as many corners points in all as specified in the attribute `draw:corners`.

The value  $s\%$  of the `draw:sharpness` attribute is a percent value. The special case 0% means that primary and secondary circle are identical. The special case 100% means that the secondary circle degenerates to a point, the center point of the polygon. The value of the `draw:sharpness` attribute may be negative and may be greater than 100%. Note that for small absolute value of the `draw:sharpness` attribute, the described polygon is in fact convex.

The basic model is scaled—thereby the circle might become an ellipse, so that the bounding box of the polygon has the width and height as specified in the `svg:width` 19.575 and `svg:height` 19.543 attributes of the `<draw:regular-polygon>` element.

Note: Because of intercept theorems the equation

$\overline{MT} = \overline{MR} \cdot (1 - s\%)$  is valid for the scaled outline too.



The `draw:sharpness` attribute is usable with the following element: `<draw:regular-polygon>` 10.3.6.

The `draw:sharpness` attribute has the data type `percent` 18.3.24.

## 19.212 draw:start

The `draw:start` attribute specifies the start value for the `<draw:opacity>` element, which interpolates between the value of this attribute and `draw:end` 19.139. The defined value range for the `draw:start` attribute is 0% to 100%, where 0% is transparent and 100% is opaque.

The `draw:start` attribute is usable with the following element: `<draw:opacity>` 16.42.7.

The `draw:start` attribute has the data type `zeroToHundredPercent` 18.3.42.

## 19.213 draw:start-angle

The `draw:start-angle` attribute specifies the start angle of a section, cut, or arc for circles where the `draw:kind` 19.188 attribute value is `section`, `cut` or `arc`.

The `draw:start-angle` attribute is usable with the following elements: `<draw:circle>` 10.3.8 and `<draw:ellipse>` 10.3.9.

The `draw:start-angle` attribute has the data type `angle` 18.3.1.

## 19.214 draw:start-color

The `draw:start-color` attribute specifies the start value for the `<draw:gradient>` element, which interpolates between the value of this attribute and `draw:end-color` 19.141.

The `draw:start-color` attribute is usable with the following element: `<draw:gradient>` 16.42.1.

The `draw:start-color` attribute has the data type `color` 18.3.9.

## 19.215 draw:start-glue-point

The `draw:start-glue-point` attribute identifies the glue point in a shape where a connector starts by its number. See 10.3.16. Glue point numbers are defined by the `draw:id` attributes of the glue point elements `<draw:glue-point>`. See 19.187.

If the connector is not connected to a shape, this attribute is ignored.

The `draw:start-glue-point` attribute is usable with the following element:  
`<draw:connector>` 10.3.10.

The `draw:start-glue-point` attribute has the data type `nonNegativeInteger` 18.2.

## 19.216 draw:start-intensity

The `draw:start-intensity` attribute specifies the intensity of the gradient's start color as a percentage value. If it is not specified, the color is used at 100% intensity. The defined value range for the `draw:start-intensity` attribute is 0% to 100%.

The `draw:start-intensity` attribute is usable with the following element:  
`<draw:gradient>` 16.42.1.

The `draw:start-intensity` attribute has the data type `zeroToHundredPercent` 18.3.42.

## 19.217 draw:start-shape

The `draw:start-shape` attribute specifies a drawing shape to which the start of a connector is connected by its ID.

The `draw:start-shape` attribute is usable with the following element:  
`<draw:connector>` 10.3.10.

The `draw:start-shape` attribute has the data type `IDREF` 18.2.

## 19.218 draw:style

### 19.218.1 General

The `draw:style` attribute specifies the rendering of graphic objects.

### 19.218.2 <draw:gradient>

The `draw:style` attribute specifies a rendering for a gradient.

The defined values for the `draw:style` attribute are:

- `axial`: defines a bi-linear gradient that is also known as a reflected gradient or mirrored linear gradient. It is created as a linear gradient that is mirrored (or reflected) along its axis.
- `ellipsoid`: defines a gradient where the colors are blend along the radius from the center of an ellipsoid as defined by the `draw:cx` 19.129 and `draw:cy` 19.130 attributes. The length of the semi major-axis is the width of the filled area and the length of the semi-minor axis is the height of the filled area. The area outside the ellipse is filled with the end color.
- `linear`: defines a gradient where the colors blend along the linear axis of the gradient. The axis of the gradient is specified with the `draw:angle` 19.112 attribute clockwise to the vertical axis.

- **radial**: defines a gradient where the colors are blend along the radius from the center of a circle as defined with the `draw:cx` and `draw:cy` attributes. The outside of the circle is filled with the end color.
- **rectangle**: defines a gradient that produces a rectangular blend from the center of the rectangle to the shortest of the 4 borders. The center of the rectangle is defined with the attributes `draw:cx` and `draw:cy`. The width of the rectangle is the width of the filled area, the height of the rectangle is the height of the filled area. The outside of the square is filled with the end color.
- **square**: defines a gradient that produces a square blend, imitating the visual perspective in a corridor or the aerial view of a pyramid. Also known as "box gradient" and "pyramidal gradient". The center of the square is defined with the `draw:cx` and `draw:cy` attributes. The width and height of the square is the minimum value of either the width or the height of the filled area. The outside of the square is filled with the end color.

The `draw:style` attribute is usable with the following element: `<draw:gradient>` 16.42.1.

The values of the `draw:style` attribute are `linear`, `axial`, `radial`, `ellipsoid`, `square` or `rectangular`.

### 19.218.3 `<draw:hatch>`

The `draw:style` attribute specifies a rendering for a hatch.

The defined values for the `draw:style` attribute are:

- **double**: defines a hatch that is made of a set of parallel lines along the axis of the hatch and a set of parallel lines perpendicular to the axis of the hatch.
- **single**: defines a hatch that is made of parallel lines along the axis of the hatch.
- **triple**: defines a hatch that is made of a set of parallel lines along the axis of the hatch, a set of parallel lines perpendicular to the axis of the hatch and a set of lines along 45 degree clockwise to the axis of the hatch.

The `draw:style` attribute is usable with the following element: `<draw:hatch>` 16.42.5.

The values of the `draw:style` attribute are `single`, `double` or `triple`.

### 19.218.4 `<draw:opacity>`

The `draw:style` attribute specifies the transparency of a graphic object.

The defined values for the `draw:style` attribute are:

- **axial**: 19.218.2
- **ellipsoid**: 19.218.2
- **linear**: 19.218.2
- **rectangle**: 19.218.2
- **square**: 19.218.2

The transparency of an object is blended in contrast to the blending of colors described for `draw:style` on `<draw:gradient>`.

The `draw:style` attribute is usable with the following element: `<draw:opacity>` 16.42.7.

The values of the `draw:style` attribute are `linear`, `axial`, `radial`, `ellipsoid`, `square` or `rectangular`.

## 19.218.5 <draw:stroke-dash>

The `draw:style` attribute specifies a rendering for a stroke-dash.

This attribute is evaluated for a shape if its style does not contain an `svg:stroke-linecap` 20.171 attribute.

The defined values for the `draw:style` attribute are:

- `rect`: dash has a rectangular shape.
- `round`: dash has a round shape.

The `draw:style` attribute is usable with the following element: <`draw:stroke-dash`> 16.42.9.

The values of the `draw:style` attribute are `rect` or `round`.

## 19.219 draw:style-name

### 19.219.1 General

The `draw:style-name` attribute specifies the name of a <`style:style`> 16.2 element whose `style:family` 19.480 attribute value varies depending upon the element where it appears.

### 19.219.2 <dr3d:cube>

The `draw:style-name` attribute specifies the name of a <`style:style`> element with a `style:family` attribute value of `graphic`.

The `draw:style-name` attribute is usable with the following element: <`dr3d:cube`> 10.5.4.

The `draw:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.219.3 <dr3d:extrude>

The `draw:style-name` attribute specifies the name of a <`style:style`> element with a `style:family` attribute value of `graphic`.

The `draw:style-name` attribute is usable with the following element: <`dr3d:extrude`> 10.5.6.

The `draw:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.219.4 <dr3d:rotate>

The `draw:style-name` attribute specifies the name of a <`style:style`> element with a `style:family` attribute value of `graphic`.

The `draw:style-name` attribute is usable with the following element: <`dr3d:rotate`> 10.5.7.

The `draw:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.219.5 <dr3d:scene>

The `draw:style-name` attribute specifies the name of a <`style:style`> element with a `style:family` attribute value of `graphic`.

The `draw:style-name` attribute is usable with the following element: <`dr3d:scene`> 10.5.2.

The `draw:style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.219.6 <dr3d:sphere>

The `draw:style-name` attribute specifies the name of a `<style:style>` element with a `style:family` attribute value of `graphic`.

The `draw:style-name` attribute is usable with the following element: `<dr3d:sphere>` 10.5.5.

The `draw:style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.219.7 <draw:caption>

The `draw:style-name` attribute specifies the name of a `<style:style>` element with a `style:family` attribute value of `graphic`.

The `draw:style-name` attribute is usable with the following element: `<draw:caption>` 10.3.11.

The `draw:style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.219.8 <draw:circle>

The `draw:style-name` attribute specifies the name of a `<style:style>` element with a `style:family` attribute value of `graphic`.

The `draw:style-name` attribute is usable with the following element: `<draw:circle>` 10.3.8.

The `draw:style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.219.9 <draw:connector>

The `draw:style-name` attribute specifies the name of a `<style:style>` element with a `style:family` attribute value of `graphic`.

The `draw:style-name` attribute is usable with the following element: `<draw:connector>` 10.3.10.

The `draw:style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.219.10 <draw:control>

The `draw:style-name` attribute specifies the name of a `<style:style>` element with a `style:family` attribute value of `graphic`.

The `draw:style-name` attribute is usable with the following element: `<draw:control>` 10.3.13.

The `draw:style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.219.11 <draw:custom-shape>

The `draw:style-name` attribute specifies the name of a `<style:style>` element with a `style:family` attribute value of `graphic`.

The `draw:style-name` attribute is usable with the following element: `<draw:custom-shape>` 10.6.1.

The `draw:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.219.12 <draw:ellipse>

The `draw:style-name` attribute specifies the name of a `<style:style>` element with a `style:family` attribute value of `graphic`.

The `draw:style-name` attribute is usable with the following element: `<draw:ellipse>` 10.3.9.

The `draw:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.219.13 <draw:frame>

The `draw:style-name` attribute specifies the name of a `<style:style>` element with a `style:family` attribute value of `graphic`.

The `draw:style-name` attribute is usable with the following element: `<draw:frame>` 10.4.2.

The `draw:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.219.14 <draw:g>

The `draw:style-name` attribute specifies the name of a `<style:style>` element with a `style:family` attribute value of `graphic`.

The `draw:style-name` attribute is usable with the following element: `<draw:g>` 10.3.15.

The `draw:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.219.15 <draw:line>

The `draw:style-name` attribute specifies the name of a `<style:style>` element with a `style:family` attribute value of `graphic`.

The `draw:style-name` attribute is usable with the following element: `<draw:line>` 10.3.3.

The `draw:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.219.16 <draw:measure>

The `draw:style-name` attribute specifies the name of a `<style:style>` element with a `style:family` attribute value of `graphic`.

The `draw:style-name` attribute is usable with the following element: `<draw:measure>` 10.3.12.

The `draw:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.219.17 <draw:page>

The `draw:style-name` attribute specifies the name of a `<style:style>` element with a `style:family` attribute value of `drawing-page`.

The `draw:style-name` attribute is usable with the following element: `<draw:page>` 10.2.4.

The `draw:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.219.18 <draw:page-thumbnail>

The `draw:style-name` attribute specifies the name of a `<style:style>` element with a `style:family` attribute value of `graphic`.

The `draw:style-name` attribute is usable with the following element: `<draw:page-thumbnail>` 10.3.14.

The `draw:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.219.19 `<draw:path>`

The `draw:style-name` attribute specifies the name of a `<style:style>` element with a `style:family` attribute value of `graphic`.

The `draw:style-name` attribute is usable with the following element: `<draw:path>` 10.3.7.

The `draw:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.219.20 `<draw:polygon>`

The `draw:style-name` attribute specifies the name of a `<style:style>` element with a `style:family` attribute value of `graphic`.

The `draw:style-name` attribute is usable with the following element: `<draw:polygon>` 10.3.5.

The `draw:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.219.21 `<draw:polyline>`

The `draw:style-name` attribute specifies the name of a `<style:style>` element with a `style:family` attribute value of `graphic`.

The `draw:style-name` attribute is usable with the following element: `<draw:polyline>` 10.3.4.

The `draw:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.219.22 `<draw:rect>`

The `draw:style-name` attribute specifies the name of a `<style:style>` element with a `style:family` attribute value of `graphic`.

The `draw:style-name` attribute is usable with the following element: `<draw:rect>` 10.3.2.

The `draw:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.219.23 `<draw:regular-polygon>`

The `draw:style-name` attribute specifies the name of a `<style:style>` element with a `style:family` attribute value of `graphic`.

The `draw:style-name` attribute is usable with the following element: `<draw:regular-polygon>` 10.3.6.

The `draw:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.219.24 `<office:annotation>`

The `draw:style-name` attribute specifies the name of a `<style:style>` element with a `style:family` attribute value of `graphic`.

The `draw:style-name` attribute is usable with the following element: `<office:annotation>` 14.1.

The `draw:style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.219.25 <presentation:notes>

The `draw:style-name` attribute specifies the name of a `<style:style>` element with a `style:family` attribute value of `drawing-page`.

The `draw:style-name` attribute is usable with the following element:  
`<presentation:notes>` 16.19.

The `draw:style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.219.26 <style:handout-master>

The `draw:style-name` attribute specifies the name of a `<style:style>` element with a `style:family` attribute value of `drawing-page`.

The `draw:style-name` attribute is usable with the following element: `<style:handout-master>` 10.2.1.

The `draw:style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.219.27 <style:master-page>

The `draw:style-name` attribute specifies the name of a `<style:style>` element with a `style:family` attribute value of `drawing-page`.

The `draw:style-name` attribute is usable with the following element: `<style:master-page>` 16.9.

The `draw:style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.220 draw:text-areas

The `draw:text-areas` attribute specifies up to two text areas. If the attribute specifies one text area, this text area is used to position and align the text. If the attribute specifies two text areas, the second text area is used if a style that is assigned to the text has a `style:writing-mode` 20.404 attribute with value `tb-rl` or `tb`. Otherwise, the first text area is used. If the `draw:text-areas` attribute is omitted, the entire area of the shape is used to position and align the text.

An area consists of four parameters:

The first parameter specifies the left side of a text area.

The second parameter specifies the top side of a text area.

The third parameter specifies the right side of a text area.

The fourth parameter specifies the bottom side of a text area.

A parameter can also have one of the following enhancements:

- A "?" (U+003F, QUESTION MARK) is used to mark the beginning of a formula name. The result of the `<draw:equation>` 10.2.5 element's `draw:formula` 19.171 attribute whose `draw:name` 19.197 attribute is the formula name that is used as the value.
- If "\$" (U+0024, DOLLAR SIGN) precedes an integer value, the value is an index to a `draw:modifiers` 19.196 attribute. The corresponding modifier value is used as the parameter value.

The value of a `draw:text-areas` attribute is formally defined as:

```

textareas ::= textareasequence
textareasequence ::= textarea ( ' ' + textarea )?
textarea ::= position ' ' + position ' ' + position ' ' + position
position ::= formula | modifier | number
formula ::= '?' name
modifier ::= '$' integer
number ::= sign? float | sign? integer
float ::= fractional exponent? | integer exponent
fractional ::= integer? '.' integer | integer '.'
exponent ::= ( 'e' | 'E' ) sign? integer
sign ::= '+' | '-'
integer ::= [0-9] +
name ::= [^ ] +

```

The `draw:text-areas` attribute is usable with the following element: `<draw:enhanced-geometry>` 10.6.2.

The `draw:text-areas` attribute has the data type `string` 18.2.

## 19.221 `draw:text-path`

The `draw:text-path` attribute specifies if text is displayed on a text path.

The defined values for the `draw:text-path` attribute are:

- `false`: text is not displayed on a text path.
- `true`: text is displayed on a text path.

The default value for this attribute is `false`.

The `draw:text-path` attribute is usable with the following element: `<draw:enhanced-geometry>` 10.6.2.

The `draw:text-path` attribute has the data type `boolean` 18.3.3.

## 19.222 `draw:text-path-allowed`

The `draw:text-path-allowed` attribute specifies whether the user interface of a consumer that supports the `draw:text-path` 19.221 attribute should allow modification of the value of the `draw:text-path` attribute.

The defined values for the `draw:text-path-allowed` attribute are:

- `false`: the value of the `draw:text-path` attribute should not be modifiable by the user interface.
- `true`: the value of the `draw:text-path` attribute should be modifiable by the user interface.

The default value for this attribute is `false`.

The `draw:text-path-allowed` attribute is usable with the following element: `<draw:enhanced-geometry>` 10.6.2.

The `draw:text-path-allowed` attribute has the data type `boolean` 18.3.3.

## 19.223 `draw:text-path-mode`

The `draw:text-path-mode` attribute specifies how text is drawn on a path.

The defined values for the `draw:text-path-mode` attribute are:

- `normal`: text is drawn along the path without scaling;
- `path`: text is fitted to a path;

- `shape`: text is fitted to the bounding box of a shape.

The default value for this attribute is `normal`.

The `draw:text-path-mode` attribute is usable with the following element:  
`<draw:enhanced-geometry>` 10.6.2.

The values of the `draw:text-path-mode` attribute are `normal`, `path` or `shape`.

## 19.224 `draw:text-path-same-letter-heights`

The `draw:text-path-same-letter-heights` attribute specifies if all letters in a custom shape have the same height.

The defined values for the `draw:text-path-same-letter-heights` attribute are:

- `false`: letters in a custom shape need not have the same height.
- `true`: letters in a custom shape have the same height.

The default value for this attribute is `false`.

The `draw:text-path-same-letter-heights` attribute is usable with the following element: `<draw:enhanced-geometry>` 10.6.2.

The `draw:text-path-same-letter-heights` attribute has the data type `boolean`  
`18.3.3`.

## 19.225 `draw:text-path-scale`

The `draw:text-path-scale` attribute specifies the scaling of a text path.

The defined values for the `draw:text-path-scale` attribute are:

- `path`: text scaling is determined by the length of the path from the `draw:enhanced-path` 19.145 attribute.
- `shape`: text scaling is determined by the width of a shape.

The default value for this attribute is `path`.

The `draw:text-path-scale` attribute is usable with the following element:  
`<draw:enhanced-geometry>` 10.6.2.

The values of the `draw:text-path-scale` attribute are `path` or `shape`.

## 19.226 `draw:text-rotate-angle`

The `draw:text-rotate-angle` attribute specifies the angle by which text within a custom shape is rotated in addition to the rotation included in the shape's `draw:transform` 19.228 attribute.

The default value for this attribute is `0`.

The `draw:text-rotate-angle` attribute is usable with the following element:  
`<draw:enhanced-geometry>` 10.6.2.

The `draw:text-rotate-angle` attribute has the data type `angle` 18.3.1.

## 19.227 `draw:text-style-name`

The `draw:text-style-name` attribute specifies a style for formatting of text in a shape.

The value of this attribute is the name of a `<style:style>` 16.2 element with a `style:family` 19.480 attribute value of `paragraph`.

The `draw:text-style-name` attribute is usable with the following elements: `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:connector>` 10.3.10, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:line>` 10.3.3, `<draw:measure>` 10.3.12, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2, `<draw:regular-polygon>` 10.3.6 and `<office:annotation>` 14.1.

The `draw:text-style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.228 `draw:transform`

The `draw:transform` attribute specifies a list of transformations that can be applied to a drawing shape.

The value of this attribute is a list of transform definitions, which are applied to the drawing shape in the order in which they are listed. The transform definitions in the list shall be separated by a white space and/or a comma "," (U+002C, COMMA). Unless otherwise stated, the parameters of the transform definitions are double values (18.2)

The defined transforms are:

- `matrix(<a> <b> <c> <d> <e> <f>)`, specifies a transformation in the form of a transformation matrix of six values. "The values describe a standard 3x2 homogeneous transformation matrix in column-major order, where the right column (`<e>`, `<f>`) describes the translation.
- `rotate(<rotate-angle>)`, specifies a rotation by `<rotate-angle>` in radians about the origin of the shape's coordinate system.
- `scale(<sx> [<sy>])`, specifies a scale operation by `<sx>` and `<sy>`. If `<sy>` is not provided, it is assumed to be equal to `<sx>`.
- `skewX(<skew-angle>)`, specifies a skew transformation by `<skew-angle>` in radians along the x-axis.
- `skewY(<skew-angle>)`, specifies a skew transformation by `<skew-angle>` in radians along the y-axis.
- `translate(<tx> [<ty>])`, specifies a translation by `<tx>` and `<ty>`, where `<tx>` and `<ty>` are lengths (18.3.19). If `<ty>` is not provided, it is assumed to be zero.

The `draw:transform` attribute is usable with the following elements: `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:connector>` 10.3.10, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:line>` 10.3.3, `<draw:measure>` 10.3.12, `<draw:page-thumbnail>` 10.3.14, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2, `<draw:regular-polygon>` 10.3.6 and `<office:annotation>` 14.1.

The `draw:transform` attribute has the data type `string` 18.2.

## 19.229 `draw:type`

### 19.229.1 General

The `draw:type` attribute specifies a drawing type.

### 19.229.2 `<draw:connector>`

The `draw:type` attribute specifies the line or series of lines that connect two glue points.

The defined values for the `draw:type` attribute are:

- `curve`: a curve connector draws a single curved line between the two glue points of the connected objects.
- `line`: a line connector draws one straight line between the two glue points of the connected objects.
- `lines`: a lines connector leaves two connecting objects with straight lines and connects them with a straight (not necessarily perpendicular) line.
- `standard`: a standard connector leaves two connecting objects with straight lines and connects them with one or more straight perpendicular lines that do not intersect the connected shapes' bounding boxes.

For a `<draw:connector>` element the default value for this attribute is `standard`.

The `draw:type` attribute is usable with the following element: `<draw:connector>` 10.3.10.

The values of the `draw:type` attribute are `standard`, `lines`, `line` or `curve`.

### 19.229.3 `<draw:enhanced-geometry>`

The `draw:type` attribute specifies the name of a shape type that is rendering engine dependent.

The defined values for the `draw:type` attribute are:

- `non-primitive`: no shape type is available.
- a value of type string: a rendering-engine-specific name of a shape.

The value of this attribute shall not influence the geometry of the shape.

**Note:** The shape type can be used to offer specialized user interfaces for classes of shapes, for example for arrows, smileys, etc.

For a `<draw:enhanced-geometry>` element the default value for this attribute is `non-primitive`.

The `draw:type` attribute is usable with the following element: `<draw:enhanced-geometry>` 10.6.2.

The values of the `draw:type` attribute are `non-primitive` or a value of type string 18.2.

### 19.230 `draw:value`

The `draw:value` attribute specifies the value of a run-time parameter specified by a `draw:name` 19.197 attribute.

The `draw:value` attribute is usable with the following element: `<draw:param>` 10.4.9.

The `draw:value` attribute has the data type string 18.2.

### 19.231 `draw:z-index`

The `draw:z-index` attribute defines a rendering order for shapes in a document instance. In the absence of this attribute, shapes are rendered in the order in which they appear in the document.

The `draw:z-index` values increase from back to front.

For a shape on which the `style:run-through` 20.351 attribute with value `foreground` is in effect, producers should not generate a `draw:z-index` value that is smaller than the value of any `draw:z-index` on a shape on which the `style:run-through` attribute with value `background` is in effect.

Producers shall not generate a `draw:z-index` for shapes that are children of a `<draw:g>` element 10.3.15 or a `<dr3d:scene>` element 10.5.2.

The `draw:z-index` attribute is usable with the following elements: `<dr3d:cube>` 10.5.4, `<dr3d:extrude>` 10.5.6, `<dr3d:rotate>` 10.5.7, `<dr3d:scene>` 10.5.2, `<dr3d:sphere>` 10.5.5, `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:connector>` 10.3.10, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:g>` 10.3.15, `<draw:line>` 10.3.3, `<draw:measure>` 10.3.12, `<draw:page-thumbnail>` 10.3.14, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2, `<draw:regular-polygon>` 10.3.6 and `<office:annotation>` 14.1.

The `draw:z-index` attribute has the data type `nonNegativeInteger` 18.2.

## 19.232 fo:column-count

The `fo:column-count` attribute specifies the number of columns in a section.

The `fo:column-count` attribute is usable with the following element: `<style:columns>` 17.12.

The `fo:column-count` attribute has the data type `positiveInteger` 18.2.

## 19.233 fo:column-gap

The `fo:column-gap` attribute specifies the gap between columns for `<style:columns>` elements that do not contain individual `<style:column>` elements.

**Note:** This attribute has the same name as an [XSL] property but it is attached to a different element.

In the OpenDocument XSL-compatible namespace, the `fo:column-gap` attribute does not support `inherit` and `percentage` values.

The `fo:column-gap` attribute is usable with the following element: `<style:columns>` 17.12.

The `fo:column-gap` attribute has the data type `length` 18.3.19.

## 19.234 fo:country

### 19.234.1 <text:alphabetical-index-source>

The `fo:country` attribute specifies the country code used for sorting an alphabetical index.

The `fo:country` attribute is usable with the following element: `<text:alphabetical-index-source>` 8.8.2.

The `fo:country` attribute has the data type `countryCode` 18.3.11.

### 19.234.2 <text:bibliography-configuration>

The `fo:country` attribute specifies the country code used for sorting a bibliographic index.

The `fo:country` attribute is usable with the following element: `<text:bibliography-configuration>` 16.31.6.

The `fo:country` attribute has the data type `countryCode` 18.3.11.

## 19.235 fo:end-indent

The `fo:end-indent` attribute specifies the right space of a column. Together with the `fo:start-indent` attribute of an immediately adjacent and following column, it establishes the gap between the columns. 19.245

See §7.10.8 of [XSL].

In the OpenDocument XSL-compatible namespace, the `fo:end-indent` attribute does not support `inherit` and percentage values.

The default value for this attribute is `0cm`.

The `fo:end-indent` attribute is usable with the following element: `<style:column>` 17.13.

The `fo:end-indent` attribute has the data type `length` 18.3.19.

## 19.236 fo:language

### 19.236.1 General

See §7.9.2 of [XSL]. Compared to the XSL `language` attribute, the value range of this attribute is extended to additional parts of ISO 639.

This attribute may be ignored if it is not specified together with a `fo:country` 20.188 attribute.

In the OpenDocument XSL-compatible namespace, the `fo:language` attribute does not support `inherit` and `none` values.

### 19.236.2 <text:alphabetical-index-source>

The `fo:language` attribute specifies the language code used for sorting an alphabetical index.

The `fo:language` attribute is usable with the following element: `<text:alphabetical-index-source>` 8.8.2.

The `fo:language` attribute has the data type `languageCode` 18.3.18.

### 19.236.3 <text:bibliography-configuration>

The `fo:language` attribute specifies the language code used for sorting a bibliographic index.

The `fo:language` attribute is usable with the following element: `<text:bibliography-configuration>` 16.31.6.

The `fo:language` attribute has the data type `languageCode` 18.3.18.

## 19.237 fo:margin-left

See §7.10.3 of [XSL].

The `fo:margin-left` attribute specifies the left margins for the text lines of a list item. The attribute has the same meaning as the formatting property attribute `fo:margin-left`. 20.207 Its value is only considered for paragraphs inside a list item whose paragraph styles do not specify a `fo:text-indent` attribute or a `fo:margin-left` attribute.

The `fo:margin-left` attribute is usable with the following element: `<style:list-level-label-alignment>` 17.20.

The `fo:margin-left` attribute has the data type `length` 18.3.19.

## 19.238 fo:max-height

See §7.14.6 of [XSL].

The `fo:max-height` attribute specifies the maximum height of a text box when the minimum height has been set using the `fo:min-height` 20.212 attribute. When this maximum value is reached, the text box stops increasing in height. The units of measure for this attribute shall match those used for the corresponding `fo:min-height` attribute.

If the anchor for the text box is in a table cell, a percentage value is relative to the size of the surrounding table cell. If the anchor for the text box is in a text box, a percentage value is relative to the size of the surrounding text box. In other cases, percentage values are relative to the width or height of the page or window.

In the OpenDocument XSL-compatible namespace, the `fo:max-height` attribute does not support the `none` value.

The `fo:max-height` attribute is usable with the following element: `<draw:text-box>`  
10.4.3.

The values of the `fo:max-height` attribute are a value of type `length` 18.3.19 or a value of type `percent` 18.3.24.

## 19.239 fo:max-width

See §7.14.7 of [XSL].

The `fo:max-width` attribute specifies the maximum width of a text box when the minimum width have been set using the `fo:min-width` 20.213 attribute. When this maximum value is reached, the text box stops increasing in width. The units of measure for this attribute shall match those used for the corresponding `fo:min-width` attribute.

If the anchor for the text box is in a table cell, a percentage value is relative to the size of the surrounding table cell. If the anchor for the text box is in a text box, a percentage value is relative to the size of the surrounding text box. In other cases, percentage values are relative to the width or height of the page or window.

The `fo:max-width` attribute is usable with the following element: `<draw:text-box>`  
10.4.3.

The values of the `fo:max-width` attribute are a value of type `length` 18.3.19 or a value of type `percent` 18.3.24.

## 19.240 fo:min-height

See §7.14.8 of [XSL].

If set, it overrides the height of a text box specified by the `svg:height` 19.543 attribute of the surrounding `<draw:frame>` 10.4.2 element.

If the anchor for a text box is in a table cell, a percentage value is relative to the surrounding table box. If the anchor for the text box is in a text box, a percentage value is relative to the surrounding text box. In other cases, percentage values are relative to the height of the page or window.

The `fo:min-height` attribute is usable with the following element: `<draw:text-box>`  
10.4.3.

The values of the `fo:min-height` attribute are a value of type `length` 18.3.19 or a value of type `percent` 18.3.24.

## 19.241 fo:min-width

See §7.14.9 of [XSL].

If set, it overrides the width of a text box specified by the `svg:width` 20.413 attribute of the surrounding `<draw:frame>` 10.4.2 element.

If the anchor for a text box is in a table cell, a percentage value is relative to the surrounding table box. If the anchor for a text box is in a text box, a percentage value is relative to the surrounding text box. In other cases, percentage values are relative to the height of the page or window.

The `fo:min-width` attribute is usable with the following element: `<draw:text-box>` 10.4.3.

The values of the `fo:min-width` attribute are a value of type `length` 18.3.19 or a value of type `percent` 18.3.24.

## 19.242 `fo:script`

### 19.242.1 General

See §7.9.3 of [XSL]. The attribute should be used only if necessary according to the rules of §2.2.3 of [RFC5646], or its successors. The attribute may be ignored if it is not specified together with a `fo:language` attribute.

In the OpenDocument XSL-compatible namespace, the `fo:script` attribute does not support `inherit` and `none` values.

### 19.242.2 `<text:alphabetical-index-source>`

The `fo:script` attribute specifies the script code used for sorting an alphabetical index. The attribute should be used only if necessary according to the rules of §2.2.3 of [RFC5646], or its successors.

The `fo:script` attribute is usable with the following element: `<text:alphabetical-index-source>` 8.8.2.

The `fo:script` attribute has the data type `scriptCode` 18.3.30.

### 19.242.3 `<text:bibliography-configuration>`

The `fo:script` attribute specifies the script code used for sorting a bibliographic index. The attribute should be used only if necessary according to the rules of §2.2.3 of [RFC5646], or its successors.

The `fo:script` attribute is usable with the following element: `<text:bibliography-configuration>` 16.31.6.

The `fo:script` attribute has the data type `scriptCode` 18.3.30.

## 19.243 `fo:space-after`

See §7.10.6 of [XSL].

The default value for this attribute is `0cm`.

The `fo:space-after` attribute is usable with the following element: `<style:column>` 17.13.

The `fo:space-after` attribute has the data type `length` 18.3.19.

## 19.244 `fo:space-before`

See §7.10.5 of [XSL].

The default value for this attribute is `0cm`.

The `fo:space-before` attribute is usable with the following element: `<style:column>` 17.13.

The `fo:space-before` attribute has the data type `length` 18.3.19.

## 19.245 fo:start-indent

See §7.10.7 of [XSL].

The default value for this attribute is `0cm`.

The `fo:start-indent` attribute is usable with the following element: `<style:column>` 17.13.

The `fo:start-indent` attribute has the data type `length` 18.3.19.

## 19.246 fo:text-indent

The `fo:text-indent` attribute specifies the indent for the text lines of a list item. The attribute has the same meaning as the formatting property attribute `fo:text-indent`. See 20.225 It is used for paragraphs inside list items whose paragraph styles do not specify an `fo:text-indent` or `fo:margin-left` attribute.

The value `fo:text-indent + fo:margin-left` specifies the indent of the first text line of the list item. This position is also the alignment position for the list label, because the value of `text:min-label-width` is treated as 0.

The `fo:text-indent` attribute is usable with the following element: `<style:list-level-label-alignment>` 17.20.

The `fo:text-indent` attribute has the data type `length` 18.3.19.

## 19.247 form:allow-deletes

The `form:allow-deletes` attribute specifies whether data records can be deleted by a form. It applies only if a form is data-aware.

The defined values for the `form:allow-deletes` attribute are:

- `false`: records cannot be deleted by a form.
- `true`: records can be deleted by a form.

The default value for this attribute is `true`.

The `form:allow-deletes` attribute is usable with the following element: `<form:form>` 13.3.

The `form:allow-deletes` attribute has the data type `boolean` 18.3.3.

## 19.248 form:allow-inserts

The `form:allow-inserts` attribute specifies whether new data records can be inserted by a form. It applies only if the form is data-aware.

The defined values for the `form:allow-inserts` attribute are:

- `false`: records cannot be inserted by a form.
- `true`: records can be inserted by a form.

The default value for this attribute is `true`.

The `form:allow-inserts` attribute is usable with the following element: `<form:form>` 13.3.

The `form:allow-inserts` attribute has the data type `boolean` 18.3.3.

## 19.249 `form:allow-updates`

The `form:allow-updates` attribute specifies whether data records in the form can be updated. It applies only if the form is data-aware.

The defined values for the `form:allow-updates` attribute are:

- `false`: records cannot be updated by a form.
- `true`: records can be updated by a form.

The default value for this attribute is `true`.

The `form:allow-updates` attribute is usable with the following element: `<form:form>` 13.3.

The `form:allow-updates` attribute has the data type `boolean` 18.3.3.

## 19.250 `form:apply-design-mode`

The `form:apply-design-mode` attribute specifies whether forms are presented in editable or non-editable state.

- `false`: forms in document presented in completed state
- `true`: forms in document presented in editable state

The default value for this attribute is `true`.

The `form:apply-design-mode` attribute is usable with the following element: `<office:forms>` 13.2.

The `form:apply-design-mode` attribute has the data type `boolean` 18.3.3.

## 19.251 `form:apply-filter`

The `form:apply-filter` attribute specifies whether the filter clause specified by the `form:filter` 19.273 attribute is applied to the SQL command constituting the form's result set.

The defined values for the `form:apply-filter` attribute are:

- `false`: the filter clause specified by the `form:filter` attribute is not applied to the SQL command constituting the form's result set.
- `true`: the filter clause specified by the `form:filter` attribute is applied to the SQL command constituting the form's result set.

The default value for this attribute is `false`.

The `form:apply-filter` attribute is usable with the following element: `<form:form>` 13.3.

The `form:apply-filter` attribute has the data type `boolean` 18.3.3.

## 19.252 `form:auto-complete`

The `form:auto-complete` attribute specifies , when the user enters text in the combo box that matches the beginning of a list item in the combo box, whether the consumer completes the text with the matched item.

The defined values for the `form:auto-complete` attribute are:

- `false`: completion of text is not enabled.

- `true`: completion of text is enabled.

The `form:auto-complete` attribute is usable with the following element:  
`<form:combobox>` 13.5.11.

The `form:auto-complete` attribute has the data type `boolean` 18.3.3.

## 19.253 `form:automatic-focus`

The `form:automatic-focus` attribute specifies whether the consumer loading the document should set the focus to a form control.

The defined values for the `form:automatic-focus` attribute are:

- `false`: the consumer does not set the focus to a form control, but to the document view. In this case, the user can start editing the document content.
- `true`: the consumer sets the focus to a form control after loading the document. In this case, the user can immediately start editing the form control content.

**Note:** If the `<office:forms>` 13.2 element and its sub-elements define multiple form controls, it is not defined as to which form control receives focus.

The default value for this attribute is `false`.

The `form:automatic-focus` attribute is usable with the following element:  
`<office:forms>` 13.2.

The `form:automatic-focus` attribute has the data type `boolean` 18.3.3.

## 19.254 `form:bound-column`

The `form:bound-column` attribute specifies the column values of the list source result set that are used to fill the data field values.

The `form:list-source` and `form:list-source-type` attributes together define an SQL command to execute against the database to which the control's form is bound. The result set of the execution of this SQL command is the list source result set.

The `form:bound-column` attribute is usable with the following element: `<form:listbox>` 13.5.13.

The `form:bound-column` attribute has the data type `string` 18.2.

## 19.255 `form:button-type`

The `form:button-type` attribute specifies the type of a button.

The defined values for the `form:button-type` attribute are:

- `push`: pressing the button does not perform any action by default. A user can add scripts to a button and the script will run when the button is pressed.
- `reset`: pressing the button resets every control in the form to its default value.
- `submit`: pressing the button submits the form.
- `url`: pressing the button loads the IRI that is specified in the `xlink:href` 19.9.16 attribute.

The default value for this attribute is `push`.

The `form:button-type` attribute is usable with the following elements: `<form:button>` 13.5.15 and `<form:image>` 13.5.16.

The values of the `form:button-type` attribute are `submit`, `reset`, `push` or `url`.

## 19.256 `form:command`

The `form:command` attribute specifies a command to execute on a data source.

The value is interpreted differently, depending to the value of the `form:command-type` 19.257 attribute of the form.

The `form:command` attribute is usable with the following element: `<form:form>` 13.3.

The `form:command` attribute has the data type `string` 18.2.

## 19.257 `form:command-type`

The `form:command-type` attribute specifies the type of command to execute on a data source.

The defined values of the `form:command-type` attribute are:

- `command`: the command contains an SQL statement. The form executes the SQL statement.
- `query`: the command contains the name of query. The form retrieves the query and executes its SQL statement.
- `table`: the command contains a table name. The form executes an SQL statement that retrieves all of the data in the table.

**Note:** The attribute value `query` is only meaningful if the `form:datasource` 19.264 attribute specifies a data source name or a database front-end document.

The default value for this attribute is `command`.

The `form:command-type` attribute is usable with the following element: `<form:form>` 13.3.

The values of the `form:command-type` attribute are `table`, `query` or `command`.

## 19.258 `form:control-implementation`

The `form:control-implementation` attribute specifies a rendition or implementation of a control that should be created. If the consumer supports the `form` element this attribute is used with, but does not support the specific concrete rendition or implementation, the consumer shall ignore the `form:control-implementation` attribute and use its own rendition of the `form` element.

The `form:control-implementation` attribute is usable with the following elements: `<form:button>` 13.5.15, `<form:checkbox>` 13.5.17, `<form:column>` 13.5.23, `<form:combobox>` 13.5.11, `<form:date>` 13.5.8, `<form:file>` 13.5.5, `<form:fixed-text>` 13.5.10, `<form:form>` 13.3, `<form:formatted-text>` 13.5.6, `<form:frame>` 13.5.19, `<form:generic-control>` 13.5.25, `<form:grid>` 13.5.22, `<form:hidden>` 13.5.21, `<form:image>` 13.5.16, `<form:image-frame>` 13.5.20, `<form:listbox>` 13.5.13, `<form:number>` 13.5.7, `<form:password>` 13.5.4, `<form:radio>` 13.5.18, `<form:text>` 13.5.2, `<form:textarea>` 13.5.3, `<form:time>` 13.5.9 and `<form:value-range>` 13.5.24.

The `form:control-implementation` attribute has the data type `namespacedToken` 18.3.20.

## 19.259 form:convert-empty-to-null

The `form:convert-empty-to-null` attribute specifies whether empty current values are treated as NULL.

The defined values for the `form:convert-empty-to-null` attribute are:

- `false`: an empty string in the control is treated as an empty string.
- `true`: an empty string in the control is treated as a NULL value.

The default value for this attribute is `false`.

The `form:convert-empty-to-null` attribute is usable with the following elements:  
`<form:combobox> 13.5.11`, `<form:date> 13.5.8`, `<form:formatted-text> 13.5.6`,  
`<form:number> 13.5.7`, `<form:password> 13.5.4`, `<form:text> 13.5.2`,  
`<form:textarea> 13.5.3` and `<form:time> 13.5.9`.

The `form:convert-empty-to-null` attribute has the data type `boolean` 18.3.3.

## 19.260 form:current-selected

The `form:current-selected` attribute specifies the state of a radio button or option control.

The defined values for the `form:current-selected` attribute are:

- `false`: radio button or option control is not selected.
- `true`: radio button or option control is selected.

The default value for this attribute is `false`.

The `form:current-selected` attribute is usable with the following elements:  
`<form:option> 13.5.14` and `<form:radio> 13.5.18`.

The `form:current-selected` attribute has the data type `boolean` 18.3.3.

## 19.261 form:current-state

The `form:current-state` attribute specifies the state of a check box control.

The defined values for the `form:current-state` attribute are:

- `checked`: The check box is checked. The value of the control is submitted with the form.
- `unchecked`: The check box is not checked.
- `unknown`: This value is available only if the `form:is-tristate` 19.281 attribute for a `<form:checkbox> 13.5.17` element is `true`. It indicates that a check box control is neither checked nor unchecked.

The `form:current-state` attribute is usable with the following element:  
`<form:checkbox> 13.5.17`.

The values of the `form:current-state` attribute are `unchecked`, `checked` or `unknown`.

## 19.262 form:current-value

The `form:current-value` attribute specifies the current status of an input control. It overrides the value of a `form:value` attribute, if one is present.

The defined value types for `form:current-value` attribute values are:

- `<form:combobox> 13.5.11`: string.
- `<form:date> 13.5.8`: date.

- `<form:file>` 13.5.5: string.
- `<form:formatted-text>` 13.5.6: string.
- `<form:number>` 13.5.7: double.
- `<form:text>` 13.5.2: string.
- `<form:textarea>` 13.5.3: string.
- `<form:time>` 13.5.9: time.

The `form:current-value` attribute is usable with the following elements:

`<form:checkbox>` 13.5.11, `<form:date>` 13.5.8, `<form:file>` 13.5.5, `<form:formatted-text>` 13.5.6, `<form:number>` 13.5.7, `<form:text>` 13.5.2, `<form:textarea>` 13.5.3 and `<form:time>` 13.5.9.

The `form:current-value` attribute has the data type string 18.2.

## 19.263 form:data-field

The `form:data-field` attribute specifies the name of a column in the result set defined by a (data-aware) form. A control interchanges content with its column when:

- the current row of a form changes: the control is initialized with the value from its column.
- the user changes the content of a control: this changed value is written back to its column.

**Note:** consumers may delay this write-back, for instance until the control loses the focus, instead of updating the column value with every change in the control.

The `form:data-field` attribute is usable with the following elements: `<form:checkbox>` 13.5.17, `<form:checkbox>` 13.5.11, `<form:date>` 13.5.8, `<form:formatted-text>` 13.5.6, `<form:image-frame>` 13.5.20, `<form:listbox>` 13.5.13, `<form:number>` 13.5.7, `<form:radio>` 13.5.18, `<form:text>` 13.5.2, `<form:textarea>` 13.5.3 and `<form:time>` 13.5.9.

The `form:data-field` attribute has the data type string 18.2.

## 19.264 form:datasource

The `form:datasource` attribute specifies the name of a data source to use for a form.

The defined values for the `form:datasource` attribute are:

- An IRI specifying a database connection.
- A data source name that a consumer can use to establish a database connection.
- The IRI of an OpenDocument database front-end document.

**Note:** The data source name is a programmatic name for a database front end document which allows it to reference data sources regardless of their real location. The mapping from data source names to specific instances of database front end documents is implementation-defined.

The `form:datasource` attribute is usable with the following element: `<form:form>` 13.3.

The values of the `form:datasource` attribute are a value of type `anyIRI` 18.3.2 or a value of type string 18.2.

## 19.265 form:default-button

The `form:default-button` attribute specifies whether a button is the default button on a form. If a user clicks 19.406 the default button or presses the Return key while an input control is focused, the consumer takes the same action.

If multiple buttons with `form:default-button` attributes with values of `true` are present in a form, it is implementation-dependent which button is the default for the form.

The defined values for the `form:default-button` attribute are:

- `false`: button is not the default button for a form.
- `true`: button is the default button for a form.

The default value for this attribute is `false`.

The `form:default-button` attribute is usable with the following element:

`<form:button>` 13.5.15.

The `form:default-button` attribute has the data type `boolean` 18.3.3.

## 19.266 form:delay-for-repeat

The `form:delay-for-repeat` attribute specifies a time-out to be used before a pressed mouse button results in repeating an action. The `form:delay-for-repeat` attribute is ignored unless a `form:repeat` 19.302 attribute is present, and evaluates to `true`.

The default value for this attribute is `PT0.050S`.

The `form:delay-for-repeat` attribute is usable with the following elements:

`<form:button>` 13.5.15, `<form:date>` 13.5.8, `<form:formatted-text>` 13.5.6, `<form:number>` 13.5.7, `<form:time>` 13.5.9 and `<form:value-range>` 13.5.24.

The `form:delay-for-repeat` attribute has the data type `duration` 18.2.

## 19.267 form:detail-fields

The `form:detail-fields` attribute specifies the names of the columns in contained forms that are related to columns in a containing form. The columns are used as parameters in the command for the contained form to retrieve the details from its containing form.

This attribute contains a comma “,” (U+002C, COMMA) separated list of field names.

The `form:detail-fields` attribute is usable with the following element: `<form:form>` 13.3.

The `form:detail-fields` attribute has the data type `string` 18.2.

## 19.268 form:disabled

The `form:disabled` attribute specifies whether a control can accept user input.

Controls that are disabled are not included in the tabbing navigation sequence and cannot be focused.

The defined values for the `form:disabled` attribute are:

- `false`: a control cannot accept user input.
- `true`: a control can accept user input.

The default value for this attribute is `false`.

The `form:disabled` attribute is usable with the following elements: `<form:button>` 13.5.15, `<form:checkbox>` 13.5.17, `<form:combobox>` 13.5.11, `<form:date>` 13.5.8,

```
<form:file> 13.5.5, <form:fixed-text> 13.5.10, <form:formatted-text> 13.5.6,  
<form:frame> 13.5.19, <form:grid> 13.5.22, <form:image> 13.5.16, <form:image-  
frame> 13.5.20, <form:listbox> 13.5.13, <form:number> 13.5.7, <form:password>  
13.5.4, <form:radio> 13.5.18, <form:text> 13.5.2, <form:textarea> 13.5.3,  
<form:time> 13.5.9 and <form:value-range> 13.5.24.
```

The `form:disabled` attribute has the data type [boolean 18.3.3](#).

## 19.269 `form:dropdown`

The `form:dropdown` attribute specifies whether a list in a `<form:combobox>` or `<form:listbox>` element is always visible or is only visible when the user clicks [19.406](#) the drop-down button.

The defined values for the `form:dropdown` attribute are:

- `false`: list is visible only when a user selects a drop-down button.
- `true`: list is always visible.

The default value for this attribute is `false`.

The `form:dropdown` attribute is usable with the following elements: `<form:combobox>` [13.5.11](#) and `<form:listbox>` [13.5.13](#).

The `form:dropdown` attribute has the data type [boolean 18.3.3](#).

## 19.270 `form:echo-char`

The `form:echo-char` attribute specifies the character that a form uses to mask text that a user inputs in a password control.

The default value for this attribute is `*` (U+002A, ASTERISK).

The `form:echo-char` attribute is usable with the following element: `<form:password>` [13.5.4](#).

The `form:echo-char` attribute has the data type [character 18.3.7](#).

## 19.271 `form:enctype`

The `form:enctype` attribute, when the value of the `form:method` [19.291](#) attribute is `post`, specifies the content type used to submit the form to the server. Other MIME types may be acceptable. See §17.3.4 of [HTML4].

The default value for this attribute is `application/x-www-form-urlencoded`.

The `form:enctype` attribute is usable with the following element: `<form:form>` [13.3](#).

The `form:enctype` attribute has the data type [string 18.2](#).

## 19.272 `form:escape-processing`

The `form:escape-processing` attribute, when the value of the `form:command-type` [19.257](#) attribute is `command`, specifies whether the consumer processes the command before passing it to a database driver.

The defined values for the `form:escape-processing` attribute are:

- `false`: consumer does not process commands before passing them to a database driver.
- `true`: consumer does process commands before passing them to a database driver.

The default value for this attribute is `true`.

The `form:escape-processing` attribute is usable with the following element:  
`<form:form>` 13.3.

The `form:escape-processing` attribute has the data type `boolean` 18.3.3.

## 19.273 `form:filter`

The `form:filter` attribute specifies a filter for the SQL clause determined by the `form:command-type` 19.257 and `form:command` 19.256 attributes. If that SQL statement contains a filter (WHERE clause), the filter specified by the attribute value is conjunctively added to the existing filter.

The filter is only applied if the `form:apply-filter` attribute has the value `true`. 19.251.

**Note:** This allows OpenDocument consumers to toggle a filter on and off without losing the content of the `form:filter` attribute.

The `form:filter` attribute is usable with the following element: `<form:form>` 13.3.

The `form:filter` attribute has the data type `string` 18.2.

## 19.274 `form:focus-on-click`

The `form:focus-on-click` attribute specifies if a `<form:button>` element is given the focus in a form when the form button control for that element is operated.

The defined values for the `form:focus-on-click` attribute are:

- `false`: element not given focus.
- `true`: element given focus.

The `form:focus-on-click` attribute is usable with the following element:  
`<form:button>` 13.5.15.

The `form:focus-on-click` attribute has the data type `boolean` 18.3.3.

## 19.275 `form:for`

The `form:for` attribute specifies a comma "," (U+002C, COMMA) separated list of `xml:ids` of control elements that are the targets of the elements where this attribute appears.

The `form:for` attribute is usable with the following elements: `<form:fixed-text>` 13.5.10 and `<form:frame>` 13.5.19.

The `form:for` attribute has the data type `string` 18.2.

## 19.276 `form:id (deprecated)`

The `form:id` attribute specifies an identifier for a control element.

OpenDocument consumers shall ignore a `form:id` attribute if it occurs on an element with an `xml:id` attribute value.

OpenDocument producers may write `form:id` attributes for elements in addition to an `xml:id` attribute.

The value of a `form:id` attribute shall equal the value of an `xml:id` attribute on the same element.

The `form:id` attribute is deprecated in favor of `xml:id`. 19.920

The `form:id` attribute is usable with the following elements: `<form:button>` 13.5.15, `<form:checkbox>` 13.5.17, `<form:combobox>` 13.5.11, `<form:date>` 13.5.8,

```
<form:file> 13.5.5, <form:fixed-text> 13.5.10, <form:formatted-text> 13.5.6,  
<form:frame> 13.5.19, <form:generic-control> 13.5.25, <form:grid> 13.5.22,  
<form:hidden> 13.5.21, <form:image> 13.5.16, <form:image-frame> 13.5.20,  
<form:listbox> 13.5.13, <form:number> 13.5.7, <form:password> 13.5.4,  
<form:radio> 13.5.18, <form:text> 13.5.2, <form:textarea> 13.5.3, <form:time>  
13.5.9 and <form:value-range> 13.5.24.
```

## 19.277 form:ignore-result

The `form:ignore-result` attribute specifies whether to discard all results that are retrieved from an underlying data source.

The defined values for the `form:ignore-result` attribute are:

- `false`: data from query of data source not discarded.
- `true`: data from query of data source is discarded. Only new data can be inserted into a data source.

The default value for this attribute is `false`.

The `form:ignore-result` attribute is usable with the following element: `<form:form> 13.3`.

The `form:ignore-result` attribute has the data type `boolean` 18.3.3.

## 19.278 form:image-align

The `form:image-align` attribute specifies which border (`start`, `end`) or axis (`center`) of an image and a text are to be aligned.

The defined values for the `form:image-align` attribute are:

- `center`: image and text are centered.
- `end`: image and text are aligned on the end of the image.
- `start`: image and text are aligned on the start of the image.

If a `form:image-align` attribute is not present, it is assumed to be `center`.

The default value for this attribute is `center`.

The `form:image-align` attribute is usable with the following elements: `<form:button> 13.5.15, <form:checkbox> 13.5.17` and `<form:radio> 13.5.18`.

The values of the `form:image-align` attribute are `start`, `center` or `end`.

## 19.279 form:image-data

The `form:image-data` attribute specifies the location of a file containing image data.

The `form:image-data` attribute is usable with the following elements: `<form:button> 13.5.15, <form:image> 13.5.16` and `<form:image-frame> 13.5.20`.

The `form:image-data` attribute has the data type `anyIRI` 18.3.2.

## 19.280 form:image-position

If the `form:image-position` attribute specifies the location of an image relative to the text of a control.

The defined values for the `form:image-position` attribute are:

- `bottom`: image is placed below the text.

- `center`: image centered relative to the text.
- `end`: image is placed after the text.
- `start`: image is placed before the text.
- `top`: image is placed above the text.

The default value for this attribute is `center`.

The `form:image-position` attribute is usable with the following elements: `<form:button>` 13.5.15, `<form:checkbox>` 13.5.17 and `<form:radio>` 13.5.18.

## 19.281 `form:is-tristate`

The `form:is-tristate` attribute specifies a check box can have the values of `checked`, `unchecked` and `unknown`.

The defined values for the `form:is-tristate` attribute are:

- `false`: a check box can only have the values of `checked` and `unchecked`.
- `true`: a check box can have the values of `checked`, `unchecked` and `unknown`.

The default value for this attribute is `false`.

The `form:is-tristate` attribute is usable with the following element: `<form:checkbox>` 13.5.17.

The `form:is-tristate` attribute has the data type `boolean` 18.3.3.

## 19.282 `form:label`

The `form:label` attribute specifies the text for a control.

The `form:label` attribute is usable with the following elements: `<form:button>` 13.5.15, `<form:checkbox>` 13.5.17, `<form:column>` 13.5.23, `<form:fixed-text>` 13.5.10, `<form:frame>` 13.5.19, `<form:item>` 13.5.12, `<form:option>` 13.5.14 and `<form:radio>` 13.5.18.

The `form:label` attribute has the data type `string` 18.2.

## 19.283 `form:linked-cell`

The `form:linked-cell` attribute specifies a table cell to which the form control content should be linked. Linking means that the cell value and control content are equal at all times.

The table cell to link the control to can be specified as a `cellAddress`, or as a named cell range. In the latter case, if the named range contains more than one cell, the cell in the first row and first column of the range is linked to the form control.

If the attribute value matches the name of a named range as well as a cell address, it is interpreted as the name of a named range.

The `form:linked-cell` attribute is usable with the following elements:

`<form:checkbox>` 13.5.17, `<form:combobox>` 13.5.11, `<form:date>` 13.5.8, `<form:file>` 13.5.5, `<form:formatted-text>` 13.5.6, `<form:listbox>` 13.5.13, `<form:number>` 13.5.7, `<form:password>` 13.5.4, `<form:radio>` 13.5.18, `<form:text>` 13.5.2, `<form:textarea>` 13.5.3, `<form:time>` 13.5.9 and `<form:value-range>` 13.5.24.

The values of the `form:linked-cell` attribute are a value of type `cellAddress` 18.3.4 or a value of type `string` 18.2.

## 19.284 form:list-linkage-type

The `form:list-linkage-type` attribute specifies how a control defined by a `<form:listbox>` 13.5.13 elements interacts with a linked table cell.

The defined values for the `form:list-linkage-type` attribute are:

- `selection`: The text selected in the control is propagated to its linked cell. If text is entered into a cell linked to a `<form:listbox>` element, the list item whose label matches the cell text is selected, or all items are deselected if no such item exists.
- `selection-indices`: If a list item is selected in the control, its index value on a 1-based index is propagated to its linked cell. If the value in the linked cell changes, the control should select the list item with the index described by the cell value, or deselect all items if the cell value cannot be interpreted as integer value, or does not specify a valid list index item.

The `form:list-linkage-type` attribute is usable with the following element:  
`<form:listbox>` 13.5.13.

The values of the `form:list-linkage-type` attribute are `selection` or `selection-indices`.

## 19.285 form:list-source

The `form:list-source` attribute specifies a source for the list entries in a list or combo box.

Obtaining those list entries may be a multi-step process. The process depends upon the value of the `form:list-source-type` attribute on the same element.

If the `form:list-source-type` 19.286 attribute has the value `value-list`, the content of the `form:list-source` attribute is ignored.

In all other cases, the content of the `form:list-source` attribute is resolved relative to a database connection or database front-end document. The `<form:form>` 13.3 element containing the control defines this database connection or document by its `form:datasource` 19.264 attribute.

- For retrieving the list entries depending on the `form:list-source-type` attribute, the following algorithms are to be used:
  - If the `form:list-source-type` attribute has the value `table-fields`, the `form:list-source` attribute specifies the name of a database table. The names of the columns of this table, in the order as returned by the database driver, are used as list entries.
  - For all other values of `form:list-source-type`, the value of the `form:list-source` attribute is used to create a result set. For this, the first step is to transform the value of the `form:list-source` attribute into an SQL command as follows:
    - `sql, sql-pass-through`: The SQL command is the content of the `form:list-source` attribute.
    - `query`: The `form:list-source` attribute specifies the name of a query object in the database, whose underlying SQL command is to be used.
    - `table`: The `form:list-source` attribute specifies the name of a database table, and the SQL command is the default SQL command to select all data from that table. This is "SELECT \* FROM <table\_name>", but consumers may find it necessary to add escape characters to a table name, if required by the database driver.

In a next step, the obtained SQL command is executed against the database. If the `form:list-source-type` attribute has the value `sql-pass-through` or `query`, and if the `db:escape-processing` 19.51 attribute of the referred query has the value `false`, then

the SQL command shall be passed directly to the database driver, without being interpreted and modified or normalized by the consumer.

This execution produces a two-dimensional homogeneous result set, whose first-column values form the entry list to be used.

The `form:list-source` attribute is usable with the following elements:  
`<form:combobox>` 13.5.11 and `<form:listbox>` 13.5.13.

The `form:list-source` attribute has the data type `string` 18.2.

## 19.286 `form:list-source-type`

The `form:list-source-type` attribute specifies how to populate the entry list in a combo box or list box control.

The defined values for the `form:list-source-type` attribute are:

- `query`: evaluated against the database front-end document or database connection defined by the `<form:form>` element which the control belongs to. The `form:list-source` attribute defines that evaluation. 19.285
- `sql`: evaluated against the database front-end document or database connection defined by the `<form:form>` element which the control belongs to. The `form:list-source` attribute defines that evaluation. 19.285
- `sql-pass-through`: evaluated against the database front-end document or database connection defined by the `<form:form>` element which the control belongs to. The `form:list-source` attribute defines that evaluation. 19.285
- `table`: evaluated against the database front-end document or database connection defined by the `<form:form>` element which the control belongs to. The `form:list-source` attribute defines that evaluation. 19.285
- `table-fields`: evaluated against the database front-end document or database connection defined by the `<form:form>` element which the control belongs to. The `form:list-source` attribute defines that evaluation. 19.285
- `value-list`: entry list populated by values specified by the `form:value` 19.316 attributes of `<form:option>` 13.5.14 child elements. This value is applicable to list boxes only.

The `form:list-source-type` attribute is usable with the following elements:  
`<form:combobox>` 13.5.11 and `<form:listbox>` 13.5.13.

The values of the `form:list-source-type` attribute are `table`, `query`, `sql`, `sql-pass-through`, `value-list` or `table-fields`.

## 19.287 `form:master-fields`

The `form:master-fields` attribute specifies the names of the columns in the result set represented by a parent form. Each time the parent form changes the current row, the nested form queries the database again based on the values of the master fields.

The attribute contains a comma "," (U+002C, COMMA) separated list of field names.

The `form:master-fields` attribute is usable with the following element: `<form:form>` 13.3.

The `form:master-fields` attribute has the data type `string` 18.2.

## 19.288 form:maxLength

The `form:maxLength` attribute specifies the maximum number of characters that a user can enter in an input control.

The `form:maxLength` attribute is usable with the following elements: `<form:combobox>` 13.5.11, `<form:date>` 13.5.8, `<form:file>` 13.5.5, `<form:formatted-text>` 13.5.6, `<form:number>` 13.5.7, `<form:password>` 13.5.4, `<form:text>` 13.5.2, `<form:textarea>` 13.5.3 and `<form:time>` 13.5.9.

The `form:maxLength` attribute has the data type `nonNegativeInteger` 18.2.

## 19.289 form: maxValue

### 19.289.1 General

The `form: maxValue` attribute specifies the maximum value that a user can enter.

### 19.289.2 <form:date>

The value type of the `form: maxValue` attribute is `date`.

The `form: maxValue` attribute is usable with the following element: `<form:date>` 13.5.8.

The `form: maxValue` attribute has the data type `date` 18.2.

### 19.289.3 <form:formatted-text>

The value type of the `form: maxValue` attribute is `string`.

The `form: maxValue` attribute is usable with the following element: `<form:formatted-text>` 13.5.6.

The `form: maxValue` attribute has the data type `string` 18.2.

### 19.289.4 <form:number> and <form:value-range>

The value type of the `form: maxValue` attribute is `double`.

The `form: maxValue` attribute is usable with the following elements: `<form:number>` 13.5.7 and `<form:value-range>` 13.5.24.

The `form: maxValue` attribute has the data type `double` 18.2.

### 19.289.5 <form:time>

The value type of the `form: maxValue` attribute is `time`.

The `form: maxValue` attribute is usable with the following element: `<form:time>` 13.5.9.

The `form: maxValue` attribute has the data type `time` 18.2.

## 19.290 form:minValue

### 19.290.1 General

The `form: minValue` attribute specifies the minimum value that a user can enter.

### 19.290.2 <form:date>

The value type of the `form: minValue` attribute is `date`.

The `form:min-value` attribute is usable with the following element: `<form:date>` 13.5.8.

The `form:min-value` attribute has the data type `date` 18.2.

### 19.290.3 `<form:formatted-text>`

The value type of the `form:min-value` attribute is `string`.

The `form:min-value` attribute is usable with the following element: `<form:formatted-text>` 13.5.6.

The `form:min-value` attribute has the data type `string` 18.2.

### 19.290.4 `<form:number>` and `<form:value-range>`

The value type of the `form:min-value` attribute is `double`.

The `form:min-value` attribute is usable with the following elements: `<form:number>` 13.5.7 and `<form:value-range>` 13.5.24.

The `form:min-value` attribute has the data type `double` 18.2.

### 19.290.5 `<form:time>`

The value type of the `form:min-value` attribute is `time`.

The `form:min-value` attribute is usable with the following element: `<form:time>` 13.5.9.

The `form:min-value` attribute has the data type `time` 18.2.

## 19.291 `form:method`

The `form:method` attribute specifies the HTTP method [RFC2616] used to submit data in a form to a server.

The defined values for the `form:method` attribute are:

- `get`: HTTP get method.
- `post`: HTTP post method.
- a value of type `string`

These values are not case sensitive.

The default value for this attribute is `get`.

The `form:method` attribute is usable with the following element: `<form:form>` 13.3.

The values of the `form:method` attribute are `get`, `post` or a value of type `string` 18.2.

## 19.292 `form:multi-line`

The `form:multi-line` attribute specifies whether a label is displayed on multiple lines.

The defined values for the `form:multi-line` attribute are:

- `false`: label is not displayed over multiple lines.
- `true`: label is displayed over multiple lines.

The default value for this attribute is `false`.

The `form:multi-line` attribute is usable with the following element: `<form:fixed-text>` 13.5.10.

The `form:multi-line` attribute has the data type `boolean` 18.3.3.

## 19.293 `form:multiple`

The `form:multiple` attribute specifies whether a user can select multiple items from a list box.

The defined values for the `form:multiple` attribute are:

- `false`: multiple items cannot be selected from a list box.
- `true`: multiple items can be selected from a list box.

The default value for this attribute is `false`.

The `form:multiple` attribute is usable with the following element: `<form:listbox>` 13.5.13.

The `form:multiple` attribute has the data type `boolean` 18.3.3.

## 19.294 `form:name`

The `form:name` attribute specifies the name of a form or control element.

**Note:** This may be used to give a form or control element an identity, which can be used for scripting or for submitting the content of controls.

The `form:name` attribute is usable with the following elements: `<form:button>` 13.5.15, `<form:checkbox>` 13.5.17, `<form:column>` 13.5.23, `<form:combobox>` 13.5.11, `<form:date>` 13.5.8, `<form:file>` 13.5.5, `<form:fixed-text>` 13.5.10, `<form:form>` 13.3, `<form:formatted-text>` 13.5.6, `<form:frame>` 13.5.19, `<form:generic-control>` 13.5.25, `<form:grid>` 13.5.22, `<form:hidden>` 13.5.21, `<form:image>` 13.5.16, `<form:image-frame>` 13.5.20, `<form:listbox>` 13.5.13, `<form:number>` 13.5.7, `<form:password>` 13.5.4, `<form:radio>` 13.5.18, `<form:text>` 13.5.2, `<form:textarea>` 13.5.3, `<form:time>` 13.5.9 and `<form:value-range>` 13.5.24.

The `form:name` attribute has the data type `string` 18.2.

## 19.295 `form:navigation-mode`

The `form:navigation-mode` attribute specifies methods for navigation of a database aware form.

The defined values of the `form:navigation-mode` attribute are:

- `current`: an user interface is provided and the navigation is performed on the current form.
- `none`: a user interface is not provided, but the form may be navigated using the TAB and SHIFT/TAB keys on the keyboard.
- `parent`: a user interface is provided and the navigation is performed on the parent form of the current form.

The `form:navigation-mode` attribute is usable with the following element: `<form:form>` 13.3.

The values of the `form:navigation-mode` attribute are `none`, `current` or `parent`.

## 19.296 `form:order`

The `form:order` attribute specifies a sort criteria for a command.

The `form:order` attribute is usable with the following element: `<form:form>` 13.3.

The `form:order` attribute has the data type `string` 18.2.

## 19.297 `form:orientation`

The `form:orientation` attribute specifies the orientation of a control,

The defined values for the `form:orientation` attribute are:

- `horizontal`: control is oriented horizontally.
- `vertical`: control is oriented vertically.

The `form:orientation` attribute is usable with the following element: `<form:value-range>` 13.5.24.

The values of the `form:orientation` attribute are `horizontal` or `vertical`.

## 19.298 `form:page-step-size`

The `form:page-step-size` attribute specifies the preferred major value change offered to the user (if applicable for the control used).

The `form:page-step-size` attribute is usable with the following element: `<form:value-range>` 13.5.24.

The `form:page-step-size` attribute has the data type `positiveInteger` 18.2.

## 19.299 `form:printable`

The `form:printable` attribute specifies whether a control is printed when a user prints a document containing the control.

The defined values for the `form:printable` attribute are:

- `false`: control not printed when a document containing the control is printed.
- `true`: control printed when a document containing the control is printed.

The default value for this attribute is `true`.

The `form:printable` attribute is usable with the following elements: `<form:button>` 13.5.15, `<form:checkbox>` 13.5.17, `<form:combobox>` 13.5.11, `<form:date>` 13.5.8, `<form:file>` 13.5.5, `<form:fixed-text>` 13.5.10, `<form:formatted-text>` 13.5.6, `<form:frame>` 13.5.19, `<form:grid>` 13.5.22, `<form:image>` 13.5.16, `<form:image-frame>` 13.5.20, `<form:listbox>` 13.5.13, `<form:number>` 13.5.7, `<form:password>` 13.5.4, `<form:radio>` 13.5.18, `<form:text>` 13.5.2, `<form:textarea>` 13.5.3, `<form:time>` 13.5.9 and `<form:value-range>` 13.5.24.

The `form:printable` attribute has the data type `boolean` 18.3.3.

## 19.300 `form:property-name`

The `form:property-name` attribute specifies the name of a property element. The name is unique within all `<form:property>` and `<form:list-property>` child elements of a `<form:properties>` 13.7 element.

The `form:property-name` attribute is usable with the following elements: `<form:list-property>` 13.9 and `<form:property>` 13.8.

The `form:property-name` attribute has the data type `string` 18.2.

## 19.301 form:readonly

The `form:readonly` attribute specifies whether a user can modify the value of a control.

Read-only controls are included in a tabbing navigation sequence.

The defined values for the `form:readonly` attribute are:

- `false`: a user cannot modify the value of a control.
- `true`: a user can modify the value of a control.

The default value for this attribute is `false`.

The `form:readonly` attribute is usable with the following elements: `<form:combobox>` 13.5.11, `<form:date>` 13.5.8, `<form:file>` 13.5.5, `<form:formatted-text>` 13.5.6, `<form:image-frame>` 13.5.20, `<form:number>` 13.5.7, `<form:text>` 13.5.2, `<form:textarea>` 13.5.3 and `<form:time>` 13.5.9.

The `form:readonly` attribute has the data type `boolean` 18.3.3.

## 19.302 form:repeat

The `form:repeat` attribute specifies whether a form control, or a part of a form control, repeats a defined action in response a continuous action by a user.

The defined values for the `form:repeat` attribute are:

- `false`: continuous action results in one defined action.
- `true`: continuous action results in repeating the defined action.

**Note:** Pressing and holding any key is an example of a continuous action by a user.

The `form:repeat` attribute is usable with the following elements: `<form:button>` 13.5.15, `<form:date>` 13.5.8, `<form:formatted-text>` 13.5.6, `<form:number>` 13.5.7, `<form:time>` 13.5.9 and `<form:value-range>` 13.5.24.

The `form:repeat` attribute has the data type `boolean` 18.3.3.

## 19.303 form:selected

The `form:selected` attribute specifies the default state of a radio button or option. A control initializes to the default state specified by this attribute control.

In a group of radio buttons that share the same name, only one radio button can have a `form:selected` attribute set to `true`.

The defined values for the `form:selected` attribute are:

- `false`: not the default state of a radio button or option.
- `true`: default state of a radio button or option.

The default value for this attribute is `false`.

The `form:selected` attribute is usable with the following elements: `<form:option>` 13.5.14 and `<form:radio>` 13.5.18.

The `form:selected` attribute has the data type `boolean` 18.3.3.

## 19.304 form:size

The `form:size` attribute specifies the number of rows that are visible at a time in a combo box list or a list box list.

The `form:size` attribute is usable with the following elements: `<form:combobox>` 13.5.11 and `<form:listbox>` 13.5.13.

The `form:size` attribute has the data type `nonNegativeInteger` 18.2.

## 19.305 `form:source-cell-range`

The `form:source-cell-range` attribute specifies a rectangular cell range, whose content should be synchronized with a form control's item list. The form control reacts to changes in the cell range, that is, it adjusts its item list as the cell range's content changes.

The attribute specifies a cell range using `cellRangeAddress` or with the name of a named cell range. It displays the left-most columns of a range, if the range contains more columns than a form control can display.

The `form:source-cell-range` attribute is usable with the following elements: `<form:combobox>` 13.5.11 and `<form:listbox>` 13.5.13.

The values of the `form:source-cell-range` attribute are a value of type `cellRangeAddress` 18.3.5 or a value of type `string` 18.2.

## 19.306 `form:spin-button`

The `form:spin-button` attribute specifies whether a form control for input has spin buttons. Spin buttons are used to increment or decrement all or part of an input value.

The defined values for the `form:spin-button` attribute are:

- `false`: form control does not have spin buttons.
- `true`: form control has spin buttons.

The default value for this attribute is `false`.

The `form:spin-button` attribute is usable with the following elements: `<form:date>` 13.5.8, `<form:formatted-text>` 13.5.6, `<form:number>` 13.5.7 and `<form:time>` 13.5.9.

The `form:spin-button` attribute has the data type `boolean` 18.3.3.

## 19.307 `form:state`

The `form:state` attribute specifies the default state of a check box control. A check box control initializes to the default state specified by this attribute.

The defined values for the `form:state` attribute are:

- `checked`: default state of a check box control.
- `unchecked`: check box control has no default state.
- `unknown`: state of check box control is not known. This value is available only if the value of the `form:is-tristate` 19.281 attribute on this check box control is `true`.

The default value for this attribute is `unchecked`.

The `form:state` attribute is usable with the following element: `<form:checkbox>` 13.5.17.

The values of the `form:state` attribute are `unchecked`, `checked` or `unknown`.

## 19.308 `form:step-size`

The `form:step-size` attribute specifies the preferred minor value change offered the user (if applicable for the control used).

The default value for this attribute is 1.

The `form:step-size` attribute is usable with the following element: `<form:value-range>` 13.5.24.

The `form:step-size` attribute has the data type `positiveInteger` 18.2.

## 19.309 `form:tab-cycle`

The `form:tab-cycle` attribute specifies how the consumer responds when the user presses the TAB key in the controls in a form. The behavior of the consumer depends on whether the form is bound to a data source.

The defined values for the `form:tab-cycle` attribute are:

- `current`: If a user presses the TAB key in the last control of the form, the focus moves to the first control specified in the tab order of the same form. If the form is data aware (13.1), then the current row of the result set subject to the form is not changed.
- `page`: If a user presses the TAB key in the last control of a form, the focus moves to the first control specified in the tab order for the next form.
- `records`: If a user presses the TAB key in the last control of the form, the focus moves to the first control specified in the tab order of the same form. If the form is data aware (13.1), then the current row of the result set subject to the form moves to the next row.

The `form:tab-cycle` attribute is usable with the following element: `<form:form>` 13.3.

The values of the `form:tab-cycle` attribute are `records`, `current` or `page`.

## 19.310 `form:tab-index`

The `form:tab-index` attribute specifies the tabbing navigation order of controls within a form. Tabbing order is the order of focus on controls when a user navigates the form using the TAB key on the keyboard. Elements within other elements can be included in a tabbing order. Values do not have to be sequential and they do not have to begin with a particular value.

The rules of tab navigation are:

- 1) Controls that have the `form:disabled` 19.268 attribute set to `true` are not included in the navigation, independent of their `form:tab-index` value.
- 2) Controls that have the same values for the `form:tab-index` attribute are navigated in order of their position in the form.
- 3) The navigation starts with the controls with lowest non-zero `form:tab-index` value.
- 4) When the controls with a certain `form:tab-index` value have been navigated, navigation continues with the controls with the next highest value.
- 5) After the controls with the highest value have been navigated, navigation continues with the controls that do not contain the `form:tab-index` attribute or contain the attribute with a value of 0.

The default value for this attribute is 0.

The `form:tab-index` attribute is usable with the following elements: `<form:button>` 13.5.15, `<form:checkbox>` 13.5.17, `<form:combobox>` 13.5.11, `<form:date>` 13.5.8, `<form:file>` 13.5.5, `<form:formatted-text>` 13.5.6, `<form:grid>` 13.5.22, `<form:image>` 13.5.16, `<form:listbox>` 13.5.13, `<form:number>` 13.5.7, `<form:password>` 13.5.4, `<form:radio>` 13.5.18, `<form:text>` 13.5.2, `<form:textarea>` 13.5.3, `<form:time>` 13.5.9 and `<form:value-range>` 13.5.24.

The `form:tab-index` attribute has the data type `nonNegativeInteger` 18.2.

### 19.311 form:tab-stop

The `form:tab-stop` attribute specifies whether a control is included in the tabbing navigation order.

The defined values for the `form:tab-stop` attribute are:

- `false`: control is not included in tabbing navigation order.
- `true`: control is included in tabbing navigation order.

The default value for this attribute is `true`.

The `form:tab-stop` attribute is usable with the following elements: `<form:button> 13.5.15`, `<form:checkbox> 13.5.17`, `<form:combobox> 13.5.11`, `<form:date> 13.5.8`, `<form:file> 13.5.5`, `<form:formatted-text> 13.5.6`, `<form:grid> 13.5.22`, `<form:image> 13.5.16`, `<form:listbox> 13.5.13`, `<form:number> 13.5.7`, `<form:password> 13.5.4`, `<form:radio> 13.5.18`, `<form:text> 13.5.2`, `<form:textarea> 13.5.3`, `<form:time> 13.5.9` and `<form:value-range> 13.5.24`.

The `form:tab-stop` attribute has the data type `boolean` 18.3.3.

### 19.312 form:text-style-name

The `form:text-style-name` attribute specifies a paragraph style that is applied to all controls for a column. A paragraph style for a form may reference a data style.

The `form:text-style-name` attribute is usable with the following element: `<form:column> 13.5.23`.

The `form:text-style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.313 form:title

The `form:title` attribute contains additional information about a control.

**Note:** The value of the attribute may be used as a "tool tip" (a short message that appears when the pointing device pauses over the form element). Audio user agents may speak the title information in a similar context.

The `form:title` attribute is usable with the following elements: `<form:button> 13.5.15`, `<form:checkbox> 13.5.17`, `<form:combobox> 13.5.11`, `<form:date> 13.5.8`, `<form:file> 13.5.5`, `<form:fixed-text> 13.5.10`, `<form:formatted-text> 13.5.6`, `<form:frame> 13.5.19`, `<form:grid> 13.5.22`, `<form:image> 13.5.16`, `<form:image-frame> 13.5.20`, `<form:listbox> 13.5.13`, `<form:number> 13.5.7`, `<form:password> 13.5.4`, `<form:radio> 13.5.18`, `<form:text> 13.5.2`, `<form:textarea> 13.5.3`, `<form:time> 13.5.9` and `<form:value-range> 13.5.24`.

The `form:title` attribute has the data type `string` 18.2.

### 19.314 form:toggle

The `form:toggle` attribute specifies whether a form button control acts as a push button or as a toggle button.

**Note:** A toggle button is similar to a check box (in fact, it could be considered a different visual representation of a check box), in that operating it once, by either mouse or keyboard, toggles its state between "pressed" and "not pressed".

A push button triggers an action, which happens every time the user operates the button.

The defined values for the `form:toggle` attribute are:

- `false`: form button control acts as push button.
- `true`: form button control acts as a toggle button.

The default value for this attribute is `false`.

The `form:toggle` attribute is usable with the following element: `<form:button>` 13.5.15.

The `form:toggle` attribute has the data type `boolean` 18.3.3.

## 19.315 `form:validation`

The `form:validation` attribute specifies whether the text that the user enters is validated during input against the input format associated with the control.

The defined values for the `form:validation` attribute are:

- `true`: invalid content is detected during input.
- `false`: invalid content is not detected during input.

The default value for this attribute is `false`.

The `form:validation` attribute is usable with the following element: `<form:formatted-text>` 13.5.6.

The `form:validation` attribute has the data type `boolean` 18.3.3.

## 19.316 `form:value`

### 19.316.1 General

The `form:value` attribute specifies the default value of an input control.

The `form:value` attribute is usable with the following elements: `<form:button>` 13.5.15, `<form:checkbox>` 13.5.17, `<form:combobox>` 13.5.11, `<form:date>` 13.5.8, `<form:file>` 13.5.5, `<form:formatted-text>` 13.5.6, `<form:hidden>` 13.5.21, `<form:image>` 13.5.16, `<form:number>` 13.5.7, `<form:option>` 13.5.14, `<form:password>` 13.5.4, `<form:radio>` 13.5.18, `<form:text>` 13.5.2, `<form:textarea>` 13.5.3, `<form:time>` 13.5.9 and `<form:value-range>` 13.5.24.

The `form:value` attribute has the data type `string` 18.2.

### 19.316.2 `<form:button>`

The value type of the `form:value` attribute is `boolean`.

The `form:value` attribute is usable with the following element: `<form:button>` 13.5.15.

The `form:value` attribute has the data type `string` 18.2.

### 19.316.3 `<form:date>`

The value type of the `form:value` attribute is `date`.

The `form:value` attribute is usable with the following element: `<form:date>` 13.5.8.

The `form:value` attribute has the data type `date` 18.2.

### 19.316.4 `<form:number>` and `<form:value-range>`

The value type of the `form:value` attribute is `number`.

The `form:value` attribute is usable with the following elements: `<form:number>` 13.5.7 and `<form:value-range>` 13.5.24.

The `form:value` attribute has the data type `double` 18.2.

### 19.316.5 `<form:checkbox>`, `<form:combobox>`, `<form:file>`, `<form:formatted-text>`, `<form:hidden>`, `<form:image>`, `<form:option>`, `<form:password>`, `<form:radio>`, `<form:text>`, `<form:textarea>`

The value type of the `form:value` attribute is `string`.

The `form:value` attribute is usable with the following elements: `<form:checkbox>` 13.5.17, `<form:combobox>` 13.5.11, `<form:file>` 13.5.5, `<form:formatted-text>` 13.5.6, `<form:hidden>` 13.5.21, `<form:image>` 13.5.16, `<form:option>` 13.5.14, `<form:password>` 13.5.4, `<form:radio>` 13.5.18, `<form:text>` 13.5.2 and `<form:textarea>` 13.5.3.

The `form:value` attribute has the data type `string` 18.2.

### 19.316.6 `<form:time>`

The value type of the `form:value` attribute is `time`.

The `form:value` attribute is usable with the following element: `<form:time>` 13.5.9.

The `form:value` attribute has the data type `time` 18.2.

## 19.317 `form:visual-effect`

The `form:visual-effect` attribute specifies a visual effect to apply to a control.

The defined values for the `form:visual-effect` attribute are:

- `3d`: 3D visual effect.
- `flat`: flat visual effect.

The `form:visual-effect` attribute is usable with the following elements:  
`<form:checkbox>` 13.5.17 and `<form:radio>` 13.5.18.

The values of the `form:visual-effect` attribute are `flat` or `3d`.

## 19.318 `form:xforms-list-source`

The `form:xforms-list-source` attribute specifies a reference to an `xform:bind` attribute, and creates a list entry for each node in the node-set defined by that attribute.

The `form:xforms-list-source` attribute is usable with the following element:  
`<form:listbox>` 13.5.13.

The `form:xforms-list-source` attribute has the data type `string` 18.2.

## 19.319 `form:xforms-submission`

The `form:xforms-submission` attribute specifies the `id` of an `<xforms:submission>` (3.3.3 of [XForms]) child element of a `<xforms:model>` element for a `<form:button>` 13.5.15 control.

The `form:xforms-submission` attribute is usable with the following element:  
`<form:button>` 13.5.15.

The `form:xforms-submission` attribute has the data type `string` 18.2.

## 19.320 grddl:transformation

The `grddl:transformation` attributes link to a transformation for extracting RDF data from the document. See [GRDDL].

**Note:** This transformation is intended to be interpreted by consumers that do not support OpenDocument.

The referenced transformations should map OpenDocument elements that are children of the `<office:meta>` 3.2 element to RDF. They should also extract the RDF triples stored as RDFa attributes in the document content and styles.

**Note:** Please check the [OASIS Open Document Format for Office Applications \(OpenDocument\) TC's](#) web site for GRDDL transformations provided by the OASIS Open Document Format for Office Applications (OpenDocument) TC.

The `grddl:transformation` attribute is usable with the following elements:

`<office:document>` 3.1.2, `<office:document-content>` 3.1.3.2, `<office:document-meta>` 3.1.3.4, `<office:document-settings>` 3.1.3.5 and `<office:document-styles>` 3.1.3.3.

The value of the `grddl:transformation` attribute is a white space separated lists of values of type `anyIRI` 18.3.2, including the empty list.

## 19.321 meta:cell-count

The `meta:cell-count` attribute specifies the number of table cells that an OpenDocument producer has counted in a document.

The `meta:cell-count` attribute is usable with the following element: `<meta:document-statistic>` 4.3.2.18.

The `meta:cell-count` attribute has the data type `nonNegativeInteger` 18.2.

## 19.322 meta:character-count

The `meta:character-count` attribute specifies the number of characters that an OpenDocument producer has counted in a document.

The value of the `meta:character-count` attribute is implementation-dependent.

The `meta:character-count` attribute is usable with the following element: `<meta:document-statistic>` 4.3.2.18.

The `meta:character-count` attribute has the data type `nonNegativeInteger` 18.2.

## 19.323 meta:date

The `meta:date` attribute specifies the date and time when a template was last modified, prior to being used to create the current document.

The `meta:date` attribute is usable with the following element: `<meta:template>` 4.3.2.12.

The `meta:date` attribute has the data type `dateTime` 18.2.

## 19.324 meta:delay

The `meta:delay` attribute specifies a reload delay.

The `meta:delay` attribute is usable with the following element: `<meta:auto-reload>` 4.3.2.13.

The `meta:delay` attribute has the data type `duration` 18.2.

## 19.325 meta:draw-count

The `meta:draw-count` attribute specifies the number of drawing shapes that an OpenDocument producer has counted in a document.

The `meta:draw-count` attribute is usable with the following element: `<meta:document-statistic>` 4.3.2.18.

The `meta:draw-count` attribute has the data type `nonNegativeInteger` 18.2.

## 19.326 meta:frame-count

The `meta:frame-count` attribute specifies the number of text boxes that an OpenDocument producer has counted in a document.

The `meta:frame-count` attribute is usable with the following element: `<meta:document-statistic>` 4.3.2.18.

The `meta:frame-count` attribute has the data type `nonNegativeInteger` 18.2.

## 19.327 meta:image-count

The `meta:image-count` attribute specifies the number of images that an OpenDocument producer has counted in a document.

The `meta:image-count` attribute is usable with the following element: `<meta:document-statistic>` 4.3.2.18.

The `meta:image-count` attribute has the data type `nonNegativeInteger` 18.2.

## 19.328 meta:name

The `meta:name` attribute specifies the name of a metadata element. The name shall be unique within the domain of `<meta:user-defined>` 4.3.3 elements.

The `meta:name` attribute is usable with the following element: `<meta:user-defined>` 4.3.3.

The `meta:name` attribute has the data type `string` 18.2.

## 19.329 meta:non-whitespace-character-count

The `meta:non-whitespace-character-count` attribute specifies the number of non-white space characters that an OpenDocument producer has counted in a document.

The value of the `meta:non-whitespace-character-count` attribute is implementation-dependent.

The `meta:non-whitespace-character-count` attribute is usable with the following element: `<meta:document-statistic>` 4.3.2.18.

The `meta:non-whitespace-character-count` attribute has the data type `nonNegativeInteger` 18.2.

## 19.330 meta:object-count

The `meta:object-count` attribute specifies the number of embedded objects stored in OpenDocument format that the OpenDocument producer has counted for the document.

The `meta:object-count` attribute is usable with the following element:

`<meta:document-statistic> 4.3.2.18.`

The `meta:object-count` attribute has the data type `nonNegativeInteger 18.2.`

## 19.331 meta:ole-object-count

The `meta:ole-object-count` attribute specifies the number of embedded objects stored in a different format than OpenDocument that the OpenDocument producer has counted for the document.

The `meta:ole-object-count` attribute is usable with the following element:

`<meta:document-statistic> 4.3.2.18.`

The `meta:ole-object-count` attribute has the data type `nonNegativeInteger 18.2.`

## 19.332 meta:page-count

The `meta:page-count` attribute specifies the number of pages that an OpenDocument producer has calculated for a document.

The value of the `meta:page-count` attribute is implementation-dependent.

The `meta:page-count` attribute is usable with the following element: `<meta:document-statistic> 4.3.2.18.`

The `meta:page-count` attribute has the data type `nonNegativeInteger 18.2.`

## 19.333 meta:paragraph-count

The `meta:paragraph-count` attribute specifies the number of paragraphs that an OpenDocument producer has counted in a document.

The `meta:paragraph-count` attribute is usable with the following element:

`<meta:document-statistic> 4.3.2.18.`

The `meta:paragraph-count` attribute has the data type `nonNegativeInteger 18.2.`

## 19.334 meta:row-count

The `meta:row-count` attribute specifies the number of lines contained in a document instance.

The value of the `meta:row-count` attribute is implementation-dependent.

The `meta:row-count` attribute is usable with the following element: `<meta:document-statistic> 4.3.2.18.`

The `meta:row-count` attribute has the data type `nonNegativeInteger 18.2.`

## 19.335 meta:sentence-count

The `meta:sentence-count` attribute specifies the number of sentences that an OpenDocument producer has counted in a document.

The value of the `meta:sentence-count` attribute is implementation-dependent.

The `meta:sentence-count` attribute is usable with the following element:  
`<meta:document-statistic>` 4.3.2.18.

The `meta:sentence-count` attribute has the data type `nonNegativeInteger` 18.2.

## 19.336 meta:syllable-count

The `meta:syllable-count` attribute specifies the number of syllables that an OpenDocument producer has counted in a document.

The value of the `meta:syllable-count` attribute is implementation-dependent.

The `meta:syllable-count` attribute is usable with the following element:  
`<meta:document-statistic>` 4.3.2.18.

The `meta:syllable-count` attribute has the data type `nonNegativeInteger` 18.2.

## 19.337 meta:table-count

The `meta:table-count` attribute specifies the number of `<table:table>` elements counted in a document.

The `meta:table-count` attribute is usable with the following element: `<meta:document-statistic>` 4.3.2.18.

The `meta:table-count` attribute has the data type `nonNegativeInteger` 18.2.

## 19.338 meta:value-type

The `meta:value-type` attribute specifies the type of a metadata element.

The defined values for the `meta:value-type` attribute are:

- `boolean`: 18.3.3
- `date`: 18.2
- `float`: 18.2
- `time`: 18.2
- `string`: 18.2

The default type for meta-data elements is `string`.

The `meta:value-type` attribute is usable with the following element: `<meta:user-defined>` 4.3.3.

## 19.339 meta:word-count

The `meta:word-count` attribute specifies the number of words that an OpenDocument producer has counted in a document.

The value of the `meta:word-count` attribute is implementation-dependent.

The `meta:word-count` attribute is usable with the following element: `<meta:document-statistic>` 4.3.2.18.

The `meta:word-count` attribute has the data type `nonNegativeInteger` 18.2.

## 19.340 number:automatic-order

The `number:automatic-order` attribute specifies whether data is ordered to match the default order for the language and country of a data style.

The defined values for the `number:automatic-order` attribute are:

- `false`: data is not ordered to match the default order for the language and country of a data style.
- `true`: data is ordered to match the default order for the language and country of a data style.

The default value for this attribute is `false`.

The `number:automatic-order` attribute is usable with the following elements: `<number:currency-style>` 16.29.8 and `<number:date-style>` 16.29.11.

The `number:automatic-order` attribute has the data type `boolean` 18.3.3.

## 19.341 number:calendar

The `number:calendar` attribute specifies the calendar system used to extract parts of a date.

The defined values for the `number:calendar` attribute are:

- `buddhist`: Buddhist calendar, identical to Gregorian calendar but add 543 years to the Gregorian date.
- `gengou`: Japanese Gengou calendar, Emperor eras. Identical to Gregorian calendar but with different eras for each emperor. Consumers may implement only the modern eras starting 1868, Meiji, Taisho, Showa and Heisei. Earlier dates are then displayed using the Gregorian calendar [JIS X 0301].
- `gregorian`: Gregorian calendar, as defined in [ISO8601], Section 4.2.1, usually specified for dates no earlier than 1582-10-15. If this calendar is specified for a date earlier than 1582-10-15, the meaning is implementation-defined.
- `hanja`: Additional Gregorian calendar in Korean locales, uses differently localized day and month names.
- `hanja_yoil`: Legacy alias for `hanja`
- `hijri`: Islamic Hijri lunar calendar (religious, non-civil).
- `jewish`: Jewish lunisolar calendar.
- `ROC`: Taiwanese Minguo calendar, identical to Gregorian calendar but subtract 1911 years from the Gregorian date.

The attribute value may also be a string value. If this attribute is not specified, the default calendar system for the locale of the data style is used.

The `number:calendar` attribute is usable with the following elements: `<number:day>` 16.29.12, `<number:day-of-week>` 16.29.16, `<number:era>` 16.29.15, `<number:month>` 16.29.13, `<number:quarter>` 16.29.18, `<number:week-of-year>` 16.29.17 and `<number:year>` 16.29.14.

The values of the `number:calendar` attribute are `gregorian`, `gengou`, `ROC`, `hanja_yoil`, `hanja`, `hijri`, `jewish`, `buddhist` or a value of type `string` 18.2.

## 19.342 number:country

The `number:country` attribute specifies a country code for a data style. The country code is used for formatting properties whose evaluation is locale-dependent.

If a country is not specified, the system settings are used.

The `number:country` attribute on a `<number:currency-symbol>` element, specifies the country of a currency symbol.

The `number:country` attribute is usable with the following elements: `<number:boolean-style>` 16.29.24, `<number:currency-style>` 16.29.8, `<number:currency-symbol>` 16.29.9, `<number:date-style>` 16.29.11, `<number:number-style>` 16.29.2, `<number:percentage-style>` 16.29.10, `<number:text-style>` 16.29.26 and `<number:time-style>` 16.29.19.

The `number:country` attribute has the data type `countryCode` 18.3.11.

## 19.343 `number:decimal-places`

### 19.343.1 General

The `number:decimal-places` attribute specifies the number of decimal places to display.

### 19.343.2 `<number:number>`

If the `number:decimal-places` attribute is not specified, the number of decimal places specified by the default table cell style is used.

The `number:decimal-places` attribute is usable with the following element:  
`<number:number>` 16.29.3.

The `number:decimal-places` attribute has the data type `integer` 18.2.

### 19.343.3 `<number:seconds>`

If the `number:decimal-places` attribute is not specified or if the value of the attribute is 0, seconds are not displayed.

For a `<number:seconds>` element the default value for this attribute is 0.

The `number:decimal-places` attribute is usable with the following element:  
`<number:seconds>` 16.29.22.

The `number:decimal-places` attribute has the data type `integer` 18.2.

### 19.343.4 `<number:scientific-number>`

The `number:decimal-places` attribute specifies the number of decimal places to display.

If the `number:decimal-places` attribute is not specified, the number of decimal places specified by the default table cell style is used.

The `number:decimal-places` attribute is usable with the following element:  
`<number:scientific-number>` 16.29.6.

The `number:decimal-places` attribute has the data type `integer` 18.2.

## 19.344 `number:decimal-replacement`

The `number:decimal-replacement` attribute specifies a replacement text for decimal places if a number style specifies that decimal places are used but the number displayed is an integer.

**Note:** What replacement text is supported is implementation-dependent.

The `number:decimal-replacement` attribute is usable with the following element:  
`<number:number>` 16.29.3.

The `number:decimal-replacement` attribute has the data type `string` 18.2.

## 19.345 number:denominator-value

The `number:denominator-value` attribute specifies an integer value that is used as the denominator of a fraction. If this attribute is not present, a denominator that is appropriate for displaying the number is used.

**Note:** The `number:denominator-value` attribute is used with `number:grouping` 19.350, `number:min-denominator-digits` 19.353, `number:min-integer-digits` 19.355 and `number:min-numerator-digits` 19.357 to define the display of a floating-point number as a fraction.

The `number:denominator-value` attribute is usable with the following element:  
`<number: fraction> 16.29.7.`

The `number:denominator-value` attribute has the data type `integer` 18.2.

## 19.346 number:display-factor

The `number:display-factor` attribute specifies a factor by which each number is scaled (divided) before displaying.

The default value for this attribute is 1.

The `number:display-factor` attribute is usable with the following element:  
`<number: number> 16.29.3.`

The `number:display-factor` attribute has the data type `double` 18.2.

## 19.347 number:exponent-interval

The `number:exponent-interval` attribute determines the valid exponents to be used: the valid exponents are the integer multiples of the value of the `number:exponent-interval` attribute.

The default value for this attribute is 1.

The `number:exponent-interval` attribute is usable with the following element:  
`<number: scientific-number> 16.29.6.`

The `number:exponent-interval` attribute has the data type `positiveInteger` 18.2.

## 19.348 number:format-source

The `number:format-source` attribute specifies the source of definitions of the `short` and `long` display formats.

The defined values for the `number:format-source` attribute are:

- `fixed`: the values `short` and `long` of the `number:style` attribute are defined by this standard.
- `language`: the meaning of the values `long` and `short` of the `number:style` attribute depend upon the `number:language` and `number:country` attributes of the date style. If neither of those attributes are specified, consumers should use their default locale for short and long date and time formats.

The default value for this attribute is `fixed`.

The `number:format-source` attribute is usable with the following elements:  
`<number: date-style> 16.29.11` and `<number: time-style> 16.29.19.`

The values of the `number:format-source` attribute are `fixed` or `language`.

## 19.349 number:forced-exponent-sign

The `number:forced-exponent-sign` attribute specifies whether the sign of the exponent for a scientific number is always displayed.

The defined values for the `number:forced-exponent-sign` attribute are:

- `false`: the exponent sign is displayed only for a negative value of the exponent, otherwise it is not displayed.
- `true`: the exponent sign is always displayed regardless of the value of exponent.

The default value for this attribute is `true`.

The `number:forced-exponent-sign` attribute is usable with the following element:  
`<number:scientific-number> 16.29.6.`

The `number:forced-exponent-sign` attribute has the data type `boolean` 18.3.3.

## 19.350 number:grouping

The `number:grouping` attribute specifies whether the integer digits of a number should be grouped using a separator character.

The grouping character that is used and the number of digits that are grouped together depends on the language and country of the style.

The defined values for the `number:grouping` attribute are:

- `false`: integer digits of a number are not grouped using a separator character.
- `true`: integer digits of a number should be grouped by a separator character.

The default value for this attribute is `false`.

The `number:grouping` attribute is usable with the following elements:  
`<number:fraction> 16.29.7, <number:number> 16.29.3 and <number:scientific-number> 16.29.6.`

The `number:grouping` attribute has the data type `boolean` 18.3.3.

## 19.351 number:language

The `number:language` attribute specifies a language code. The country code is used for formatting properties whose evaluation is locale-dependent.

If a language code is not specified, either the system settings or the setting for the system's language are used, depending on the property whose value should be evaluated.

The `number:language` attribute is usable with the following elements:  
`<number:boolean-style> 16.29.24, <number:currency-style> 16.29.8, <number:currency-symbol> 16.29.9, <number:date-style> 16.29.11, <number:number-style> 16.29.2, <number:percentage-style> 16.29.10, <number:text-style> 16.29.26 and <number:time-style> 16.29.19.`

The `number:language` attribute has the data type `languageCode` 18.3.18.

## 19.352 number:max-denominator-value

The `number:max-denominator-value` attribute specifies the maximum denominator permitted to be chosen if its `<number:fraction>` element does not have a `number:denominator-value` attribute. The `number:max-denominator-value` attribute is ignored in the presence of a `number:denominator-value` 19.345 attribute. The absence of the `number:max-denominator-value` attribute indicates that no maximum denominator is specified.

**Note:** The `number:max-denominator-value` attribute is used with `number:grouping` 19.348, `number:min-denominator-digits` 19.350, `number:min-integer-digits` 19.352 and `number:min-numerator-digits` 19.353 to define the display of a floating-point number as a fraction.

**Note:** A `number:max-denominator-value` attribute value of 99 is equivalent to permitting any 1- or 2-digit denominator. A `number:max-denominator-value` attribute value of 999 is equivalent to permitting any 1, 2, or 3 digit denominator.

The `number:max-denominator-value` attribute is usable with the following element:  
`<number: fraction> 16.29.7.`

The `number:max-denominator-value` attribute has the data type `positiveInteger` 18.2.

## 19.353 number:min-denominator-digits

The `number:min-denominator-digits` attribute specifies the minimum number of digits to use to display the denominator of a fraction.

The `number:min-denominator-digits` attribute is usable with the following element:  
`<number: fraction> 16.29.7.`

The `number:min-denominator-digits` attribute has the data type `integer` 18.2.

## 19.354 number:min-exponent-digits

The `number:min-exponent-digits` attribute specifies the minimum number of digits to use to display an exponent.

The `number:min-exponent-digits` attribute is usable with the following element:  
`<number: scientific-number> 16.29.6.`

The `number:min-exponent-digits` attribute has the data type `integer` 18.2.

## 19.355 number:min-integer-digits

The `number:min-integer-digits` attribute specifies the minimum number of integer digits to display in the integer portion of a number, a scientific number, or a fraction.

For a `<number: fraction>` element, if the `number:min-integer-digits` attribute is not present, no integer portion is displayed.

The `number:min-integer-digits` attribute is usable with the following elements:  
`<number: fraction> 16.29.7, <number: number> 16.29.3 and <number: scientific-number> 16.29.6.`

The `number:min-integer-digits` attribute has the data type `integer` 18.2.

## 19.356 number:min-decimal-places

### 19.356.1 General

The `number:min-decimal-places` attribute specifies the minimum number of digits in the decimal part.

The value of the `number:min-decimal-places` attribute shall not be greater than the value of the `number:decimal-places` 19.343 attribute.

If the value of `number:min-decimal-places` is less than the value of `number:decimal-places`, trailing zero digits in decimal places following the position indicated by the value of `number:min-decimal-places` shall not be displayed.

### 19.356.2 `<number:number>`

If the `number:min-decimal-places` attribute is not specified, it is set to:

- 0, if the value of the `number:decimal-replacement` attribute is the empty string.
- the value of the `number:decimal-places` attribute in other cases.

If both `number:decimal-replacement` 19.344 and `number:min-decimal-places` attributes are present, and the number displayed is an integer, then the value of `number:decimal-replacement` shall be displayed.

The `number:min-decimal-places` attribute is usable with the following element:  
`<number:number> 16.29.3`.

The `number:min-decimal-places` attribute has the data type integer 18.2.

### 19.356.3 `<number:scientific-number>`

If the `number:min-decimal-places` attribute is not specified, it is set to the value of the `number:decimal-places` 19.343 attribute.

The `number:min-decimal-places` attribute is usable with the following element:  
`<number:scientific-number> 16.29.6`.

The `number:min-decimal-places` attribute has the data type integer 18.2.

### 19.357 `number:min-numerator-digits`

The `number:min-numerator-digits` attribute specifies the minimum number of digits to use to display the numerator in a fraction.

The `number:min-numerator-digits` attribute is usable with the following element:  
`<number:fraction> 16.29.7`.

The `number:min-numerator-digits` attribute has the data type integer 18.2.

### 19.358 `number:position`

The `number:position` attribute specifies the position where text appears.

The index of a position starts with 1 and is counted by digits from right to left in the integer part of a number, starting left from a decimal separator if one exists, or from the last digit of the number. Text is inserted before the digit at the specified position. If the value of `number:position` attribute is greater than the value of `number:min-integer-digits` 19.355 and greater than the number of integer digits in the number, text is prepended to the number.

The `number:position` attribute is usable with the following element:  
`<number:embedded-text> 16.29.4`.

The `number:position` attribute has the data type integer 18.2.

## 19.359 number:possessive-form

The `number:possessive-form` attribute specifies whether the month is displayed as a noun or using the possessive form.

The `number:possessive-form` attribute is only applied when a `number:textual` 19.363 attribute on the same `<number:month>` element has the value of `true`.

The defined values for the `number:possessive-form` attribute are:

- `false`: the name of the month is displayed in nominative form.
- `true`: the name of the month is displayed in possessive form.

The `number:possessive-form` attribute is usable with the following element:  
`<number:month> 16.29.13`.

The `number:possessive-form` attribute has the data type `boolean` 18.3.3.

## 19.360 number:rfc-language-tag

The `number:rfc-language-tag` attribute specifies a language identifier according to the rules of [RFC5646], or its successors.

It shall only be used if its value cannot be expressed as a valid combination of the `number:language` 19.351, `number:script` 19.361 and `number:country` 19.342 attributes.

Producers may add support for consumers that don't support the `number:rfc-language-tag` attribute by specifying `number:language`, `number:script` and `number:country` attributes with values as close as possible to the actual value of the `number:rfc-language-tag` attribute. Producers shall not use values for these attributes that contradict the value of the `table:rfc-language-tag` attribute.

The `number:rfc-language-tag` attribute is usable with the following elements:  
`<number:boolean-style> 16.29.24`, `<number:currency-style> 16.29.8`,  
`<number:currency-symbol> 16.29.9`, `<number:date-style> 16.29.11`,  
`<number:number-style> 16.29.2`, `<number:percentage-style> 16.29.10`,  
`<number:text-style> 16.29.26` and `<number:time-style> 16.29.19`.

The `number:rfc-language-tag` attribute has the data type `language` 18.3.17.

## 19.361 number:script

The `number:script` attribute specifies a script code. The script code is used for formatting properties whose evaluation is locale-dependent. The attribute should be used only if necessary according to the rules of §2.2.3 of [RFC5646], or its successors.

The `number:script` attribute is usable with the following elements: `<number:boolean-style> 16.29.24`, `<number:currency-style> 16.29.8`, `<number:currency-symbol> 16.29.9`, `<number:date-style> 16.29.11`, `<number:number-style> 16.29.2`, `<number:percentage-style> 16.29.10`, `<number:text-style> 16.29.26` and `<number:time-style> 16.29.19`.

The `number:script` attribute has the data type `scriptCode` 18.3.30.

## 19.362 number:style

### 19.362.1 General

The `number:style` attribute specifies whether the content of a time element is displayed in short or long format. The value of this attribute can be `short` or `long`. The meaning of these

values depends on the value of the `number:format-source` 19.348 attribute that is attached to a date or time style.

### 19.362.2 `<number:day>`

If the value of the `number:format-source` attribute is fixed the `number:style` attribute on the same element has the defined values:

- `short`: the day of month is displayed using one or two digits.
- `long`: the day of month is displayed using two digits.

The `number:style` attribute is usable with the following element: `<number:day>` 16.29.12.

The values of the `number:style` attribute are `short` or `long`.

### 19.362.3 `<number:day-of-week>`

If the value of the `number:format-source` attribute is fixed the `number:style` attribute on the same element has the defined values:

- `short`: abbreviated name of the day is displayed.
- `long`: full name of the day is displayed.

The `number:style` attribute is usable with the following element: `<number:day-of-week>` 16.29.16.

The values of the `number:style` attribute are `short` or `long`.

### 19.362.4 `<number:era>`

If the value of the `number:format-source` attribute is fixed the `number:style` attribute on the same element has the defined values:

- `short`: abbreviated era name is used.
- `long`: full era name is used.

The `number:style` attribute is usable with the following element: `<number:era>` 16.29.15.

The values of the `number:style` attribute are `short` or `long`.

### 19.362.5 `<number:hours>`

If the value of the `number:format-source` attribute is fixed the `number:style` attribute on the same element has the defined values:

- `short`: hours are displayed using at least one digit.
- `long`: hours are displayed using at least two digits.

The `number:style` attribute is usable with the following element: `<number:hours>` 16.29.20.

The values of the `number:style` attribute are `short` or `long`.

### 19.362.6 `<number:minutes>`

If the value of the `number:format-source` attribute is fixed the `number:style` attribute on the same element has the defined values:

- `short`: minutes are displayed using at least one digit.
- `long`: minutes are displayed using at least two digits.

The `number:style` attribute is usable with the following element: `<number:minutes>`  
16.29.21.

The values of the `number:style` attribute are `short` or `long`.

### 19.362.7 `<number:month>`

If the value of the `number:format-source` attribute is fixed the `number:style` attribute on the same element has the defined values:

- `short`: abbreviated name of the month is displayed or the month is displayed using one or two digits.
- `long`: full name of the month is displayed or the month is displayed using two digits.

The `number:style` attribute is usable with the following element: `<number:month>`  
16.29.13.

The values of the `number:style` attribute are `short` or `long`.

### 19.362.8 `<number:quarter>`

If the value of the `number:format-source` attribute is fixed the `number:style` attribute on the same element has the defined values:

- `short`: abbreviated name of the quarter is displayed.
- `long`: full name of the quarter is displayed.

The `number:style` attribute is usable with the following element: `<number:quarter>`  
16.29.18.

The values of the `number:style` attribute are `short` or `long`.

### 19.362.9 `<number:seconds>`

If the value of the `number:format-source` attribute is fixed the `number:style` attribute on the same element has the defined values:

- `short`: seconds are displayed using at least one digit.
- `long`: seconds are displayed using at least two digits.

The `number:style` attribute is usable with the following element: `<number:seconds>`  
16.29.22.

The values of the `number:style` attribute are `short` or `long`.

### 19.362.10 `<number:year>`

If the value of the `number:format-source` attribute is fixed the `number:style` attribute on the same element has the defined values:

- `short`: year is displayed using two digits.
- `long`: year is displayed using four digits.

The `number:style` attribute is usable with the following element: `<number:year>`  
16.29.14.

The values of the `number:style` attribute are `short` or `long`.

## 19.363 number:textual

The `number:textual` attribute specifies whether the name or number of a month is displayed in the month portion of a date.

The defined values for the `number:textual` element are:

- `false`: the number of the month is displayed.
- `true`: the name of the month is displayed.

The name or number of a month is defined by the `number:calendar` 19.341 attribute on the same parent element as the `number:textual` attribute.

The default value for this attribute is `false`.

The `number:textual` attribute is usable with the following element: `<number:month>` 16.29.13.

The `number:textual` attribute has the data type `boolean` 18.3.3.

## 19.364 number:title

The `number:title` attribute specifies the title of a data style.

The `number:title` attribute is usable with the following elements: `<number:boolean-style>` 16.29.24, `<number:currency-style>` 16.29.8, `<number:date-style>` 16.29.11, `<number:number-style>` 16.29.2, `<number:percentage-style>` 16.29.10, `<number:text-style>` 16.29.26 and `<number:time-style>` 16.29.19.

The `number:title` attribute has the data type `string` 18.2.

## 19.365 number:transliteration-country

The `number:transliteration-country` attribute specifies a country code in conformance with [RFC5646].

If no language/country (locale) combination is specified, the locale of the data style is used.

The `number:transliteration-country` attribute is usable with the following elements: `<number:boolean-style>` 16.29.24, `<number:currency-style>` 16.29.8, `<number:date-style>` 16.29.11, `<number:number-style>` 16.29.2, `<number:percentage-style>` 16.29.10, `<number:text-style>` 16.29.26 and `<number:time-style>` 16.29.19.

The `number:transliteration-country` attribute has the data type `countryCode` 18.3.11.

## 19.366 number:transliteration-format

The `number:transliteration-format` attribute specifies which number characters to use.

The value of the `number:transliteration-format` attribute shall be a decimal "DIGIT ONE" character with numeric value 1 as listed in the Unicode Character Database file UnicodeData.txt with value 'Nd' (Numeric decimal digit) in the General\_Category/Numeric\_Type property field 6 and value '1' in the Numeric\_Value fields 7 and 8, respectively as listed in DerivedNumericValues.txt

If no format is specified the default ASCII representation of Latin-Indic digits is used, other transliteration attributes present in that case are ignored.

The default value for this attribute is 1.

The `number:transliteration-format` attribute is usable with the following elements:  
`<number:boolean-style>` 16.29.24, `<number:currency-style>` 16.29.8,  
`<number:date-style>` 16.29.11, `<number:number-style>` 16.29.2,  
`<number:percentage-style>` 16.29.10, `<number:text-style>` 16.29.26 and  
`<number:time-style>` 16.29.19.

The `number:transliteration-format` attribute has the data type `string` 18.2.

## 19.367 `number:transliteration-language`

The `number:transliteration-language` attribute specifies a language code in conformance with [RFC5646].

If no language/country (locale) combination is specified, the locale of the data style is used.

The `number:transliteration-language` attribute is usable with the following elements:  
`<number:boolean-style>` 16.29.24, `<number:currency-style>` 16.29.8,  
`<number:date-style>` 16.29.11, `<number:number-style>` 16.29.2,  
`<number:percentage-style>` 16.29.10, `<number:text-style>` 16.29.26 and  
`<number:time-style>` 16.29.19.

The `number:transliteration-language` attribute has the data type `countryCode` 18.3.11.

## 19.368 `number:transliteration-style`

The `number:transliteration-style` attribute specifies the transliteration format of a number system.

The semantics of the values of the `number:transliteration-style` attribute are locale- and implementation-dependent.

The default value for this attribute is `short`.

The `number:transliteration-style` attribute is usable with the following elements:  
`<number:boolean-style>` 16.29.24, `<number:currency-style>` 16.29.8,  
`<number:date-style>` 16.29.11, `<number:number-style>` 16.29.2,  
`<number:percentage-style>` 16.29.10, `<number:text-style>` 16.29.26 and  
`<number:time-style>` 16.29.19.

The values of the `number:transliteration-style` attribute are `short`, `medium` or `long`.

## 19.369 `number:truncate-on-overflow`

The `number:truncate-on-overflow` attribute specifies if a time or duration for which the value to be displayed by the largest time component specified in the style is too large to be displayed using the value range for `<number:hours>` 16.29.20 (0 to 23), or `<number:minutes>` 16.29.21 or `<number:seconds>` 16.29.22 (0 to 59) is truncated or if the value range of this component is extended. The largest time component is those for which a value of "1" represents the longest period of time.

If a value gets truncated, then its value is displayed modulo 24 (for `<number:hours>`) or modulo 60 (for `<number:minutes>` and `<number:seconds>`).

If the value range of a component get extended, then values larger than 23 or 59 are displayed.

The defined values for the `number:truncate-on-overflow` element are:

- `false`: the value range of the component is extended.
- `true`: the value range of the component is not extended.

The default value for this attribute is true.

The `number:truncate-on-overflow` attribute is usable with the following element:  
`<number:time-style> 16.29.19.`

The `number:truncate-on-overflow` attribute has the data type boolean 18.3.3.

## 19.370 office:automatic-update

The `office:automatic-update` attribute specifies if DDE links are automatically updated or only upon user request.

The defined values for the `office:automatic-update` element are:

- `false`: DDE links are updated on user request only.
- `true`: DDE links are updated automatically.

The default value for this attribute is true.

The `office:automatic-update` attribute is usable with the following elements:  
`<office:dde-source> 14.7.5` and `<text:dde-connection-decl> 14.7.3.`

The `office:automatic-update` attribute has the data type boolean 18.3.3.

## 19.371 office:boolean-value

The `office:boolean-value` attribute specifies the Boolean value for a table cell (`<table:change-track-table-cell>`, `<table:covered-table-cell>` and `<table:table-cell>`), a text field (`<text:expression>`, `<text:user-defined>`, `<text:user-field-decl>` and `<text:variable-set>`), or a form property (`<form:list-value>` and `<form:property>`).

It specifies a default value for database columns and database column definitions (`<db:column> 12.35` and `<db:column-definition> 12.40`).

For `<text:user-defined>` elements, this attribute is only evaluated if the corresponding `<meta:user-defined> 4.3.3` element has attribute `meta:value-type="boolean"`.

The `office:boolean-value` attribute is usable with the following elements:  
`<db:column> 12.35`, `<db:column-definition> 12.40`, `<form:list-value> 13.10`,  
`<form:property> 13.8`, `<table:change-track-table-cell> 9.9.16`,  
`<table:covered-table-cell> 9.1.5`, `<table:table-cell> 9.1.4`,  
`<text:expression> 7.4.14`, `<text:user-defined> 7.5.6`, `<text:user-field-decl> 7.4.8` and `<text:variable-set> 7.4.4`.

The `office:boolean-value` attribute has the data type boolean 18.3.3.

## 19.372 office:conversion-mode

The `office:conversion-mode` attribute specifies the method by which the DDE server converts its data into numbers.

The defined values for the `office:conversion-mode` attribute are:

- `into-default-style-data-style`: numbers are converted into the data style that is specified by the default style.
- `into-english-number`: numbers are converted into the English default format.
- `keep-text`: numbers are not converted. They are treated as text.

The default value for this attribute is `into-default-style-data-style`.

The `office:conversion-mode` attribute is usable with the following element:  
`<office:dde-source>` 14.7.5.

The values of the `office:conversion-mode` attribute are `into-default-style-data-style`, `into-english-number` or `keep-text`.

## 19.373 office:currency

The `office:currency` attribute specifies the currency for a value of a table cell (`<table:change-track-table-cell>`, `<table:covered-table-cell>` and `<table:table-cell>`), a text field (`<text:expression>`, `<text:user-field-decl>` and `<text:variable-set>`), or a form property (`<form:list-value>` and `<form:property>`).

It specifies a default currency for database columns and database column definitions (`<db:column>` 12.35 and `<db:column-definition>` 12.40).

The `office:currency` attribute is usable with the following elements: `<db:column>` 12.35, `<db:column-definition>` 12.40, `<form:list-value>` 13.10, `<form:property>` 13.8, `<table:change-track-table-cell>` 9.9.16, `<table:covered-table-cell>` 9.1.5, `<table:table-cell>` 9.1.4, `<text:expression>` 7.4.14, `<text:user-field-decl>` 7.4.8 and `<text:variable-set>` 7.4.4.

The `office:currency` attribute has the data type `string` 18.2.

## 19.374 office:date-value

The `office:date-value` attribute specifies the date or date-time value for a table cell (`<table:change-track-table-cell>`, `<table:covered-table-cell>` and `<table:table-cell>`), a text field (`<text:expression>`, `<text:user-defined>`, `<text:user-field-decl>` and `<text:variable-set>`), or a form property (`<form:list-value>` and `<form:property>`).

It specifies a default value for database columns and database column definitions (`<db:column>` 12.35 and `<db:column-definition>` 12.40).

For `<text:user-defined>` elements, this attribute is only evaluated if the corresponding `<meta:user-defined>` 4.3.3 element has attribute `meta:value-type="date"`.

The `office:date-value` attribute is usable with the following elements: `<db:column>` 12.35, `<db:column-definition>` 12.40, `<form:list-value>` 13.10, `<form:property>` 13.8, `<table:change-track-table-cell>` 9.9.16, `<table:covered-table-cell>` 9.1.5, `<table:table-cell>` 9.1.4, `<text:expression>` 7.4.14, `<text:user-defined>` 7.5.6, `<text:user-field-decl>` 7.4.8 and `<text:variable-set>` 7.4.4.

The `office:date-value` attribute has the data type `dateOrDateTime` 18.3.14.

## 19.375 office:dde-application

The `office:dde-application` attribute specifies the name of the target application to use for the DDE connection.

The `office:dde-application` attribute is usable with the following elements:  
`<office:dde-source>` 14.7.5 and `<text:dde-connection-decl>` 14.7.3.

The `office:dde-application` attribute has the data type `string` 18.2.

## 19.376 office:dde-item

The `office:dde-item` attribute specifies which information the target application should deliver.

The `office:dde-item` attribute is usable with the following elements: `<office:dde-source>` 14.7.5 and `<text:dde-connection-decl>` 14.7.3.

The `office:dde-item` attribute has the data type `string` 18.2.

## 19.377 office:dde-topic

The `office:dde-topic` attribute specifies the name of the topic to use for the DDE connection.

The `office:dde-topic` attribute is usable with the following elements: `<office:dde-source>` 14.7.5 and `<text:dde-connection-decl>` 14.7.3.

The `office:dde-topic` attribute has the data type `string` 18.2.

## 19.378 office:display

The `office:display` attribute specifies whether an annotation is visible.

The `office:display` attribute is usable with the following element:  
`<office:annotation>` 14.1.

The `office:display` attribute has the data type `boolean` 18.3.3.

## 19.379 office:mimetype

The `office:mimetype` attribute specifies the document type for a document that is contained in a single XML file. Its values are the MIME types that are used for the packaged variant of office documents.

The `office:mimetype` attribute is usable with the following element:  
`<office:document>` 3.1.2.

The `office:mimetype` attribute has the data type `string` 18.2.

## 19.380 office:name

### 19.380.1 General

The `office:name` attribute specifies a name. The use of that name is specified for each element on which it appears.

### 19.380.2 `<draw:a>`

The `office:name` attribute specifies a name for a hyperlink. The name can serve as a target for other hyperlinks. The name does not have to be unique.

The `office:name` attribute is usable with the following element: `<draw:a>` 10.4.12.

The `office:name` attribute has the data type `string` 18.2.

### 19.380.3 `<draw:area-circle>`

The `office:name` attribute specifies a name for a shape represented by this element. The name does not have to be unique.

The `office:name` attribute is usable with the following element: `<draw:area-circle>`  
10.4.13.4.

The `office:name` attribute has the data type `string` 18.2.

#### 19.380.4 `<draw:area-rectangle>`

The `office:name` attribute specifies a name for a shape represented by this element. The name does not have to be unique.

The `office:name` attribute is usable with the following element: `<draw:area-rectangle>` 10.4.13.3.

The `office:name` attribute has the data type `string` 18.2.

#### 19.380.5 `<draw:area-polygon>`

The `office:name` attribute specifies a name for an image map. The name does not have to be unique.

The `office:name` attribute is usable with the following element: `<draw:area-polygon>`  
10.4.13.5.

The `office:name` attribute has the data type `string` 18.2.

#### 19.380.6 `<office:annotation>`

The `office:name` attribute specifies a name for an annotation. It is used to match `<office:annotation>` elements with the corresponding `<office:annotation-end>` 14.2 elements.

The name that appears in the `office:name` attribute of any `<office:annotation>`, `<office:annotation-end>` pair shall be unique to that pair of elements in a document instance.

The `office:name` attribute is usable with the following element: `<office:annotation>`  
14.1.

The `office:name` attribute has the data type `string` 18.2.

#### 19.380.7 `<office:annotation-end>`

The `office:name` attribute specifies a name for an annotation. It is used to match `<office:annotation-end>` elements with the corresponding `<office:annotation>` 14.1 elements.

The `office:name` attribute is usable with the following element: `<office:annotation-end>` 14.2.

The `office:name` attribute has the data type `string` 18.2.

#### 19.380.8 `<office:dde-source>`

The `office:name` attribute specifies the name by which a connection can be referenced.

The `office:name` attribute is usable with the following element: `<office:dde-source>`  
14.7.5.

The `office:name` attribute has the data type `string` 18.2.

### 19.380.9 <text:a>

The `office:name` attribute specifies a name for a hyperlink. The name serves as a target for other hyperlinks and need not be unique.

**Note:** This attribute is specified for compatibility with [HTML4] only, where an `<a>` element may serve as a link source and target simultaneously. This attribute should not be used for any purpose other than to represent links that came from a HTML document.

The `office:name` attribute is usable with the following element: `<text:a>` 6.1.8.

The `office:name` attribute has the data type `string` 18.2.

### 19.380.10 <text:dde-connection-decl>

The `office:name` attribute specifies a name by which a connection can be referenced.

The `office:name` attribute is usable with the following element: `<text:dde-connection-decl>` 14.7.3.

The `office:name` attribute has the data type `string` 18.2.

## 19.381

### 19.382 office:server-map

The `office:server-map` attribute specifies that a link returns data to a server-side image map. The mouse coordinates of the click 19.406 position of the graphic shape are appended to the IRI of the link. The coordinates are be used by the server to determine which link to activate within the image map.

The default value for this attribute is `false`.

The `office:server-map` attribute is usable with the following element: `<draw:a>` 10.4.12.

The `office:server-map` attribute has the data type `boolean` 18.3.3.

### 19.383 office:string-value

The `office:string-value` attribute specifies the string-value for a table cell (`<table:change-track-table-cell>`, `<table:covered-table-cell>` and `<table:table-cell>`), a text field (`<text:expression>`, `<text:user-defined>`, `<text:user-field-decl>` and `<text:variable-set>`), or a form property (`<form:list-value>` and `<form:property>`).

It specifies a default value for database columns and database column definitions (`<db:column>` 12.35 and `<db:column-definition>` 12.40).

For `<text:user-defined>` elements, this attribute is only evaluated if the corresponding `<meta:user-defined>` 4.3.3 element has (default) attribute `meta:value-type="string"`.

The `office:string-value` attribute is usable with the following elements: `<db:column>` 12.35, `<db:column-definition>` 12.40, `<form:list-value>` 13.10, `<form:property>` 13.8, `<table:change-track-table-cell>` 9.9.16, `<table:covered-table-cell>` 9.1.5, `<table:table-cell>` 9.1.4, `<text:expression>` 7.4.14, `<text:user-defined>` 7.5.6, `<text:user-field-decl>` 7.4.8 and `<text:variable-set>` 7.4.4.

The `office:string-value` attribute has the data type `string` 18.2.

## 19.384 office:target-frame

The `office:target-frame` attribute specifies a target frame.

The defined values for the `office:target-frame` attribute are:

- `_blank`: The referenced document is displayed in a new frame.
- `_parent`: The referenced document is displayed in the parent frame of the current frame.
- `_self`: The referenced document replaces the content of the current frame.
- `_top`: The referenced document is displayed in the topmost frame, that is the frame that contains the current frame as a child or descendant but is not contained within another frame.
- A frame name: The referenced document is displayed in the named frame. If the named frame does not exist, a new frame with that name is created.

The default value for this attribute is `_blank`.

The `office:target-frame` attribute is usable with the following elements:

`<form:button>` 13.5.15, `<form:form>` 13.3 and `<form:image>` 13.5.16.

The `office:target-frame` attribute has the data type `targetFrameName` 18.3.35.

## 19.385 office:target-frame-name

The `office:target-frame-name` attribute specifies the name of a target frame.

The defined values for the `office:target-frame-name` attribute are:

- `_blank`: The referenced document is displayed in a new frame.
- `_parent`: The referenced document is displayed in the parent frame of the current frame.
- `_self`: The referenced document replaces the content of the current frame.
- `_top`: The referenced document is displayed in the topmost frame, that is the frame that contains the current frame as a child or descendant but is not contained within another frame.
- A frame name: The referenced document is displayed in the named frame. If the named frame does not exist, a new frame with that name is created.

The `office:target-frame-name` attribute may be used together with an `xlink:show` 19.917 attribute. In that case, if the value of the attribute is `_blank`, the `xlink:show` attribute value should be `new`. If the value of the attribute is any of the other value options, the value of the `xlink:show` attribute should be `replace`.

The `office:target-frame-name` attribute is usable with the following elements:

`<draw:a>` 10.4.12, `<draw:area-circle>` 10.4.13.4, `<draw:area-polygon>` 10.4.13.5, `<draw:area-rectangle>` 10.4.13.3, `<meta:hyperlink-behaviour>` 4.3.2.14 and `<text:a>` 6.1.8.

The `office:target-frame-name` attribute has the data type `targetFrameName` 18.3.35.

## 19.386 office:time-value

The `office:time-value` attribute specifies the time value for a table cell (`<table:change-track-table-cell>`, `<table:covered-table-cell>` and `<table:table-cell>`), a text field (`<text:expression>`, `<text:user-defined>`, `<text:user-field-decl>` and `<text:variable-set>`), or a form property (`<form:list-value>` and `<form:property>`).

It specifies a default value for database columns and database column definitions (<db:column> 12.35 and <db:column-definition> 12.40).

For <text:user-defined> elements, this attribute is only evaluated if the corresponding <meta:user-defined> 4.3.3 element has attribute meta:value-type="time".

The office:time-value attribute is usable with the following elements: <db:column> 12.35, <db:column-definition> 12.40, <form:list-value> 13.10, <form:property> 13.8, <table:change-track-table-cell> 9.9.16, <table:covered-table-cell> 9.1.5, <table:table-cell> 9.1.4, <text:expression> 7.4.14, <text:user-defined> 7.5.6, <text:user-field-decl> 7.4.8 and <text:variable-set> 7.4.4.

The office:time-value attribute has the data type duration 18.2.

## 19.387 office:title

The office:title attribute specifies a short accessible description.

**Note:** See appendix D for guidelines on using this attribute.

The office:title attribute is usable with the following elements: <draw:a> 10.4.12 and <text:a> 6.1.8.

The office:title attribute has the data type string 18.2.

## 19.388 office:value

The office:value attribute specifies the currency, float or percentage value for a table cell (<table:change-track-table-cell>, <table:covered-table-cell> and <table:table-cell>), a text field (<text:expression>, <text:user-defined>, <text:user-field-decl> and <text:variable-set>), or a form property (<form:list-value> and <form:property>).

It specifies a default value for database columns and database column definitions (<db:column> 12.35 and <db:column-definition> 12.40).

For <text:user-defined> elements, this attribute is only interpreted if the corresponding <meta:user-defined> 4.3.3 element has attribute meta:value-type="float".

The office:value attribute is usable with the following elements: <db:column> 12.35, <db:column-definition> 12.40, <form:list-value> 13.10, <form:property> 13.8, <table:change-track-table-cell> 9.9.16, <table:covered-table-cell> 9.1.5, <table:table-cell> 9.1.4, <text:expression> 7.4.14, <text:user-defined> 7.5.6, <text:user-field-decl> 7.4.8 and <text:variable-set> 7.4.4.

## 19.389 office:value-type

The office:value-type attribute specifies the value-type of a table cell (<table:change-track-table-cell>, <table:covered-table-cell> and <table:table-cell>), a text field (<text:expression>, <text:user-field-decl>, <text:variable-decl>, <text:variable-input> and <text:variable-set>), or a form property (<form:list-value> and <form:property>).

It specifies a default value type of database columns and database column definitions (<db:column> 12.35 and <db:column-definition> 12.40).

The value type of each of these elements shall be specified. If the value type is not string or if the <table:table-cell> element content differs from the value of the element, the corresponding Value Attribute(s) (Table 14 - Value attributes) shall contain the value(s) of the element.

Table 14 - Value attributes

Value Type	Value Attribute(s)	Encoded as...	Example
boolean	office:boolean-value	true or false	"true"
currency	office:value and office:currency	Numeric value and currency symbol	"100" "USD"
date	office:date-value	Date value as specified in §3.2.9 of [xmlschema-2], or date and time value as specified in §3.2.7 of [xmlschema-2]	"2003-04-17"
error	office:error	String	"error"
float	office:value	Numeric value	"12.345"
percentage	office:value	Numeric value	"0.50"
string	office:string-value	Strings	"abc def"
time	office:time-value	Duration, as specified in §3.2.6 of [xmlschema-2]	"PT03H30M00S"
void	none		

If the value type is string and the `office:string-value` attribute is not present, the element content defines the value.

**Note:** The OpenDocument concept of field values and value types and their encoding in XML is also modeled on the corresponding XML for table cell attributes.

The `office:value-type` attribute is usable with the following elements: `<db:column> 12.35, <db:column-definition> 12.40, <form:list-property> 13.9, <form:property> 13.8, <table:change-track-table-cell> 9.9.16, <table:covered-table-cell> 9.1.5, <table:table-cell> 9.1.4, <text:expression> 7.4.14, <text:user-field-decl> 7.4.8, <text:variable-decl> 7.4.3, <text:variable-input> 7.4.6 and <text:variable-set> 7.4.4.`

The `office:value-type` attribute has the data type `valueType 18.3.39`.

## 19.390 office:version

The `office:version` attribute identifies the version of the OpenDocument specification that defines the associated element, its schema, its complete content, and its interpretation.

The `office:version` attribute shall be present in every `<office:document>, <office:document-content>, <office:document-styles>, <office:document-meta>, and <office:document-settings>` element in the XML documents that comprise an OpenDocument 1.4 document. The value of the `office:version` attribute shall be "1.4".

**Note:** The `office:version` attribute value was not required by OpenDocument specifications prior to 1.2.

The `office:version` attribute is usable with the following elements:  
`<office:document>` 3.1.2, `<office:document-content>` 3.1.3.2,  
`<office:document-meta>` 3.1.3.4, `<office:document-settings>` 3.1.3.5 and  
`<office:document-styles>` 3.1.3.3.

The only value of the `office:version` attribute is 1.4.

## 19.391 presentation:action

The `presentation:action` attribute specifies the type of action that is executed when an event is triggered.

The defined values for the `presentation:action` attribute are:

- `execute`: another application is launched when this event is triggered. The application is specified with the `xlink:href` 19.916 attribute on the same `<presentation:event-listener>` element.
- `fade-out`: the object that contains this event is faded out when this event is triggered. The attributes `presentation:effect` 19.398, `presentation:direction` 19.396, `presentation:speed` 19.421 and `presentation:start-scale` 19.422 can be used to set the effect.
- `first-page`: the presentation jumps to the first page of the current document.
- `hide`: the object that contains this event is hidden if the event is triggered.
- `last-page`: the presentation jumps to the last page of the current document.
- `last-visited-page`: the presentation jumps to the page that was displayed before the current page.
- `next-page`: the presentation jumps to the next page.
- `none`: no action is performed when this event is triggered.
- `previous-page`: the presentation jumps to the previous page.
- `show`: the target of an IRI is opened when this event is triggered. The IRI is specified with the `xlink:href` 19.916 attribute on the same `<presentation:event-listener>` element.
- `sound`: an audio effect is started when the effect is triggered. The audio effect is described by a `<presentation:sound>` 10.8.2 child element.
- `stop`: if a slide show is active, it will be stopped.
- `verb`: if the object that contains this event supports the execution of [OLE] verbs, the verb with the id set in the `presentation:verb` 19.432 attribute is executed.

The `presentation:action` attribute is usable with the following element:  
`<presentation:event-listener>` 10.9.2.

The values of the `presentation:action` attribute are `none`, `previous-page`, `next-page`, `first-page`, `last-page`, `hide`, `stop`, `execute`, `show`, `verb`, `fade-out`, `sound` or `last-visited-page`.

## 19.392 presentation:animations

The `presentation:animations` attribute enables or disables the playback of animations that are specified within embedded or linked images during a presentation.

The defined values for the `presentation:animations` attribute are:

- `disabled`: image animations cannot be played during a presentation.

- `enabled`: image animations can be played during a presentation.

The default value for this attribute is `enabled`.

The `presentation:animations` attribute is usable with the following element:  
`<presentation:settings>` 10.9.4.7.

The values of the `presentation:animations` attribute are `enabled` or `disabled`.

## 19.393 `presentation:class`

The `presentation:class` attribute classifies presentation shapes by their usage within a draw page.

The defined values for the `presentation:class` attribute are:

- `chart`: presentation charts are standard object shapes.
- `graphic`: presentation graphics are standard graphic shapes
- `handout`: presentation handouts are placeholder for the drawing page in a handout page
- `notes`: presentation notes are used on notes pages
- `object`: presentation objects are standard object shapes
- `orgchart`: presentation organization charts are standard object shapes.
- `outline`: outlines are standard text shapes
- `page`: presentation pages are used on notes pages
- `subtitle`: subtitles are standard text shapes
- `table`: presentation tables are standard object shapes
- `text`: presentation texts are standard text shapes
- `title`: titles are standard text shapes

The `date-time`, `footer`, `header`, and `page-number` classes can be used only for drawing shapes that are contained in master pages. Depending on the settings of the `<style:master-page>` 16.9 element, they are displayed automatically on drawing pages that use the master page.

When used with drawing shapes contained in master pages, the defined values for the `presentation:class` attribute are:

- `date-time`: drawing shape is used as a date and/or time shape. Date and Time shapes are standard text shapes.
- `footer`: drawing shape is used as a footer. Footer shapes are standard text shapes.
- `header`: drawing shape is used as a header. Header shapes are standard text shapes.
- `page-number`: drawing shape is used as a page number shape. Page Number shapes are standard text shapes.

The `presentation:class` attribute is usable with the following elements: `<draw:frame>` 10.4.2 and `<draw:page-thumbnail>` 10.3.14.

The values of the `presentation:class` attribute are `title`, `outline`, `subtitle`, `text`, `graphic`, `object`, `chart`, `table`, `orgchart`, `page`, `notes`, `handout`, `header`, `footer`, `date-time` or `page-number`.

## 19.394 presentation:class-names

The `presentation:class-names` attribute specifies a white space separated list of styles with the family value of `presentation`. The referenced styles are applied in the order they are contained in the list.

If both `presentation:style-name` and `presentation:class-names` are present, the style referenced by the `presentation:style-name` 19.426 attribute is applied before the styles referenced by the `presentation:class-names` attribute.

The `presentation:class-names` attribute is usable with the following elements:

```
<dr3d:cube> 10.5.4, <dr3d:extrude> 10.5.6, <dr3d:rotate> 10.5.7, <dr3d:scene> 10.5.2, <dr3d:sphere> 10.5.5, <draw:caption> 10.3.11, <draw:circle> 10.3.8, <draw:connector> 10.3.10, <draw:control> 10.3.13, <draw:custom-shape> 10.6.1, <draw:ellipse> 10.3.9, <draw:frame> 10.4.2, <draw:g> 10.3.15, <draw:line> 10.3.3, <draw:measure> 10.3.12, <draw:page-thumbnail> 10.3.14, <draw:path> 10.3.7, <draw:polygon> 10.3.5, <draw:polyline> 10.3.4, <draw:rect> 10.3.2, <draw:regular-polygon> 10.3.6 and <office:annotation> 14.1.
```

The `presentation:class-names` attribute has the data type `styleNameRefs` 18.3.34.

## 19.395 presentation:delay (deprecated)

The `presentation:delay` attribute specifies the delay before a presentation effect starts after the previous one has finished.

The `presentation:delay` attribute is usable with the following elements:

```
<presentation:hide-shape> 10.8.5, <presentation:hide-text> 10.8.6, <presentation:show-shape> 10.8.3 and <presentation:show-text> 10.8.4.
```

The `presentation:delay` attribute has the data type `duration` 18.2.

## 19.396 presentation:direction

The `presentation:direction` attribute specifies the direction of an effect.

The defined values for the `presentation:direction` attribute are:

- `clockwise`: effect moves in a clockwise direction.
- `counter-clockwise`: effect moves in a counter-clockwise direction.
- `from-bottom`: effect moves from the bottom of the presentation area.
- `from-center`: effect moves from the center of the presentation area.
- `from-left`: effect moves from the left of the presentation area.
- `from-right`: effect moves from the right of the presentation area.
- `from-top`: effect moves from the top of the presentation area.
- `from-lower-left`: effect moves from the lower-left of the presentation area.
- `from-lower-right`: effect moves from the lower-right of the presentation area.
- `from-upper-left`: effect moves from the upper-left of the presentation area.
- `from-upper-right`: effect moves from the upper-right of the presentation area.
- `horizontal`: effect moves from the bottom of the presentation area.
- `none`: no presentation effect direction
- `path`: effect moves from the bottom of the presentation area.
- `spiral-inward-left`: effect spirals inward from the left of the presentation area.

- **spiral-inward-right**: effect spirals inward from the right of the presentation area.
- **spiral-outward-left**: effect spirals outward to the left of the presentation area.
- **spiral-outward-right**: effect spirals outward to the right of the presentation area.
- **to-left**: effect moves to the left of the presentation area.
- **to-top**: effect moves to the top of the presentation area.
- **to-right**: effect moves to the right of the presentation area.
- **to-bottom**: effect moves to the bottom of the presentation area.
- **to-center**: effect moves from the center of the presentation area.
- **to-lower-left**: effect moves to the lower-left of the presentation area.
- **to-upper-left**: effect moves to the upper-left of the presentation area.
- **to-lower-right**: effect moves to the lower-right of the presentation area.
- **to-upper-right**: effect moves to the upper-right of the presentation area.
- **vertical**: effect moves vertically in the presentation area.

The default value for this attribute is `none`.

The `presentation:direction` attribute is usable with the following elements:  
`<presentation:event-listener>` 10.9.2, `<presentation:hide-shape>` 10.8.5,  
`<presentation:hide-text>` 10.8.6, `<presentation:show-shape>` 10.8.3 and  
`<presentation:show-text>` 10.8.4.

The values of the `presentation:direction` attribute are `none`, `from-left`, `from-top`, `from-right`, `from-bottom`, `from-center`, `from-upper-left`, `from-upper-right`, `from-lower-left`, `from-lower-right`, `to-left`, `to-top`, `to-right`, `to-bottom`, `to-upper-left`, `to-upper-right`, `to-lower-right`, `to-lower-left`, `path`, `spiral-inward-left`, `spiral-inward-right`, `spiral-outward-left`, `spiral-outward-right`, `vertical`, `horizontal`, `to-center`, `clockwise` or `counter-clockwise`.

## 19.397 `presentation:endless`

The `presentation:endless` attribute specifies whether a presentation repeats indefinitely or not.

The defined values for the `presentation:endless` attribute are:

- **false**: presentation does not repeat indefinitely.
- **true**: presentation repeats indefinitely.

The default value for this attribute is `false`.

The `presentation:endless` attribute is usable with the following element:  
`<presentation:settings>` 10.9.4.7.

The `presentation:endless` attribute has the data type `boolean` 18.3.3.

## 19.398 `presentation:effect`

The `presentation:effect` attribute specifies the type of an effect.

The defined values for the `presentation:effect` attribute are:

- **appear**: the shape is faded in by just switching its state from invisible to visible.

- `checkerboard`: the shape is faded in or out by drawing or removing checkerboard-like blocks that increase in size over time.
- `close`: the shape is drawn or removed line by line, either horizontally or vertically, starting at the edge of the shape.
- `dissolve`: the shape is faded in or out by drawing or removing small blocks in a random fashion.
- `fade`: the shape fades from its visible or hidden state to a hidden or visible state.
- `hide`: the shape is faded out by just switching its state from visible to invisible.
- `laser`: this effect is only available for text shapes; the characters of the text are moved one by one from the top edge of the screen to their final position.
- `lines`: the shape is faded in and out by drawing or removing line by line, either horizontally or vertically, in a random fashion.
- `move`: the shape moves from or to its final position.
- `move-short`: like the move effect, but the moving shape is clipped to its final bounding rectangle during fade.
- `none`: no effect is used.
- `open`: the shape is drawn or removed line by line, either horizontally or vertically, starting at the center of the shape.
- `random`: an effect is chosen at random to fade the shape in or out.
- `rotate`: the shape rotates horizontally or vertically for a short amount of time during this effect.
- `stretch`: the shape is faded in or out by changing its size during this effect.
- `stripes`: the shape is faded in or out by drawing or removing horizontal or vertical stripes that change their size.
- `wavyline`: the shape is faded in and out by drawing or removing small blocks in a snake like fashion.

The default value for this attribute is `none`.

The `presentation:effect` attribute is usable with the following elements:  
`<presentation:event-listener>` 10.9.2, `<presentation:hide-shape>` 10.8.5,  
`<presentation:hide-text>` 10.8.6, `<presentation:show-shape>` 10.8.3 and  
`<presentation:show-text>` 10.8.4.

The values of the `presentation:effect` attribute are `none`, `fade`, `move`, `stripes`, `open`, `close`, `dissolve`, `wavyline`, `random`, `lines`, `laser`, `appear`, `hide`, `move-short`, `checkerboard`, `rotate` or `stretch`.

## 19.399 `presentation:force-manual`

The `presentation:force-manual` attribute specifies that the value of the `presentation:transition-type` 20.240 attribute is set to `manual`.

The defined values for the `presentation:force-manual` attribute are:

- `false`: `presentation:transition-type` attribute is not set to `manual`.
- `true`: `presentation:transition-type` attribute is set to `manual`.

The default value for this attribute is `false`.

The `presentation:force-manual` attribute is usable with the following element:  
`<presentation:settings>` 10.9.4.7.

The `presentation:force-manual` attribute has the data type `boolean` 18.3.3.

## 19.400 presentation:full-screen

The `presentation:full-screen` attribute specifies whether the presentation is displayed in full-screen mode or in a window.

The defined values for the `presentation:full-screen` attribute are:

- `false`: presentation is displayed in a window.
- `true`: presentation is displayed in full-screen mode.

The default value for this attribute is `true`.

The `presentation:full-screen` attribute is usable with the following element:  
`<presentation:settings>` 10.9.4.7.

The `presentation:full-screen` attribute has the data type `boolean` 18.3.3.

## 19.401 presentation:group-id

The `presentation:group-id` attribute specifies a group id. This id can be used to group animation elements within a user interface, where a group consists of all animation elements that have the same group id.

The `presentation:group-id` attribute is usable with the following elements:  
`<anim:audio>` 15.5, `<anim:command>` 15.6.1, `<anim:iterate>` 15.4.4, `<anim:par>` 15.4.2 and `<anim:seq>` 15.4.3.

The `presentation:group-id` attribute has the data type `string` 18.2.

## 19.402 presentation:master-element

The `presentation:master-element` attribute specifies the id of an animation element.

**Note:** Consumer user interfaces may consider animation elements that have a `presentation:master-element` attribute to be a part of the animation element that is referenced, and may therefore exclude them from any lists of defined animations.

The `presentation:master-element` attribute is usable with the following elements:  
`<anim:audio>` 15.5, `<anim:command>` 15.6.1, `<anim:iterate>` 15.4.4, `<anim:par>` 15.4.2 and `<anim:seq>` 15.4.3.

The `presentation:master-element` attribute has the data type `IDREF` 18.2.

## 19.403 presentation:mouse-as-pen

The `presentation:mouse-as-pen` attribute specifies if a mouse pointer is displayed as a pen or a pointer.

The defined values for the `presentation:mouse-as-pen` attribute are:

- `false`: mouse pointer is displayed as a pointer.
- `true`: mouse pointer is displayed as a pen for drawing.

The default value for this attribute is `false`.

The `presentation:mouse-as-pen` attribute is usable with the following element:  
`<presentation:settings>` 10.9.4.7.

The `presentation:mouse-as-pen` attribute has the data type `boolean` 18.3.3.

## 19.404 presentation:mouse-visible

The `presentation:mouse-visible` attribute specifies whether a mouse pointer is visible during a presentation.

The defined values for the `presentation:mouse-visible` attribute are:

- `false`: mouse not visible during presentation.
- `true`: mouse visible during presentation.

The default value for this attribute is `true`.

The `presentation:mouse-visible` attribute is usable with the following element:  
`<presentation:settings>` 10.9.4.7.

The `presentation:mouse-visible` attribute has the data type `boolean` 18.3.3.

## 19.405 presentation:name

The `presentation:name` attribute identifies the element on which it appears inside a presentation.

The `presentation:name` attribute is usable with the following elements:  
`<presentation:date-time-decl>` 10.9.4.6, `<presentation:footer-decl>` 10.9.4.4, `<presentation:header-decl>` 10.9.4.2 and `<presentation:show>` 10.9.4.8.

The `presentation:name` attribute has the data type `string` 18.2.

## 19.406 presentation:node-type

The `presentation:node-type` attribute specifies a node type for an animation element.

The defined values for the `presentation:node-type` attribute are:

- `after-previous`: this animation element is the root element of an effect that starts after the previous effect.
- `default`: no node type is defined. This is the default setting.
- `interactive-sequence`: this animation element is the root element for a sequence of effects that are started when the user selects an element inside a page.
- `main-sequence`: this animation element is the root element for the main sequence of effects of a page
- `on-click`: this animation element is the root element of an effect that starts with a user click.
- `timing-root`: this animation element is the root element for the animation of a page.
- `with-previous`: this animation element is the root element of an effect that starts with the previous effect.

**Note:** Click is the action of a computer user moving a cursor or focus to a location on a screen (point) and then pressing a mouse button, usually the left button(click), or other pointing device. A click can be used with any number of input devices varying from mice, touch pads, keyboards, joysticks, scroll buttons, and roller balls.

The default value for this attribute is `default`.

The `presentation:node-type` attribute is usable with the following elements:  
`<anim:audio>` 15.5, `<anim:command>` 15.6.1, `<anim:iterate>` 15.4.4, `<anim:par>` 15.4.2 and `<anim:seq>` 15.4.3.

The values of the `presentation:node-type` attribute are `default`, `on-click`, `with-previous`, `after-previous`, `timing-root`, `main-sequence` or `interactive-sequence`.

## 19.407 presentation:object

The `presentation:object` attribute specifies the type of object that a `<presentation:placeholder>` element represents.

The defined values for the `presentation:object` attribute are the same as those defined for the `presentation:class` attribute. 19.393

The `presentation:object` attribute is usable with the following element:  
`<presentation:placeholder>` 16.44.

The values of the `presentation:object` attribute are `title`, `outline`, `subtitle`, `text`, `graphic`, `object`, `chart`, `table`, `orgchart`, `page`, `notes`, `handout`, `header`, `footer`, `date-time` or `page-number`.

## 19.408 presentation:pages

The `presentation:pages` attribute specifies a comma "," (U+002C, COMMA) separated list of page names as given by `draw:name` 19.197 attributes on `<draw:page>` 10.2.4 elements. The pages are displayed in the order in which they are listed during a presentation that uses this show. Pages can be included more than once.

The `presentation:pages` attribute is usable with the following element:  
`<presentation:show>` 10.9.4.8.

The `presentation:pages` attribute has the data type `string` 18.2.

## 19.409 presentation:path-id (deprecated)

The `presentation:path-id` attribute specifies a polygon shape by the value of its `draw:shape-id` 19.210 attribute. An effect moves along the lines of the specified polygon. The referenced polygon is not visible during the presentation.

The `presentation:path-id` attribute is usable with the following elements:  
`<presentation:hide-shape>` 10.8.5, `<presentation:hide-text>` 10.8.6,  
`<presentation:show-shape>` 10.8.3 and `<presentation:show-text>` 10.8.4.

The `presentation:path-id` attribute has the data type `string` 18.2.

## 19.410 presentation:pause

The `presentation:pause` attribute specifies a time duration for displaying a pause screen before the presentation is played again if the presentation is to be repeated indefinitely. If this attribute is not set or has a value of `P0s`, a pause screen is not displayed in endless mode.

The `presentation:pause` attribute is usable with the following element:  
`<presentation:settings>` 10.9.4.7.

The `presentation:pause` attribute has the data type `duration` 18.2.

## 19.411 presentation:placeholder

The `presentation:placeholder` attribute specifies if a shape is a placeholder or a presentation object with content.

- `true`: the shape is a placeholder.

- `false`: the shape is a presentation object with content.

The default value is `false`.

The `presentation:placeholder` attribute is usable with the following elements: `<draw:frame>` 10.4.2 and `<draw:page-thumbnail>` 10.3.14.

The `presentation:placeholder` attribute has the data type `boolean` 18.3.3.

## 19.412 presentation:play-full

The `presentation:play-full` attribute specifies when an effect starts.

The defined values for the `presentation:play-full` attribute are:

- `false`: effect starts after current effect ends.
- `true`: effect starts after the sound specified by its parent `<presentation:sound>` element ends.

The `presentation:play-full` attribute is usable with the following element: `<presentation:sound>` 10.8.2.

The `presentation:play-full` attribute has the data type `boolean` 18.3.3.

## 19.413 presentation:presentation-page-layout-name

The `presentation:presentation-page-layout-name` attribute references the `style:name` 19.502 attribute of a `<style:presentation-page-layout>` 16.43 element.

The `presentation:presentation-page-layout-name` attribute is usable with the following elements: `<draw:page>` 10.2.4 and `<style:handout-master>` 10.2.1.

The `presentation:presentation-page-layout-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.414 presentation:preset-class

The `presentation:preset-class` attribute specifies the class of the preset that was used to create an animation element.

The defined values for the `presentation:preset-class` attribute are:

- `custom`: the preset was a user-defined one. This is the default setting.
- `entrance`: the preset was an entrance effect.
- `exit`: the preset was an exit effect.
- `emphasis`: the preset was an emphasis effect.
- `motion-path`: the preset was a motion path.
- `ole-action`: the preset was an ole action.
- `media-call`: the preset was a media call.

The default value for this attribute is `custom`.

The `presentation:preset-class` attribute is usable with the following elements: `<anim:audio>` 15.5, `<anim:command>` 15.6.1, `<anim:iterate>` 15.4.4, `<anim:par>` 15.4.2 and `<anim:seq>` 15.4.3.

The values of the `presentation:preset-class` attribute are `custom`, `entrance`, `exit`, `emphasis`, `motion-path`, `ole-action` or `media-call`.

## 19.415 presentation:preset-id

The `presentation:preset-id` attribute specifies the name of the preset that was used to create an animation element.

The `presentation:preset-id` attribute is usable with the following elements:

`<anim:audio>` 15.5, `<anim:command>` 15.6.1, `<anim:iterate>` 15.4.4, `<anim:par>` 15.4.2 and `<anim:seq>` 15.4.3.

The `presentation:preset-id` attribute has the data type `string` 18.2.

## 19.416 presentation:preset-sub-type

The `presentation:preset-sub-type` attribute specifies the sub-type of the preset that was used to create an animation element.

The `presentation:preset-sub-type` attribute is usable with the following elements:

`<anim:audio>` 15.5, `<anim:command>` 15.6.1, `<anim:iterate>` 15.4.4, `<anim:par>` 15.4.2 and `<anim:seq>` 15.4.3.

The `presentation:preset-sub-type` attribute has the data type `string` 18.2.

## 19.417 presentation:show

The `presentation:show` attribute specifies the name of a `<presentation:show>` element that is used for the presentation. If the `presentation:start-page` 19.423 attribute is set, it overrides the value of this attribute.

The `presentation:show` attribute is usable with the following element:

`<presentation:settings>` 10.9.4.7.

The `presentation:show` attribute has the data type `string` 18.2.

## 19.418 presentation:show-end-of-presentation-slide

The `presentation:show-end-of-presentation-slide` attribute specifies whether an additional slide should be shown at the end of the presentation, indicating the presentation is finished.

The slide content is not defined within the document, but is implementation-defined.

The defined values for the `presentation:show-end-of-presentation-slide` attribute are:

- `false`: end-of-presentation slide not shown at end of presentation.
- `true`: end-of-presentation slide shown at end of presentation.

The default value for this attribute is `true`.

The `presentation:show-end-of-presentation-slide` attribute is usable with the following element: `<presentation:settings>` 10.9.4.7.

The `presentation:show-end-of-presentation-slide` attribute has the data type `boolean` 18.3.3.

## 19.419 presentation:show-logo

The `presentation:show-logo` attribute specifies whether an implementation-dependent logo is shown on a pause screen.

The defined values for the `presentation:show-logo` attribute are:

- `false`: logo not shown on pause screen.

- `true`: logo shown on pause screen.

The default value for this attribute is `false`.

The `presentation:show-logo` attribute is usable with the following element:  
`<presentation:settings>` 10.9.4.7.

The `presentation:show-logo` attribute has the data type `boolean` 18.3.3.

## 19.420 presentation:source

The `presentation:source` attribute specifies whether the current date/time or the fixed content of the field declaration is displayed.

- `fixed`: display content of the attribute's `<presentation:date-time-decl>` element
- `current-date`: display the current date/time as reported by the operating system

The `presentation:source` attribute is usable with the following element:  
`<presentation:date-time-decl>` 10.9.4.6.

The values of the `presentation:source` attribute are `fixed` or `current-date`.

## 19.421 presentation:speed

The `presentation:speed` attribute specifies the speed of an effect.

The defined values for the `presentation:speed` attribute are:

- `fast`: The `fast` speed should be faster than the speed the implementation chooses for the values `medium` and `slow`.
- `medium`: The `medium` speed should be faster than the speed the implementation chooses for the value `slow`, and slower than the speed the implementation chooses for the value `fast`.
- `slow`: The `slow` speed should be slower than the speed the implementation chooses for the values `medium` and `fast`.

The rates specified by these values are implementation-defined.

The default value for this attribute is `medium`.

The `presentation:speed` attribute is usable with the following elements:  
`<presentation:event-listener>` 10.9.2, `<presentation:hide-shape>` 10.8.5,  
`<presentation:hide-text>` 10.8.6, `<presentation:play>` 10.8.8,  
`<presentation:show-shape>` 10.8.3 and `<presentation:show-text>` 10.8.4.

The values of the `presentation:speed` attribute are `slow`, `medium` or `fast`.

## 19.422 presentation:start-scale

The `presentation:start-scale` attribute specifies the start size of a shape as a percentage of its original size, which is given by the `svg:width` 19.575 and `svg:height` 19.543 attributes of the element specifying the shape.

The default value for this attribute is `100%`.

The `presentation:start-scale` attribute is usable with the following elements:  
`<presentation:event-listener>` 10.9.2, `<presentation:hide-shape>` 10.8.5,  
`<presentation:hide-text>` 10.8.6, `<presentation:show-shape>` 10.8.3 and  
`<presentation:show-text>` 10.8.4.

The `presentation:start-scale` attribute has the data type `percent` 18.3.24.

## 19.423 presentation:start-page

The `presentation:start-page` attribute specifies the name of the page on which the presentation starts. If this attribute is set, it overrides the `presentation:show` 19.417 attribute.

The `presentation:start-page` attribute is usable with the following element:  
`<presentation:settings>` 10.9.4.7.

The `presentation:start-page` attribute has the data type `string` 18.2.

## 19.424 presentation:start-with-navigator

The `presentation:start-with-navigator` attribute specifies whether a navigator window is initially displayed during a presentation.

The defined values for the `presentation:start-with-navigator` attribute are:

- `false`: navigator window not displayed initially during a presentation.
- `true`: navigator window displayed initially during a presentation.

The default value for this attribute is `false`.

The `presentation:start-with-navigator` attribute is usable with the following element: `<presentation:settings>` 10.9.4.7.

The `presentation:start-with-navigator` attribute has the data type `boolean` 18.3.3.

## 19.425 presentation:stay-on-top

The `presentation:stay-on-top` attribute specifies whether a presentation window is displayed on top of other windows during a presentation.

The defined values for the `presentation:stay-on-top` attribute are:

- `false`: presentation window need not be displayed on top of other windows during a presentation.
- `true`: presentation window shall be displayed on top of other windows during a presentation.

The default value for this attribute is `false`.

The `presentation:stay-on-top` attribute is usable with the following element:  
`<presentation:settings>` 10.9.4.7.

The `presentation:stay-on-top` attribute has the data type `boolean` 18.3.3.

## 19.426 presentation:style-name

The `presentation:style-name` attribute specifies a style for a presentation shape.

The value of the attribute is the name of a `<style:style>` 16.2 element that has a family value of `presentation`. The formatting properties of the specified style and its parent styles are used to format the shape.

The `presentation:style-name` attribute is usable with the following elements:  
`<dr3d:cube>` 10.5.4, `<dr3d:extrude>` 10.5.6, `<dr3d:rotate>` 10.5.7, `<dr3d:scene>` 10.5.2, `<dr3d:sphere>` 10.5.5, `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:connector>` 10.3.10, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:g>` 10.3.15, `<draw:line>` 10.3.3, `<draw:measure>` 10.3.12, `<draw:thumbnail>` 10.3.14, `<draw:path>`

10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2, `<draw:regular-polygon>` 10.3.6 and `<office:annotation>` 14.1.

The `presentation:style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.427 `presentation:transition-on-click`

The `presentation:transition-on-click` attribute specifies whether a mouse click 19.406 on a slide during a presentation triggers a transition.

The defined values for the `presentation:transition-on-click` attribute are:

- `disabled`: mouse click on a slide during a presentation does not trigger a transition.
- `enabled`: mouse click on a slide during a presentation triggers a transition.

The default value for this attribute is `enabled`.

The `presentation:transition-on-click` attribute is usable with the following element: `<presentation:settings>` 10.9.4.7.

The values of the `presentation:transition-on-click` attribute are `enabled` or `disabled`.

## 19.428 `presentation:use-date-time-name`

The `presentation:use-date-time-name` attribute specifies the name of the `<presentation:date-time-decl>` 10.9.4.6 that is used for all `<presentation:date-time>` 10.9.4.5 elements that are displayed on a page.

The `presentation:use-date-time-name` attribute is usable with the following elements: `<draw:page>` 10.2.4, `<presentation:notes>` 16.19 and `<style:handout-master>` 10.2.1.

The `presentation:use-date-time-name` attribute has the data type `string` 18.2.

## 19.429 `presentation:use-footer-name`

The `presentation:use-footer-name` attribute specifies the name of the `<presentation:footer-decl>` 10.9.4.4 element that is used for all `<presentation:footer>` 10.9.4.3 elements that are displayed on a page.

The `presentation:use-footer-name` attribute is usable with the following elements: `<draw:page>` 10.2.4, `<presentation:notes>` 16.19 and `<style:handout-master>` 10.2.1.

The `presentation:use-footer-name` attribute has the data type `string` 18.2.

## 19.430 `presentation:use-header-name`

The `presentation:use-header-name` attribute specifies the name of the `<presentation:header-decl>` 10.9.4.2 element that is used for all `<presentation:header>` 10.9.4.1 elements that are displayed on a page.

The `presentation:use-header-name` attribute is usable with the following elements: `<draw:page>` 10.2.4, `<presentation:notes>` 16.19 and `<style:handout-master>` 10.2.1.

The `presentation:use-header-name` attribute has the data type `string` 18.2.

## 19.431 presentation:user-transformed

The `presentation:user-transformed` attribute specifies whether the size and position of the shape is set by the user or is set by the corresponding presentation shape on the master page.

The defined values for the `presentation:user-transformed` attribute are:

- `false`: size and position of shape set by corresponding presentation shape on master page.
- `true`: size and position of shape set by user.

The `presentation:user-transformed` attribute is usable with the following elements: `<draw:frame>` 10.4.2 and `<draw:page-thumbnail>` 10.3.14.

The `presentation:user-transformed` attribute has the data type `boolean` 18.3.3.

## 19.432 presentation:verb

The `presentation:verb` attribute specifies the [OLE] verb that is executed for event listeners of type `verb` at the object that contains this event.

The `presentation:verb` attribute is usable with the following element: `<presentation:event-listener>` 10.9.2.

The `presentation:verb` attribute has the data type `nonNegativeInteger` 18.2.

## 19.433 script:event-name

The `script:event-name` attribute specifies the name of an event.

Event names should begin with a namespace prefix followed by a ":" (U+003A, COLON).

Where appropriate, producers should use the event names described in [DOMEvents2]. The corresponding namespace is "<http://www.w3.org/2001/xml-events>".

**Note:** Event names defined in [DOMEvents2] are not namespaced. If used in OpenDocument, they should be preceded by a namespace prefix as described above

Table 15 specifies events defined in [DOMEvents2] that have an equivalent event in HTML. The namespace used for these events is "<http://www.w3.org/2001/xml-events>".

*Table 15 - Script Events*

<b><i>Value of <code>script:event-name</code> Attribute</i></b>	<b><i>Equivalent HTML Event</i></b>	<b><i>Description of Event</i></b>
<code>dom:change</code>	<code>onchange</code>	Occurs when a control is no longer focused and the value of the control was modified since it was given focus.
<code>dom:DOMFocusIn</code>	<code>onfocus</code>	Occurs when a control is given focus using the mouse or the TAB key.
<code>dom:DOMFocusOut</code>	<code>onblur</code>	Occurs when a control is no longer focused as a result of moving the mouse or by tabbing navigation.
<code>dom:mousedown</code>	<code>onmousedown</code>	Occurs when a mouse button is pressed on a control.
<code>dom:mousemove</code>	<code>onmousemove</code>	Occurs when the mouse pointer is moved onto

<b>Value of <code>script:event-name</code> Attribute</b>	<b>Equivalent HTML Event</b>	<b>Description of Event</b>
		a control.
dom:mouseover	onmouseover	Occurs when the mouse pointer is moved over the control.
dom:mouseup	onmouseup	Occurs when a mouse button is released on a control.
dom:mouseout	onmouseout	Occurs when the mouse pointer is moved away from a control.
dom:reset	onreset	Occurs when a form is reset.
dom:submit	onsubmit	Occurs when a form is submitted.

The `script:event-name` attribute is usable with the following elements:

<presentation:event-listener> 10.9.2 and <script:event-listener> 14.5.2.

The `script:event-name` attribute has the data type `string` 18.2.

## 19.434 `script:language`

The attribute `script:language` specifies the name of a script. Script language names are implementation-dependent. The names identifying script languages should begin with a namespace prefix, followed by a ":" (U+003A, COLON) separator. If a namespace prefix is present, the local name of the attribute value is considered to be a name in the XML namespace bound to the namespace prefix.

The `script:language` attribute is usable with the following elements: <office:script> 3.13, <script:event-listener> 14.5.2 and <text:script> 7.7.9.

The `script:language` attribute has the data type `string` 18.2.

## 19.435 `script:macro-name`

The `script:macro-name` attribute specifies the name of a macro. The interpretation of the name is script-language-dependent.

The `script:macro-name` attribute is usable with the following element: <script:event-listener> 14.5.2.

The `script:macro-name` attribute has the data type `string` 18.2.

## 19.436 `smil:accelerate`

See §11.1.2 of [SMIL20].

The defined range for the `smil:accelerate` attribute is 0 to 1, inclusive.

The default value for a `smil:accelerate` attribute is 0, meaning no acceleration.

The `smil:accelerate` attribute is usable with the following elements: <anim:animate> 15.2.2, <anim:animateColor> 15.2.6, <anim:animateMotion> 15.2.5, <anim:animateTransform> 15.2.3, <anim:iterate> 15.4.4, <anim:par> 15.4.2, <anim:seq> 15.4.3, <anim:set> 15.2.4 and <anim:transitionFilter> 15.2.7.

The `smil:accelerate` attribute has values of type `decimal` 18.2 in the range [0,1].

## 19.437 smil:accumulate

See §3.4.3 of [SMIL20].

The `smil:accumulate` attribute is usable with the following elements: `<anim:animate>` 15.2.2, `<anim:animateColor>` 15.2.6, `<anim:animateMotion>` 15.2.5, `<anim:animateTransform>` 15.2.3, `<anim:set>` 15.2.4 and `<anim:transitionFilter>` 15.2.7.

The values of the `smil:accumulate` attribute are `none` or `sum`.

## 19.438 smil:additive

See §3.4.3 of [SMIL20].

The `smil:additive` attribute is usable with the following elements: `<anim:animate>` 15.2.2, `<anim:animateColor>` 15.2.6, `<anim:animateMotion>` 15.2.5, `<anim:animateTransform>` 15.2.3, `<anim:set>` 15.2.4 and `<anim:transitionFilter>` 15.2.7.

The values of the `smil:additive` attribute are `replace` or `sum`.

## 19.439 smil:attributeName

See §3.4.1 of [SMIL20].

The defined values for the `smil:attributeName` attribute are:

- `charColor`: animates the element's text color.
- `charFontName`: animates the element's text font.
- `charHeight`: animates the element's text height.
- `charPosture`: animates the element's text posture.
- `charUnderline`: animates the element's text underline.
- `charWeight`: animates the element's text weight.
- `color`: animates the element's color, this animates both fill, line and text color. Values can be RGB or HSL.
- `fillColor`: animates the element's fill color.
- `fillStyle`: animates the element's fill style.
- `height`: animates the element's height, values are given in screen space where 0 is no height and 1 is the same height as the screen.
- `lineColor`: animates the element's line color.
- `lineStyle`: animates the element's line style.
- `opacity`: animates the element's opacity.
- `rotate`: animates the element's rotation, this animates both the shapes and text animation.
- `skewX`: animates the element's horizontal skew.
- `skewY`: animates the element's vertical skew.
- `visibility`: animates the element's visibility.
- `width`: animates the element's width, values are given in screen space where 0 is no width and 1 is the same width as the screen.

- `x`: animates the element's x position, values are given in screen space where 0 is the left edge and 1 is the right edge.
- `y`: animates the element's y position, values are given in screen space where 0 is the top and 1 is the bottom.

The `smil:attributeName` attribute is usable with the following elements:

`<anim:animate>` 15.2.2, `<anim:animateColor>` 15.2.6, `<anim:animateMotion>` 15.2.5, `<anim:animateTransform>` 15.2.3 and `<anim:set>` 15.2.4.

The `smil:attributeName` attribute has the data type `string` 18.2.

## 19.440 `smil:autoReverse`

See §11.1.2 of [SMIL20].

The default value for this attribute is `false`.

The `smil:autoReverse` attribute is usable with the following elements: `<anim:animate>` 15.2.2, `<anim:animateColor>` 15.2.6, `<anim:animateMotion>` 15.2.5, `<anim:animateTransform>` 15.2.3, `<anim:iterate>` 15.4.4, `<anim:par>` 15.4.2, `<anim:seq>` 15.4.3, `<anim:set>` 15.2.4 and `<anim:transitionFilter>` 15.2.7.

The `smil:autoReverse` attribute has the data type `boolean` 18.3.3.

## 19.441 `smil:begin`

See §10.3.1 of [SMIL20].

The `smil:begin` attribute is usable with the following elements: `<anim:animate>` 15.2.2, `<anim:animateColor>` 15.2.6, `<anim:animateMotion>` 15.2.5, `<anim:animateTransform>` 15.2.3, `<anim:audio>` 15.5, `<anim:command>` 15.6.1, `<anim:iterate>` 15.4.4, `<anim:par>` 15.4.2, `<anim:seq>` 15.4.3, `<anim:set>` 15.2.4 and `<anim:transitionFilter>` 15.2.7.

The `smil:begin` attribute has the data type `string` 18.2.

## 19.442 `smil:by`

See §3.4.4 of [SMIL20].

The `smil:by` attribute is usable with the following elements: `<anim:animate>` 15.2.2, `<anim:animateColor>` 15.2.6, `<anim:animateMotion>` 15.2.5, `<anim:animateTransform>` 15.2.3 and `<anim:transitionFilter>` 15.2.7.

The `smil:by` attribute has the data type `string` 18.2.

## 19.443 `smil:calcMode`

See §3.4.2 and §3.7.1 of [SMIL20].

For `<anim:animate>` 15.2.2 and `<anim:animateColor>` 15.2.6 elements the default value for this attribute is `discrete`.

For a `<anim:animateMotion>` 15.2.5 element the default value for this attribute is `paced`.

The `smil:calcMode` attribute is usable with the following elements: `<anim:animate>` 15.2.2, `<anim:animateColor>` 15.2.6, `<anim:animateMotion>` 15.2.5 and `<anim:transitionFilter>` 15.2.7.

The values of the `smil:calcMode` attribute are `discrete`, `linear`, `paced` or `spline`.

## 19.444 smil:decelerate

See §11.1.2 of [SMIL20].

The defined range for the `smil:decelerate` attribute is 0 to 1, inclusive.

The default value for this attribute is 0.

The `smil:decelerate` attribute is usable with the following elements: `<anim:animate>` 15.2.2, `<anim:animateColor>` 15.2.6, `<anim:animateMotion>` 15.2.5, `<anim:animateTransform>` 15.2.3, `<anim:iterate>` 15.4.4, `<anim:par>` 15.4.2, `<anim:seq>` 15.4.3, `<anim:set>` 15.2.4 and `<anim:transitionFilter>` 15.2.7.

The `smil:decelerate` attribute has values of type `decimal` 18.2 in the range [0,1].

## 19.445 smil:direction

See §12.4.1 of [SMIL20].

For a `<anim:transitionFilter>` 15.2.7 element the default value for this attribute is `forward`.

The `smil:direction` attribute is usable with the following element:  
`<anim:transitionFilter>` 15.2.7.

The values of the `smil:direction` attribute are `forward` or `reverse`.

## 19.446 smil:dur

See §10.3.1 of [SMIL20].

The `smil:dur` attribute is usable with the following elements: `<anim:animate>` 15.2.2, `<anim:animateColor>` 15.2.6, `<anim:animateMotion>` 15.2.5, `<anim:animateTransform>` 15.2.3, `<anim:audio>` 15.5, `<anim:iterate>` 15.4.4, `<anim:par>` 15.4.2, `<anim:seq>` 15.4.3, `<anim:set>` 15.2.4 and `<anim:transitionFilter>` 15.2.7.

The `smil:dur` attribute has the data type `string` 18.2.

## 19.447 smil:end

See §10.3.1 of [SMIL20].

The `smil:end` attribute is usable with the following elements: `<anim:animate>` 15.2.2, `<anim:animateColor>` 15.2.6, `<anim:animateMotion>` 15.2.5, `<anim:animateTransform>` 15.2.3, `<anim:audio>` 15.5, `<anim:command>` 15.6.1, `<anim:iterate>` 15.4.4, `<anim:par>` 15.4.2, `<anim:seq>` 15.4.3, `<anim:set>` 15.2.4 and `<anim:transitionFilter>` 15.2.7.

The `smil:end` attribute has the data type `string` 18.2.

## 19.448 smil:endsync

See §10.3.1 of [SMIL20].

The `smil:endsync` attribute is usable with the following elements: `<anim:iterate>` 15.4.4, `<anim:par>` 15.4.2 and `<anim:seq>` 15.4.3.

The values of the `smil:endsync` attribute are `first`, `last`, `all`, `media` or a value of type `IDREF` 18.2.

## 19.449 smil:fadeColor

See §12.5.1 of [SMIL20].

The default value for this attribute is #000000.

The `smil:fadeColor` attribute is usable with the following element:  
`<anim:transitionFilter>` 15.2.7.

The `smil:fadeColor` attribute has the data type `color` 18.3.9.

## 19.450 smil:fill

See §10.3.1 of [SMIL20].

The `smil:fill` attribute is usable with the following elements: `<anim:animate>` 15.2.2, `<anim:animateColor>` 15.2.6, `<anim:animateMotion>` 15.2.5, `<anim:animateTransform>` 15.2.3, `<anim:audio>` 15.5, `<anim:iterate>` 15.4.4, `<anim:par>` 15.4.2, `<anim:seq>` 15.4.3, `<anim:set>` 15.2.4 and `<anim:transitionFilter>` 15.2.7.

The values of the `smil:fill` attribute are `remove`, `freeze`, `hold`, `auto`, `default` or `transition`.

## 19.451 smil:fillDefault

See §10.3.1 of [SMIL20].

The `smil:fillDefault` attribute is usable with the following elements: `<anim:animate>` 15.2.2, `<anim:animateColor>` 15.2.6, `<anim:animateMotion>` 15.2.5, `<anim:animateTransform>` 15.2.3, `<anim:audio>` 15.5, `<anim:iterate>` 15.4.4, `<anim:par>` 15.4.2, `<anim:seq>` 15.4.3, `<anim:set>` 15.2.4 and `<anim:transitionFilter>` 15.2.7.

The values of the `smil:fillDefault` attribute are `remove`, `freeze`, `hold`, `transition`, `auto` or `inherit`.

## 19.452 smil:from

See §3.4.4 of [SMIL20].

The `smil:from` attribute is usable with the following elements: `<anim:animate>` 15.2.2, `<anim:animateColor>` 15.2.6, `<anim:animateMotion>` 15.2.5, `<anim:animateTransform>` 15.2.3 and `<anim:transitionFilter>` 15.2.7.

The `smil:from` attribute has the data type `string` 18.2.

## 19.453 smil:keySplines

See §3.7.1 of [SMIL20].

The `smil:keySplines` attribute is usable with the following elements: `<anim:animate>` 15.2.2, `<anim:animateColor>` 15.2.6 and `<anim:animateMotion>` 15.2.5.

The `smil:keySplines` attribute has the data type `string` 18.2.

## 19.454 smil:keyTimes

See §3.7.1 of [SMIL20].

The `smil:keyTimes` attribute is usable with the following elements: `<anim:animate>` 15.2.2, `<anim:animateColor>` 15.2.6 and `<anim:animateMotion>` 15.2.5.

The `smil:keyTimes` attribute has the data type `string` 18.2.

## 19.455 `smil:mode`

See §12.5.1 of [SMIL20].

The default value for this attribute is `in`.

The `smil:mode` attribute is usable with the following element:  
`<anim:transitionFilter>` 15.2.7.

The values of the `smil:mode` attribute are `in` or `out`.

## 19.456 `smil:repeatCount`

See §10.3.1 of [SMIL20].

The `smil:repeatCount` attribute is usable with the following elements: `<anim:animate>` 15.2.2, `<anim:animateColor>` 15.2.6, `<anim:animateMotion>` 15.2.5, `<anim:animateTransform>` 15.2.3, `<anim:audio>` 15.5, `<anim:iterate>` 15.4.4, `<anim:par>` 15.4.2, `<anim:seq>` 15.4.3, `<anim:set>` 15.2.4 and `<anim:transitionFilter>` 15.2.7.

The values of the `smil:repeatCount` attribute are a non negative value of type `decimal` 18.2 or `indefinite`.

## 19.457 `smil:repeatDur`

See §10.3.1 of [SMIL20].

The value of the `smil:repeatDur` attribute can be `indefinite`, or a `clock-value` as defined in §10.3.1 of SMIL

The `smil:repeatDur` attribute is usable with the following elements: `<anim:animate>` 15.2.2, `<anim:animateColor>` 15.2.6, `<anim:animateMotion>` 15.2.5, `<anim:animateTransform>` 15.2.3, `<anim:audio>` 15.5, `<anim:iterate>` 15.4.4, `<anim:par>` 15.4.2, `<anim:seq>` 15.4.3, `<anim:set>` 15.2.4 and `<anim:transitionFilter>` 15.2.7.

The `smil:repeatDur` attribute has the data type `string` 18.2.

## 19.458 `smil:restart`

See §10.3.1 of [SMIL20].

The default value for this attribute is `default`.

The `smil:restart` attribute is usable with the following elements: `<anim:animate>` 15.2.2, `<anim:animateColor>` 15.2.6, `<anim:animateMotion>` 15.2.5, `<anim:animateTransform>` 15.2.3, `<anim:audio>` 15.5, `<anim:iterate>` 15.4.4, `<anim:par>` 15.4.2, `<anim:seq>` 15.4.3, `<anim:set>` 15.2.4 and `<anim:transitionFilter>` 15.2.7.

The values of the `smil:restart` attribute are `never`, `always`, `whenNotActive` or `default`.

## 19.459 `smil:restartDefault`

See §10.3.1 of [SMIL20].

The default value for this attribute is `inherit`.

The `smil:restartDefault` attribute is usable with the following elements:  
`<anim:animate>` 15.2.2, `<anim:animateColor>` 15.2.6, `<anim:animateMotion>` 15.2.5, `<anim:animateTransform>` 15.2.3, `<anim:audio>` 15.5, `<anim:iterate>` 15.4.4, `<anim:par>` 15.4.2, `<anim:seq>` 15.4.3, `<anim:set>` 15.2.4 and `<anim:transitionFilter>` 15.2.7.

The values of the `smil:restartDefault` attribute are `never`, `always`, `whenNotActive` or `inherit`.

## 19.460 smil:subtype

See §12.4.1 of [SMIL20].

**Note:** See §12.8 of [SMIL20] for a list of supported subtypes.

The `smil:subtype` attribute is usable with the following element:  
`<anim:transitionFilter>` 15.2.7.

The `smil:subtype` attribute has the data type `string` 18.2.

## 19.461 smil:targetElement

See §3.4.1 of [SMIL20].

Drawing shapes are specified by their `xml:id` or `draw:id` attribute values and paragraphs by their `xml:id` or `text:id` attribute values.

If the `anim:sub-item` attribute of a `<anim:iterate>` element has the value `whole`, the iteration includes the drawing shape's background and its text. If the `anim:sub-item` 19.11 attribute value is `text`, only the shape's text is iterated.

The `smil:targetElement` attribute is usable with the following elements:  
`<anim:animate>` 15.2.2, `<anim:animateColor>` 15.2.6, `<anim:animateMotion>` 15.2.5, `<anim:animateTransform>` 15.2.3, `<anim:command>` 15.6.1, `<anim:iterate>` 15.4.4, `<anim:set>` 15.2.4 and `<anim:transitionFilter>` 15.2.7.

The `smil:targetElement` attribute has the data type `IDREF` 18.2.

## 19.462 smil:to

See §3.4.4 of [SMIL20].

The `smil:to` attribute is usable with the following elements: `<anim:animate>` 15.2.2, `<anim:animateColor>` 15.2.6, `<anim:animateMotion>` 15.2.5, `<anim:animateTransform>` 15.2.3, `<anim:set>` 15.2.4 and `<anim:transitionFilter>` 15.2.7.

The `smil:to` attribute has the data type `string` 18.2.

## 19.463 smil:type

The [SMIL20] `smil:type` attribute specifies a transition type or family.

See §12.8 of [SMIL20] for a list of supported types.

If this attribute is present, the attributes `presentation:transition-type` and `presentation:transition-style` attributes should be ignored.

The `smil:type` attribute is usable with the following element:  
`<anim:transitionFilter>` 15.2.7.

The `smil:type` attribute has the data type `string` 18.2.

## 19.464 smil:values

See §3.4.2 of [SMIL20].

The `smil:values` attribute is usable with the following elements: `<anim:animate>` 15.2.2, `<anim:animateColor>` 15.2.6, `<anim:animateMotion>` 15.2.5, `<anim:animateTransform>` 15.2.3 and `<anim:transitionFilter>` 15.2.7.

The `smil:values` attribute has the data type `string` 18.2.

## 19.465 style:adjustment

The `style:adjustment` attribute specifies how a footnote separator line is aligned on a page.

The defined values for the `style:adjustment` attribute are:

- `center`: footnote separator line centered on page.
- `left`: footnote separator line aligned to left on page.
- `right`: footnote separator line aligned to right on page.

The default value for this attribute is `left`.

The `style:adjustment` attribute is usable with the following element:

`<style:footnote-sep>` 17.4.

The values of the `style:adjustment` attribute are `left`, `center` or `right`.

## 19.466 style:apply-style-name

The `style:apply-style-name` attribute specifies the style to apply when the condition specified by the `style:condition` 19.472 attribute is `true`. If the referenced style is `undefined` or is an automatic style, an error occurs.

The referenced style shall be defined by the same type of element as the conditional style that contains the condition. If the referenced style element has a `style:family` 19.472 attribute, its value further shall be equal the value of the `style:family` attribute of the conditional style.

The `style:apply-style-name` attribute is usable with the following element:

`<style:map>` 16.3.

The `style:apply-style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.467 style:auto-update

The `style:auto-update` attribute specifies whether styles are automatically updated when the formatting properties of an object that has the style assigned to it are changed.

The defined values for the `style:auto-update` attribute are:

- `false`: a change to a formatting property is applied for the object where the change was made. If necessary, a new automatic style will be created which is applied to the object where the change was made.
- `true`: a change to a formatting property results in the updating of the common style that is applied to an object. The formatting change is applied to all objects subject to the common style where the change was made.

The default value for this attribute is `false`.

The `style:auto-update` attribute is usable with the following element: `<style:style>` 16.2.

The `style:auto-update` attribute has the data type `boolean` 18.3.3.

## 19.468 `style:base-cell-address`

The `style:base-cell-address` attribute specifies as an absolute cell address with a table name the base cell for relative addresses in formulas. This attribute applies only to cell styles where the condition contains a formula.

The `style:base-cell-address` attribute is usable with the following element:

`<style:map> 16.3.`

The `style:base-cell-address` attribute has the data type `cellAddress` 18.3.4.

## 19.469 `style:char`

The `style:char` attribute specifies the delimiter character for tab stops of type `char`.

The `style:char` attribute is usable with the following element: `<style:tab-stop> 17.8.`

The `style:char` attribute has the data type `character` 18.3.7.

## 19.470 `style:class`

The `style:class` attribute specifies a style class name.

A style may belong to an arbitrary class of styles. The style class name is an arbitrary string. The style class name has no meaning within the file format itself, but it can for instance be evaluated by user interfaces to show a list of styles where the styles are grouped by its name.

The `style:class` attribute is usable with the following element: `<style:style> 16.2.`

The `style:class` attribute has the data type `string` 18.2.

## 19.471 `style:color`

The `style:color` attribute specifies the color of a column or footnote separator line.

For a `<style:column-sep> 17.14` element the default value for this attribute is `#000000`.

The `style:color` attribute is usable with the following elements: `<style:column-sep> 17.14` and `<style:footnote-sep> 17.4`.

The `style:color` attribute has the data type `color` 18.3.9.

## 19.472 `style:condition`

The `style:condition` attribute specifies a condition that triggers a style map.

The value of this attribute should be a namespace prefix, followed by a ":" (U+003A, COLON), followed by one of the defined conditions. If the namespace prefix is missing it defaults to the "urn:oasis:names:tc:opendocument:xmlns:of:1.2" namespace. The XML namespace that applies to the `style:condition` attribute specifies the syntax and semantics of any expression occurrences in the `style:condition` syntax.

If a consumer does not recognize a condition, it shall ignore the `<style:map>` element containing the condition.

The defined conditions that may be used by paragraph styles are:

- `footnote()` and `endnote()`: true if the style is applied to a paragraph or heading that is contained in a footnote or endnote.
- `header()` and `footer()`: true if the style is applied to a paragraph or heading that is contained in a page header or footer.

- `list-level ()=n`, where `n` is a number: true if the style is applied to a paragraph or heading that is contained in a list on level `n`, or to a numbered paragraph on level `n`.
- `outline-level ()=n`, where `n` is a number: true if the style is applied to a heading on level `n`.
- `section ()`: true if the style is applied to a paragraph or heading that is contained in a text section.
- `table ()` and `table-header ()`: true if the style is applied to a paragraph or heading that is contained in a table cell or table header cell.

Within paragraph style conditions `n` is a positive integer in the lexical form of the [xmlschema-2] positiveInteger datatype.

The defined conditions that may be used by table cell styles are:

- `cell-content () op value`: where `op` is one of the relational operators "`<`", "`>`", "`<=`", "`>=`", "`=`" or "`!=`": true if the style is applied to a table cell where the cell's value compared to the value specified in the condition by `value` using the relational operator `op` evaluates to true.
- `cell-content-is-between (value1, value2)`: true if the style is applied to a table cell whose value is between `value1` and `value2`.
- `cell-content-is-not-between (value, value)`: true if the style is applied to a table cell whose value is not between `value1` and `value2`.
- `is-true-formula (expression)`: true if the style is applied to a table cell for which formula evaluates true.

Within table cell style conditions:

- `value`, `value1` and `value2` are a `numberValue`, a `string` or an `expression`.
- `numberValue` is a whole or decimal number in the lexical form of the [xmlschema-2] decimal datatype.
- `string` expresses a value of the [xmlschema-2] string datatype by surrounding the sequence of string characters in quotation marks (U+0022, QUOTATION MARK). A quotation mark within the string itself is expressed with two consecutive quotation marks.
- `expression` is a lexical form that is neither `numberValue` nor `string` and that is a well-formed expression determined by the namespace applicable to the `style:condition` attribute value.

The defined condition that may be used by data styles is:

- `value () op n`, where, `op` is one of the relational operators "`<`", "`>`", "`<=`", "`>=`", "`=`" or "`!=`", and `n` is a number for non-Boolean data styles and `true` or `false` for Boolean data styles: true if the style is applied to a table cell where the cell's value compared to the value specified in the condition by `n` using the relational operator `op` evaluates to true.

Relative cell range addresses that occur in a condition for a table cell style define an offset which shall be interpreted relative to the cell for which a condition is calculated. The offset is the same as the offsets between the table cell addressed by the relative cell range address and the table cell specified by the `style:base-cell-address` attribute.

An OpenDocument Consumer when hosting a formula evaluator makes available the host-dependent properties defined by Table 17 Host-dependent Properties in 19.646.

The `style:condition` attribute is usable with the following element: `<style:map>` 16.3.

The `style:condition` attribute has the data type `string` 18.2.

## 19.473 style:data-style-name

### 19.473.1 General

The `style:data-style-name` attribute specifies the name of a data style.

### 19.473.2 <presentation:date-time-decl>

The `style:data-style-name` attribute specifies a data style to format a date and time.

The `style:data-style-name` attribute is usable with the following element:  
`<presentation:date-time-decl>` 10.9.4.6.

The `style:data-style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.473.3 <style:style>

The `style:data-style-name` attribute specifies a data style for a `<style:style>` element that defines the styles for table cell styles and chart styles.

The `style:data-style-name` attribute is usable with the following element:  
`<style:style>` 16.2.

The `style:data-style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.473.4 <text:creation-date>

The `style:data-style-name` attribute specifies a data style to format a numeric value.

The `style:data-style-name` attribute is usable with the following element:  
`<text:creation-date>` 7.5.3.

The `style:data-style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.473.5 <text:creation-time>

The `style:data-style-name` attribute specifies a data style to format a numeric value.

The `style:data-style-name` attribute is usable with the following element:  
`<text:creation-time>` 7.5.4.

The `style:data-style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.473.6 <text:database-display>

The `style:data-style-name` attribute specifies a data style to format a numeric value.

The `style:data-style-name` attribute is usable with the following element:  
`<text:database-display>` 7.6.3.

The `style:data-style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.473.7 <text:date>

The `style:data-style-name` attribute specifies a data style to format a numeric value.

The `style:data-style-name` attribute is usable with the following element:  
`<text:date>` 7.3.2.

The `style:data-style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.473.8 <text:editing-duration>

The `style:data-style-name` attribute specifies a data style to format a numeric value.

The `style:data-style-name` attribute is usable with the following element:  
`<text:editing-duration>` 7.5.14.

The `style:data-style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.473.9 <text:expression>

The `style:data-style-name` attribute specifies a data style to format a field value. This attribute shall be omitted for fields whose value type is `string`. It shall be present for fields whose value type is not `string`.

The `style:data-style-name` attribute is usable with the following element:  
`<text:expression>` 7.4.14.

The `style:data-style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.473.10 <text:meta-field>

The `style:data-style-name` attribute specifies a data style to format a numeric value.

The `style:data-style-name` attribute is usable with the following element:  
`<text:meta-field>` 7.5.19.

The `style:data-style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.473.11 <text:modification-date>

The `style:data-style-name` attribute specifies a data style to format a numeric value.

The `style:data-style-name` attribute is usable with the following element:  
`<text:modification-date>` 7.5.16.

The `style:data-style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.473.12 <text:modification-time>

The `style:data-style-name` attribute specifies a data style to format a numeric value.

The `style:data-style-name` attribute is usable with the following element:  
`<text:modification-time>` 7.5.15.

The `style:data-style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.473.13 <text:print-date>

The `style:data-style-name` attribute specifies a data style to format a numeric value.

The `style:data-style-name` attribute is usable with the following element:  
`<text:print-date>` 7.5.8.

The `style:data-style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.473.14 <text:print-time>

The `style:data-style-name` attribute specifies a data style to format a numeric value.

The `style:data-style-name` attribute is usable with the following element:  
`<text:print-time>` 7.5.7.

The `style:data-style-name` attribute has the data type [styleNameRef 18.3.33](#).

### 19.473.15 `<text:table-formula>`

The `style:data-style-name` attribute specifies a data style to format a numeric value.

The `style:data-style-name` attribute is usable with the following element:  
`<text:table-formula>` 7.7.14.

The `style:data-style-name` attribute has the data type [styleNameRef 18.3.33](#).

### 19.473.16 `<text:time>`

The `style:data-style-name` attribute specifies a data style to format a numeric value.

The `style:data-style-name` attribute is usable with the following element:  
`<text:time>` 7.3.3.

The `style:data-style-name` attribute has the data type [styleNameRef 18.3.33](#).

### 19.473.17 `<text:user-defined>`

The `style:data-style-name` attribute specifies a data style to format a field value. This attribute shall be omitted for fields whose value type is `string`. It shall be present for fields whose value type is not `string`.

The `style:data-style-name` attribute is usable with the following element:  
`<text:user-defined>` 7.5.6.

The `style:data-style-name` attribute has the data type [styleNameRef 18.3.33](#).

### 19.473.18 `<text:user-field-get>`

The `style:data-style-name` attribute specifies a data style to format a field value. This attribute shall be omitted for fields whose value type is `string`. It shall be present for fields whose value type is not `string`.

The `style:data-style-name` attribute is usable with the following element:  
`<text:user-field-get>` 7.4.9.

The `style:data-style-name` attribute has the data type [styleNameRef 18.3.33](#).

### 19.473.19 `<text:user-field-input>`

The `style:data-style-name` attribute specifies a data style to format a field value. This attribute shall be omitted for fields whose value type is `string`. It shall be present for fields whose value type is not `string`.

The `style:data-style-name` attribute is usable with the following element:  
`<text:user-field-input>` 7.4.10.

The `style:data-style-name` attribute has the data type [styleNameRef 18.3.33](#).

### 19.473.20 `<text:variable-get>`

The `style:data-style-name` attribute specifies a data style to format a field value. This attribute shall be omitted for fields whose value type is `string`. It shall be present for fields whose value type is not `string`.

The `style:data-style-name` attribute is usable with the following element:  
`<text:variable-get>` 7.4.5.

The `style:data-style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.473.21 `<text:variable-input>`

The `style:data-style-name` attribute specifies a data style to format a field value. This attribute shall be omitted for fields whose value type is `string`. It shall be present for fields whose value type is not `string`.

The `style:data-style-name` attribute is usable with the following element:

`<text:variable-input>` 7.4.6.

The `style:data-style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.473.22 `<text:variable-set>`

The `style:data-style-name` attribute specifies a data style to format a field value. This attribute shall be omitted for fields whose value type is `string`. It shall be present for fields whose value type is not `string`.

The `style:data-style-name` attribute is usable with the following element:

`<text:variable-set>` 7.4.4.

The `style:data-style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.474 `style:default-outline-level`

The `style:default-outline-level` attribute specifies a default outline level for a style with the `style:family` 19.480 attribute value `paragraph`.

If the `style:default-outline-level` attribute is present in a paragraph style, and if this paragraph style is assigned to a paragraph or heading by user action, then the consumer should replace the paragraph or heading with a heading of the specified level, which has the same content and attributes as the original paragraph or heading.

**Note:** This attribute does not modify the behavior of `<text:p>` 5.1.3 or `<text:h>` 5.1.2 elements, but only instructs a consumer to create one or the other when assigning a paragraph style as a result of user interface action while the document is edited.

The `style:default-outline-level` attribute value can be empty. If empty, this attribute does not inherit a list style value from a parent style.

The `style:default-outline-level` attribute is usable with the following element:  
`<style:style>` 16.2.

The values of the `style:default-outline-level` attribute are a value of type `positiveInteger` 18.2 or an empty string.

## 19.475 `style:display`

The `style:display` attribute specifies whether the header or footer is displayed or not.

The defined values for the `style:display` attribute are:

- `false`: header or footer is not displayed.
- `true`: header or footer is displayed.

The default value for this attribute is `true`.

The `style:display` attribute is usable with the following elements: `<style:footer>` 16.12, `<style:footer-first>` 16.13, `<style:footer-left>` 16.15,

<style:header> 16.10, <style:header-first> 16.11 and <style:header-left> 16.14.

The `style:display` attribute has the data type [boolean](#) 18.3.3.

## 19.476 style:display-name

The `style:display-name` attribute specifies the name of a style as it should appear in the user interface. If this attribute is not present, the display name should be the same as the style name.

The `style:display-name` attribute is usable with the following elements:

<number:boolean-style> 16.29.24, <number:currency-style> 16.29.8, <number:date-style> 16.29.11, <number:number-style> 16.29.2, <number:percentage-style> 16.29.10, <number:text-style> 16.29.26, <number:time-style> 16.29.19, <style:master-page> 16.9, <style:presentation-page-layout> 16.43, <style:style> 16.2 and <text:list-style> 16.32.

The `style:display-name` attribute has the data type `string` 18.2.

## 19.477 style:distance

The `style:distance` attribute specifies the distance between the last of the characters using the larger font and the first of the remaining characters of each line.

**Note:** If the leading characters for a paragraph span the line size of a portion of the remaining text of a paragraph, the `style:distance` attribute specifies the distance between the end of the text written in the larger font and the smaller text of any lines that it spans.

The default value for this attribute is `0cm`.

The `style:distance` attribute is usable with the following element: <style:drop-cap> 17.9.

The `style:distance` attribute has the data type `length` 18.3.19.

## 19.478 style:distance-before-sep

The `style:distance-before-sep` attribute specifies the space between the body-text area and a footnote separator line.

The `style:distance-before-sep` attribute is usable with the following element: <style:footnote-sep> 17.4.

The `style:distance-before-sep` attribute has the data type `length` 18.3.19.

## 19.479 style:distance-after-sep

The `style:distance-after-sep` attribute specifies the space between a footnote separator line and the footnote text.

The `style:distance-after-sep` attribute is usable with the following element: <style:footnote-sep> 17.4.

The `style:distance-after-sep` attribute has the data type `length` 18.3.19.

## 19.480 style:family

The `style:family` attribute specifies the family of a style.

The defined values for the `style:family` attribute are:

- `chart`: family name of styles for charts.
- `drawing-page`: family name of styles for drawing pages.
- `graphic`: family name of styles for graphic elements.
- `paragraph`: family name of styles for paragraphs.
- `presentation`: family name of styles for presentations.
- `ruby`: family name of styles for ruby text.
- `section`: family name of styles for sections.
- `table`: family name of styles for tables.
- `table-cell`: family name of styles for table cells.
- `table-column`: family name of styles for table columns.
- `table-row`: family name of styles for table rows.
- `text`: family name of styles for text.

The `style:family` attribute is usable with the following elements: `<style:default-style>` 16.4 and `<style:style>` 16.2.

## 19.481 style:filter-name

The `style:filter-name` attribute specifies the implementation-dependent filter name that has been used to load an image into the document.

This attribute is attached to a `<style:background-image>` element.

The `style:filter-name` attribute is usable with the following element: `<style:background-image>` 17.3.

The `style:filter-name` attribute has the data type `string` 18.2.

## 19.482 style:font-adornments

The `style:font-adornments` attribute specifies a characteristic of a font.

**Note:** A font characteristic includes bold or italic, which when used with a font family name, specifies a font.

The `style:font-adornments` attribute is usable with the following element: `<style:font-face>` 16.23.

The `style:font-adornments` attribute has the data type `string` 18.2.

## 19.483 style:font-charset

The `style:font-charset` attribute specifies whether a font defines glyphs according to the semantics of [UNICODE] or not.

The value of this attribute can be `x-symbol` or a character encoding in the notation described in the §4.3.3 of [XML1.0]. If the value is `x-symbol`, the font does not define glyphs according to the semantics of [UNICODE]. If the value is one of the encodings or transformations of [UNICODE], the font does define glyphs according to the semantics of [UNICODE]. The use of other values is deprecated.

**Note:** Fonts for which the attribute has the value `x-symbol` may define glyphs for code points in the private use area of [UNICODE].

The `style:font-charset` attribute is usable with the following element: `<style:font-face>` 16.23.

The `style:font-charset` attribute has the data type `textEncoding` 18.3.36.

## 19.484 `style:font-family-generic`

The `style:font-family-generic` attribute specifies a generic font family name.

The defined values for the `style:font-family-generic` attribute are:

- `decorative`: the family of decorative fonts.
- `modern`: the family of modern fonts.
- `roman`: the family roman fonts (with serifs).
- `script`: the family of script fonts.
- `swiss`: the family roman fonts (without serifs).
- `system`: the family system fonts.

The `style:font-family-generic` attribute is usable with the following element: `<style:font-face>` 16.23.

The values of the `style:font-family-generic` attribute are `roman`, `swiss`, `modern`, `decorative`, `script` or `system`.

## 19.485 `style:font-pitch`

The `style:font-pitch` attribute specifies whether a font has a fixed or variable width.

The defined values for the `style:font-pitch` attribute are:

- `fixed`: font has a fixed width.
- `variable`: font has a variable width.

The `style:font-pitch` attribute is usable with the following element: `<style:font-face>` 16.23.

The values of the `style:font-pitch` attribute are `fixed` or `variable`.

## 19.486 `style:height`

The `style:height` attribute specifies the height of a column separator line. The value of this attribute is a percentage of the height of the columned area.

The defined value range for the `style:height` attribute is 0% to 100%, inclusive.

The default value for this attribute is 100%.

The `style:height` attribute is usable with the following element: `<style:column-sep>` 17.14.

The `style:height` attribute has the data type `zeroToHundredPercent` 18.3.42.

## 19.487 `style:leader-char`

The `style:leader-char` attribute specifies a leader character.

The `style:leader-char` attribute is usable with the following element: `<text:index-entry-tab-stop>` 8.13.6.

The `style:leader-char` attribute has the data type `character` 18.3.7.

## 19.488 style:leader-color

The `style:leader-color` attribute specifies the color of a leader line. The value of this attribute is either `font-color` or a color. If the value is `font-color`, the current text color is used for the leader line.

The `style:leader-color` attribute is usable with the following element: `<style:tab-stop> 17.8`.

The values of the `style:leader-color` attribute are `font-color` or a value of type `color` 18.3.9.

## 19.489 style:leader-style

The `style:leader-style` attribute specifies a style for a leader line.

The defined values for the `style:leader-style` attribute are:

- `none`: tab stop has no leader line.
- `dash`: tab stop has a dashed leader line.
- `dot-dash`: tab stop has a leader line whose repeating pattern is a dot followed by a dash.
- `dot-dot-dash`: tab stop has a leader line whose repeating pattern has two dots followed by a dash.
- `dotted`: tab stop has a dotted leader line.
- `long-dash`: tab stop has a dashed leader line whose dashes are longer than the ones from the dashed line for value `dash`.
- `solid`: tab stop has a solid leader line.
- `wave`: tab stop has a wavy leader line.

**Note:** The definitions of the values of the `style:leader-style` attribute are based on the text decoration style 'text-underline-style' from [CSS3Text], §9.2.

The `style:leader-style` attribute is usable with the following element: `<style:tab-stop> 17.8`.

The values of the `style:leader-style` attribute are `none`, `solid`, `dotted`, `dash`, `long-dash`, `dot-dash`, `dot-dot-dash` or `wave`.

## 19.490 style:leader-text

The `style:leader-text` attribute specifies a single Unicode character for use as leader text for tab stops.

An consumer may support only specific characters as textual leaders. If a character that is not supported by a consumer is specified by this attribute, the consumer should display a leader character that it supports instead of the one specified by this attribute.

If both `style:leader-text` and `style:leader-style` 19.480 attributes are specified, the value of the `style:leader-text` sets the leader text for tab stops.

The default value for this attribute is “ ” (U+0020, SPACE).

The `style:leader-text` attribute is usable with the following element: `<style:tab-stop> 17.8`.

The `style:leader-text` attribute has the data type `character` 18.3.7.

## 19.491 style:leader-text-style

The `style:leader-text-style` specifies a text style that is applied to a textual leader. It is not applied to leader lines. If the attribute appears in an automatic style, it may reference either an automatic text style or a common style. If the attribute appears in a common style, it may reference a common style only.

The `style:leader-text-style` attribute is usable with the following element:  
`<style:tab-stop> 17.8.`

The `style:leader-text-style` attribute has the data type `styleNameRef` 18.3.33.

## 19.492 style:leader-type

The `style:leader-type` attribute specifies whether a leader line should be drawn, and if so, whether a single or double line will be used.

The defined values for the `style:leader-type` attribute are:

- `double`: a double line is drawn.
- `none`: no line is drawn.
- `single`: a single line is drawn.

The `style:leader-type` attribute is usable with the following element: `<style:tab-stop> 17.8.`

The values of the `style:leader-type` attribute are `none`, `single` or `double`.

## 19.493 style:leader-width

The `style:leader-width` attribute specifies the width (i.e., thickness) of a leader line.

The defined values for the `style:leader-width` attribute are:

- `auto`: the width of a leader line should be calculated from the font size of the text where the leader line will appear.
- `bold`: the width of a leader line should be calculated from the font size of the text where the leader line will appear but is wider than for the value of `auto`.
- a value of type `percent` 18.3.24
- a value of type `positiveInteger` 18.2
- a value of type `positiveLength` 18.3.27

The line widths referenced by the values `medium`, `normal`, `thick` and `thin` are implementation-defined.

The `style:leader-width` attribute is usable with the following element: `<style:tab-stop> 17.8.`

The values of the `style:leader-width` attribute are `auto`, `normal`, `bold`, `thin`, `medium`, `thick`, a value of type `positiveInteger` 18.2, a value of type `percent` 18.3.24 or a value of type `positiveLength` 18.3.27.

## 19.494 style:legend-expansion

The `style:legend-expansion` attribute specifies the direction in which a legend expands.

The defined values for the `style:legend-expansion` attribute are:

- `balanced`: legend expands horizontally and vertically.

- `custom`: with a `style:legend-expansion-aspect-ratio` 19.495 attribute legend expands according to the specified ratio.
- `high`: legend expands vertically.
- `wide`: legend expands horizontally.

The `style:legend-expansion` attribute is usable with the following element:  
`<chart:legend>` 11.4.

## 19.495 style:legend-expansion-aspect-ratio

The `style:legend-expansion-aspect-ratio` attribute specifies the ratio between width and height for a `style:legend` 19.494 attribute with value `custom`.

The `style:legend-expansion-aspect-ratio` attribute is usable with the following element: `<chart:legend>` 11.4.

The `style:legend-expansion-aspect-ratio` attribute has the data type `double` 18.2.

## 19.496 style:length

The `style:length` attribute specifies the number of characters that are displayed in a larger font.

The defined values for the `style:length` attribute are:

- `word`: specifies a set of characters to be displayed in a larger font.
- a value of type `positiveInteger` 18.2

**Note:** What will be treated as a “word” by a consumer is locale-specific.

The default value for this attribute is 1.

The `style:length` attribute is usable with the following element: `<style:drop-cap>` 17.9.

The values of the `style:length` attribute are `word` or a value of type `positiveInteger` 18.2.

## 19.497 style:line-style

The `style:line-style` attribute specifies the style of a footnote separator line.

The defined value for the `style:line-style` attribute is `none`: no footnote separator line is drawn.

The footnote separator components referenced by the values `dash`, `dot-dash`, `dot-dot-dash`, `dotted`, `long-dash`, `solid` and `wave`, are implementation-defined.

The `style:line-style` attribute is usable with the following element:  
`<style:footnote-sep>` 17.4.

The values of the `style:line-style` attribute are `none`, `solid`, `dotted`, `dash`, `long-dash`, `dot-dash`, `dot-dot-dash` or `wave`.

## 19.498 style:lines

The `style:lines` attribute specifies the number of text lines which a font spans. If the value of this attribute is 1, `<style:drop-cap>` is disabled.

The default value for this attribute is 1.

The `style:lines` attribute is usable with the following element: `<style:drop-cap>` 17.9.

The `style:lines` attribute has the data type `positiveInteger` 18.2.

## 19.499 style:list-level

The `style:list-level` attribute specifies the list level value of a list style that may be applied to any paragraph style. It does not directly specify the paragraph's list level value, but consumers can change the paragraph's list level value to the specified value when the paragraph style is applied. It is valid only if the `style:list-style-name` 19.480 attribute is specified and is not empty, and if the `style:family` 19.480 attribute has the value `paragraph`.

The `style:list-level` attribute is usable with the following element: `<style:style>` 16.2.

The values of the `style:list-level` attribute are a value of type `positiveInteger` 18.2 or an empty string.

## 19.500 style:list-style-name

The `style:list-style-name` attribute specifies a list style for style families with paragraph formatting properties. This applies to automatic and common styles.

The list style specified by this attribute is applied to headings and paragraphs that are contained in a list, where the list does not specify a list style itself, and the list has no list style specification for any of its parents.

The `style:list-style-name` attribute value can be empty. If empty, this attribute does not inherit a list style value from a parent style.

The `style:list-style-name` attribute is usable with the following element: `<style:style>` 16.2.

The values of the `style:list-style-name` attribute are a value of type `styleName` 18.3.32 or an empty string.

## 19.501 style:master-page-name

The `style:master-page-name` attribute defines a master page for a paragraph or table style. This applies to automatic and common styles. `<style:master-page>` 16.9.

If this attribute is associated with a style, a page break is inserted when the style is applied and the specified master page is applied to the resulting page.

This attribute is ignored if it is associated with a paragraph style that is applied to a paragraph within a table.

The `style:master-page-name` attribute is usable with the following element: `<style:style>` 16.2.

The `style:master-page-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.502 style:name

### 19.502.1 General

The `style:name` attribute specifies names that reference style mechanisms.

## 19.502.2 <number:boolean-style>, <number:currency-style>, <number:date-style>, <number:num-list-format>, <number:number-style>, <number:percentage-style>, <number:text-style>, <number:time-style>, <style:master-page>, <style:page-layout>, <style:presentation-page-layout>, <style:style>, <text:list-style>, <text:outline-style>

The `style:name` attribute specifies the name of a:

- data style - It can be used with all data style elements.
- displayed list number string format
- list style
- master page - Each master page is referenced using the page name. The name specified shall be unique to the document instance.
- outline style - The outline style as a list style can be referenced by the `style:list-style-name` 19.500 attribute.
- page layout
- style – With the `style:family` 19.480 attribute, the `style:name` attribute uniquely identifies a style. The `<office:styles>` 3.15.2, `<office:automatic-styles>` 3.15.3 and `<office:master-styles>` 3.15.4 elements each shall not contain two styles with the same family and the same name.

For automatic styles, the name may be generated by OpenDocument producers. For each style family or style element, producers should generate distinct sets of names for automatic styles stored in the `content.xml` 3.1.3.2 and `styles.xml` 3.1.3.3 files. The names should also be distinct from the names used in the `<office:styles>` element inside the `styles.xml` files.

**Note:** If the document is produced multiple times, it cannot be assumed that the same name is generated each time.

The `style:name` attribute is usable with the following elements: `<number:boolean-style>` 16.29.24, `<number:currency-style>` 16.29.8, `<number:date-style>` 16.29.11, `<number:num-list-format>` 16.29.30, `<number:number-style>` 16.29.2, `<number:percentage-style>` 16.29.10, `<number:text-style>` 16.29.26, `<number:time-style>` 16.29.19, `<style:master-page>` 16.9, `<style:page-layout>` 16.5, `<style:presentation-page-layout>` 16.43, `<style:style>` 16.2, `<text:list-style>` 16.32 and `<text:outline-style>` 16.36.

The `style:name` attribute has the data type `styleName` 18.3.32.

## 19.502.3 <style:font-face>

The `style:name` attribute specifies a unique name for a font declaration. This name can be used inside styles as an attribute of the `<style:text-properties>` element) as the value of a `style:font-name` attribute to immediately select a font face declaration.

The `style:name` attribute is usable with the following element: `<style:font-face>` 16.23.

The `style:name` attribute has the data type `string` 18.2.

## 19.503 style:next-style-name

### 19.503.1 General

The `style:next-style-name` attribute specifies a style to be used with pages and paragraphs.

### 19.503.2 <style:master-page>

The `style:next-style-name` attribute specifies the name of the master page that is used for the next page if the current page is entirely filled. If the next style name is not specified, the current master page is used for the next page. The value of this attribute shall be the name of a `<style:master-page>` element.

The `style:next-style-name` attribute is usable with the following element:

`<style:master-page>` 16.9.

The `style:next-style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.503.3 <style:style>

Within styles for paragraphs, `style:next-style-name` attribute specifies the style to be used for the next paragraph if a paragraph break is inserted in the user interface. By default, the current style is used as the next style.

The `style:next-style-name` attribute is usable with the following element:

`<style:style>` 16.2.

The `style:next-style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.504 style:num-format

The `style:num-format` attribute specifies a numbering sequence.

The defined values for the `style:num-format` attribute are:

- 1: Hindu-Arabic number sequence starts with 1.
- a: number sequence of lowercase Modern Latin basic alphabet characters starts with “a”.
- A: number sequence of uppercase Modern Latin basic alphabet characters starts with “A”.
- i: number sequence of lowercase Roman numerals starts with “i”.
- I: number sequence of uppercase Roman numerals start with “I”.
- a value of type `string` 18.2.
- an empty string: no number sequence displayed.

If no value is given, no number sequence is displayed.

The `style:num-format` attribute is usable with the following elements:

`<text:character-count>` 7.5.18.5, `<text:database-row-number>` 7.6.6, `<text:image-count>` 7.5.18.7, `<text:linenumbering-configuration>` 16.31.1, `<text:list-level-style-number>` 16.34, `<text:notes-configuration>` 16.31.3, `<text:object-count>` 7.5.18.8, `<text:outline-level-style>` 16.37, `<text:page-count>` 7.5.18.2, `<text:page-number>` 7.3.4, `<text:page-variable-get>` 7.7.1.3, `<text:paragraph-count>` 7.5.18.3, `<text:sequence>` 7.4.13, `<text:table-count>` 7.5.18.6 and `<text:word-count>` 7.5.18.4.

The values of the `style:num-format` attribute are 1, i, I, a value of type `string` 18.2, an empty string, a or A.

## 19.505 style:num-letter-sync

The `style:num-letter-sync` attribute specifies whether letter synchronization shall take place. If letters are used in alphabetical order for numbering, there are two ways to process overflows within a digit, as follows:

- `false`: A new digit is inserted. Its start value is "a" or "A", and it is incremented every time an overflow occurs in the following digit. The numbering sequence (for lower case numberings) in that case is a,b,c, ..., z, aa, ab, ac, ...,az, ba, ..., and so on.
- `true`: A new digit is inserted that always has the same value as the following digit. The numbering sequence (for lower case numberings) in that case is a, b, c, ..., z, aa, bb, cc, ..., zz, aaa, ..., and so on. This is called letter synchronization.

The `style:num-letter-sync` attribute is usable with the following elements:  
`<text:character-count> 7.5.18.5, <text:database-row-number> 7.6.6,`  
`<text:image-count> 7.5.18.7, <text:linenumbering-configuration> 16.31.1,`  
`<text:list-level-style-number> 16.34, <text:notes-configuration> 16.31.3,`  
`<text:object-count> 7.5.18.8, <text:outline-level-style> 16.37,`  
`<text:page-count> 7.5.18.2, <text:page-number> 7.3.4, <text:page-variable-get> 7.7.1.3, <text:paragraph-count> 7.5.18.3, <text:sequence> 7.4.13,`  
`<text:table-count> 7.5.18.6 and <text:word-count> 7.5.18.4.`

The `style:num-letter-sync` attribute has the data type `boolean` 18.3.3.

## 19.506 style:num-prefix

The `style:num-prefix` attribute specifies what to display before a number.

If the `style:num-prefix` and `style:num-suffix` 19.507 values do not contain any character that has a Unicode category of Nd, Ni, No, Lu, Li, Lt, Lm or Lo, an [XSLT] format attribute can be created from the OpenDocument attributes by concatenating the values of the `style:num-prefix`, `style:num-format` 19.504, and `style:num-suffix` attributes.

The `style:num-prefix` attribute can also specify a character before the value of a `text:bullet-char` 19.765 attribute.

The `style:num-prefix` attribute is usable with the following elements: `<text:list-level-style-bullet> 16.33, <text:list-level-style-number> 16.34,`  
`<text:notes-configuration> 16.31.3 and <text:outline-level-style> 16.37.`

The `style:num-prefix` attribute has the data type `string` 18.2.

## 19.507 style:num-suffix

The `style:num-suffix` attribute specifies what to display after a number.

If the `style:num-prefix` 19.506 and `style:num-suffix` values do not contain any character that has a Unicode category of Nd, Ni, No, Lu, Li, Lt, Lm or Lo, an [XSLT] format attribute can be created from the OpenDocument attributes by concatenating the values of the `style:num-prefix`, `style:num-format` 19.504, and `style:num-suffix` attributes.

The `style:num-suffix` attribute can also specify a character after the value of a `text:bullet-char` 19.765 attribute.

The `style:num-suffix` attribute is usable with the following elements: `<text:list-level-style-bullet> 16.33, <text:list-level-style-number> 16.34,`  
`<text:notes-configuration> 16.31.3 and <text:outline-level-style> 16.37.`

The `style:num-suffix` attribute has the data type `string` 18.2.

## 19.508 style:page-layout-name

The `style:page-layout-name` attribute specifies a page layout style that contains sizes, border and orientation attributes.

The `style:page-layout-name` attribute is usable with the following elements:

`<presentation:notes>` 16.19, `<style:handout-master>` 10.2.1 and  
`<style:master-page>` 16.9.

The `style:page-layout-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.509 style:page-usage

The `style:page-usage` attribute specifies the type of pages that a master page should generate.

The defined values for the `style:page-usage` attribute are:

- `all`: if there are no `<style:header-left>` or `<style:footer-left>` elements, the header and footer content is the same for left and right pages.
- `left`: `<style:header-right>` and `<style:footer-right>` elements are ignored.
- `mirrored`: if there are no `<style:header-left>` or `<style:footer-left>` elements, the header and footer content is the same for left and right pages.
- `right`: `<style:header-left>` and `<style:footer-left>` elements are ignored.

The default value for this attribute is `all`.

The `style:page-usage` attribute determines together with the `style:writing-mode` 20.404.3 attribute the “binding edge” for documents. Table 16 specifies, which physical edge of a page is used as “binding edge”. For writing-mode values `rl-tb`, `tb-rl`, `rl`, and `tb` it is specified in column RTL and for all other writing-mode values in column LTR.

Illustration of binding edge and reading direction for books.

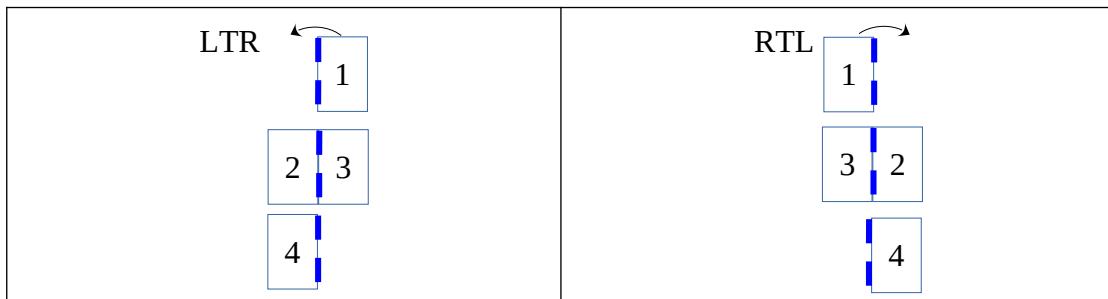


Table 16: Overview of which physical edge is the "binding edge."

value of attribute style:page-usage	physical edge of the page for binding	
	LTR	RTL
all	left	right
left	right	left
right	left	right
mirrored	for pages at <b>odd</b> position in the physical sequence of pages left	right
	for pages at <b>even</b> position in the physical sequence of pages right	left

The `style:page-usage` attribute is usable with the following element: `<style:page-layout>` 16.5.

The values of the `style:page-usage` attribute are `all`, `left`, `right` or `mirrored`.

## 19.510 style:parent-style-name

The `style:parent-style-name` attribute specifies the name of a parent style. The parent style cannot be an automatic style and shall exist.

If a parent style is not specified, the default style which has the same `style:family` 19.480 attribute value as the current style is used.

The `style:parent-style-name` attribute is usable with the following element: `<style:style>` 16.2.

The `style:parent-style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.511 style:percentage-data-style-name

The `style:percentage-data-style-name` attribute references the name of a percentage data style.

This attribute should be used together with `chart:data-label-number` attribute when the `chart:data-label-number` attribute has the value `percentage` or the value `value-and-percentage`.

The `style:percentage-data-style-name` attribute is usable with the following element: `<style:style>` 16.2.

The `style:percentage-data-style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.512 style:position

### 19.512.1 General

This attribute defines a position.

### 19.512.2 <style:background-image>

The `style:position` attribute specifies the position of a background image. Its value can be a space-separated combination of `top`, `center` or `bottom` for the vertical position and `left`, `center` or `right` for the horizontal position. The vertical and horizontal positions can be specified in any order. If one position is specified, the other position defaults to `center`.

For a `<style:background-image>` 17.3 element the default value for this attribute is `center`.

The `style:position` attribute is usable with the following element:

`<style:background-image>` 17.3.

The values of the `style:position` attribute are `left`, `center`, `right`, `top`, `bottom`, or two white space separated values, that may appear in any order. One of these values is one of: `left`, `center` or `right`. The other value is one of: `top`, `center` or `bottom`.

### 19.512.3 <style:tab-stop>

The `style:position` attribute specifies the position of a tab stop. Depending on the value of the `text:relative-tab-stop-position` 19.861 attribute in the `<text:table-of-content-source>` 8.3.2, `<text:illustration-index-source>` 8.4.2, `<text:object-index-source>` 8.6.2, `<text:user-index-source>` 8.7.2 or `<text:alphabetical-index-source>` 8.8.2 parent element, the position of the tab is interpreted as being relative to the left margin or the left indent.

The `style:position` attribute is usable with the following element: `<style:tab-stop>` 17.8.

The `style:position` attribute has the data type `length` 18.3.19.

### 19.512.4 <text:index-entry-tab-stop>

The `style:position` attribute specifies the position of a tab stop. If the value of the `style:type` attribute is `left`, then this attribute shall be present. Otherwise, it shall be omitted.

Depending on the value of the `text:relative-tab-stop-position` 19.861 attribute in the `<text:table-of-content-source>` 8.5.2, `<text:illustration-index-source>` 8.4.2, `<text:object-index-source>` 8.6.2, `<text:user-index-source>` 8.7.2 or `<text:alphabetical-index-source>` 8.8.2 parent element, the position of the tab is interpreted as being relative to the left margin or the left indent.

The `style:position` attribute is usable with the following element: `<text:index-entry-tab-stop>` 8.13.6.

The `style:position` attribute has the data type `length` 18.3.19.

## 19.513 style:rel-height

The `style:rel-height` attribute specifies height of a drawing object as a relative value within a frame.

The defined values for the `style:rel-height` attribute are:

- `scale`: the height should be calculated depending on the width, so that the ratio of width and height of the original image or object size is preserved.
- `scale-min`: the height should be calculated as for value `scale`, but the calculated height is a minimum height rather than an absolute one.
- a value of type `percent` 18.3.24.

The interpretation of percentage values depends on the anchor of the drawing object. If the anchor for the drawing object is in a table cell, the percentage value is relative to the surrounding table box. If the anchor for the drawing object is in a text box, the percentage value is relative to the surrounding text box. In other cases, the percentage values is relative to the width of the page or window.

To support consumers that do not support relative width and heights, producers should also provide the height in `svg:height` and `fo:min-height` attributes.

The `style:rel-height` attribute is usable with the following element: `<draw:frame>`  
10.4.2.

The values of the `style:rel-height` attribute are a value of type `percent` 18.3.24, `scale` or `scale-min`.

## 19.514 style:rel-width

### 19.514.1 General

The `style:rel-width` attribute specifies widths.

### 19.514.2 <draw:frame>

The `style:rel-width` attribute specifies the width of a drawing object as a relative value within a frame.

The defined values for the `style:rel-width` attribute are:

- `scale`: the width should be calculated depending on the height, so that the ratio of width and height of the original image or object size is preserved.
- `scale-min`: the width should be calculated as for value `scale`, but the calculated width is a minimum width rather than an absolute one.
- a value of type `percent` 18.3.24.

The interpretation of percentage values depends on the anchor of the drawing object. If the anchor for the drawing object is in a table cell, the percentage value is relative to the surrounding table box. If the anchor for the drawing object is in a text box, the percentage value is relative to the surrounding text box. In other cases, the percentage values is relative to the width of the page or window.

To support consumers that do not support relative width, producers should also provide the width and heights in the `svg:width` and `fo:min-width` attributes.

The `style:rel-width` attribute is usable with the following element: `<draw:frame>`  
10.4.2.

The values of the `style:rel-width` attribute are a value of type `percent` 18.3.24, `scale` or `scale-min`.

### 19.514.3 <style:column>

The `style:rel-width` attribute specifies the width of a column.

Column widths are specified as relative widths, that is, a number followed by a "\*" (U+002A, ASTERISK) character. The space available for columns is distributed according to their relative widths."

The `style:rel-width` attribute is usable with the following element: `<style:column>` 17.13.

The `style:rel-width` attribute has the data type `relativeLength` 18.3.28.

#### 19.514.4 `<style:footnote-sep>`

The `style:rel-width` attribute specifies the length of the footnote separator line as a percentage of the body-text area.

The `style:rel-width` attribute is usable with the following element: `<style:footnote-sep>` 17.4.

The `style:rel-width` attribute has the data type `percent` 18.3.24.

#### 19.515 `style:repeat`

The `style:repeat` attribute specifies if an image can be repeated or stretched over an area.

The defined values for the `style:repeat` attribute are:

- `no-repeat`: image should not be repeated.
- `repeat`: image should be repeated.
- `stretch`: image should be stretched over an area.

For a `<style:background-image>` 17.3 element the default value for this attribute is `repeat`.

The `style:repeat` attribute is usable with the following elements: `<style:background-fill>` 17.26 and `<style:background-image>` 17.3.

The values of the `style:repeat` attribute are `no-repeat`, `repeat`, `stretch` or `scale`.

#### 19.516 `style:rfc-language-tag`

The `style:rfc-language-tag` attribute specifies a language identifier according to the rules of [RFC5646], or its successors.

It shall only be used if its value cannot be expressed as a valid combination of the `fo:language` 19.871, `fo:script` 19.242 and `fo:country` 19.234 attributes.

**Note.** Producers may choose to assist consumers that don't support the `style:rfc-language-tag` attribute by specifying `fo:language`, `fo:script` and `fo:country` attributes with values whose combination is otherwise invalid and whose interpretation is therefore implementation-dependent.

The `style:rfc-language-tag` attribute is usable with the following elements: `<text:alphabetical-index-source>` 8.8.2 and `<text:bibliography-configuration>` 16.31.6.

The `style:rfc-language-tag` attribute has the data type `language` 18.3.17.

#### 19.517 `style:style`

The `style:style` attribute specifies the line style of a column separator line.

The defined values for the `style:style` attribute are:

- **none**: no separator line.
- **dashed**: dashed separator line.
- **dot-dashed**: separator line whose repeating pattern is a dot followed by a dash.
- **dotted**: dotted separator line.
- **solid**: solid separator line.

The default value for this attribute is **solid**.

The `style:style` attribute is usable with the following element: `<style:column-sep>` 17.14.

The values of the `style:style` attribute are `none`, `solid`, `dotted`, `dashed` or `dot-dashed`.

## 19.518 style:style-name

The `style:style-name` attribute specifies the text style to apply to characters specified by a `<style:drop-cap>` element.

The `style:style-name` attribute is usable with the following element: `<style:drop-cap>` 17.9.

The `style:style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.519 style:type

### 19.519.1 General

The `style:type` attribute specifies a tab stop type.

### 19.519.2 <text:index-entry-tab-stop>

The `style:type` attribute specifies the type of a tab stop within an index entry template. If the value of this attribute is `left`, the `style:position` attribute shall also be present. Otherwise, this attribute shall be omitted.

The defined values for the `style:type` attribute are:

- `left`: text is left aligned with a tab stop.
- `right`: text is right aligned with a tab stop.

The `style:type` attribute is usable with the following element: `<text:index-entry-tab-stop>` 8.13.6.

The values of the `style:type` attribute are `right` or `left`.

### 19.519.3 <style:tab-stop>

The `style:type` attribute specifies the type of a tab stop within paragraph formatting properties.

The defined values for the `style:type` attribute are:

- `center`: text is centered on a tab stop.
- `char`: character appears at a tab stop position.
- `left`: text is left aligned with a tab stop.
- `right`: text is right aligned with a tab stop.

For a `<style:tab-stop> 17.8` element the default value for this attribute is `left`.

The `style:type` attribute is usable with the following element: `<style:tab-stop> 17.8`.

The values of the `style:type` attribute are `left`, `center`, `right` or `char`.

## 19.520 `style:vertical-align`

The `style:vertical-align` attribute specifies how to vertically align a line in a columned area. The value of this attribute can be either `top`, `middle`, or `bottom`.

The defined values for the `style:vertical-align` attribute are:

- `bottom`: line is aligned with the bottom of a columned area.
- `middle`: line is aligned with the middle of a columned area.
- `top`: line is aligned with the top of a columned area.

For a `<style:column-sep>` element the default value for this attribute is `top`.

The `style:vertical-align` attribute is usable with the following element:  
`<style:column-sep> 17.14`.

The values of the `style:vertical-align` attribute are `top`, `middle` or `bottom`.

## 19.521 `style:volatile`

The `style:volatile` attribute specifies whether unused style in a document are retained or discarded by consumers.

The defined values for the `style:volatile` attribute are:

- `false`: consumers should discard the unused styles.
- `true`: consumers should keep unused styles.

The `style:volatile` attribute is usable with the following elements: `<number:boolean-style> 16.29.24`, `<number:currency-style> 16.29.8`, `<number:date-style> 16.29.11`, `<number:number-style> 16.29.2`, `<number:percentage-style> 16.29.10`, `<number:text-style> 16.29.26` and `<number:time-style> 16.29.19`.

The `style:volatile` attribute has the data type `boolean` [18.3.3](#).

## 19.522 `style:width`

### 19.522.1 General

The `style:width` attribute specifies a width.

### 19.522.2 `<style:column-sep>`

The `style:width` attribute specifies the width of a column separation.

The `style:width` attribute is usable with the following element: `<style:column-sep> 17.14`.

The `style:width` attribute has the data type `length` [18.3.19](#).

### 19.522.3 `<style:footnote-sep>`

The `style:width` attribute width or thickness of a line.

The `style:width` attribute is usable with the following element: `<style:footnote-sep>` 17.4.

The `style:width` attribute has the data type `length` 18.3.19.

## 19.523 `svg:accent-height`

See §20.8.3 of [SVG].

The `svg:accent-height` attribute is usable with the following element: `<style:font-face>` 16.23.

The `svg:accent-height` attribute has the data type `integer` 18.2.

## 19.524 `svg:alphabetic`

See §20.8.3 of [SVG].

The `svg:alphabetic` attribute is usable with the following element: `<style:font-face>` 16.23.

The `svg:alphabetic` attribute has the data type `integer` 18.2.

## 19.525 `svg:ascent`

See §20.8.3 of [SVG].

The `svg:ascent` attribute is usable with the following element: `<style:font-face>` 16.23.

The `svg:ascent` attribute has the data type `integer` 18.2.

## 19.526 `svg:bbox`

See §20.8.3 of [SVG].

The `svg:bbox` attribute is usable with the following element: `<style:font-face>` 16.23.

The `svg:bbox` attribute has the data type `string` 18.2.

## 19.527 `svg:cap-height`

See §20.8.3 of [SVG].

The `svg:cap-height` attribute is usable with the following element: `<style:font-face>` 16.23.

The `svg:cap-height` attribute has the data type `integer` 18.2.

## 19.528 `svg:cx`

### 19.528.1 `<draw:area-circle>`

The `svg:cx` attribute defines the x-axis coordinate of the center of a circular image map area.

The `svg:cx` attribute is usable with the following element: `<draw:area-circle>` 10.4.13.4.

The `svg:cx` attribute has the data type `coordinate` 18.3.10.

### 19.528.2 `<draw:circle>`

See §9.3 of [SVG].

The `svg:cx` attribute is usable with the following element: `<draw:circle>` 10.3.8.

The `svg:cx` attribute has the data type `coordinate` 18.3.10.

### 19.528.3 `<draw:ellipse>`

See §9.4 of [SVG].

The `svg:cx` attribute is usable with the following element: `<draw:ellipse>` 10.3.9.

The `svg:cx` attribute has the data type `coordinate` 18.3.10.

### 19.528.4 `<svg:radialGradient>`

See §13.3.2 of [SVG].

For a `<svg:radialGradient>` 16.42.3 element the default value for this attribute is 50%.

The `svg:cx` attribute is usable with the following element: `<svg:radialGradient>` 16.42.3.

The values of the `svg:cx` attribute are a value of type `coordinate` 18.3.10 or a value of type `percent` 18.3.24.

## 19.529 `svg:cy`

### 19.529.1 `<draw:area-circle>`

The `svg:cy` attribute defines the y-axis coordinate of the center of a circular image map area.

The `svg:cy` attribute is usable with the following element: `<draw:area-circle>` 10.4.13.4.

The `svg:cy` attribute has the data type `coordinate` 18.3.10.

### 19.529.2 `<draw:circle>`

See §9.3 of [SVG].

The `svg:cy` attribute is usable with the following element: `<draw:circle>` 10.3.8.

The `svg:cy` attribute has the data type `coordinate` 18.3.10.

### 19.529.3 `<draw:ellipse>`

See §9.4 of [SVG].

The `svg:cy` attribute is usable with the following element: `<draw:ellipse>` 10.3.9.

The `svg:cy` attribute has the data type `coordinate` 18.3.10.

### 19.529.4 `<svg:radialGradient>`

See §13.3.2 of [SVG].

For a `<svg:radialGradient>` 16.42.3 element the default value for this attribute is 50%.

The `svg:cy` attribute is usable with the following element: `<svg:radialGradient>` 16.42.3.

The values of the `svg:cy` attribute are a value of type `coordinate` 18.3.10 or a value of type `percent` 18.3.24.

## 19.530 `svg:d`

See §8 of [SVG].

The `svg:d` attribute is usable with the following elements: `<dr3d:extrude>` 10.5.6, `<dr3d:rotate>` 10.5.7, `<draw:connector>` 10.3.10, `<draw:contour-path>` 10.4.11.3, `<draw:marker>` 16.42.8 and `<draw:path>` 10.3.7.

The `svg:d` attribute has the data type `pathData` 18.3.23.

## 19.531 `svg:descent`

See §20.8.3 of [SVG].

The `svg:descent` attribute is usable with the following element: `<style:font-face>` 16.23.

The `svg:descent` attribute has the data type `integer` 18.2.

## 19.532 `svg:font-family`

See §20.8.3 of [SVG].

The `svg:font-family` attribute is usable with the following element: `<style:font-face>` 16.23.

The `svg:font-family` attribute has the data type `string` 18.2.

## 19.533 `svg:font-size`

See §20.8.3 of [SVG].

The `svg:font-size` attribute is usable with the following element: `<style:font-face>` 16.23.

The `svg:font-size` attribute has the data type `positiveLength` 18.3.27.

## 19.534 `svg:font-stretch`

See §20.8.3 of [SVG].

The `svg:font-stretch` attribute is usable with the following element: `<style:font-face>` 16.23.

The values of the `svg:font-stretch` attribute are `normal`, `ultra-condensed`, `extra-condensed`, `condensed`, `semi-condensed`, `semi-expanded`, `expanded`, `extra-expanded` or `ultra-expanded`.

## 19.535 `svg:font-style`

See §20.8.3 of [SVG].

The `svg:font-style` attribute is usable with the following element: `<style:font-face>` 16.23.

The values of the `svg:font-style` attribute are `normal`, `italic` or `oblique`.

## 19.536 `svg:font-variant`

See §20.8.3 of [SVG].

The `svg:font-variant` attribute is usable with the following element: `<style:font-face>` 16.23.

The values of the `svg:font-variant` attribute are `normal` or `small-caps`.

## 19.537 `svg:font-weight`

See §20.8.3 of [SVG].

The `svg:font-weight` attribute is usable with the following element: `<style:font-face>` 16.23.

The values of the `svg:font-weight` attribute are `normal`, `bold`, `100`, `200`, `300`, `400`, `500`, `600`, `700`, `800` or `900`.

## 19.538 `svg:fx`

See §13.2.3 of [SVG].

The `svg:fx` attribute is usable with the following element: `<svg:radialGradient>` 16.42.3.

The values of the `svg:fx` attribute are a value of type coordinate 18.3.10 or a value of type percent 18.3.24.

## 19.539 `svg:fy`

See §13.2.3 of [SVG].

The `svg:fy` attribute is usable with the following element: `<svg:radialGradient>` 16.42.3.

The values of the `svg:fy` attribute are a value of type coordinate 18.3.10 or a value of type percent 18.3.24.

## 19.540 `svg:gradientTransform`

See §13.2.2 and §13.2.3 of [SVG].

The `svg:gradientTransform` attribute is usable with the following elements: `<svg:linearGradient>` 16.42.2 and `<svg:radialGradient>` 16.42.3.

The `svg:gradientTransform` attribute has the data type string 18.2.

## 19.541 `svg:gradientUnits`

See §13.2.2 and §13.2.3 of [SVG].

The default value for this attribute is `objectBoundingBox`.

The `svg:gradientUnits` attribute is usable with the following elements: `<svg:linearGradient>` 16.42.2 and `<svg:radialGradient>` 16.42.3.

The only value of the `svg:gradientUnits` attribute is `objectBoundingBox`.

## 19.542 `svg:hanging`

See §20.8.3 of [SVG].

The `svg:hanging` attribute is usable with the following element: `<style:font-face>` 16.23.

The `svg:hanging` attribute has the data type integer 18.2.

## 19.543 `svg:height`

### 19.543.1 `<chart:chart>`

The `svg:height` attribute defines the height of the entire chart. If it is omitted, the size of the chart is determined by the size of the window in which the chart is displayed.

The `svg:height` attribute is usable with the following element: `<chart:chart>` 11.1.

The `svg:height` attribute has the data type `length` 18.3.19.

### 19.543.2 `<chart:legend>`

The `svg:height` attribute defines the height of a legend.

The `svg:height` attribute is usable with the following element: `<chart:legend>` 11.4.

The `svg:height` attribute has the data type `length` 18.3.19.

### 19.543.3 `<chart:coordinate-region>`

The `svg:height` attribute defines the height of a positioning rectangle, see 11.3.

The `svg:height` attribute is usable with the following element: `<chart:coordinate-region>` 11.3.

The `svg:height` attribute has the data type `length` 18.3.19.

### 19.543.4 `<chart:plot-area>`

The `svg:height` attribute defines the height of the plot-area including axis labels.

The `svg:height` attribute is usable with the following element: `<chart:plot-area>` 11.5.

The `svg:height` attribute has the data type `length` 18.3.19.

### 19.543.5 `<dr3d:scene>`

The `svg:height` attribute specifies the height of the viewport.

The `svg:height` attribute is usable with the following element: `<dr3d:scene>` 10.5.2.

The `svg:height` attribute has the data type `length` 18.3.19.

### 19.543.6 `<draw:area-polygon>`

The `svg:height` attribute defines the height of a polygonal image map area's bounding box.

The `svg:height` attribute is usable with the following element: `<draw:area-polygon>` 10.4.13.5.

The `svg:height` attribute has the data type `length` 18.3.19.

### 19.543.7 `<draw:area-rectangle>`

The `svg:height` attribute defines the height of a rectangular image map area.

The `svg:height` attribute is usable with the following element: `<draw:area-rectangle>` 10.4.13.3.

The `svg:height` attribute has the data type `length` 18.3.19.

### 19.543.8 <draw:contour-polygon>, <draw:contour-path>

The `svg:height` attribute defines the height of a contour's bounding box.

The `svg:height` attribute is usable with the following elements: `<draw:contour-path>` 10.4.11.3 and `<draw:contour-polygon>` 10.4.11.2.

The `svg:height` attribute has the data type `length` 18.3.19.

### 19.543.9 <draw:fill-image>

The `svg:height` attribute may be used to specify the height of a linked image. Its value is overridden by the height of the linked image resource.

**Note:** This attribute can be used to estimate the size of a fill image without loading the image data.

The `svg:height` attribute is usable with the following element: `<draw:fill-image>` 16.42.6.

The `svg:height` attribute has the data type `length` 18.3.19.

### 19.543.10 <draw:caption>, <draw:custom-shape>, <draw:circle>, <draw:control>, <draw:ellipse>, <draw:frame>, <draw:rect>, <draw:page-thumbnail>, <draw:path>, <draw:polygon>, <draw:polyline>, <draw:regular-polygon>

See §5.1.2 of [SVG]. For drawing shapes that have a non-rectangular shape, the length refers to the drawing shape's bounding box.

If the `svg:width` and `svg:height` attributes specify different values for a `<draw:circle>` element, the radius of the circle is derived from the smaller of the two values, and the circle is centered within the bounding box.

The `svg:height` attribute is usable with the following elements: `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:page-thumbnail>` 10.3.14, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2 and `<draw:regular-polygon>` 10.3.6.

The `svg:height` attribute has the data type `length` 18.3.19.

### 19.543.11 <office:annotation>

The `svg:height` attribute specifies the height of the rectangular which displays the annotation. Its meaning is the same as for a `<draw:caption>` element. See 19.575.12.

The `svg:height` attribute is usable with the following element: `<office:annotation>` 14.1.

The `svg:height` attribute has the data type `length` 18.3.19.

### 19.543.12 <presentation:placeholder>

See 19.575.12. Percentage values are relative to the height of the drawing page.

The `svg:height` attribute is usable with the following element: `<presentation:placeholder>` 16.44.

The values of the `svg:height` attribute are a value of type `length` 18.3.19 or a value of type `percent` 18.3.24.

## 19.544 `svg:ideographic`

See §20.8.3 of [SVG].

The `svg:ideographic` attribute is usable with the following element: `<style:font-face>` 16.23.

The `svg:ideographic` attribute has the data type `integer` 18.2.

## 19.545 `svg:mathematical`

See §20.8.3 of [SVG].

The `svg:mathematical` attribute is usable with the following element: `<style:font-face>` 16.23.

The `svg:mathematical` attribute has the data type `integer` 18.2.

## 19.546 `svg:name`

See §20.8.3 of [SVG].

The `svg:name` attribute is usable with the following element: `<svg:font-face-name>` 16.25.

The `svg:name` attribute has the data type `string` 18.2.

## 19.547 `svg:offset`

See §13.2.4 of [SVG].

The `svg:offset` attribute is usable with the following element: `<svg:stop>` 16.42.4.

The values of the `svg:offset` attribute are a value of type `double` 18.2 or a value of type `percent` 18.3.24.

## 19.548 `svg:origin`

See §19.2.12 of [SVG].

**Note:** SVG relies upon the definition of origin in SMIL, see:

<http://www.w3.org/TR/2001/REC-smil-animation-20010904/#MotionOriginAttribute>

The `svg:origin` attribute is usable with the following element: `<anim:animateMotion>` 15.2.5.

The `svg:origin` attribute has the data type `string` 18.2.

## 19.549 `svg:overline-position`

See §20.8.3 of [SVG].

The `svg:overline-position` attribute is usable with the following element: `<style:font-face>` 16.23.

The `svg:overline-position` attribute has the data type `integer` 18.2.

## 19.550 `svg:overline-thickness`

See §20.8.3 of [SVG].

The `svg:overline-thickness` attribute is usable with the following element:

`<style:font-face> 16.23.`

The `svg:overline-thickness` attribute has the data type `integer 18.2.`

## 19.551 `svg:panose-1`

See §20.8.3 of [SVG].

The `svg:panose-1` attribute is usable with the following element: `<style:font-face> 16.23.`

The `svg:panose-1` attribute has the data type `string 18.2.`

## 19.552 `svg:path`

See §19.2.12 of [SVG].

The `svg:path` attribute is usable with the following element: `<anim:animateMotion> 15.2.5.`

The `svg:path` attribute has the data type `pathData 18.3.23.`

## 19.553 `svg:r`

### 19.553.1 `<draw:area-circle>`

The `svg:r` attribute defines the radius of a circular image map area.

The `svg:r` attribute is usable with the following element: `<draw:area-circle> 10.4.13.4.`

The `svg:r` attribute has the data type `length 18.3.19.`

### 19.553.2 `<draw:circle>`

The `svg:r` attribute defines the radius of a circle. The use of this attribute is defined by §9.3 of [SVG].

**Note:** If a `<draw:circle>` element does not have a value for the `svg:r` attribute, then its `svg:x`, `svg:y`, `svg:height`, and `svg:width` attributes can be used to define the geometry of a circle.

The `svg:r` attribute is usable with the following element: `<draw:circle> 10.3.8.`

The `svg:r` attribute has the data type `length 18.3.19.`

### 19.553.3 `<svg:radialGradient>`

See §13.2.3 of [SVG].

For a `<svg:radialGradient> 16.42.3` element the default value for this attribute is 50%.

The `svg:r` attribute is usable with the following element: `<svg:radialGradient> 16.42.3.`

The values of the `svg:r` attribute are a value of type `coordinate 18.3.10` or a value of type `percent 18.3.24.`

## 19.554 `svg:rx`

See §9.4 of [SVG].

The `svg:rx` and `svg:ry` attributes can be used to round off the corners of a rectangle. The `svg:rx` attribute specifies the x-axis radius of the ellipse used to round off the corners of a

rectangle. The `svg:ry` attribute specifies the y-axis radius of that ellipse. If only the `svg:rx` attribute is present then its value will be used for `svg:ry`. If only a `svg:ry` attribute is present then its value will be used for `svg:rx`.

For use of this attribute with `<draw:rect>` see §9.2 of [SVG].

For use of this attribute with `<draw:ellipse>` see §9.4 of [SVG].

The `svg:rx` attribute is usable with the following elements: `<draw:caption>` 10.3.11, `<draw:ellipse>` 10.3.9, `<draw:rect>` 10.3.2, `<draw:text-box>` 10.4.3 and `<office:annotation>` 14.1.

The `svg:rx` attribute has the data type `nonNegativeLength` 18.3.21.

## 19.555 `svg:ry`

See §9.4 of [SVG].

The `svg:rx` and `svg:ry` attributes can be used to round off the corners of a rectangle. The `svg:rx` attribute specifies the x-axis radius of the ellipse used to round off the corners of a rectangle. The `svg:ry` attribute specifies the y-axis radius of that ellipse. If only the `svg:rx` attribute is present then its value will be used for `svg:ry`. If only a `svg:ry` attribute is present then its value will be used for `svg:rx`.

The `svg:ry` attribute is usable with the following elements: `<draw:caption>` 10.3.11, `<draw:ellipse>` 10.3.9, `<draw:rect>` 10.3.2, `<draw:text-box>` 10.4.3 and `<office:annotation>` 14.1.

The `svg:ry` attribute has the data type `nonNegativeLength` 18.3.21.

## 19.556 `svg:slope`

See §20.8.3 of [SVG].

The `svg:slope` attribute is usable with the following element: `<style:font-face>` 16.23.

The `svg:slope` attribute has the data type `integer` 18.2.

## 19.557 `svg:spreadMethod`

See §13.2.2 and §13.2.3 of [SVG].

The default value for this attribute is `pad`.

The `svg:spreadMethod` attribute is usable with the following elements: `<svg:linearGradient>` 16.42.2 and `<svg:radialGradient>` 16.42.3.

The values of the `svg:spreadMethod` attribute are `pad`, `reflect` or `repeat`.

## 19.558 `svg:stemh`

See §20.8.3 of [SVG].

The `svg:stemh` attribute is usable with the following element: `<style:font-face>` 16.23.

The `svg:stemh` attribute has the data type `integer` 18.2.

## 19.559 `svg:stemv`

See §20.8.3 of [SVG].

The `svg:stemv` attribute is usable with the following element: `<style:font-face>` 16.23.

The `svg:stemv` attribute has the data type `integer` 18.2.

## 19.560 `svg:stop-color`

See §13.2.4 of [SVG].

The `svg:stop-color` attribute is usable with the following element: `<svg:stop>` 16.42.4.

The `svg:stop-color` attribute has the data type `color` 18.3.9.

## 19.561 `svg:stop-opacity`

See §13.2.4 of [SVG].

The `svg:stop-opacity` attribute is usable with the following element: `<svg:stop>` 16.42.4.

The `svg:stop-opacity` attribute has the data type `double` 18.2.

## 19.562 `svg:strikethrough-position`

See §20.8.3 of [SVG].

The `svg:strikethrough-position` attribute is usable with the following element: `<style:font-face>` 16.23.

The `svg:strikethrough-position` attribute has the data type `integer` 18.2.

## 19.563 `svg:strikethrough-thickness`

See §20.8.3 of [SVG].

The `svg:strikethrough-thickness` attribute is usable with the following element: `<style:font-face>` 16.23.

The `svg:strikethrough-thickness` attribute has the data type `integer` 18.2.

## 19.564 `svg:string`

See §20.8.3 of [SVG].

The `svg:string` attribute is usable with the following element: `<svg:font-face-format>` 16.28.

The `svg:string` attribute has the data type `string` 18.2.

## 19.565 `svg:type`

See §19.2.14 of [SVG].

The `svg:type` attribute is usable with the following element: `<anim:animateTransform>` 15.2.3.

The values of the `svg:type` attribute are `translate`, `scale`, `rotate`, `skewX` or `skewY`.

## 19.566 `svg:underline-position`

See §20.8.3 of [SVG].

The `svg:underline-position` attribute is usable with the following element: `<style:font-face>` 16.23.

The `svg:underline-position` attribute has the data type `integer` 18.2.

## 19.567 `svg:underline-thickness`

See §20.8.3 of [SVG].

The `svg:underline-thickness` attribute is usable with the following element:  
`<style:font-face> 16.23.`

The `svg:underline-thickness` attribute has the data type `integer 18.2`.

## 19.568 `svg:unicode-range`

See §20.8.3 of [SVG].

The `svg:unicode-range` attribute is usable with the following element: `<style:font-face> 16.23.`

The `svg:unicode-range` attribute has the data type `string 18.2`.

## 19.569 `svg:units-per-em`

See §20.8.3 of [SVG].

The `svg:units-per-em` attribute is usable with the following element: `<style:font-face> 16.23.`

The `svg:units-per-em` attribute has the data type `integer 18.2`.

## 19.570 `svg:v-alphabetic`

See §20.8.3 of [SVG].

The `svg:v-alphabetic` attribute is usable with the following element: `<style:font-face> 16.23.`

The `svg:v-alphabetic` attribute has the data type `integer 18.2`.

## 19.571 `svg:v-hanging`

See §20.8.3 of [SVG].

The `svg:v-hanging` attribute is usable with the following element: `<style:font-face> 16.23.`

The `svg:v-hanging` attribute has the data type `integer 18.2`.

## 19.572 `svg:v-ideographic`

See §20.8.3 of [SVG].

The `svg:v-ideographic` attribute is usable with the following element: `<style:font-face> 16.23.`

The `svg:v-ideographic` attribute has the data type `integer 18.2`.

## 19.573 `svg:v-mathematical`

See §20.8.3 of [SVG].

The `svg:v-mathematical` attribute is usable with the following element: `<style:font-face> 16.23.`

The `svg:v-mathematical` attribute has the data type `integer 18.2`.

## 19.574 `svg:viewBox`

The `svg:viewBox` attribute specifies a rectangle in a local coordinate system of a shape. The syntax for using this attribute is the same as the [SVG] syntax. The value of the attribute are four numbers separated by white spaces, which define the left, top, width, and height of the rectangle. Such local coordinate system is used by the `draw:points` 19.206 and `svg:d` 19.530 attributes, as well as by the attributes of `<draw:enhanced-geometry>` 10.6.2 element and its child elements which specify coordinates. The `svg:viewBox` attribute and its value shall be present in case of the `draw:points` 19.206 and `svg:d` 19.530 attributes. The `svg:viewBox` attribute and its value shall be present in case of `<draw:enhanced-geometry>` 10.6.2, if there exists any position or length value without unit in its attributes or child elements.

The rectangle specified by the `svg:viewBox` attribute is mapped to the rectangle in the outer coordinate system, specified by the position and size attributes of the shape.

The `svg:viewBox` attribute is usable with the following elements: `<dr3d:extrude>` 10.5.6, `<dr3d:rotate>` 10.5.7, `<draw:area-polygon>` 10.4.13.5, `<draw:connector>` 10.3.10, `<draw:contour-path>` 10.4.11.3, `<draw:contour-polygon>` 10.4.11.2, `<draw:enhanced-geometry>` 10.6.2, `<draw:marker>` 16.42.8, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5 and `<draw:polyline>` 10.3.4.

The values of the `svg:viewBox` attribute are four white space separated values of type integer 18.2.

## 19.575 `svg:width`

### 19.575.1 `<chart:chart>`

The `svg:width` attribute defines the width of a chart. If it is omitted, the size of a chart is determined by the size of its `<chart:chart>` 11.1 document or its containing element in another document.

The `svg:width` attribute is usable with the following element: `<chart:chart>` 11.1.

The `svg:width` attribute has the data type `length` 18.3.19.

### 19.575.2 `<chart:coordinate-region>`

The `svg:width` attribute defines the width of a positioning rectangle, see 11.3.

The `svg:width` attribute is usable with the following element: `<chart:coordinate-region>` 11.3.

The `svg:width` attribute has the data type `length` 18.3.19.

### 19.575.3 `<chart:floor>`

The `svg:width` attribute defines the thickness of a floor.

The `svg:width` attribute is usable with the following element: `<chart:floor>` 11.8.

The `svg:width` attribute has the data type `length` 18.3.19.

### 19.575.4 `<chart:legend>`

The `svg:width` attribute defines the width of a legend.

The `svg:width` attribute is usable with the following element: `<chart:legend>` 11.4.

The `svg:width` attribute has the data type `length` 18.3.19.

### 19.575.5 <chart:plot-area>

The `svg:width` attribute defines the width of a plot-area including axis labels.

The `svg:width` attribute is usable with the following element: <chart:plot-area> 11.5.

The `svg:width` attribute has the data type `length` 18.3.19.

### 19.575.6 <chart:wall>

The `svg:width` attribute specifies the thickness of a wall for three-dimensional charts.

The `svg:width` attribute is usable with the following element: <chart:wall> 11.7.

The `svg:width` attribute has the data type `length` 18.3.19.

### 19.575.7 <dr3d:scene>

The `svg:width` attribute specifies the width of the viewport.

The `svg:width` attribute is usable with the following element: <dr3d:scene> 10.5.2.

The `svg:width` attribute has the data type `length` 18.3.19.

### 19.575.8 <draw:area-polygon>

The `svg:width` attribute defines the width of a polygonal image map area's bounding box.

The `svg:width` attribute is usable with the following element: <draw:area-polygon> 10.4.13.5.

The `svg:width` attribute has the data type `length` 18.3.19.

### 19.575.9 <draw:area-rectangle>

The `svg:width` attribute defines the width of a rectangular image map area.

The `svg:width` attribute is usable with the following element: <draw:area-rectangle> 10.4.13.3.

The `svg:width` attribute has the data type `length` 18.3.19.

### 19.575.10 <draw:contour-polygon>, <draw:contour-path>

The `svg:width` attribute defines the width of a contour's bounding box.

The `svg:width` attribute is usable with the following elements: <draw:contour-path> 10.4.11.3 and <draw:contour-polygon> 10.4.11.2.

The `svg:width` attribute has the data type `length` 18.3.19.

### 19.575.11 <draw:fill-image>

The `svg:width` attribute may be used to specify the width of a linked image. Its value is overridden by the width of the linked image resource.

**Note:** This attribute can be used to estimate the size of a fill image without loading the image data.

The `svg:width` attribute is usable with the following element: <draw:fill-image> 16.42.6.

The `svg:width` attribute has the data type `length` 18.3.19.

### 19.575.12 `<draw:caption>`, `<draw:custom-shape>`, `<draw:circle>`, `<draw:control>`, `<draw:ellipse>`, `<draw:frame>`, `<draw:rect>`, `<draw:page-thumbnail>`, `<draw:path>`, `<draw:polygon>`, `<draw:polyline>`, `<draw:regular-polygon>`

See §5.1.2 of [SVG]. For drawing shapes that have a non-rectangular shape, the length refers to the drawing shape's bounding box.

If the `svg:width` and `svg:height` 19.234 attributes specify different values for a `<draw:circle>` element, the radius of the circle is derived from the smaller of the two values, and the circle is centered within the bounding box.

The `svg:width` attribute is usable with the following elements: `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:page-thumbnail>` 10.3.14, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2 and `<draw:regular-polygon>` 10.3.6.

The `svg:width` attribute has the data type `length` 18.3.19.

### 19.575.13 `<office:annotation>`

The `svg:width` attribute specifies the width of the rectangular which displays the annotation. Its meaning is the same as for a `<draw:caption>` element. See 19.575.12.

The `svg:width` attribute is usable with the following element: `<office:annotation>` 14.1.

The `svg:width` attribute has the data type `length` 18.3.19.

### 19.575.14 `<presentation:placeholder>`

See 19.575.12. Percentage values are relative to the width of the drawing page.

The `svg:width` attribute is usable with the following element:  
`<presentation:placeholder>` 16.44.

The values of the `svg:width` attribute are a value of type `length` 18.3.19 or a value of type `percent` 18.3.24.

## 19.576 `svg:widths`

See §20.8.3 of [SVG].

The `svg:widths` attribute is usable with the following element: `<style:font-face>` 16.23.

The `svg:widths` attribute has the data type `string` 18.2.

## 19.577 `svg:x`

### 19.577.1 `<chart:coordinate-region>`

The `svg:x` attribute specifies the horizontal distance of the upper left corner of the `<chart:coordinate-region>` element from the upper left corner of the `<chart:chart>` element.

The `svg:x` attribute is usable with the following element: `<chart:coordinate-region>` 11.3.

The `svg:x` attribute has the data type `coordinate` 18.3.10.

## 19.577.2 `<chart:data-label>, <chart:equation>, <chart:footer>, <chart:legend>, <chart:plot-area>, <chart:subtitle>, <chart:title>`

The `svg:x` attribute specifies the horizontal distance of the upper-left corner of the given element from the upper-left corner of the `<chart:chart>` element.

The `svg:x` attribute is usable with the following elements: `<chart:data-label>` 11.15, `<chart:equation>` 11.19, `<chart:footer>` 11.2.3, `<chart:legend>` 11.4, `<chart:plot-area>` 11.5, `<chart:subtitle>` 11.2.2 and `<chart:title>` 11.2.1.

The `svg:x` attribute has the data type `coordinate` 18.3.10.

## 19.577.3 `<draw:area-polygon>`

The `svg:x` attribute defines the x-axis coordinate of a polygonal image map area's bounding box.

The `svg:x` attribute is usable with the following element: `<draw:area-polygon>` 10.4.13.5.

The `svg:x` attribute has the data type `coordinate` 18.3.10.

## 19.577.4 `<draw:area-rectangle>`

The `svg:x` attribute defines the x-axis coordinate of a rectangular image map area.

The `svg:x` attribute is usable with the following element: `<draw:area-rectangle>` 10.4.13.3.

The `svg:x` attribute has the data type `coordinate` 18.3.10.

## 19.577.5 `<dr3d:scene>`

The `svg:x` attribute specifies the horizontal position of the left edge of the viewport.

The `svg:x` attribute is usable with the following element: `<dr3d:scene>` 10.5.2.

The `svg:x` attribute has the data type `coordinate` 18.3.10.

## 19.577.6 `<draw:caption>, <draw:custom-shape>, <draw:circle>, <draw:control>, <draw:ellipse>, <draw:frame>, <draw:rect>, <draw:page-thumbnail>, <draw:path>, <draw:polygon>, <draw:polyline>, <draw:regular-polygon>`

See §5.1.2 of [SVG]. For drawing shapes that have a non-rectangular shape, the coordinate refers to the drawing shape's bounding box.

The `svg:x` attribute is usable with the following elements: `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:page-thumbnail>` 10.3.14, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2 and `<draw:regular-polygon>` 10.3.6.

The `svg:x` attribute has the data type `coordinate` 18.3.10.

## 19.577.7 <draw:glue-point>

The `svg:x` attribute specifies the horizontal position of the glue point. The coordinate is either a percentage value relative to the drawing shape's center measured horizontally, or, if the `draw:align` 19.111 attribute is also specified, an absolute distance value relative to the edge specified by the `draw:align` attribute.

The `svg:x` attribute is usable with the following element: <draw:glue-point> 10.3.16.

The values of the `svg:x` attribute are a value of type `distance` 18.3.15 or a value of type `percent` 18.3.24.

## 19.577.8 <office:annotation>

The `svg:x` attribute specifies the horizontal position of an annotation. Its meaning is the same as for a <draw:caption> element. See 19.577.6.

The `svg:x` attribute is usable with the following element: <office:annotation> 14.1.

The `svg:x` attribute has the data type `coordinate` 18.3.10.

## 19.577.9 <presentation:placeholder>

See 19.577.6. Percentage values are relative to the height of the drawing page.

The `svg:x` attribute is usable with the following element: <presentation:placeholder> 16.44.

The values of the `svg:x` attribute are a value of type `coordinate` 18.3.10 or a value of type `percent` 18.3.24.

## 19.578 `svg:x1`

### 19.578.1 <draw:connector>, <draw:line> and <draw:measure>

See §9.5 of [SVG].

The `svg:x1` attribute may be omitted on <draw:connector> elements when the connector is connected to a shape by a `draw:start-shape` 19.217 attribute. The start position then is the position of the glue point to which the connector is connected.

The `svg:x1` attribute is usable with the following elements: <draw:connector> 10.3.10, <draw:line> 10.3.3 and <draw:measure> 10.3.12.

The `svg:x1` attribute has the data type `coordinate` 18.3.10.

### 19.578.2 <svg:linearGradient>

See §13.3.2 of [SVG].

For a <svg:linearGradient> 16.42.2 element the default value for this attribute is 0%.

The `svg:x1` attribute is usable with the following element: <svg:linearGradient> 16.42.2.

The values of the `svg:x1` attribute are a value of type `coordinate` 18.3.10 or a value of type `percent` 18.3.24.

## 19.579 `svg:x2`

### 19.579.1 <draw:connector>, <draw:line> and <draw:measure>

See §9.5 of [SVG].

The `svg:x2` attribute may be omitted on `<draw:connector>` elements when the connector is connected to a shape by a `draw:end-shape` 19.144 attribute. The end position then is the position of the glue point to which the connector is connected.

The `svg:x2` attribute is usable with the following elements: `<draw:connector>` 10.3.10, `<draw:line>` 10.3.3 and `<draw:measure>` 10.3.12.

The `svg:x2` attribute has the data type `coordinate` 18.3.10.

## 19.579.2 `<svg:linearGradient>`

See §13.3.2 of [SVG].

For a `<svg:linearGradient>` 16.42.2 element the default value for this attribute is 100%.

The `svg:x2` attribute is usable with the following element: `<svg:linearGradient>` 16.42.2.

The values of the `svg:x2` attribute are a value of type `coordinate` 18.3.10 or a value of type `percent` 18.3.24.

## 19.580 `svg:x-height`

See §20.8.3 of [SVG].

The `svg:x-height` attribute is usable with the following element: `<style:font-face>` 16.23.

The `svg:x-height` attribute has the data type `integer` 18.2.

## 19.581 `svg:y`

### 19.581.1 `<chart:coordinate-region>`

The `svg:y` attribute specifies the vertical distance of the upper left corner of the `<chart:coordinate-region>` element from the upper left-corner of the `<chart:chart>` element.

The `svg:y` attribute is usable with the following element: `<chart:coordinate-region>` 11.3.

The `svg:y` attribute has the data type `coordinate` 18.3.10.

### 19.581.2 `<chart:data-label>`, `<chart:equation>`, `<chart:footer>`, `<chart:legend>`, `<chart:plot-area>`, `<chart:subtitle>`, `<chart:title>`

The `svg:y` attribute specifies the vertical distance of the upper-left corner of the given element from the upper-left corner of the `<chart:chart>` element.

The `svg:y` attribute is usable with the following elements: `<chart:data-label>` 11.15, `<chart:equation>` 11.19, `<chart:footer>` 11.2.3, `<chart:legend>` 11.4, `<chart:plot-area>` 11.5, `<chart:subtitle>` 11.2.2 and `<chart:title>` 11.2.1.

The `svg:y` attribute has the data type `coordinate` 18.3.10.

### 19.581.3 `<draw:area-polygon>`

The `svg:y` attribute defines the y-axis coordinate of a polygonal image map area's bounding box.

The `svg:y` attribute is usable with the following element: `<draw:area-polygon>` 10.4.13.5.

The `svg:y` attribute has the data type `coordinate` 18.3.10.

#### 19.581.4 `<draw:area-rectangle>`

The `svg:y` attribute defines the y-axis coordinate of a rectangular image map area.

The `svg:y` attribute is usable with the following element: `<draw:area-rectangle>` 10.4.13.3.

The `svg:y` attribute has the data type `coordinate` 18.3.10.

#### 19.581.5 `<dr3d:scene>`

The `svg:y` attribute specifies the vertical position of the top edge of the viewport.

The `svg:y` attribute is usable with the following element: `<dr3d:scene>` 10.5.2.

The `svg:y` attribute has the data type `coordinate` 18.3.10.

#### 19.581.6 `<draw:caption>, <draw:custom-shape>, <draw:circle>, <draw:control>, <draw:ellipse>, <draw:frame>, <draw:g>, <draw:rect>, <draw:page-thumbnail>, <draw:path>, <draw:polygon>, <draw:polyline>, <draw:regular-polygon>`

See §5.1.2 of [SVG]. For drawing shapes that have a non-rectangular shape, the coordinate refers to the drawing shape's bounding box.

The `svg:y` attribute is usable with the following elements: `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:g>` 10.3.15, `<draw:page-thumbnail>` 10.3.14, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2 and `<draw:regular-polygon>` 10.3.6.

The `svg:y` attribute has the data type `coordinate` 18.3.10.

#### 19.581.7 `<draw:glue-point>`

The `svg:y` attribute specifies the vertical position of the glue point. The coordinate is either a percentage value relative to the drawing shape's center measured vertically, or, if the `draw:align` 19.111 attribute is also specified, an absolute distance value relative to the edge specified by the `draw:align` attribute.

The `svg:y` attribute is usable with the following element: `<draw:glue-point>` 10.3.16.

The values of the `svg:y` attribute are a value of type `distance` 18.3.15 or a value of type `percent` 18.3.24.

#### 19.581.8 `<office:annotation>`

The `svg:y` attribute specifies the position where the annotation is displayed. Its meaning is the same as for a `<draw:caption>` element. See 19.577.6.

The `svg:y` attribute is usable with the following element: `<office:annotation>` 14.1.

The `svg:y` attribute has the data type `coordinate` 18.3.10.

#### 19.581.9 `<presentation:placeholder>`

See 19.581.6. Percentage values are relative to the height of the drawing page.

The `svg:y` attribute is usable with the following element: `<presentation:placeholder>` 16.44.

The values of the `svg:y` attribute are a value of type `coordinate` 18.3.10 or a value of type `percent` 18.3.24.

## 19.582 `svg:y1`

### 19.582.1 `<draw:connector>`, `<draw:line>` and `<draw:measure>`

See §9.5 of [SVG].

The `svg:y1` attribute may be omitted on `<draw:connector>` elements when the connector is connected to a shape by a `draw:start-shape` 19.217 attribute. The start position then is the position of the glue point to which the connector is connected.

The `svg:y1` attribute is usable with the following elements: `<draw:connector>` 10.3.10, `<draw:line>` 10.3.3 and `<draw:measure>` 10.3.12.

The `svg:y1` attribute has the data type `coordinate` 18.3.10.

### 19.582.2 `<svg:linearGradient>`

See §13.3.2 of [SVG].

For a `<svg:linearGradient>` 16.42.2 element the default value for this attribute is 0%.

The `svg:y1` attribute is usable with the following element: `<svg:linearGradient>` 16.42.2.

The values of the `svg:y1` attribute are a value of type `coordinate` 18.3.10 or a value of type `percent` 18.3.24.

## 19.583 `svg:y2`

### 19.583.1 `<draw:connector>`, `<draw:line>` and `<draw:measure>`

See §9.5 of [SVG].

The `svg:y2` attribute may be omitted on `<draw:connector>` elements when the connector is connected to a shape by a `draw:end-shape` 19.144 attribute. The end position then is the position of the glue point to which the connector is connected.

The `svg:y2` attribute is usable with the following elements: `<draw:connector>` 10.3.10, `<draw:line>` 10.3.3 and `<draw:measure>` 10.3.12.

The `svg:y2` attribute has the data type `coordinate` 18.3.10.

### 19.583.2 `<svg:linearGradient>`

See §13.3.2 of [SVG].

For a `<svg:linearGradient>` 16.42.2 element the default value for this attribute is 100%.

The `svg:y2` attribute is usable with the following element: `<svg:linearGradient>` 16.42.2.

The values of the `svg:y2` attribute are a value of type `coordinate` 18.3.10 or a value of type `percent` 18.3.24.

## 19.584 table:acceptance-state

The `table:acceptance-state` attribute specifies whether a tracked change has been accepted or rejected already, or whether an acceptance or rejection is still pending.

The defined values for the `table:acceptance-state` attribute are:

- `accepted`: a tracked change has been accepted.
- `pending`: a tracked change is pending approval or rejection.
- `rejected`: a tracked change has been rejected.

The default value for this attribute is `pending`.

The `table:acceptance-state` attribute is usable with the following elements:

`<table:cell-content-change>` 9.9.17, `<table:deletion>` 9.9.9,

`<table:insertion>` 9.9.3 and `<table:movement>` 9.9.13.

The values of the `table:acceptance-state` attribute are `accepted`, `rejected` or `pending`.

## 19.585 table:algorithm

The `table:algorithm` attribute specifies the algorithm used to compare sort keys. To avoid name conflicts between different consumers, the name of the algorithm should begin with a namespace prefix, followed by a ":" (U+003A, COLON) separator.

The `table:algorithm` attribute is usable with the following element: `<table:sort>` 9.4.19.

The `table:algorithm` attribute has the data type `string` 18.2.

## 19.586 table:add-empty-lines

The `table:add-empty-lines` attribute specifies whether an empty row is inserted in the data pilot table after the data (including the subtotals) for each member of the field.

The defined values for the `table:add-empty-lines` attribute are:

- `false`: an empty line should not be inserted after the data for a member, including any subtotal of that member.
- `true`: an empty line should not be inserted after the data for a member, including any subtotal of that member.

The `table:add-empty-lines` attribute is usable with the following element:

`<table:data-pilot-layout-info>` 9.6.15.

The `table:add-empty-lines` attribute has the data type `boolean` 18.3.3.

## 19.587 table:allow-empty-cell

The `table:allow-empty-cell` attribute specifies whether a cell can be empty.

The defined values for the `table:allow-empty-cell` attribute are:

- `false`: cell cannot be empty.
- `true`: cell can be empty.

The default value for this attribute is `true`.

The `table:allow-empty-cell` attribute is usable with the following element:

`<table:content-validation>` 9.4.5.

The `table:allow-empty-cell` attribute has the data type `boolean` 18.3.3.

## 19.588 `table:application-data`

The `table:application-data` attribute specifies extra information about a data pilot table, which can be used by a consumer. This data shall not influence the behavior of the data pilot.

The `table:application-data` attribute is usable with the following element:  
`<table:application-data>` 9.6.3.

The `table:application-data` attribute has the data type `string` 18.2.

## 19.589 `table:automatic-find-labels`

The `table:automatic-find-labels` attribute specifies whether a consumer should attempt to find labels of rows and columns.

The defined values for the `table:automatic-find-labels` attribute are:

- `false`: consumers should not attempt to find labels of rows and columns.
- `true`: consumers should attempt find labels of rows and columns.

**Note:** The `table:automatic-find-labels` attribute enables the use of the automatic lookup of labels capability defined by OpenFormula.  
OpenFormula, 5.10.2. (\*update\*)

The default value for this attribute is `true`.

The `table:automatic-find-labels` attribute is usable with the following element:  
`<table:calculation-settings>` 9.4.1.

The `table:automatic-find-labels` attribute has the data type `boolean` 18.3.3.

## 19.590 `table:base-cell-address`

The `table:base-cell-address` attribute specifies the address of the base cell for relative addresses in formulas that occur within a condition. This attribute is only necessary when the condition contains a formula. The value of this attribute shall be an absolute cell address that contains a table name. The dollar signs "\$" (U+0024, DOLLAR SIGN) that indicate an absolute address may be omitted.

The `table:base-cell-address` attribute is usable with the following elements:  
`<table:content-validation>` 9.4.5, `<table:named-expression>` 9.4.13 and  
`<table:named-range>` 9.4.12.

The `table:base-cell-address` attribute has the data type `cellAddress` 18.3.4.

## 19.591 `table:bind-styles-to-content`

The `table:bind-styles-to-content` attribute specifies whether cells retain their style after a subtotal calculation. This attribute is only evaluated if a `<table:sort-groups>` element is present.

The defined values for the `table:bind-styles-to-content` attribute are:

- `false`: cells need not retain their style after a subtotal calculation.
- `true`: cells should retain their style after a subtotal calculation.

The default value for this attribute is `true`.

The `table:bind-styles-to-content` attribute is usable with the following elements:  
`<table:sort>` 9.4.19 and `<table:subtotal-rules>` 9.4.21.

The `table:bind-styles-to-content` attribute has the data type `boolean` 18.3.3.

## 19.592 `table:border-color`

The `table:border-color` attribute specifies the color of a border that is displayed around a scenario that belongs to a scenario table.

The `table:border-color` attribute is usable with the following element:  
`<table:scenario>` 9.2.7.

The `table:border-color` attribute has the data type `color` 18.3.9.

## 19.593 `table:buttons`

The `table:buttons` attribute specifies the cells in a data pilot table that should be displayed as buttons to trigger interactive operations on the table such as changing the order of columns. Its value is a list of white-space-separated cell-addresses. The value of this attribute shall be an absolute cell address that contains a table name. The dollar signs "\$" (U+0024, DOLLAR SIGN) that indicate an absolute address may be omitted.

The `table:buttons` attribute is usable with the following element: `<table:data-pilot-table>` 9.6.3.

The `table:buttons` attribute has the data type `cellRangeAddressList` 18.3.6.

## 19.594 `table:case-sensitive`

The `table:case-sensitive` attribute specifies whether to distinguish between upper and lower case when comparing, sorting or filtering content.

That attribute is only evaluated if the operations take place on strings.

The defined values for the `table:case-sensitive` attribute are:

- `false`: upper and lower case are not distinguished when comparing, sorting or filtering content.
- `true`: upper and lower case are distinguished when comparing, sorting or filtering content.

For a `<table:calculation-settings>` 9.4.1 element the default value for this attribute is `true`.

For `<table:filter-condition>` 9.5.5, `<table:sort>` 9.4.19 and `<table:subtotal-rules>` 9.4.21 elements the default value for this attribute is `false`.

The `table:case-sensitive` attribute is usable with the following elements:  
`<table:calculation-settings>` 9.4.1, `<table:filter-condition>` 9.5.5,  
`<table:sort>` 9.4.19 and `<table:subtotal-rules>` 9.4.21.

The `table:case-sensitive` attribute has the data type `boolean` 18.3.3.

## 19.595 `table:cell-address`

The `table:cell-address` attribute specifies the original address of a cell used in a calculation.

The `table:cell-address` attribute is usable with the following element:  
`<table:change-track-table-cell>` 9.9.16.

The `table:cell-address` attribute has the data type `cellAddress` 18.3.4.

## 19.596 table:cell-range

The `table:cell-range` attribute specifies a list of ranges of cells.

The `table:cell-range` attribute is usable with the following elements: `<chart:footer>` 11.2.3, `<chart:subtitle>` 11.2.2 and `<chart:title>` 11.2.1.

The `table:cell-range` attribute has the data type `cellRangeAddressList` 18.3.6.

## 19.597 table:cell-range-address

### 19.597.1 General

The `table:cell-range-address` attribute specifies a range of cells or a list of ranges of cells.

### 19.597.2 <chart:categories>

The `table:cell-range-address` attribute specifies the source from which the category labels are taken. If this attribute or the `<chart:categories>` element is omitted the `chart:data-source-has-labels` 19.17 attribute of the `<chart:plot-area>` 11.5 element should be evaluated.

The `table:cell-range-address` attribute is usable with the following element: `<chart:categories>` 11.10.

The `table:cell-range-address` attribute has the data type `cellRangeAddressList` 18.3.6.

### 19.597.3 <chart:domain> - bubble

The `table:cell-range-address` attribute specifies the x and y-coordinate values for bubble charts when the `chart:class` 11.5 attribute of its parent `<chart:series>` 11.12 element has the value `chart:bubble`.

The `table:cell-range-address` attribute is usable with the following element: `<chart:domain>` 11.13.

The `table:cell-range-address` attribute has the data type `cellRangeAddressList` 18.3.6.

### 19.597.4 <chart:domain>- scatter

The `table:cell-range-address` attribute specifies the x-coordinate values for scatter charts when the `chart:class` 19.15 attribute of its parent `<chart:series>` 11.12 element has the value `chart:scatter`.

The `table:cell-range-address` attribute is usable with the following element: `<chart:domain>` 11.13.

The `table:cell-range-address` attribute has the data type `cellRangeAddressList` 18.3.6.

### 19.597.5 <chart:domain> - surface

The `table:cell-range-address` attribute of the first `<chart:domain>` element specifies the y-coordinate values and the `table:cell-range-address` attribute of the second `<chart:domain>` element specifies the x-coordinate values for surface charts when the `chart:class` 19.15 attribute of its parent `<chart:series>` 11.12 element has the value `chart:surface` and the `chart:values-cell-range-address` 19.28 attribute of this

`<chart:series>` element specifies a range with more than one row and more than one column.

The `table:cell-range-address` attribute is usable with the following element:  
`<chart:domain>` 11.13.

The `table:cell-range-address` attribute has the data type `cellRangeAddressList` 18.3.6.

## 19.597.6 `<chart:plot-area>` (removed)

## 19.597.7 `<table:highlighted-range>`

The `table:cell-range-address` attribute specifies the address of a range that is highlighted.

The `table:cell-range-address` attribute is usable with the following element:  
`<table:highlighted-range>` 9.3.4.

The `table:cell-range-address` attribute has the data type `cellRangeAddress` 18.3.5.

## 19.597.8 `<table:named-range>`

The `table:cell-range-address` attribute specifies the address of a named range. The address can be either absolute or relative. If the cell range address is relative, the parent `<table:named-range>` element shall have a `table:base-cell-address` 19.590 attribute with a value. A relative cell address defines an offset which shall be interpreted relative to the cell where the named range is used. The offset is the same as the offsets between the table cell addressed by the relative cell range address and the table cell specified by the `table:base-cell-address` attribute.

The `table:cell-range-address` attribute is usable with the following element:  
`<table:named-range>` 9.4.12.

The `table:cell-range-address` attribute has the data type `cellRangeAddress` 18.3.5.

## 19.597.9 `<table:source-cell-range>`

The `table:cell-range-address` attribute specifies the cell range containing the source data. The value of this attribute shall be an absolute cell address that contains a table name. The dollar signs "\$" (U+0024, DOLLAR SIGN) that indicate an absolute address may be omitted. If a `table:name` 19.677 attribute on an element specifies a named range, the cell range specified by the `table:cell-range` 19.596 attribute on the same element, shall match the cell range specified by the named range.

The `table:cell-range-address` attribute is usable with the following element:  
`<table:source-cell-range>` 9.6.5.

The `table:cell-range-address` attribute has the data type `cellRangeAddress` 18.3.5.

The `table:cell-range-address` attribute has the data type `cellRangeAddress` 18.3.5.

## 19.598 `table:column`

The `table:column` attribute specifies the column number of a cell.

The `table:column` attribute is usable with the following elements: `<table:cell-address>` 9.9.18, `<table:source-range-address>` 9.9.14 and `<table:target-range-address>` 9.9.15.

The `table:column` attribute has the data type `integer` 18.2.

## 19.599 `table:comment`

The `table:comment` attribute specifies a comment about a scenario.

The `table:comment` attribute is usable with the following element: `<table:scenario>` 9.2.7.

The `table:comment` attribute has the data type `string` 18.2.

## 19.600 `table:condition`

The `table:condition` attribute specifies a condition that is used to validate the value of a table cell. When evaluation of the condition results in false, any action taken is determined by the other attributes and child elements of the `<table:content-validation>` element.

**Note:** Possible actions include presentation of a message, triggering an event, and performance of a script, with or without invalidating the value. The `table:allow-empty-cell` 19.587 attribute value influences whether the `table:condition` is applicable.

The value of this attribute should be a namespace prefix, followed by a ":" (U+003A, COLON), followed by Boolean expression. If the namespace prefix is missing the namespace defaults to the "urn:oasis:names:tc:opendocument:xmlns:of:1.2" namespace. The XML namespace that applies to the `table:condition` attribute specifies the syntax and semantics of the `expression` occurrences in the `table:condition` syntax.

**Note:** The value of the `table:base-cell-address` 19.590 attribute of the `<table:content-validation>` 9.4.5 element can influence how references are resolved in an `expression` anywhere within the defined conditions.

An OpenDocument Consumer when hosting a formula evaluator makes available the host-dependent properties defined by Table 17 Host-dependent Properties in 19.646.

The defined conditions are:

- `cell-content-text-length() op value`, where `op` is one of the relational operators ""<,">,"<=",">=","=" or "!=": true if the length of the cell's text compared to the value specified in the condition by `value` using the relational operator `op` evaluates to true.
- `cell-content-text-length-is-between(value1, value2)`: true if the length of the cell's text is between `value1` and `value2`.
- `cell-content-text-length-is-not-between(value1, value2)`: true if the length of the cell's text is not between `value1` and `value2`.
- `cell-content-is-in-list(list)`, where `list` is one or more string entries, separated by ";" (U+003B, SEMICOLON), or an `expression`: true if the cell's content is in `list`.
- `cell-content-is-decimal-number()` and `condition`, where `condition` is one of the value conditions below: true if the cell has a numeric value and `condition` is true.
- `cell-content-is-whole-number()` and `condition`, where `condition` is one of the value conditions below: true if the cell's cell has a whole-number value and `condition` is true.

- `cell-content-is-date()` and `condition`, where `condition` is one of the value conditions below: true if the cell has a date value and `condition` is true.
- `cell-content-is-time()` and `condition`, where `condition` is one of the value conditions below: true if the cell has a time value and `condition` is true.
- `is-true-formula(expression)`: true if evaluation of the `expression` yields a value that converts to logical type value true in the semantics for the `expression`; false otherwise.

The defined value conditions are:

- `cell-content() op value`, where `op` is one of the relational operators "`<`", "`>`", "`<=`", "`>=`", "`=`" or "`!=`": true if the cell's value compared to the value specified in the condition by `value` using the relational operator `op` evaluates to true.
- `cell-content-is-between(value1, value2)`: true if the cell's value is between `value1` and `value2`.
- `cell-content-is-not-between(value1, value2)`: true if the cell's value is not between `value1` and `value2`.
- `is-true-formula(expression)`: true if evaluation of the `expression` yields a value that converts to logical value true in the semantics for the `expression`; false otherwise.

Within the conditions:

- `value`, `value1` and `value2` are a `numberValue`, a `string` or an `expression`.
- `numberValue` is a whole or decimal number in the lexical form of the [xmlschema-2] decimal datatype.
- `string` expresses a value of the [xmlschema-2] string datatype by surrounding the sequence of string characters in quotation marks (U+0022, QUOTATION MARK). A quotation mark within the string itself is expressed with two consecutive quotation marks.
- `expression` is a lexical form that is neither `numberValue` nor `string` and that is a well-formed expression determined by the namespace applicable to the `table:condition` value.

The `table:condition` attribute is usable with the following element: `<table:content-validation> 9.4.5.`

The `table:condition` attribute has the data type `string 18.2`.

## 19.601 table:condition-source

The `table:condition-source` attribute specifies whether the condition is contained in a filter or encoded in a cell range.

The defined values for the `table:condition-source` attribute are:

- `cell-range`: condition encoded into the cell range specified by a `table:condition-source-range-address` 19.602 attribute.
- `self`: condition specified by the `<table:filter>` element's child elements.

The default value for this attribute is `self`.

The `table:condition-source` attribute is usable with the following element: `<table:filter> 9.5.2.`

The values of the `table:condition-source` attribute are `self` or `cell-range`.

## 19.602 table:condition-source-range-address

The `table:condition-source-range-address` attribute specifies a cell range that contains encoded conditions. The first row of the cell range shall contain the labels of the columns whose content should be filtered. The following rows contain conditions that have to evaluate to true for the cells contained in the columns. The conditions in each row are connected by an “and” operation, while the rows are connected by an “or” operation. This means that a row is of the source table is displayed if there is at least one row in the condition range where all conditions evaluate to true if they are applied to the columns specified in the first row of the condition range.

The `table:condition-source-range-address` attribute is usable with the following element: `<table:filter>` 9.5.2.

The `table:condition-source-range-address` attribute has the data type `cellRangeAddress` 18.3.5.

## 19.603 table:contains-error

The `table:contains-error` attribute specifies whether a cell range contains an error at the point of evaluation.

The defined values for the `table:contains-error` attribute are:

- `false`: cell range contains no error at point of evaluation.
- `true`: cell range contains an error at point of evaluation.

The default value for this attribute is `false`.

The `table:contains-error` attribute is usable with the following element: `<table:highlighted-range>` 9.3.4.

The `table:contains-error` attribute has the data type `boolean` 18.3.3.

## 19.604 table:contains-header

The `table:contains-header` attribute specifies whether the content of a database range's first row or column defines labels which may be used to reference a whole row or column.

The defined values for the `table:contains-header` attribute are:

- `false`: the first row or column of the database range does not define labels.
- `true`: the first row or column of the database range defines labels.

The default value for this attribute is `true`.

The `table:contains-header` attribute is usable with the following element: `<table:database-range>` 9.4.15.

The `table:contains-header` attribute has the data type `boolean` 18.3.3.

## 19.605 table:content-validation-name

The `table:content-validation-name` attribute specifies the name of a validity check.

The value of this attribute is a name found in the `table:name` 19.677 attribute of a `<table:content-validation>` 9.4.5 element. If this attribute is not present, a cell may have arbitrary content.

The `table:content-validation-name` attribute is usable with the following elements: `<table:covered-table-cell>` 9.1.5 and `<table:table-cell>` 9.1.4.

The `table:content-validation-name` attribute has the data type `string` 18.2.

## 19.606 table:copy-back

The `table:copy-back` attribute specifies whether data is copied back into a scenario table if another scenario is activated.

The defined values for the `table:copy-back` attribute are:

- `false`: data is not copied back into a scenario table if another scenario is activated.
- `true`: data copied back into a scenario table if another scenario is activated.

The default value for this attribute is `true`.

The `table:copy-back` attribute is usable with the following element: `<table:scenario>` 9.2.7.

The `table:copy-back` attribute has the data type `boolean` 18.3.3.

## 19.607 table:copy-formulas

The `table:copy-formulas` attribute specifies whether formulas are copied from a scenario table to a destination table.

The defined values for the `table:copy-formulas` attribute are:

- `false`: only values resulting from formulas are copied.
- `true`: formulas are copied.

The default value for this attribute is `true`.

The `table:copy-formulas` attribute is usable with the following element: `<table:scenario>` 9.2.7.

The `table:copy-formulas` attribute has the data type `boolean` 18.3.3.

## 19.608 table:copy-styles

The `table:copy-styles` attribute specifies whether styles are copied from a scenario table to a destination table together with the data.

The defined values for the `table:copy-styles` attribute are:

- `false`: styles are not copied with data.
- `true`: styles are copied with data.

The default value for this attribute is `true`.

The `table:copy-styles` attribute is usable with the following element: `<table:scenario>` 9.2.7.

The `table:copy-styles` attribute has the data type `boolean` 18.3.3.

## 19.609 table:count

The `table:count` attribute specifies the count of inserted rows, columns or tables.

The default value for this attribute is 1.

The `table:count` attribute is usable with the following element: `<table:insertion>` 9.9.3.

The `table:count` attribute has the data type `positiveInteger` 18.2.

## 19.610 table:country

The `table:country` attribute specifies the country information for the natural language in which comparisons will occur.

The `table:country` attribute is usable with the following element: `<table:sort>` 9.4.19.

The `table:country` attribute has the data type `countryCode` 18.3.11.

## 19.611 table:data-field

The `table:data-field` attribute specifies the data field whose values are taken into account.

The `table:data-field` attribute is usable with the following elements: `<table:pilot-display-info>` 9.6.13 and `<table:pilot-sort-info>` 9.6.14.

The `table:data-field` attribute has the data type `string` 18.2.

## 19.612 table:data-cell-range-address

The `table:data-cell-range-address` attribute specifies the cell range address of data.

The `table:data-cell-range-address` attribute is usable with the following element: `<table:label-range>` 9.4.9.

The `table:data-cell-range-address` attribute has the data type `cellRangeAddress` 18.3.5.

## 19.613 table:database-name

A `table:database-name` attribute specifies the name of an SQL database for the importation of data.

The `table:database-name` attribute is usable with the following elements: `<table:database-source-query>` 9.4.18, `<table:database-source-sql>` 9.4.16 and `<table:database-source-table>` 9.4.17.

The `table:database-name` attribute has the data type `string` 18.2.

## 19.614 table:database-table-name

A `table:database-table-name` attribute specifies a database table for the importation of data.

The `table:database-table-name` attribute is usable with the following element: `<table:database-source-table>` 9.4.17.

The `table:database-table-name` attribute has the data type `string` 18.2.

## 19.615 table:data-type

### 19.615.1 General

The `table:data-type` attribute specifies a data type for tables.

### 19.615.2 `<table:filter-condition>`

The `table:data-type` attribute specifies whether a comparison shall take place as text, numeric values, or as color.

The defined values for the `table:data-type` attribute are:

- **number**: comparison as numeric values.
- **text**: comparison as text values.
- **text-color**: comparison based on the text color. The value "window-font-color" specifies the default text color, as specified in `style:use-window-font-color 20.395`. Text color assigned by data styles 16.29 is not considered.
- **data-style-color**: comparison based on the text color applied by a data style 16.29. The value "window-font-color" specifies the default text color, as specified in `style:use-window-font-color 20.395`.
- **background-color**: comparison based on the cell background color, as determined by the applicable `<style:table-cell-properties> 17.18` element.

A `<table:filter-condition> 9.5.5` element that is a descendant of `<table: data-pilot-table> 9.6.3` and has a `table:data-type` attribute with value `text-color`, `data-style-color` or `background-color` is ignored.

For a `<table:filter-condition> 9.5.5` element the default value for this attribute is `text`.

The `table:data-type` attribute is usable with the following element: `<table:filter-condition> 9.5.5`.

The values of the `table:data-type` attribute are `text` or `number`.

The values of the `table:data-type` attribute are `text-color` or `data-style-color`.

The only value of the `table:data-type` attribute is `background-color`.

### 19.615.3 `<table:sort-by>`

The `table:data-type` attribute specifies the data type of a field to be sorted.

The defined values for the `table:data-type` attribute are:

- `automatic`: consumer determines the type of data in a field.
- `number`: numeric data type.
- `text`: text data type.
- a value of type `string` (name of a user-defined sort order, implementation-dependent)

For a `<table:sort-by> 9.4.20` element the default value for this attribute is `automatic`.

The `table:data-type` attribute is usable with the following element: `<table:sort-by> 9.4.20`.

The values of the `table:data-type` attribute are `text`, `number`, `automatic` or a value of type `string` 18.2.

### 19.615.4 `<table:sort-groups>`

The `table:data-type` attribute specifies the data type of a field to be sorted.

If the attribute value is `automatic`, the consumer shall determine what type of data is in the field. User-defined sort orders are implementation specific.

The defined values for the `table:data-type` attribute are:

- `automatic`: consumer determines the type of data in a field.
- `number`: numeric data type.
- `text`: text data type.

- a value of type `string` (name of a user-defined sort order, implementation-dependent)

For a `<table:sort-groups>` 9.4.22 element the default value for this attribute is `automatic`.

The `table:data-type` attribute is usable with the following element: `<table:sort-groups>` 9.4.22.

The values of the `table:data-type` attribute are `text`, `number`, `automatic` or a value of type `string` 18.2.

## 19.616 `table:date-end`

The `table:date-end` attribute specifies the end value for a grouping of date values.

All values that are higher than the end value are contained in a single group, while values that are equal to or lower than the end value are grouped as specified by the `table:grouped-by` 19.650 and `table:step` 19.650 attributes.

The defined values for the `table:date-end` attribute are:

- `auto`: the highest value of the field is taken as the end value.
- a value of type `dateOrDateTime` 18.3.14.

The `table:date-end` attribute is usable with the following element: `<table: data-pilot-groups>` 9.6.17.

The values of the `table:date-end` attribute are a value of type `dateOrDateTime` 18.3.14 or `auto`.

## 19.617 `table:date-start`

The `table:date-start` attribute specifies the start value for a grouping of date values.

The defined values for the `table:date-start` attribute are:

- `auto`: the lowest value of the field is used as the start value.
- a value of type `dateOrDateTime`: 18.3.14

The `table:date-start` attribute is usable with the following element: `<table: data-pilot-groups>` 9.6.17.

The values of the `table:date-start` attribute are a value of type `dateOrDateTime` 18.3.14 or `auto`.

## 19.618 `table:date-value`

The `table:date-value` attribute specifies the null date. The null date is the date that results in the value “0” if a date value is converted into a numeric value.

The default value for this attribute is `1899-12-30`.

The `table:date-value` attribute is usable with the following element: `<table: null-date>` 9.4.2.

The `table:date-value` attribute has the data type `date` 18.2.

## 19.619 `table:default-cell-style-name`

The `table:default-cell-style-name` attribute specifies a default cell style. Cells defined by a `<table:table-cell>` 18.3.14 element that do not have a `table:style-name` 19.730 attribute value use the specified default cell style.

If an individual cell has a default style specified by a `table:default-cell-style-name` attribute on a `<table:table-column>` element and by a `style:default-cell-style-name` on a `<table:table-row>` element, the default style specified by the `<table:table-row>` element shall be applied to the cell and the default style specified by the `<table:table-column>` element shall be ignored.

The `table:default-cell-style-name` attribute is usable with the following elements: `<table:table-column>` 9.1.6 and `<table:table-row>` 9.1.3.

The `table:default-cell-style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.620 table:direction

The `table:direction` attribute specifies the direction of a relation between a cell and a highlighted range.

The defined values for the `table:direction` attribute are:

- `from-another-table`: relationship begins at another table.
- `from-same-table`: relationship begins at the same table.
- `to-another-table`: relationship ends at another table.

The `table:direction` attribute is usable with the following element: `<table:highlighted-range>` 9.3.4.

The values of the `table:direction` attribute are `from-another-table`, `to-another-table` or `from-same-table`.

## 19.621 table:display

The `table:display` attribute specifies whether a component of a table is displayed.

The defined values for the `table:display` attribute are:

- `false`: component is not displayed.
- `true`: component is displayed.

Components of a table for the `table:display` attribute are: `<table:data-pilot-member>` 9.6.12, `<table:error-message>` 9.4.7, `<table:help-message>` 9.4.6, `<table:table-column-group>` 9.1.10 and `<table:table-row-group>` 9.1.9.

For `<table:table-column-group>` 9.1.10 and `<table:table-row-group>` 9.1.9 elements the default value for this attribute is `true`.

For `<table:error-message>` 9.4.7 and `<table:help-message>` 9.4.6 elements the default value for this attribute is `false`.

For `<table:data-pilot-member>` 9.6.12 the default value for this attribute is `true`.

The `table:display` attribute is usable with the following elements: `<table:data-pilot-member>` 9.6.12, `<table:error-message>` 9.4.7, `<table:help-message>` 9.4.6, `<table:table-column-group>` 9.1.10 and `<table:table-row-group>` 9.1.9.

The `table:display` attribute has the data type `boolean` 18.3.3.

## 19.622 table:display-border

The `table:display-border` attribute specifies whether to display a border around a scenario that belongs to a scenario table.

The defined values for the `table:display-border` attribute are:

- `false`: border is not displayed.
- `true`: border is displayed.

The default value for this attribute is `true`.

The `table:display-border` attribute is usable with the following element:  
`<table:scenario>` 9.2.7.

The `table:display-border` attribute has the data type `boolean` 18.3.3.

## 19.623 table:display-duplicates

The `table:display-duplicates` attribute specifies whether to display duplicate matches in a result.

The defined values for the `table:display-duplicates` attribute are:

- `false`: duplicate matches not displayed in results.
- `true`: duplicate matches displayed in results.

The default value for this attribute is `true`.

The `table:display-duplicates` attribute is usable with the following element:  
`<table:filter>` 9.5.2.

The `table:display-duplicates` attribute has the data type `boolean` 18.3.3.

## 19.624 table:display-filter-buttons

The `table:display-filter-buttons` attribute specifies whether to display filter buttons. Filter buttons are list box controls displayed in the label cells whose list entries are the values that exist in the labeled row or column. Selecting one of these entries is the equivalent of applying a filter to the database range that selects all row or columns where the cells in the labeled row or column have the selected value.

The defined values for the `table:display-filter-buttons` attribute are:

- `false`: filter buttons are not displayed.
- `true`: filter buttons are displayed.

The default value for this attribute is `false`.

The `table:display-filter-buttons` attribute is usable with the following element:  
`<table:database-range>` 9.4.15.

The `table:display-filter-buttons` attribute has the data type `boolean` 18.3.3.

## 19.625 table:display-list

The `table:display-list` attribute specifies whether a list of values that occurs within a condition is displayed when a user is entering a cell value.

The defined values for the `table:display-list` attribute are:

- `none`: the list values are not displayed.
- `sort-ascending`: the list values are displayed in ascending order.
- `unsorted`: the list values are displayed in the order they occur in the condition.

The default value for this attribute is `unsorted`.

The `table:display-list` attribute is usable with the following element:  
`<table:content-validation>` 9.4.5.

The values of the `table:display-list` attribute are `none`, `unsorted` or `sort-ascending`.

## 19.626 `table:display-member-mode`

The `table:display-member-mode` attribute specifies whether the value count specified by `table:member-count` 19.730 attribute should be taken from the top or from the bottom of a data field's column.

The defined values for the `table:display-member-mode` attribute are:

- `from-bottom`: the value count is taken from the bottom of a data field's column.
- `From-top`: the value count is taken from the top of a data field's column.

The `table:display-member-mode` attribute is usable with the following element:  
`<table:data-pilot-display-info>` 9.6.13.

The values of the `table:display-member-mode` attribute are `from-top` or `from-bottom`.

## 19.627 `table:drill-down-on-double-click`

The `table:drill-down-on-double-click` attribute specifies how the data pilot table reacts to a double-click 19.406 in the data pilot table. A double click other than as specified in a data pilot table has no effect.

The defined values for the `table:drill-down-on-double-click` attribute are:

- `false`: a double click on a member label or the empty area next to it starts the edit mode of the table cell, like for cells outside of the data pilot table.

**Note:** This can be used to rename group fields or members.

- `true`: a double click on a member label or the empty area next to it shows or hides details for that member.

The default value for this attribute is `true`.

The `table:drill-down-on-double-click` attribute is usable with the following element:  
`<table:data-pilot-table>` 9.6.3.

The `table:drill-down-on-double-click` attribute has the data type `boolean` 18.3.3.

## 19.628 `table:embedded-number-behavior`

The `table:embedded-number-behavior` attribute specifies how string values that contain digits are sorted. If the value is `alpha-numeric`, string comparison as specified by the other attributes of `<table:sort>` element is used for sorting.

The defined values for the `table:embedded-number-behavior` attribute are:

- `alpha-numeric`: other attributes are used to specify the string comparison for sorting.
- `double`: string values are compared as defined below
- `integer`: string values are compared as defined below

**Note:** If the attribute value is `integer` or `double`, string-prefixed numbers will be sorted in a "natural", number-aware way, i.e. A1, A2, A3, ..., A19, A20, instead of the normal, alpha-numeric behavior, i.e. A1, A10, A11, A12, ..., A19, A2, A20, A3, A4, ..., A8, A9.

The following illustrates how two strings shall be compared if the attribute value is `integer` or `double`.

In the following, the term "alpha-numeric comparison" means the string comparison that would be used if the `table:embedded-number-behavior` attribute had the value `alpha-numeric`.

Step 1: The two strings are compared by using the alpha-numeric comparison to test whether they are equal. If they are equal, the comparison will stop immediately returning an equality as result.

Step 2: The prefix substrings of the two strings are determined by locating the first occurrence of a digit character in the two strings; the substrings from the very first characters through the characters preceding the first digits are the prefix substrings. If a string starts with a digit, the prefix substring of this string is empty. If there is no digit in either one of the compared strings, the natural sort process will end and the alpha-numeric comparison will be performed instead.

Step 3: After the prefix substrings have been determined for both strings, an alpha-numeric comparison is performed on the two prefix substrings. If they differ, the result is returned and the process will end.

Step 4. The numeric substrings are determined by locating the first occurrences of a non-digit character after the first digit characters; the substrings from the first digit characters through the characters preceding the first non-digit character are the numeric substrings. These substrings are converted into double-precision values. The converted values are compared by numeric comparison. If these values differ, then the result will be returned and the process will end.

Step 5. The suffix substrings, which are the strings that start after the last digits of the numeric substring, will be determined. This suffix substrings replace the original strings, and the whole process will start again with step 1.

Decimal separators are treated as follows: If the attribute value is `integer`, then a decimal separator is not considered as a digit. If the attribute value is `double`, the treatment of a decimal separator is context-dependent: If a decimal separator occurs adjacent to one or two digit characters, it is considered a digit character as long as it is the only occurrence in that given numeric substring. In other words, a second occurrence of a decimal separator in sequence of digits and decimal separators is treated as a non-digit character. Therefore the character immediately preceding the separator becomes the last character of the numeric substring, while the separator itself becomes the first character of the suffix substring.

This sorting process is illustrated by the following figure:

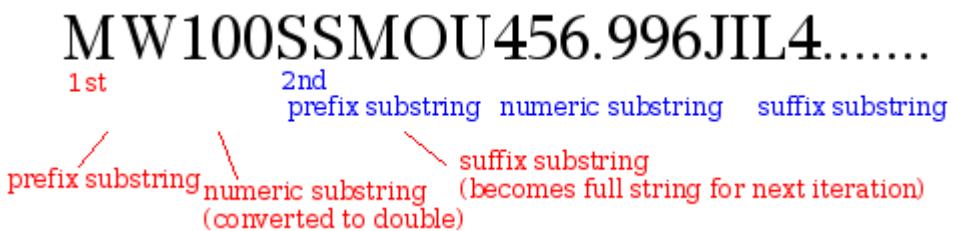


Figure 6 - Natural sort

The default value for this attribute is `alpha-numeric`.

The `table:embedded-number-behavior` attribute is usable with the following element:  
`<table:sort> 9.4.19.`

The values of the `table:embedded-number-behavior` attribute are `alpha-numeric`, `integer` or `double`.

## 19.629 table:enabled

The `table:enabled` attribute specifies whether the `<table:data-pilot-display-info>` element is evaluated or not.

The defined values for the `table:enabled` attribute are:

- `false`: the `<table:data-pilot-display-info>` element is not evaluated.
- `true`: the `<table:data-pilot-display-info>` element is evaluated.

The `table:enabled` attribute is usable with the following element: `<table:data-pilot-display-info>` 9.6.13.

The `table:enabled` attribute has the data type `boolean` 18.3.3.

## 19.630 table:end

The `table:end` attribute specifies the end value for a grouping of numeric values. All values that are higher than the end value are contained in a single group, while values that are equal to or lower than the end value are grouped as specified by `table:grouped-by` 19.671 and `table:step` 19.650 attributes.

The defined values for the `table:end` attribute are:

- `auto`: the highest value of the field is taken as the end value.
- a value of type `double`

The `table:end` attribute is usable with the following element: `<table:data-pilot-groups>` 9.6.17.

The values of the `table:end` attribute are a value of type `double` 18.2 or `auto`.

## 19.631 table:end-cell-address

The `table:end-cell-address` attribute specifies the end position of the shape if it is included in a spreadsheet document.

The `table:end-cell-address` attribute is usable with the following elements:  
`<dr3d:scene>` 10.5.2, `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:connector>` 10.3.10, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:g>` 10.3.15, `<draw:line>` 10.3.3, `<draw:measure>` 10.3.12, `<draw:page-thumbnail>` 10.3.14, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2, `<draw:regular-polygon>` 10.3.6 and `<office:annotation>` 14.1.

The `table:end-cell-address` attribute has the data type `cellAddress` 18.3.4.

## 19.632 table:end-column

The `table:end-column` attribute specifies the end column of a range if the range address is a cell range address. The value of a `table:end-column` attribute is inclusive.

The `table:end-column` attribute is usable with the following elements: `<table:source-range-address>` 9.9.14 and `<table:target-range-address>` 9.9.15.

The `table:end-column` attribute has the data type `integer` 18.2.

## 19.633 table:end-position

The `table:end-position` attribute specifies the number of the last deleted row or column. The value of a `table:end-position` attribute is exclusive.

The `table:end-position` attribute is usable with the following element:

`<table:movement-cut-off>` 9.9.12.

The `table:end-position` attribute has the data type `integer` 18.2.

## 19.634 `table:end-row`

The `table:end-row` attribute specifies the end row of a range if the range address is a cell range address. The value of a `table:end-row` attribute is inclusive.

The `table:end-row` attribute is usable with the following elements: `<table:source-range-address>` 9.9.14 and `<table:target-range-address>` 9.9.15.

The `table:end-row` attribute has the data type `integer` 18.2.

## 19.635 `table:end-table`

The `table:end-table` attribute specifies the end table of a range if the range address is a cell range address. The value of a `table:end-table` attribute is inclusive.

The `table:end-table` attribute is usable with the following elements: `<table:source-range-address>` 9.9.14 and `<table:target-range-address>` 9.9.15.

The `table:end-table` attribute has the data type `integer` 18.2.

## 19.636 `table:end-x`

The `table:end-x` attribute specifies the x-coordinate of the end position of a shape relative to the top-left edge of a cell. The size attributes of the shape are ignored.

The `table:end-x` attribute is usable with the following elements: `<dr3d:scene>` 10.5.2, `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:connector>` 10.3.10, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:g>` 10.3.15, `<draw:line>` 10.3.3, `<draw:measure>` 10.3.12, `<draw:page-thumbnail>` 10.3.14, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2, `<draw:regular-polygon>` 10.3.6 and `<office:annotation>` 14.1.

The `table:end-x` attribute has the data type `coordinate` 18.3.10.

## 19.637 `table:end-y`

The `table:end-y` attribute specifies the y-coordinate of the end position of a shape relative to the top-left edge of a cell. The size attributes of the shape are ignored.

The `table:end-y` attribute is usable with the following elements: `<dr3d:scene>` 10.5.2, `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:connector>` 10.3.10, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:g>` 10.3.15, `<draw:line>` 10.3.3, `<draw:measure>` 10.3.12, `<draw:page-thumbnail>` 10.3.14, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2, `<draw:regular-polygon>` 10.3.6 and `<office:annotation>` 14.1.

The `table:end-y` attribute has the data type `coordinate` 18.3.10.

## 19.638 `table:execute`

The `table:execute` attribute specifies whether a macro should be executed or not.

The defined values for the `table:execute` attribute are:

- `false`: macro is not executed.
- `true`: macro is executed.

The default value for this attribute is `true`.

The `table:execute` attribute is usable with the following element: `<table:error-macro>` 9.4.8.

The `table:execute` attribute has the data type `boolean` 18.3.3.

## 19.639 `table:expression`

The `table:expression` attribute specifies an expression. If the expression contains a named range or another named expression, the named range or named expression shall be specified first, before the containing expression. If the expression contains a relative cell range address, a `table:base-cell-address` attribute shall also appear on the `<table:named-expression>` element.

Relative cell range addresses that occur in an expression define an offset which shall be interpreted relative to the cell where a named expression is used. The offset is the same as the offsets between the table cell addressed by the relative cell range address and the table cell specified by the `table:base-cell-address` attribute.

The value of this attribute may comprise a namespace prefix, followed by a ":" (U+003A, COLON), followed by an expression. If the namespace prefix is missing the namespace defaults to the "urn:oasis:names:tc:opendocument:xmlns:of:1.2" namespace. If a namespace prefix is present, the expression shall start with an '=' equal sign. If no namespace prefix is present, the '=' equal sign may be omitted. The XML namespace name bound to the namespace prefix specifies the syntax and semantics of the formulas and values occurring within the expression.

An OpenDocument Consumer when hosting a formula evaluator makes available the host-dependent properties defined by Table 17 Host-dependent Properties in 19.646.

The `table:expression` attribute is usable with the following element: `<table:named-expression>` 9.4.13.

The `table:expression` attribute has the data type `string` 18.2.

## 19.640 `table:field-name`

The `table:field-name` attribute specifies a category column whose members' values are used in operations defined by a `table:type` attribute that determines the display of values of the data field of which the `<table:data-pilot-field-reference>` element is a part.

The `table:field-name` attribute is usable with the following element: `<table:data-pilot-field-reference>` 9.6.16.

The `table:field-name` attribute has the data type `string` 18.2.

## 19.641 `table:field-number`

The `table:field-number` attribute specifies a row or column number to sort, subtotal or filter. It is the number of a row or column within a database range.

The `table:field-number` attribute is usable with the following elements:  
`<table:filter-condition>` 9.5.5, `<table:sort-by>` 9.4.20 and  
`<table:subtotal-field>` 9.4.24.

The `table:field-number` attribute has the data type `nonNegativeInteger` 18.2.

## 19.642 table:filter-name

The `table:filter-name` attribute specifies the file type of the document containing the original table. The value of this attribute is implementation-dependent.

The `table:filter-name` attribute is usable with the following elements: `<table:cell-range-source>` 9.3.1 and `<table:table-source>` 9.2.6.

The `table:filter-name` attribute has the data type `string` 18.2.

## 19.643 table:filter-options

The `table:filter-options` attribute specifies implementation-dependent settings for a file type.

The `table:filter-options` attribute is usable with the following elements: `<table:cell-range-source>` 9.3.1 and `<table:table-source>` 9.2.6.

The `table:filter-options` attribute has the data type `string` 18.2.

## 19.644 table:first-row-end-column (deprecated)

The `table:first-row-end-column` attribute specifies the whether the cell in the first row, end column gets its style from its row or column.

The use of this attribute is deprecated. The `table:use-first-row-styles` 19.744 or `table:use-last-column-styles` 19.745 attributes should be used instead.

The defined values for the `table:first-row-end-column` attribute are:

- `column`: cell in first row, end column should inherit its style from the column where it appears.
- `row`: cell in first row, end column should inherit its style from the row where it appears.

The `table:first-row-end-column` attribute is usable with the following element: `<table:table-template>` 16.20.

The values of the `table:first-row-end-column` attribute are `row` or `column`.

## 19.645 table:first-row-start-column (deprecated)

The `table:first-row-start-column` attribute specifies the whether the cell in the first row, start column gets its style from its row or column.

The use of this attribute is deprecated. The `table:use-first-row-styles` 19.744 or `table:use-first-column-styles` 19.743 attributes should be used instead.

The defined values for the `table:first-row-start-column` attribute are:

- `column`: cell in first row, start column should inherit its style from the column where it appears.
- `row`: cell in first row, start column should inherit its style from the row where it appears.

The `table:first-row-start-column` attribute is usable with the following element: `<table:table-template>` 16.20.

The values of the `table:first-row-start-column` attribute are `row` or `column`.

## 19.646 table:formula

The `table:formula` attribute specifies a formula for a table cell.

Formulas specify calculations to be performed within table cells. The attribute value should begin with a namespace prefix followed by ":" (U+003A, COLON), followed by the text of the formula. The namespace bound to the prefix determines the syntax and semantics of the formula.

Whenever the initial text of a formula has the appearance of an NCName followed by ":" (U+003A, COLON), an OpenDocument producer shall provide a valid namespace prefix and separating ":" (U+003A, COLON) separator before the text of the formula in order to eliminate any ambiguity.

If a namespace prefix is not specified, the namespace defaults to the "urn:oasis:names:tc:opendocument:xmlns:of:1.2" namespace.

An OpenDocument Consumer when hosting a formula evaluator has the host-dependent properties specified in Table 17.

*Table 17 Host-dependent Properties*

<b>Property</b>	<b>Attribute</b>	<b>Default Value</b>
HOST-CASE-SENSITIVE	table:case-sensitive	true
HOST-PRECISION-AS-SHOWN	table:precision-as-shown	false
HOST-SEARCH-CRITERIA-MUST-APPLY-TO-WHOLE-CELL	table:search-criteria-must-apply-to-whole-cell	true
HOST-AUTOMATIC-FIND-LABELS	table:automatic-find-labels	true
HOST-USE-REGULAR-EXPRESSIONS	table:use-regular-expressions	true
HOST-USE-WILDCARDS	table:use-wildcards	false
HOST-NULL-YEAR	table:null-year	1930
HOST-NULL-DATE	table:null-date	1899-12-30
HOST-LOCALE	N/A	Implementation-defined
HOST-ITERATION-STATUS	table:status	disable
HOST-ITERATION-MAXIMUM-DIFFERENCE	table:maximum-difference	0
HOST-ITERATION-STEPS	table:steps	100

**Note:** Host-dependent properties are applicable to `style:condition` (19.472), `table:condition` (19.600) and, `table:expression` (19.639) attributes.

The `table:formula` attribute is usable with the following elements: `<table:change-track-table-cell>` 9.9.16, `<table:covered-table-cell>` 9.1.5 and `<table:table-cell>` 9.1.4.

The `table:formula` attribute has the data type string 18.2.

## 19.647 table:function

### 19.647.1 General

The `table:function` attribute specifies functions that are applied to tables.

The `table:function` attribute on all elements has the following defined values:

- `average`: the average of all numeric values.
- `count`: the count of all non-empty values, including text.
- `countnums`: the count of all numeric values.
- `max`: the maximum of all numeric values.
- `min`: the minimum of all numeric values.
- `product`: the product of all numeric values.
- `stdev`: the standard deviation, treating all numeric values as a sample from a population.
- `stdevp`: the standard deviation, treating all numeric values as a whole population.
- `sum`: the sum of all numeric values.
- `var`: the variance, treating all numeric values as a sample from a population.
- `varp`: the variance, treating all numeric values as a whole population.
- a value of type `string`: an implementation-dependent function name.

## 19.647.2 <table:consolidation>

The `table:function` attribute specifies a function.

The `table:function` attribute is usable with the following element:  
`<table:consolidation> 9.7.`

The values of the `table:function` attribute are `average`, `count`, `countnums`, `max`, `min`, `product`, `stdev`, `stdevp`, `sum`, `var`, `varp` or a value of type `string` `18.2`.

## 19.647.3 <table:data-pilot-field>

The `table:function` attribute specifies the function which is applied to the cell values of data columns.

For `<table:data-pilot-field>` elements whose `table:orientation` 19.743 attribute has the value `data`, the defined values for the `table:function` attribute are those defined in 19.647.1.

For `<table:data-pilot-field>` elements whose `table:orientation` attribute has the value `column` or `row`, the only defined value for the `table:function` attribute is:

- `auto`: no function is applied to that category field.

The `table:function` attribute is usable with the following element: `<table:data-pilot-field> 9.6.7.`

The values of the `table:function` attribute are `auto`, `average`, `count`, `countnums`, `max`, `min`, `product`, `stdev`, `stdevp`, `sum`, `var`, `varp` or a value of type `string` `18.2`.

## 19.647.4 <table:data-pilot-subtotal>

The `table:function` attribute specifies the type of subtotals to calculate.

In addition to the values defined in 19.647.1 the following value is defined:

- `auto`: specifies that subtotals shall use the same function(s) that are used to calculate the `<table:data-pilot-field> 9.6.7` elements with `table:orientation="data"` 19.690.

The `table:funciton` attribute is usable with the following element: `<table:subtot-pilot-subtotal>` 9.6.10.

The values of the `table:funciton` attribute are `auto`, `average`, `count`, `countnums`, `max`, `min`, `product`, `stdev`, `stdevp`, `sum`, `var`, `varp` or a value of type `string` 18.2.

## 19.647.5 `<table:subtotal-field>`

The `table:funciton` attribute specifies the type of subtotals to calculate.

The `table:funciton` attribute is usable with the following element: `<table:subtot-field>` 9.4.24.

The values of the `table:funciton` attribute are `average`, `count`, `countnums`, `max`, `min`, `product`, `stdev`, `stdevp`, `sum`, `var`, `varp` or a value of type `string` 18.2.

## 19.648 `table:grand-total`

The `table:grand-total` attribute specifies whether a grand total column, row, or both should be displayed in addition to values calculated for each combination of values in the category columns.

The defined values for the `table:grand-total` attribute are:

- `both`: grand totals for rows and columns are displayed in addition to values calculated for other combinations of values.
- `column`: grand totals for columns are displayed in addition to values calculated for other combinations of values.
- `none`: grand totals for rows and columns are not displayed in addition to values calculated for other combinations of values.
- `row`: grand totals for rows are displayed in addition to values calculated for other combinations of values.

The default value for this attribute is `both`.

The `table:grand-total` attribute is usable with the following element: `<table:subtot-pilot-table>` 9.6.3.

The values of the `table:grand-total` attribute are `none`, `row`, `column` or `both`.

## 19.649 `table:group-by-field-number`

The `table:group-by-field-number` attribute specifies a field, that is to be grouped. It is the number of a row or column within a database range.

The `table:group-by-field-number` attribute is usable with the following element: `<table:subtot-rule>` 9.4.23.

The `table:group-by-field-number` attribute has the data type `nonNegativeInteger` 18.2.

## 19.650 `table:grouped-by`

The `table:grouped-by` attribute specifies the grouping of the date values. Date values can be grouped by seconds, minutes, hours, days, months, quarters or years.

**Note:** If date values are for instance grouped by minutes, all dates or times that are within the same minute are within one group.

The defined values for the `table:grouped-by` attribute are:

- `seconds`: a value expressed as seconds.
- `minutes`: a value expressed as minutes.
- `hours`: a value expressed as hours.
- `days`: a value expressed as days.
- `months`: a value expressed as months.
- `quarters`: a value expressed as quarters (4 equal 3 month segments).
- `years`: a value expressed as years.

The `table:grouped-by` attribute is usable with the following element: `<table:database-range>` 9.6.17.

The values of the `table:grouped-by` attribute are `seconds`, `minutes`, `hours`, `days`, `months`, `quarters` or `years`.

## 19.651 table:has-persistent-data

The `table:has-persistent-data` attribute specifies whether the current data in a database range is saved when the document itself is saved.

The defined values for the `table:has-persistent-data` attribute are:

- `false`: data in a database range is not saved when the document itself is saved.
- `true`: data in a database range is saved when the document itself is saved.

The default value for this attribute is `true`.

The `table:has-persistent-data` attribute is usable with the following element: `<table:database-range>` 9.4.15.

The `table:has-persistent-data` attribute has the data type `boolean` 18.3.3.

## 19.652 table:id

The `table:id` attribute specifies the id of an insertion or other tracked change.

The `table:id` attribute is usable with the following elements: `<table:cell-content-change>` 9.9.17, `<table:cell-content-deletion>` 9.9.7, `<table:change-deletion>` 9.9.8, `<table:deletion>` 9.9.9, `<table:dependency>` 9.9.5, `<table:insertion>` 9.9.3, `<table:insertion-cut-off>` 9.9.11, `<table:movement>` 9.9.13 and `<table:previous>` 9.9.19.

The `table:id` attribute has the data type `string` 18.2.

## 19.653 table:identify-categories

The `table:identify-categories` attribute specifies whether rows that do not contain a value in one of the category columns should use the value of the nearest ancestor row that has a value, or should be moved into a group (or category) of its own.

The defined values for the `table:identify-categories` attribute are:

- `false`: empty rows form a group or category.
- `true`: empty rows use the value of the nearest ancestor row that has a value for classification into a group or category.

The default value for this attribute is `false`.

The `table:identify-categories` attribute is usable with the following element:

`<table:data-pilot-table>` 9.6.3.

The `table:identify-categories` attribute has the data type `boolean` 18.3.3.

## 19.654 `table:ignore-empty-rows`

The `table:ignore-empty-rows` attribute specifies whether empty rows in a source range should be ignored.

The defined values for the `table:ignore-empty-rows` attribute are:

- `false`: empty rows in a source range are not ignored.
- `true`: empty rows in a source range are ignored.

The default value for this attribute is `false`.

The `table:ignore-empty-rows` attribute is usable with the following element:

`<table:data-pilot-table>` 9.6.3.

The `table:ignore-empty-rows` attribute has the data type `boolean` 18.3.3.

## 19.655 `table:index`

The `table:index` attribute specifies for the `<table:operation>` element on which it appears its order in a series of `<table:operation>` elements under a `<table:detective>` element.

The `table:index` attribute is usable with the following element: `<table:operation>` 9.3.3.

The `table:index` attribute has the data type `nonNegativeInteger` 18.2.

## 19.656 `table:is-active`

The `table:is-active` attribute specifies whether a scenario that belongs to a scenario table is active.

The defined values for the `table:is-active` attribute are:

- `false`: scenario that belongs to a scenario table is inactive.
- `true`: scenario that belongs to a scenario table is active.

The `table:is-active` attribute is usable with the following element: `<table:scenario>` 9.2.7.

The `table:is-active` attribute has the data type `boolean` 18.3.3.

## 19.657 `table:is-data-layout-field`

The `table:is-data-layout-field` attribute specifies whether a field is a data layout field.

The defined values for the `table:is-data-layout-field` attribute are:

- `false`: field is not a data layout field.
- `true`: field is a data layout field.

The default value for this attribute is `false`.

The `table:is-data-layout-field` attribute is usable with the following element:

`<table:data-pilot-field>` 9.6.7.

The `table:is-data-layout-field` attribute has the data type [string 18.2](#).

## 19.658 `table:is-selection`

The `table:is-selection` attribute specifies whether the database range includes a complete database, or a selection of records from a database only.

The defined values for the `table:is-selection` attribute are:

- `false`: database range includes a section of records from a database.
- `true`: database range includes a complete database.

The default value for this attribute is `false`.

The `table:is-selection` attribute is usable with the following element:  
`<table:database-range>` [9.4.15](#).

The `table:is-selection` attribute has the data type [boolean 18.3.3](#).

## 19.659 `table:is-sub-table`

The `table:is-sub-table` attribute specifies if a nested table is a subtable.

If a table cell only contains a single table but no paragraphs or other content, the table can be specified to be a subtable. In which case it occupies the whole cell and no other content can appear in the cell.

The borders of a subtable merge with the borders of the cell where it is specified. A subtable does not have its own style.

A nested table that is not specified to be a subtable appears as a table within a table, that is, it has borders distinct from those of the parent cell and respects the padding of the parent cell.

The defined values for the `table:is-sub-table` attribute are:

- `false`: nested table is not a subtable.
- `true`: nested table is a subtable.

The default value for this attribute is `false`.

The `table:is-sub-table` attribute is usable with the following element: `<table:table>` [9.1.2](#).

The `table:is-sub-table` attribute has the data type [boolean 18.3.3](#).

## 19.660 `table:label-cell-range-address`

The `table:label-cell-range-address` attribute specifies a cell range address for labels.

The `table:label-cell-range-address` attribute is usable with the following element:  
`<table:label-range>` [9.4.9](#).

The `table:label-cell-range-address` attribute has the data type [cellRangeAddress 18.3.5](#).

## 19.661 `table:language`

The `table:language` attribute specifies the natural language in which a comparison will occur.

The `table:language` attribute is usable with the following element: `<table:sort>` [9.4.19](#).

The `table:language` attribute has the data type [languageCode 18.3.18](#).

## 19.662 table:last-column-spanned

The `table:last-column-spanned` attribute together with the `table:last-row-spanned` 19.664 attribute specifies the last known size of a range. If the size of the range has changed since the determination of its size, the value of these attributes will be incorrect.

The `table:last-column-spanned` attribute is usable with the following element:  
`<table:cell-range-source>` 9.3.1.

The `table:last-column-spanned` attribute has the data type `positiveInteger` 18.2.

## 19.663 table:last-row-end-column (deprecated)

The `table:last-row-end-column` attribute specifies the whether the cell in the last row, end column gets its style from its row or column.

The use of this attribute is deprecated. The `table:use-last-row-styles` 19.746 or `table:use-last-column-styles` 19.745 attributes should be used instead.

The defined values for the `table:last-row-end-column` attribute are:

- `column`: cell in last row, end column should inherit its style from the column where it appears.
- `row`: cell in last row, end column should inherit its style from the column where it appears.

The `table:last-row-end-column` attribute is usable with the following element:  
`<table:table-template>` 16.20.

The values of the `table:last-row-end-column` attribute are `row` or `column`.

## 19.664 table:last-row-spanned

The `table:last-row-spanned` attribute together with the `table:last-column-spanned` 19.662 attribute specifies the last known size of a range. If the size of the range has changed since the determination of its size, the value of these attributes will be incorrect.

The `table:last-row-spanned` attribute is usable with the following element:  
`<table:cell-range-source>` 9.3.1.

The `table:last-row-spanned` attribute has the data type `positiveInteger` 18.2.

## 19.665 table:last-row-start-column (deprecated)

The `table:last-row-start-column` attribute specifies the whether the cell in the last row, start column gets its style from its row or column.

The use of this attribute is deprecated. The `table:use-last-row-styles` 19.746 or `table:use-first-column-styles` 19.743 attributes should be used instead.

The defined values for the `table:last-row-start-column` attribute are:

- `column`: cell in last row, start column should inherit its style from the column where it appears.
- `row`: cell in last row, start column should inherit its style from the column where it appears.

The `table:last-row-start-column` attribute is usable with the following element:  
`<table:table-template>` 16.20.

The values of the `table:last-row-start-column` attribute are `row` or `column`.

## 19.666 table:layout-mode

The `table:layout-mode` attribute specifies how to layout a field.

The defined values for the `table:layout-mode` attribute are:

- `outline-subtotals-bottom`: Same as `outline-subtotals-top`, except that subtotals are shown at the bottom (below the member's data, as in tabular layout mode).
- `outline-subtotals-top`: In outline layout mode, the members from the following field start in the row below a member's name. Subtotals are shown at the top (in the same row as the member's name). When the subtotals take up more than one row (manually selected, or because there are multiple data fields), they are always shown below the member's data, regardless of the setting.
- `tabular-layout`: Tabular layout mode is the layout, where each member's name is on the same row as the first member from the following field. Subtotals are always shown below a member's data in this mode.

The `table:layout-mode` attribute is usable with the following element: `<table: data-pilot-layout-info>` 9.6.15.

The values of the `table:layout-mode` attribute are `tabular-layout`, `outline-subtotals-top` or `outline-subtotals-bottom`.

## 19.667 table:link-to-source-data

The `table:link-to-source-data` attribute specifies whether the data in a consolidation table range should be linked to the source data, so that it is automatically updated if any changes are made to the source data.

The defined values for the `table:link-to-source-data` attribute are:

- `false`: data in consolidated table range not linked to source data.
- `true`: data in consolidated table range is linked to source data.

The default value for this attribute is `false`.

The `table:link-to-source-data` attribute is usable with the following element: `<table:consolidation>` 9.7.

The `table:link-to-source-data` attribute has the data type `boolean` 18.3.3.

## 19.668 table:marked-invalid

The `table:marked-invalid` attribute specifies whether the current cell is marked invalid. This attribute cannot be used together with any other attributes.

The defined values for the `table:marked-invalid` attribute are:

- `false`: current cell not marked invalid.
- `true`: current cell marked invalid.

The `table:marked-invalid` attribute is usable with the following element: `<table:highlighted-range>` 9.3.4.

The `table:marked-invalid` attribute has the data type `boolean` 18.3.3.

## 19.669 table:matrix-covered

The `table:matrix-covered` attribute specifies if a cell is contained in a matrix.

The defined values for the `table:matrix-covered` attribute are:

- `false`: cell is not included in a matrix.
- `true`: cell is included in a matrix.

The default value for this attribute is `false`.

The `table:matrix-covered` attribute is usable with the following element:  
`<table:change-track-table-cell>` 9.9.16.

The `table:matrix-covered` attribute has the data type `boolean` 18.3.3.

## 19.670 `table:maximum-difference`

The `table:maximum-difference` attribute specifies the maximum difference between two iterative calculation results. The iteration is stopped if the result is less than the value of this attribute.

The default value for this attribute is `0.001`.

The `table:maximum-difference` attribute is usable with the following element:  
`<table:iteration>` 9.4.3.

The `table:maximum-difference` attribute has the data type `double` 18.2.

## 19.671 `table:member-count`

The `table:member-count` attribute specifies the number of values, from the top or from the bottom of a data field's column, that are shown.

The `table:member-count` attribute is usable with the following element: `<table:data-pilot-display-info>` 9.6.13.

The `table:member-count` attribute has the data type `nonNegativeInteger` 18.2.

## 19.672 `table:member-name`

The `table:member-name` attribute specifies the value of a data pilot member.

The `table:member-name` attribute is usable with the following element: `<table:data-pilot-field-reference>` 9.6.16.

The `table:member-name` attribute has the data type `string` 18.2.

## 19.673 `table:member-type`

The `table:member-type` attribute specifies the member of the referenced category column, whose value within the current data field is taken into account.

The defined values for the `table:member-type` attribute are:

- `named`: the `table:member-name` 19.672 attribute on the same `<table:data-pilot-field-reference>` element specifies the member whose value within the data field is taken into account.
- `next`: the value of the data field for the next visible member of the referenced category column is taken into account. Empty members are skipped.
- `previous`: the value of the data field for the next visible member of the referenced category column is taken into account. Empty members are skipped.

The `table:member-type` attribute is usable with the following element: `<table:data-pilot-field-reference>` 9.6.16.

## 19.674 `table:message-type`

The `table:message-type` attribute specifies the display of messages.

The defined values for the `table:message-type` attribute are:

- **information**: message is displayed as information only.
- **stop**: message is displayed as an error and the operation that caused the validation check is stopped.
- **warning**: message is displayed as a warning.

The default value for this attribute is `stop`.

The `table:message-type` attribute is usable with the following element: `<table:error-message>` 9.4.7.

The values of the `table:message-type` attribute are `stop`, `warning` or `information`.

## 19.675 `table:mode`

The `table:mode` attribute specifies what data should be copied from a source table to a destination table.

The defined values for the `table:mode` attribute are:

- `copy-all`: formulas and styles are copied.
- `copy-results-only`: only formula results and non-calculated cell content are copied.

The default value for this attribute is `copy-all`.

The `table:mode` attribute is usable with the following element: `<table:table-source>` 9.2.6.

The values of the `table:mode` attribute are `copy-all` or `copy-results-only`.

## 19.676 `table:multi-deletion-spanned`

The `table:multi-deletion-spanned` attribute specifies the total number of deleted rows or columns when multiple columns or rows were deleted simultaneously. Each deleted row or column is represented by a `<table:deletion>` element. The first `<table:deletion>` element in a set representing a simultaneous deletion shall carry a `table:multi-deletion-spanned` attribute that specifies the total number of deleted rows or columns.

The `table:multi-deletion-spanned` attribute is usable with the following element: `<table:deletion>` 9.9.9.

The `table:multi-deletion-spanned` attribute has the data type `integer` 18.2.

## 19.677 `table:name`

### 19.677.1 General

The `table:name` attribute specifies a name.

### 19.677.2 `<table:cell-range-source>`

The `table:name` attribute specifies the name of a source database range or named range.

The `table:name` attribute is usable with the following element: `<table:cell-range-source>` 9.3.1.

The `table:name` attribute has the data type `string` 18.2.

### 19.677.3 `<table:content-validation>`

The `table:name` attribute specifies the name of a content validation rule. It is used to reference the validation rule from the cell the rule should applied.

The `table:name` attribute is usable with the following element: `<table:content-validation>` 9.4.5.

The `table:name` attribute has the data type `string` 18.2.

#### 19.677.4 `<table:data-pilot-group>`

The `table:name` attribute specifies the name of a group.

The `table:name` attribute is usable with the following element: `<table:data-pilot-group>` 9.6.18.

The `table:name` attribute has the data type `string` 18.2.

#### 19.677.5 `<table:data-pilot-group-member>`

The `table:name` attribute specifies the name of a member.

The `table:name` attribute is usable with the following element: `<table:data-pilot-group-member>` 9.6.19.

The `table:name` attribute has the data type `string` 18.2.

#### 19.677.6 `<table:data-pilot-member>`

The `table:name` attribute specifies the value for which display information is specified.

The `table:name` attribute is usable with the following element: `<table:data-pilot-member>` 9.6.12.

The `table:name` attribute has the data type `string` 18.2.

#### 19.677.7 `<table:data-pilot-table>`

The `table:name` attribute specifies the name of a data pilot table.

The `table:name` attribute is usable with the following element: `<table:data-pilot-table>` 9.6.3.

The `table:name` attribute has the data type `string` 18.2.

#### 19.677.8 `<table:database-range>`

The `table:name` attribute specifies the name of a database range on which to perform operations. Within a single document, only one database range may have no name. This database range is created by the consumer and is used to filter or sort data in a cell range without the user creating a database range.

The `table:name` attribute is usable with the following element: `<table:database-range>` 9.4.15.

The `table:name` attribute has the data type `string` 18.2.

#### 19.677.9 `<table:named-expression>`

The `table:name` attribute specifies the name of an expression.

The `table:name` attribute is usable with the following element: `<table:named-expression>` 9.4.13.

The `table:name` attribute has the data type `string` 18.2.

## 19.677.10 <table:named-range>

The `table:name` attribute specifies the name of a database range on which to perform operations. Within a single document, only one database range may have no name.

The `table:name` attribute is usable with the following element: `<table:named-range>` 9.4.12.

The `table:name` attribute has the data type `string` 18.2.

## 19.677.11 <table:operation>

The `table:name` attribute specifies the name of a detective operation.

The defined values for the `table:name` attribute are:

- `remove-dependents`: removes highlighting from cells that use the value of the current cell in their formula.
- `remove-precedents`: removes highlighting from cells whose values are used in the formula of the current cell.
- `trace-dependents`: highlights cells that use the value of the current cell in their formula.
- `trace-errors`: highlights cells that cause an error while calculating the result of the current cell's formula.
- `trace-precedents`: highlights cells whose values are used in the formula of the current cell.

The nature of the highlighting imposed on or removed from cells as the result of detective operations is implementation-dependent.

The `table:name` attribute is usable with the following element: `<table:operation>` 9.3.3.

The values of the `table:name` attribute are `trace-dependents`, `remove-dependents`, `trace-precedents`, `remove-precedents` or `trace-errors`.

## 19.677.12 <table:source-cell-range>

The `table:name` attribute specifies the name of a cell range containing source data. The value of this attribute shall be a string matching the value of a `table:name` attribute of a `table:named-range` element.

The `table:name` attribute is usable with the following element: `<table:source-cell-range>` 9.6.5.

The `table:name` attribute has the data type `string` 18.2.

## 19.677.13 <table:source-service>

The `table:name` attribute specifies the name of a service. The value of this attribute is implementation-dependent.

The `table:name` attribute is usable with the following element: `<table:source-service>` 9.6.6.

The `table:name` attribute has the data type `string` 18.2.

## 19.677.14 <table:table>

The `table:name` attribute specifies the name of a table.

The `table:name` attribute is usable with the following element: `<table:table>` 9.1.2.

The `table:name` attribute has the data type `string` 18.2.

### 19.677.15 `<table:table-template>`

The `table:name` attribute specifies the name of a table template.

The `table:name` attribute is usable with the following element: `<table:table-template>` 16.20.

The `table:name` attribute has the data type `string` 18.2.

### 19.678 `table:null-year`

The `table:null-year` attribute specifies the start year for year values that contain only two digits. All two-digit year values are interpreted as a year that is the same as or follows the start year.

The default value for this attribute is 1930.

The `table:null-year` attribute is usable with the following element:  
`<table:calculation-settings>` 9.4.1.

The `table:null-year` attribute has the data type `positiveInteger` 18.2.

### 19.679 `table:number-columns-repeated`

#### 19.679.1 General

The `table:number-columns-repeated` attribute specifies the repetition of columns.

#### 19.679.2 `<table:covered-table-cell>`

The `table:number-columns-repeated` attribute specifies the number of successive columns in which a cell is repeated. It may be used to describe two or more adjoining cells with a single cell element, if they meet the following conditions:

- The cells contain the same content and style.
- The cells are not merged horizontally or vertically.

For a `<table:covered-table-cell>` 9.1.5 element the default value for this attribute is 1.

The `table:number-columns-repeated` attribute is usable with the following element:  
`<table:covered-table-cell>` 9.1.5.

The `table:number-columns-repeated` attribute has the data type `positiveInteger` 18.2.

#### 19.679.3 `<table:table-cell>`

The `table:number-columns-repeated` attribute specifies the number of successive columns in which a cell is repeated. It may be used to describe two or more adjoining cells with a single cell element, if they meet the following conditions:

- The cells contain the same content and style.
- The cells are not merged horizontally or vertically.

For a `<table:table-cell>` 9.1.4 element the default value for this attribute is 1.

The `table:number-columns-repeated` attribute is usable with the following element:  
`<table:table-cell>` 9.1.4.

The `table:number-columns-repeated` attribute has the data type `positiveInteger`  
18.2.

#### 19.679.4 `<table:table-column>`

The `table:number-columns-repeated` attribute specifies the number of columns to which a column description applies. If two or more columns are adjoining, and have the same style, this attribute may be used to describe them with a single `<table:table-column>` element.

For a `<table:table-column>` 9.1.6 element the default value for this attribute is 1.

The `table:number-columns-repeated` attribute is usable with the following element:  
`<table:table-column>` 9.1.6.

The `table:number-columns-repeated` attribute has the data type `positiveInteger`  
18.2.

#### 19.680 `table:number-columns-spanned`

The `table:number-columns-spanned` attribute specifies the number of columns that a cell spans.

When a cell covers another cell because of a column or row span value greater than one, a `<table:covered-table-cell>` 9.1.5 element shall appear in the table to represent the covered cell.

The default value for this attribute is 1.

The `table:number-columns-spanned` attribute is usable with the following element:  
`<table:table-cell>` 9.1.4.

The `table:number-columns-spanned` attribute has the data type `positiveInteger`  
18.2.

#### 19.681 `table:number-rows-repeated`

The `table:number-rows-repeated` attribute specifies the number of rows to which a row element applies. If two or more rows are adjoining, and have the same content and style, and do not contain vertically merged cells, they may be described by a single `<table:table-row>` element that has a `table:number-rows-repeated` attribute with a value greater than 1.

When a cell covers another cell because of a column or row span value greater than one, a `<table:covered-table-cell>` 9.1.5 element shall appear in the table to represent the covered cell.

The default value for this attribute is 1.

The `table:number-rows-repeated` attribute is usable with the following element:  
`<table:table-row>` 9.1.3.

The `table:number-rows-repeated` attribute has the data type `positiveInteger` 18.2.

#### 19.682 `table:number-rows-spanned`

The `table:number-rows-spanned` attribute specifies the number of rows that a cell spans.

The default value for this attribute is 1.

The `table:number-rows-spanned` attribute is usable with the following element:  
`<table:table-cell>` 9.1.4.

The `table:number-rows-spanned` attribute has the data type `positiveInteger` 18.2.

## 19.683 `table:number-matrix-columns-spanned`

The `table:number-matrix-columns-spanned` attribute specifies the number of columns spanned by a matrix.

The `table:number-matrix-columns-spanned` attribute is usable with the following elements: `<table:change-track-table-cell>` 9.9.16 and `<table:table-cell>` 9.1.4.

The `table:number-matrix-columns-spanned` attribute has the data type `positiveInteger` 18.2.

## 19.684 `table:number-matrix-rows-spanned`

The `table:number-matrix-rows-spanned` attribute specifies the number of rows spanned by a matrix.

The `table:number-matrix-rows-spanned` attribute is usable with the following elements: `<table:change-track-table-cell>` 9.9.16 and `<table:table-cell>` 9.1.4.

The `table:number-matrix-rows-spanned` attribute has the data type `positiveInteger` 18.2.

## 19.685 `table:object-name`

The `table:object-name` attribute specifies the name of the object in the source which contains the data and is passed to the service implementation. Its value is implementation-dependent and service-specific.

The `table:object-name` attribute is usable with the following element: `<table:source-service>` 9.6.6.

The `table:object-name` attribute has the data type `string` 18.2.

## 19.686 `table:on-update-keep-size`

The `table:on-update-keep-size` attribute specifies the behavior of a database range if the size of the data in the data source changes.

The defined values for the `table:on-update-keep-size` attribute are:

- `false`: range does not retain its size.
- `true`: range retains its size.

The default value for this attribute is `true`.

The `table:on-update-keep-size` attribute is usable with the following element: `<table:database-range>` 9.4.15.

The `table:on-update-keep-size` attribute has the data type `boolean` 18.3.3.

## 19.687 `table:on-update-keep-styles`

The `table:on-update-keep-styles` attribute specifies the behavior if the database range is updated. If the attribute value is `true`, the cell styles that are assigned to the cells in the first non-label row of the database range are used for all rows with in the database range. If the attribute value is `false`, all cells in the database range are assigned the default cell style of the document assigned.

The defined values for the `table:on-update-keep-styles` attribute are:

- `false`: all cells in the database range get the default cell style of the document assigned.
- `true`: cell styles that are assigned to the cells in the first non-label row of the database range are used for all rows within the database range.

The default value for this attribute is `false`.

The `table:on-update-keep-styles` attribute is usable with the following element:  
`<table:database-range>` 9.4.15.

The `table:on-update-keep-styles` attribute has the data type `boolean` 18.3.3.

## 19.688 `table:operator`

The `table:operator` attribute specifies the operator to use in a filter condition. Each cell contained in columns specified by a field number (`table:field-number` 19.641 attribute) is compared to a value (`table:value` 19.641 attribute) using the specified operator. The result of this comparison is the result of the filter sub-conditions specified by a `<table:filter-condition>` element.

The defined operators that use regular expressions are:

- `match` (`matches`)
- `!match` (`does not match`)

The `table:value` attribute contains the regular expression.

Regular expressions are implementation-dependent expressions that, at a minimum, conform to the requirements of Conformance Clause C1 of [UTR18].

The defined operators that do not use regular expressions are:

- `=` (`equal to`)
- `!=` (`not equal to`)
- `<` (`less than`)
- `>` (`greater than`)
- `<=` (`less than or equal to`)
- `>=` (`greater than or equal to`)
- `begins` (`begins with`)
- `contains` (`contains`)
- `!contains` (`does not contain`)
- `ends` (`ends with`)
- `!begins` (`does not begin with`)
- `!ends` (`does not end with`)
- `bottom percent` (`same as bottom values, except that the office:value` 19.752 attribute specifies the number of cells for which the condition is true as a percentage)
- `bottom values` (`true for the n cells that have the smallest value, where n is the value of the office:value attribute`)
- `empty` (`true for empty cells`)
- `!empty` (`true for non-empty cells`)
- `top percent` (`same as bottom percent, but for the largest values`)

- `top` values (same as `bottom` values, but for the largest values)

Depending on the value of the `table:data-type` 19.388 attribute, all operations take place on string or numeric, or color values. Only `=` and `!=` operations are allowed on color values.

The `table:operator` attribute is usable with the following element: `<table:filter-condition>` 9.5.5.

The `table:operator` attribute has the data type `string` 18.2.

## 19.689 `table:order`

The `table:order` attribute specifies whether a sort order should be in ascending or descending order.

The defined values for the `table:order` attribute are:

- `ascending`: sorting of values should be in ascending order.
- `descending`: sorting of values should be in descending order.

**Note:** Sorting is locale- and implementation-dependent.

For `<table:sort-by>` 9.4.20 and `<table:sort-groups>` 9.4.22 elements the default value for this attribute is `ascending`.

The `table:order` attribute is usable with the following elements: `<table:data-pilot-sort-info>` 9.6.14, `<table:sort-by>` 9.4.20 and `<table:sort-groups>` 9.4.22.

The values of the `table:order` attribute are `ascending` or `descending`.

## 19.690 `table:orientation`

### 19.690.1 General

The `table:orientation` attribute specifies the organizing of tables.

### 19.690.2 `<table:data-pilot-field>`

The `table:orientation` attribute specifies the orientation of a source field.

The defined values for the `table:orientation` attribute are:

- `column`: field specifies a category column
- `data`: field specifies a data column
- `hidden`: field has a corresponding column in a data pilot's source but are not visible in the data pilot table.
- `page`: specifies that an automatic filter (one that allows to choose one of the values that are contained in the column) should be generated for the corresponding column. In that case, an additional field with row, column or data orientation shall exist for the column.  
The `table:selected-page` attribute specifies which value is selected for the filter.
- `row`: specifies a category row

The `table:orientation` attribute is usable with the following element: `<table:data-pilot-field>` 9.6.7.

The values of the `table:orientation` attribute are `row`, `column`, `data`, `hidden` or `page`.

### 19.690.3 <table:database-range>

The `table:orientation` attribute specifies whether data fields are organized in row or columns.

The defined values for the `table:orientation` attribute are:

- `column`: organization is by columns
- `row`: organization is by rows

For a `<table:database-range>` 9.4.15 element the default value for this attribute is `row`.

The `table:orientation` attribute is usable with the following element:  
`<table:database-range>` 9.4.15.

The values of the `table:orientation` attribute are `column` or `row`.

### 19.690.4 <table:label-range>

The `table:orientation` attribute specifies whether labels label rows or columns.

The defined values for the `table:orientation` attribute are:

- `column`: label range oriented by column
- `row`: label range oriented by row

The `table:orientation` attribute is usable with the following element: `<table:label-range>` 9.4.9.

The values of the `table:orientation` attribute are `column` or `row`.

## 19.691 table:page-breaks-on-group-change

The `table:page-breaks-on-group-change` attribute specifies whether to insert a page break after the subtotal for each group.

The defined values for the `table:page-breaks-on-group-change` attribute are:

- `false`: page breaks are not inserted after subtotals for a group.
- `true`: page breaks are inserted after subtotals for a group.

The default value for this attribute is `false`.

The `table:page-breaks-on-group-change` attribute is usable with the following element: `<table:subtotal-rules>` 9.4.21.

The `table:page-breaks-on-group-change` attribute has the data type `boolean` 18.3.3.

## 19.692 table:paragraph-style-name

The `table:paragraph-style-name` attribute specifies the paragraph style which should be applied to the empty paragraphs created in cells for tables created from a template.

The `table:paragraph-style-name` attribute is usable with the following elements:  
`<table:body>` 16.21.6, `<table:even-columns>` 16.21.9, `<table:even-rows>` 16.21.7, `<table:first-column>` 16.21.4, `<table:first-row>` 16.21.2, `<table:last-column>` 16.21.5, `<table:last-row>` 16.21.3, `<table:odd-columns>` 16.21.10 and `<table:odd-rows>` 16.21.8.

The `table:paragraph-style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.693 table:parse-sql-statement

A `table:parse-sql-statement` attribute specifies whether the consumer will parse SQL statements.

The defined values for the `table:parse-sql-statement` attribute are:

- `false`: consumer does not parse SQL statements.
- `true`: consumer parses SQL statements.

The default value for this attribute is `false`.

The `table:parse-sql-statement` attribute is usable with the following element:  
`<table:database-source-sql>` 9.4.16.

The `table:parse-sql-statement` attribute has the data type `boolean` 18.3.3.

## 19.694 table:password

The `table:password` attribute specifies the password needed to access the source. It is passed to the service implementation. Its value is implementation-dependent and service-specific.

The `table:password` attribute is usable with the following element: `<table:source-service>` 9.6.6.

The `table:password` attribute has the data type `string` 18.2.

## 19.695 table:position

The `table:position` attribute specifies the position where an insertion or deletion was made in a table. Depending on the type of insertion or deletion, the value is the number of a row, a column or a table.

If multiple rows or columns get deleted, the `table:start-position` 19.723 and `table:end-position` 19.633 attributes contain the number of the first (inclusive) and last (exclusive) deleted rows or columns.

The `table:position` attribute is usable with the following elements: `<table:deletion>` 9.9.9, `<table:insertion>` 9.9.3, `<table:insertion-cut-off>` 9.9.11 and `<table:movement-cut-off>` 9.9.12.

The `table:position` attribute has the data type `integer` 18.2.

## 19.696 table:precision-as-shown

The `table:precision-as-shown` attribute specifies whether to perform a calculation using the rounded values displayed in the spreadsheet or using all of the digits in a value.

The defined values for the `table:precision-as-shown` attribute are:

- `false`: calculations are performed using all of the digits in a value, but the result is displayed as a rounded number.
- `true`: calculations are performed using the rounded values displayed in the spreadsheet.

The default value for this attribute is `false`.

The `table:precision-as-shown` attribute is usable with the following element:  
`<table:calculation-settings>` 9.4.1.

The `table:precision-as-shown` attribute has the data type `boolean` 18.3.3.

## 19.697 table:print

The `table:print` attribute specifies if a table will be printed. The `table:print` attribute value is overridden by the `table:display` 19.621 attribute. If the table is not displayed, it will also not be printed.

If a table is printed, the table range printed can be specified by the `table:print-ranges` attribute. If there is no `table:print-ranges` 19.698 attribute or it has no value specified, the non-empty portion of a table will be printed.

The defined values for the `table:print` attribute are:

- `false`: table cannot be printed.
- `true`: table can be printed.

The default value for this attribute is `true`.

The `table:print` attribute is usable with the following element: `<table:table>` 9.1.2.

The `table:print` attribute has the data type `boolean` 18.3.3.

## 19.698 table:print-ranges

The `table:print-ranges` attribute specifies print ranges of a table. Print ranges define the cells that will be printed. It contains a list of cell addresses or cell range addresses.

The `table:print-ranges` attribute is usable with the following element: `<table:table>` 9.1.2.

The `table:print-ranges` attribute has the data type `cellRangeAddressList` 18.3.6.

## 19.699 table:protect (deprecated)

The `table:protect` attribute specifies whether a table cell is protected. Users cannot edit the content of a cell that is marked as protected.

This attribute is deprecated in favor of the `table:protected` 19.700 attribute.

The defined values for the `table:protect` attribute are:

- `false`: cell is not protected.
- `true`: cell is protected.

The default value for this attribute is `false`.

The `table:protect` attribute is usable with the following elements: `<table:covered-table-cell>` 9.1.5 and `<table:table-cell>` 9.1.4.

The `table:protect` attribute has the data type `boolean` 18.3.3.

## 19.700 table:protected

### 19.700.1 General

The `table:protected` attribute specifies protection for tables.

### 19.700.2 <table:covered-table-cell>

The `table:protected` attribute specifies whether a table cell is protected. Users cannot edit the content of a cell that is marked as protected.

The defined values for the `table:protected` attribute are:

- `false`: cell is not protected.

- `true`: cell is protected.

For a `<table:covered-table-cell>` 9.1.5 element the default value for this attribute is `false`.

The `table:protected` attribute is usable with the following element: `<table:covered-table-cell>` 9.1.5.

The `table:protected` attribute has the data type `boolean` 18.3.3.

### 19.700.3 `<table:scenario>`

The `table:protected` attribute specifies whether the data that is displayed within the scenario is protected from being edited. The attribute is only evaluated if the table on which the scenario displayed is also protected.

The defined values for the `table:protected` attribute are:

- `false`: data is not protected.
- `true`: data is protected.

The `table:protected` attribute is usable with the following element: `<table:scenario>` 9.2.7.

The `table:protected` attribute has the data type `boolean` 18.3.3.

### 19.700.4 `<table:table>`

The `table:protected` attribute specifies whether a table is protected from editing. If the table is protected, the `table:protection-key` 19.701 attribute can specify an authorization requirement for resetting the protection to enable editing. For a protected table, the protection of table cells may be controlled individually by the `style:cell-protect` 20.253 attribute. Table cells are protected only if they have a table cell style assigned whose `style:cell-protect` attribute value is different than `none`.

The defined values for the `table:protected` attribute are:

- `false`: table is not protected.
- `true`: table is protected.

For a `<table:table>` 9.1.2 element the default value for this attribute is `false`.

The `table:protected` attribute is usable with the following element: `<table:table>` 9.1.2.

The `table:protected` attribute has the data type `boolean` 18.3.3.

### 19.700.5 `<table:table-cell>`

The `table:protected` attribute specifies whether a table cell is protected. Users cannot edit the content of a cell that is marked as protected.

The defined values for the `table:protected` attribute are:

- `false`: cell is not protected.
- `true`: cell is protected.

For a `<table:table-cell>` 9.1.4 element the default value for this attribute is `false`.

The `table:protected` attribute is usable with the following element: `<table:table-cell>` 9.1.4.

The `table:protected` attribute has the data type `boolean` 18.3.3.

## 19.701 table:protection-key

The `table:protection-key` attribute, when present, specifies that an authorization is required for removing the protection of a table, table cell or scenario. The authentication procedure is identified by the `table:protection-key-digest-algorithm` attribute.

19.702 The attribute value is binary data that may be used by the authentication procedure.

The password shall be provided as a sequence of bytes in UTF-8 encoding.

The `table:protection-key` attribute is usable with the following elements:  
`<office:spreadsheet>` 3.7 and `<table:table>` 9.1.2.

The `table:protection-key` attribute has the data type `string` 18.2.

## 19.702 table:protection-key-digest-algorithm

The `table:protection-key-digest-algorithm` attribute value is an IRI that identifies an authentication procedure for removing a protection.

If the IRI identifies a message-digest algorithm specified in §5.7 of [xmlenc-core], the value of `table:protection-key` 20.253 attribute shall be the hash value of the password that is required to authorize removal of the protection. The password shall be provided as a sequence of bytes in UTF-8 encoding.

Any other procedures, their identifying IRIs, and their application of `table:protection-key` values are implementation-defined.

Consumers shall support <http://www.w3.org/2000/09/xmldsig#sha1>, which is the default, and <http://www.w3.org/2000/09/xmldsig#sha256>. They may support other algorithms described in §5.7 of [xmlenc-core] or alternative procedures. Producers should use <http://www.w3.org/2000/09/xmldsig#sha256>.

The default value for this attribute is <http://www.w3.org/2000/09/xmldsig#sha1>.

The `table:protection-key-digest-algorithm` attribute is usable with the following elements: `<office:spreadsheet>` 3.7 and `<table:table>` 9.1.2.

The `table:protection-key-digest-algorithm` attribute has the data type `anyIRI` 18.3.2.

## 19.703 table:query-name

A `table:query-name` attribute specifies a query to perform on a database whose data is being imported.

The `table:query-name` attribute is usable with the following element:  
`<table:database-source-query>` 9.4.18.

The `table:query-name` attribute has the data type `string` 18.2.

## 19.704 table:range-usable-as

The `table:range-usable-as` attribute specifies whether the name of the range can be used within the specification of a print range, a filter, a repeating row, or a repeat column.

The defined values for the `table:range-usable-as` attribute are:

- `none`
- a white space-separated list that consists of any of the values `print-range`, `filter`, `repeat-row` or `repeat-column`.

The default value for this attribute is `none`.

The `table:range-usable-as` attribute is usable with the following element:  
`<table:named-range>` 9.4.12.

The values of the `table:range-usable-as` attribute are `none`, or white space separated non-empty lists of one of these values: `print-range`, `filter`, `repeat-row`, or `repeat-column`.

## 19.705 table:refresh-delay

The `table:refresh-delay` attribute specifies a time delay between automatic refresh actions.

The `table:refresh-delay` attribute is usable with the following elements:  
`<table:cell-range-source>` 9.3.1, `<table:database-range>` 9.4.15 and  
`<table:table-source>` 9.2.6.

The `table:refresh-delay` attribute has the data type `duration` 18.2.

## 19.706 table:rejecting-change-id

The `table:rejecting-change-id` attribute specifies the id of a previously-tracked change that has been rejected.

The `table:rejecting-change-id` attribute is usable with the following elements:  
`<table:cell-content-change>` 9.9.17, `<table:deletion>` 9.9.9,  
`<table:insertion>` 9.9.3 and `<table:movement>` 9.9.13.

The `table:rejecting-change-id` attribute has the data type `string` 18.2.

## 19.707 table:rfc-language-tag

The `table:rfc-language-tag` attribute specifies a language identifier according to the rules of [RFC5646], or its successors.

It shall only be used if its value cannot be expressed as a valid combination of the `table:language`, `table:script` and `table:country` attributes.

Producers may add support for consumers that don't support the `table:rfc-language-tag` attribute by specifying `table:language` 19.701, `table:script` 19.661 and `table:country` 19.710 attributes with values that are implementation-dependent.

The `table:rfc-language-tag` attribute augments the `table:language`, `table:script` and `table:country` attributes. It shall only be used if its value could not be expressed as a valid combination of those. The value shall be a language identifier according to the rules of [RFC5646], or its successors. If a fall-back is provided for consumers that do not support the `table:rfc-language-tag` attribute, producers should add `table:language`, `table:script` and `table:country` attributes whose values are as close as possible to the value of the `table:rfc-language-tag` attribute. Producers shall not use values for these attributes that contradict the value of the `table:rfc-language-tag` attribute.

The `table:rfc-language-tag` attribute is usable with the following element:  
`<table:sort>` 9.4.19.

The `table:rfc-language-tag` attribute has the data type `language` 18.3.17.

## 19.708 table:row

The `table:row` attribute specifies the row number of a cell.

The `table:row` attribute is usable with the following elements: `<table:cell-address>` 9.9.18, `<table:source-range-address>` 9.9.14 and `<table:target-range-address>` 9.9.15.

The `table:row` attribute has the data type `integer` 18.2.

## 19.709 `table:scenario-ranges`

The `table:scenario-ranges` attribute specifies the table range that is displayed as a scenario. The value of this attribute is a white-space-separated list of cell range addresses.

The `table:scenario-ranges` attribute is usable with the following element: `<table:scenario>` 9.2.7.

The `table:scenario-ranges` attribute has the data type `cellRangeAddressList` 18.3.6.

## 19.710 `table:script`

The `table:script` attribute specifies the script information for the natural language in which comparisons will occur. The attribute should be used only if necessary according to the rules of §2.2.3 of [RFC5646], or its successors.

The `table:script` attribute is usable with the following element: `<table:sort>` 9.4.19.

The `table:script` attribute has the data type `scriptCode` 18.3.30.

## 19.711 `table:search-criteria-must-apply-to-whole-cell`

The `table:search-criteria-must-apply-to-whole-cell` attribute specifies whether a search pattern matches the entire content of a cell.

**Note:** The `table:search-criteria-must-apply-to-whole-cell` is used with the `<table:filter-condition>` 19.600 element when the `table:data-type` 19.615 attribute has the value `text` or `text-color` or `data-style-color` and the `table:operator` 9.5.5 attribute has a value of: `match`, `!match`, `=`, or `!=`.

The defined values for the `table:search-criteria-must-apply-to-whole-cell` are:

- `false`: search pattern can match a substring at any position within a cell.
- `true`: search pattern matches entire content of a cell.

The default value for this attribute is `true`.

The `table:search-criteria-must-apply-to-whole-cell` attribute is usable with the following element: `<table:calculation-settings>` 9.4.1.

The `table:search-criteria-must-apply-to-whole-cell` attribute has the data type `boolean` 18.3.3.

## 19.712 `table:selected-page`

The `table:selected-page` attribute specifies which value is selected for an automatic filter.

The `table:selected-page` attribute is usable with the following element: `<table:data-pilot-field>` 9.6.7.

The `table:selected-page` attribute has the data type `string` 18.2.

## 19.713 table:show-details

The `table:show-details` attribute specifies whether additional fields are displayed for a member. This attribute changes the behavior of a data pilot only if there are multiple fields with the orientation row or column.

The defined values for the `table:show-details` attribute are:

- `false`: field with a row or column orientation that is not the last field with this orientation, no members are displayed for following fields with the same orientation. Data displayed for these fields is summarized using a `function` specified by the data pilot's `table:function` 19.647 attribute.
- `true`: field with a row or column orientation that is not the last field with this orientation, members are displayed for following fields with the same orientation.

The `table:show-details` attribute is usable with the following element: `<table:data-pilot-member>` 9.6.12.

The `table:show-details` attribute has the data type `boolean` 18.3.3.

## 19.714 table:show-empty

The `table:show-empty` attribute specifies whether fields that do not have any members should be displayed. If this attribute is not present, it is implementation-defined whether fields without members are displayed.

The defined values for the `table:show-empty` attribute are:

- `false`: fields without members are not displayed.
- `true`: fields without members are displayed.

The `table:show-empty` attribute is usable with the following element: `<table:data-pilot-level>` 9.6.8.

The `table:show-empty` attribute has the data type `boolean` 18.3.3.

## 19.715 table:show-filter-button

The `table:show-filter-button` attribute specifies whether a filter button is displayed for a data pilot table. A filter button displays a user interface for a filter if pushed.

The defined values for the `table:show-filter-button` attribute are:

- `false`: filter button is not shown in UI.
- `true`: filter button is shown in UI.

The default value for this attribute is `true`.

The `table:show-filter-button` attribute is usable with the following element: `<table:data-pilot-table>` 9.6.3.

The `table:show-filter-button` attribute has the data type `boolean` 18.3.3.

## 19.716 table:sort-mode

The `table:sort-mode` attribute specifies how to sort the members of a single data pilot field.

The defined values for the `table:sort-mode` attribute are:

- `data`: fields are sorted by values in the data field specified by a `table:data-field` 19.611 attribute.

- **manual**: fields are sorted by the user
- **name**: fields are sorted by name of the field
- **none**: fields are not sorted.

The `table:sort-mode` attribute is usable with the following element: `<table:data-pilot-sort-info>` 9.6.14.

## 19.717 table:source-cell-range-addresses

The `table:source-cell-range-addresses` attribute specifies source cell ranges.

The `table:source-cell-range-addresses` attribute is usable with the following element: `<table:consolidation>` 9.7.

The `table:source-cell-range-addresses` attribute has the data type `cellRangeAddressList` 18.3.6.

## 19.718 table:source-field-name

The `table:source-field-name` attribute specifies the name or label of a column that connects to a field.

There can be multiple `<table:data-pilot-field>` elements with the same value for this attribute.

The `table:source-field-name` attribute is usable with the following elements: `<table:data-pilot-field>` 9.6.7 and `<table:data-pilot-groups>` 9.6.17.

The `table:source-field-name` attribute has the data type `string` 18.2.

## 19.719 table:source-name

The `table:source-name` attribute specifies a source name that is passed to the service implementation. Its value is implementation-dependent and service-specific.

The `table:source-name` attribute is usable with the following element: `<table:source-service>` 9.6.6.

The `table:source-name` attribute has the data type `string` 18.2.

## 19.720 table:sql-statement

An `table:sql-statement` attribute specifies the SQL statement to use when importing data from an SQL database.

The `table:sql-statement` attribute is usable with the following element: `<table:database-source-sql>` 9.4.16.

The `table:sql-statement` attribute has the data type `string` 18.2.

## 19.721 table:start

The `table:start` attribute specifies the start value for a grouping of numeric values. All values that are lower than the start value are contained in a single group, while values that are equal to or higher than the start value are grouped as specified by the `table:grouped-by` 19.650 and `table:step` 19.650 attributes.

The defined values for the `table:start` attribute are:

- **auto**: the lowest value of the field is used as the start value.

- a value of type `double`.

The `table:start` attribute is usable with the following element: `<table:source-range-address> 9.9.14` and `<table:target-range-address> 9.9.15`.

The values of the `table:start` attribute are a value of type `double` **18.2** or `auto`.

## 19.722 `table:start-column`

The `table:start-column` attribute specifies the start-column for a cell range address. The value of a `table:start-column` attribute is inclusive.

The `table:start-column` attribute is usable with the following elements: `<table:source-range-address> 9.9.14` and `<table:target-range-address> 9.9.15`.

The `table:start-column` attribute has the data type `integer` **18.2**.

## 19.723 `table:start-position`

The `table:start-position` attribute specifies the number of the first deleted row or column if multiple rows or columns are deleted. The value of a `table:start-position` attribute is inclusive.

The `table:start-position` attribute is usable with the following element: `<table:movement-cut-off> 9.9.12`.

The `table:start-position` attribute has the data type `integer` **18.2**.

## 19.724 `table:start-row`

The `table:start-row` attribute specifies the start-row for a cell range address. The value of a `table:start-row` attribute is inclusive.

The `table:start-row` attribute is usable with the following elements: `<table:source-range-address> 9.9.14` and `<table:target-range-address> 9.9.15`.

The `table:start-row` attribute has the data type `integer` **18.2**.

## 19.725 `table:start-table`

The `table:start-table` attribute specifies the start-table for a cell range address. The value of a `table:start-table` attribute is inclusive.

The `table:start-table` attribute is usable with the following elements: `<table:source-range-address> 9.9.14` and `<table:target-range-address> 9.9.15`.

The `table:start-table` attribute has the data type `integer` **18.2**.

## 19.726 `table:status`

The `table:status` attribute specifies whether iterative calculations are enabled.

The defined values for the `table:status` attribute are:

- `enable`: iterative calculations will occur.
- `disable`: iterative calculations will not occur.

The default value for this attribute is `disable`.

The `table:status` attribute is usable with the following element: `<table:iteration>` 9.4.3.

The values of the `table:status` attribute are `enable` or `disable`.

## 19.727 `table:step`

The `table:step` attribute specifies the grouping of numeric values, by specifying the distance between the groups.

The `table:step` attribute is usable with the following element: `<table:data-pilot-groups>` 9.6.17.

The `table:step` attribute has the data type `double` 18.2.

## 19.728 `table:steps`

The `table:steps` attribute specifies the maximum number of iterative calculations.

The default value for this attribute is 100.

The `table:steps` attribute is usable with the following element: `<table:iteration>` 9.4.3.

The `table:steps` attribute has the data type `positiveInteger` 18.2.

## 19.729 `table:structure-protected`

The `table:structure-protected` attribute specifies whether a table is protected from the insertion, deletion, moving or renaming of tables in the document. If the table structure is protected and the `table:protection-key` 19.701 attribute is present, an authorization is required for resetting the protection to enable editing.

The defined values for the `table:show-details` attribute are:

- `false`: table is not protected.
- `true`: table is protected.

The default value for this attribute is `false`.

The `table:structure-protected` attribute is usable with the following element: `<office:spreadsheet>` 3.7.

The `table:structure-protected` attribute has the data type `boolean` 18.3.3.

## 19.730 `table:style-name`

### 19.730.1 General

The `table:style-name` attribute specifies styles that are assigned to elements. The style can be either an automatic or common style.

### 19.730.2 `<table:background>`

The `table:style-name` attribute specifies a `<style:style>` element for the table. The style specified by the `<style:style>` has a `style:family` 19.480 attribute with the value `table`.

The `table:style-name` attribute is usable with the following element: `<table:background>` 16.22.

The `table:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.730.3 <table:body>

The `table:style-name` attribute specifies a `<style:style>` element of type `table-cell`.

The `table:style-name` attribute is usable with the following element: `<table:body>` 16.21.6.

The `table:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.730.4 <table:covered-table-cell>

The `table:style-name` attribute specifies a `<style:style>` element of type `table-cell`.

If a cell does not have a cell style assigned, then the `table:default-cell-style-name` attribute of the table row specifies the cell style. If that does not exist, the `table:default-cell-style-name` attribute of the table column specifies the cell style. If that does not exist, the default style with family `table-cell` specifies the style for the cell.

The `table:style-name` attribute is usable with the following element: `<table:covered-table-cell>` 9.1.5.

The `table:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.730.5 <table:even-columns>

The `table:style-name` attribute specifies a `<style:style>` element of type `table-cell`.

The `table:style-name` attribute is usable with the following element: `<table:even-columns>` 16.21.9.

The `table:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.730.6 <table:even-rows>

The `table:style-name` attribute specifies a `<style:style>` element of type `table-cell`.

The `table:style-name` attribute is usable with the following element: `<table:even-rows>` 16.21.7.

The `table:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.730.7 <table:first-column>

The `table:style-name` attribute specifies a `<style:style>` element of type `table-cell`.

The `table:style-name` attribute is usable with the following element: `<table:first-column>` 16.21.4.

The `table:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.730.8 <table:first-row>

The `table:style-name` attribute specifies a `<style:style>` element of type `table-cell`.

The `table:style-name` attribute is usable with the following element: `<table:first-row>` 16.21.2.

The `table:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.730.9 `<table:last-column>`

The `table:style-name` attribute specifies a `<style:style>` element of type `table-cell`.

The `table:style-name` attribute is usable with the following element: `<table:last-column>` 16.21.5.

The `table:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.730.10 `<table:last-row>`

The `table:style-name` attribute specifies a `<style:style>` element of type `table-cell`.

The `table:style-name` attribute is usable with the following element: `<table:last-row>` 16.21.3.

The `table:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.730.11 `<table:odd-columns>`

The `table:style-name` attribute specifies a `<style:style>` element of type `table-cell`.

The `table:style-name` attribute is usable with the following element: `<table:odd-columns>` 16.21.10.

The `table:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.730.12 `<table:odd-rows>`

The `table:style-name` attribute specifies a `<style:style>` element of type `table-cell`.

The `table:style-name` attribute is usable with the following element: `<table:odd-rows>` 16.21.8.

The `table:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.730.13 `<table:table-cell>`

The `table:style-name` attribute specifies a `<style:style>` element of type `table-cell`.

If a cell does not have a cell style assigned, then the `table:default-cell-style-name` attribute of the table row specifies the cell style. If that does not exist, the `table:default-cell-style-name` attribute of the table column specifies the cell style. If that does not exist, the default style with family `table-cell` specifies the style for the cell.

The `table:style-name` attribute is usable with the following element: `<table:table-cell>` 9.1.4.

The `table:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.730.14 `<table:table-column>`

The `table:style-name` attribute specifies a `<style:style>` element of type `table-column`.

The `table:style-name` attribute is usable with the following element: `<table:table-column>` 9.1.6.

The `table:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.730.15 `<table:table-row>`

The `table:style-name` attribute specifies a `<style:style>` element of type `table-row`.

The `table:style-name` attribute is usable with the following element: `<table:table-row>` 9.1.3.

The `table:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.730.16 `<table:table>`

The `table:style-name` attribute specifies a `<style:style>` element of type `table`.

The `table:style-name` attribute is usable with the following element: `<table:table>` 9.1.2.

The `table:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.731 `table:tab-color` (moved to 20.442)

### 19.732 `table:table`

The `table:table` attribute specifies a table number for insertions, deletions and for table cells.

The `table:table` attribute is usable with the following elements: `<table:cell-address>` 9.9.18, `<table:deletion>` 9.9.9, `<table:insertion>` 9.9.3, `<table:source-range-address>` 9.9.14 and `<table:target-range-address>` 9.9.15.

The `table:table` attribute has the data type `integer` 18.2.

### 19.733 `table:table-background`

The `table:table-background` attribute specifies whether a shape is in the table background if the drawing shape is included in a spreadsheet document.

The defined values for the `table:table-background` attribute are:

- `false`: shape is included in foreground of a table.
- `true`: shape is included in background of a table.

The `table:table-background` attribute is usable with the following elements: `<dr3d:scene>` 10.5.2, `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:connector>` 10.3.10, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:g>` 10.3.15, `<draw:line>` 10.3.3, `<draw:measure>` 10.3.12, `<draw:page-thumbnail>` 10.3.14, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2, `<draw:regular-polygon>` 10.3.6 and `<office:annotation>` 14.1.

The `table:table-background` attribute has the data type `boolean` 18.3.3.

## 19.734 table:table-name

The `table:table-name` attribute specifies the name of the table in the source document to which the current table is linked. If the attribute is omitted, the current table is linked to the first table in the source document.

The `table:table-name` attribute is usable with the following element: `<table:table-source>` 9.2.6.

The `table:table-name` attribute has the data type `string` 18.2.

## 19.735 table:target-cell-address

The `table:target-cell-address` attribute specifies a target cell address.

The `table:target-cell-address` attribute is usable with the following element: `<table:consolidation>` 9.7.

The `table:target-cell-address` attribute has the data type `cellAddress` 18.3.4.

## 19.736 table:target-range-address

### 19.736.1 General

The `table:target-range-address` attribute specifies a cell range address. The value of this attribute shall be an absolute cell address that contains a table name. The dollar signs "\$" (U+0024, DOLLAR SIGN) that indicate an absolute address may be omitted.

### 19.736.2 <table:data-pilot-table>

The `table:target-range-address` attribute specifies the cell range where a data pilot table will be displayed.

The `table:target-range-address` attribute is usable with the following element: `<table:data-pilot-table>` 9.6.3.

The `table:target-range-address` attribute has the data type `cellRangeAddress` 18.3.5.

### 19.736.3 <table:database-range>

The `table:target-range-address` attribute specifies the cell range address of a database range.

The `table:target-range-address` attribute is usable with the following element: `<table:database-range>` 9.4.15.

The `table:target-range-address` attribute has the data type `cellRangeAddress` 18.3.5.

### 19.736.4 <table:filter>

The `table:target-range-address` attribute specifies the cell range for the result of a filter operation, but all table rows remain visible. If the attribute is not present, the rows that do not match the filter conditions are not displayed.

The `table:target-range-address` attribute is usable with the following element: `<table:filter>` 9.5.2.

The `table:target-range-address` attribute has the data type `cellRangeAddress` 18.3.5.

## 19.736.5 <table:sort>

The `table:target-range-address` attribute specifies the cell range for the result of a sort operation. If the attribute is not present, the cell range for the result is identical to the cell range of the database range.

The `table:target-range-address` attribute is usable with the following element:  
`<table:sort>` 9.4.19.

The `table:target-range-address` attribute has the data type `cellRangeAddress` 18.3.5.

## 19.737 table:template-name

The `table:template-name` attribute specifies a table template by name.

The `table:template-name` attribute is usable with the following element:  
`<table:table>` 9.1.2.

The `table:template-name` attribute has the data type `string` 18.2.

## 19.738 table:title

The `table:title` attribute specifies a title of a help or error message.

The `table:title` attribute is usable with the following elements: `<table:error-message>` 9.4.7 and `<table:help-message>` 9.4.6.

The `table:title` attribute has the data type `string` 18.2.

## 19.739 table:track-changes

The `table:track-changes` attribute specifies whether change tracking is enabled.

The defined values for the `table:track-changes` attribute are:

- `false`: changes are not tracked.
- `true`: changes are tracked.

The default value for this attribute is `false`.

The `table:track-changes` attribute is usable with the following element:  
`<table:tracked-changes>` 9.9.2.

The `table:track-changes` attribute has the data type `boolean` 18.3.3.

## 19.740 table:type

### 19.740.1 General

The `table:type` attribute specifies a type for a table.

### 19.740.2 <table:data-pilot-field-reference>

The `table:type` attribute specifies the processing of a referenced category column for display.

The defined values for the `table:type` attribute are:

- `column-percentage`: Same as `row-percentage`, but the total for the result's column is used.

- **index:** The row and column totals and the grand total are calculated as described for **row-percentage**, and then are used to calculate the following expression: (original result \* grand total) / (row total \* column total). Division by zero is an error.
- **member-difference:** From each result, the value calculated for the category column member specified by the `table:member-type` and `table:member-name` attributes is subtracted.
- **member-percentage:** Each result is divided by the value calculated for the category column member specified by the `table:member-type` 19.673 and `table:member-name` 19.672 attributes. Division by zero is an error. Empty results are shown as "0". If the `table:member-type` attribute has the value `previous`, "1" is displayed as first value. If the `table:member-type` attribute has the value `next`, "1" is displayed as last value.
- **member-percentage-difference:** From each result, the value calculated for the category column member specified by the `table:member-type` 19.673 and `table:member-name` 19.672 attributes is subtracted, and the result is divided by this value again. Division by zero is an error. Otherwise, the rules for **member-difference** apply.
- **none:** This value means that the results in the data fields are displayed unmodified.
- **row-percentage:** Each result is divided by the total result for its row in the data pilot table. If there are multiple data fields, the total for the result's data field is used. If there are subtotals with manually-selected summary functions, the total is calculated with the data field's summary function. Division by zero results in an error.
- **running-total:** Each result is added to the sum of the results for preceding members in the referenced category column, in the reference field's sort order, and the total sum is shown.
- **total-percentage:** Same as **row-percentage**, but the grand total for the result's data field is used.

The `table:type` attribute is usable with the following element: `<table: data-pilot-field-reference>` 9.6.16.

The values of the `table:type` attribute are `none`, `member-difference`, `member-percentage`, `member-percentage-difference`, `running-total`, `row-percentage`, `column-percentage`, `total-percentage` or `index`.

### 19.740.3 `<table:deletion>`

The `table:type` attribute specifies the type of a deletion.

The defined values for the `table:type` attribute are:

- `column`: column deleted.
- `row`: row deleted.
- `table`: table deleted.

The `table:type` attribute is usable with the following element: `<table:deletion>` 9.9.9.

The values of the `table:type` attribute are `row`, `column` or `table`.

### 19.740.4 `<table:insertion>`

The `table:type` attribute specifies the type of an insertion.

The defined values for the `table:type` attribute are:

- `column`: column inserted.
- `row`: row inserted.

- `table:table` inserted.

The `table:type` attribute is usable with the following element: `<table:insertion>` 9.9.3.

The values of the `table:type` attribute are `row`, `column` or `table`.

## 19.741 `table:use-banding-columns-styles`

The `table:use-banding-columns-styles` attribute specifies if the styles referenced by the `<table:even-columns>` 16.21.9 and `<table:odd-columns>` 16.21.10 elements of a table template referenced by the `table:template-name` 19.737 attribute are used for cells that are contained in even and odd columns.

The defined values for the `table:use-banding-columns-styles` attribute are:

- `false`: cells are not styled based on even or odd column location.
- `true`: cells are styled based on even or odd column location.

The default value for this attribute is `false`.

The `table:use-banding-columns-styles` attribute is usable with the following element: `<table:table>` 9.1.2.

The `table:use-banding-columns-styles` attribute has the data type `boolean` 18.3.3.

## 19.742 `table:use-banding-rows-styles`

The `table:use-banding-rows-styles` attribute specifies if the styles referenced by the `<table:even-rows>` 16.21.7 and `<table:odd-rows>` 16.21.8 elements of a table template referenced by the `table:template-name` 19.737 attribute are used for cells that are contained in even and odd rows.

The defined values for the `table:use-banding-rows-styles` attribute are:

- `false`: cells not styled based on even or odd row location.
- `true`: cells styled based on even or odd row location.

The default value for this attribute is `false`.

The `table:use-banding-rows-styles` attribute is usable with the following element: `<table:table>` 9.1.2.

The `table:use-banding-rows-styles` attribute has the data type `boolean` 18.3.3.

## 19.743 `table:use-first-column-styles`

The `table:use-first-column-styles` attribute specifies if the styles referenced by the `<table:first-column>` 16.21.4 element of the table template referenced by the `table:template-name` 19.737 attribute are used for cells that are contained in the table's first column.

The defined values for the `table:use-first-column-styles` attribute are:

- `false`: cells not styled based on first column location.
- `true`: cells styled based on first column location.

The default value for this attribute is `false`.

The `table:use-first-column-styles` attribute is usable with the following element: `<table:table>` 9.1.2.

The `table:use-first-column-styles` attribute has the data type `boolean` 18.3.3.

## 19.744 table:use-first-row-styles

The `table:use-first-row-styles` attribute specifies if the styles referenced by the `<table:first-row>` 16.21.2 element of the table template referenced by the `table:template-name` 19.737 attribute are used for cells that are contained in the table's first row.

The defined values for the `table:use-first-row-styles` attribute are:

- `false`: cells not styled based on first row location.
- `true`: cells styled based on first row location.

The default value for this attribute is `false`.

The `table:use-first-row-styles` attribute is usable with the following element:  
`<table:table>` 9.1.2.

The `table:use-first-row-styles` attribute has the data type `boolean` 18.3.3.

## 19.745 table:use-last-column-styles

The `table:use-last-column-styles` attribute specifies if the styles referenced by the `<table:last-column>` 16.21.5 element of the table template referenced by the `table:template-name` 19.737 attribute are used for cells that are contained in the table's last column.

The defined values for the `table:use-last-column-styles` attribute are:

- `false`: cells not styled based on last column location.
- `true`: cells styled based on last column location.

The default value for this attribute is `false`.

The `table:use-last-column-styles` attribute is usable with the following element:  
`<table:table>` 9.1.2.

The `table:use-last-column-styles` attribute has the data type `boolean` 18.3.3.

## 19.746 table:use-last-row-styles

The `table:use-last-row-styles` attribute specifies if the styles referenced by the `<table:last-row>` 16.21.3 element of the table template referenced by the `table:template-name` 19.737 attribute are used for cells that are contained in the table's last row.

The defined values for the `table:use-last-row-styles` attribute are:

- `false`: cells not styled based on last row location.
- `true`: cells styled based on last row location.

The default value for this attribute is `false`.

The `table:use-last-row-styles` attribute is usable with the following element:  
`<table:table>` 9.1.2.

The `table:use-last-row-styles` attribute has the data type `boolean` 18.3.3.

## 19.747 table:use-labels

The `table:use-labels` attribute specifies whether labels should be used by the consolidation for rows, columns or both. If labels are used for rows or columns, mathematical functions are applied to cells with equally-labeled rows or columns rather than to cells with the same relative cell address.

The defined values for the `table:type` attribute are:

- `both`: column and row labels used for consolidation.
- `column`: column labels used for consolidation.
- `none`: labels not used for consolidation.
- `row`: row labels used for consolidation.

The default value for this attribute is `none`.

The `table:use-labels` attribute is usable with the following element:  
`<table:consolidation>` 9.7.

The values of the `table:use-labels` attribute are `none`, `row`, `column` or `both`.

## 19.748 table:use-regular-expressions

The `table:use-regular-expressions` attribute specifies whether regular expressions are enabled for character-string comparisons and when searching.

Regular expressions are implementation-dependent expressions that, at a minimum, conform to the requirements of Conformance Clause C1 of [UTR18].

The `table:use-wildcard` 19.749 attribute and the `table:use-regular-expressions` attribute are mutually exclusive. The attribute values cannot be `true` for both attributes.

The defined values for the `table:use-regular-expressions` attribute are:

- `false`: regular expressions not enabled for string comparisons and searches.
- `true`: regular expressions enabled for string comparisons and searches.

The default value for this attribute is `true`.

The `table:use-regular-expressions` attribute is usable with the following element:  
`<table:calculation-settings>` 9.4.1.

The `table:use-regular-expressions` attribute has the data type `boolean` 18.3.3.

## 19.749 table:use-wildcards

The `table:use-wildcards` attribute specifies whether wildcards are enabled for character-string comparisons and when searching.

If enabled, in a query or search string of a function, the "?" (U+003F, QUESTION MARK), "\*" (U+002A, ASTERISK), and "~" (U+007E, TILDE) are defined as:

- "?" (U+003F, QUESTION MARK): matches any single character;
- "\*" (U+002A, ASTERISK): matches any sequence of characters, including an empty string;
- "~" (U+007E, TILDE): escapes the special meaning of a QUESTION MARK, ASTERISK or TILDE character that follows immediately after the TILDE character.

The `table:use-regular-expressions` 19.748 attribute and the `table:use-wildcards` attribute are mutually exclusive. The attribute values cannot be `true` for both attributes.

The defined values for the `table:use-wildcards` attribute are:

- `false`: wildcards are not enabled for character-string comparisons and searching.
- `true`: wildcards are enabled for character-string comparisons and searching.

The default value for this attribute is `false`.

The `table:use-wildcards` attribute is usable with the following element:  
`<table:calculation-settings>` 9.4.1.

The `table:use-wildcards` attribute has the data type `boolean` 18.3.3.

## 19.750 `table:used-hierarchy`

The `table:used-hierarchy` attribute specifies a unique number assigned by an external component to select the hierarchy that should be applied to a source field.

If the data pilot source is provided by an external component or service, the data contained within category columns may be grouped by its value and it may be further divided into sub-groups or hierarchies. If an external component supports hierarchies, it shall assign unique numbers to those hierarchies.

The default value for this attribute is `-1`. The default value of `-1` means no hierarchy should be applied to a source field.

The `table:used-hierarchy` attribute is usable with the following element:  
`<table:data-pilot-field>` 9.6.7.

The `table:used-hierarchy` attribute has the data type `integer` 18.2.

## 19.751 `table:user-name`

The `table:user-name` attribute specifies the user name needed to access a source. It is passed to a service implementation. Its value is implementation-dependent and service-specific.

The `table:user-name` attribute is usable with the following element: `<table:source-service>` 9.6.6.

The `table:user-name` attribute has the data type `string` 18.2.

## 19.752 `table:value`

### 19.752.1 `<table:filter-condition>`

The `table:value` attribute specifies a value for the filter condition.

The `table:value` attribute is usable with the following element: `<table:filter-condition>` 9.5.5.

The values of the `table:value` attribute are a value of type `string` 18.2 or a value of type `double` 18.2.

The values of the `table:value` attribute are a value of type `color` 18.3.9 or `window-font-color`.

The values of the `table:value` attribute are a value of type `color` 18.3.9 or `transparent`.

### 19.752.2 `<table:filter-set-item>`

The `table:value` attribute specifies a single value within a set of filter values.

The `table:value` attribute is usable with the following element: `<table:filter-set-item>` 9.5.6.

The `table:value` attribute has the data type `string` 18.2.

## 19.753 `table:value-type`

The `table:value-type` attribute specifies the value type of a null date.

The default value for this attribute is `date`.

The `table:value-type` attribute is usable with the following element: `<table:null-date>` 9.4.2.

The only value of the `table:value-type` attribute is `date`.

## 19.754 `table:visibility`

The `table:visibility` attribute specifies whether a row or column is visible.,

The defined values for the `table:visibility` attribute are:

- `collapse`: a row or column is not visible.
- `filter`: a row or column is not visible as the result of applying a filter.
- `visible`: a row or column is visible.

The default value for this attribute is `visible`.

The `table:visibility` attribute is usable with the following elements: `<table:table-column>` 9.1.6 and `<table:table-row>` 9.1.3.

The values of the `table:visibility` attribute are `visible`, `collapse` or `filter`.

## 19.755 `text:active`

The `text:active` attribute specifies the setting of the page variable.

The value of an active page variable is increased on each page.

An inactive page variable is not incremented between pages.

The defined values for the `text:active` attribute are:

- `false`: page variable is inactive.
- `true`: page variable is active.

The default value for this attribute is `false`.

The `text:active` attribute is usable with the following element: `<text:page-variable-set>` 7.7.1.2.

The `text:active` attribute has the data type `boolean` 18.3.3.

## 19.756 `text:address`

The `text:address` attribute specifies the address for a bibliography index entry.

The `text:address` attribute is usable with the following element: `<text:bibliography-mark>` 8.1.11.

The `text:address` attribute has the data type `string` 18.2.

## 19.757 `text:alphabetical-separators`

The `text:alphabetical-separators` attribute specifies whether entries beginning with the same letter are grouped and separated from the entries beginning with the next letter.

The defined values for the `text:alphabetical-separators` attribute are:

- `false`: entries beginning with the same letter are not grouped together and separated from entries beginning with the next letter.
- `true`: entries beginning with the same letter are grouped together and separated from entries beginning with the next letter.

The default value for this attribute is `false`.

The `text:alphabetical-separators` attribute is usable with the following element:  
`<text:alphabetical-index-source>` 8.8.2.

The `text:alphabetical-separators` attribute has the data type `boolean` 18.3.3.

## 19.758 text:anchor-page-number

The `text:anchor-page-number` attribute specifies the physical page number of an anchor if the drawing object is bound to a page within a text document.

The `text:anchor-page-number` attribute is usable with the following elements:  
`<dr3d:scene>` 10.5.2, `<draw:caption>` 10.3.11, `<draw:circle>` 10.3.8, `<draw:connector>` 10.3.10, `<draw:control>` 10.3.13, `<draw:custom-shape>` 10.6.1, `<draw:ellipse>` 10.3.9, `<draw:frame>` 10.4.2, `<draw:g>` 10.3.15, `<draw:line>` 10.3.3, `<draw:measure>` 10.3.12, `<draw:page-thumbnail>` 10.3.14, `<draw:path>` 10.3.7, `<draw:polygon>` 10.3.5, `<draw:polyline>` 10.3.4, `<draw:rect>` 10.3.2, `<draw:regular-polygon>` 10.3.6 and `<office:annotation>` 14.1.

The `text:anchor-page-number` attribute has the data type `positiveInteger` 18.2.

## 19.759 text:anchor-type

The `text:anchor-type` attribute specifies how a drawing shape is bound to a text document. The anchor position is the point at which a drawing shape is bound to a text document. The defined values for the `text:anchor-type` attribute are shown in Table 18.

The same attribute is available as graphic property 20.418.

If the attribute is given both as element attribute and as graphic property 20.418, then the element attribute has precedence over the graphic property.

If neither an element attribute nor a graphic property is given or can be found by inheritance, then the attribute `text:anchor-type` defaults to value "paragraph".

Table 18 - Text anchor positions

<b>If the value of the <code>text:anchor-type</code> attribute is ...</b>	<b>The anchor position is...</b>	<b>The drawing shape element shall be located in file text ...</b>	<b>Notes</b>
as-char	There is no anchor position. The drawing shape behaves like a character.	At the position where the character appears in the document.	
char	The character after the drawing shape element.	Just before the character.	
frame	The parent text box that the current drawing shape element is contained in.	In the element representing the text box to which the drawing object is bound.	

<b>If the value of the text:anchor-type attribute is ...</b>	<b>The anchor position is...</b>	<b>The drawing shape element shall be located in file text ...</b>	<b>Notes</b>
		If an image is bound to a text box, the image element is located in the text box element.	
page	The page that has the same physical page number as the value of the text:anchor-page-number 19.758 attribute that is attached to the drawing shape element. If no text:anchor-page-number attribute is given, the anchor position is the page at which the character behind the drawing object element appears.	<p>Either</p> <ul style="list-style-type: none"> <li>At the start of the document body, outside any paragraph or frame, provided a text:anchor-page-number attribute is given.</li> </ul> <p>Or</p> <ul style="list-style-type: none"> <li>Inside any paragraph element that is not contained in a header, footer, footnote, or text box, if a text:anchor-page-number attribute is not given.</li> </ul>	The physical page number is the number assigned to the page if all pages in the document are counted starting with page 1.
paragraph	The paragraph that the current drawing shape element is contained in.	At the start of the paragraph element.	

The text:anchor-type attribute is usable with the following elements: <dr3d:scene> 10.5.2, <draw:caption> 10.3.11, <draw:circle> 10.3.8, <draw:connector> 10.3.10, <draw:control> 10.3.13, <draw:custom-shape> 10.6.1, <draw:ellipse> 10.3.9, <draw:frame> 10.4.2, <draw:g> 10.3.15, <draw:line> 10.3.3, <draw:measure> 10.3.12, <draw:page-thumbnail> 10.3.14, <draw:path> 10.3.7, <draw:polygon> 10.3.5, <draw:polyline> 10.3.4, <draw:rect> 10.3.2, <draw:regular-polygon> 10.3.6 and <office:annotation> 14.1.

The values of the text:anchor-type attribute are page, frame, paragraph, char or as-char.

## 19.760 text:annotate

The text:annotate attribute specifies the annotation for a bibliography index entry.

The text:annotate attribute is usable with the following element: <text:bibliography-mark> 8.1.11.

The text:annotate attribute has the data type string 18.2.

## 19.761 text:author

The text:author attribute specifies the author or authors for a bibliography index entry.

The text:author attribute is usable with the following element: <text:bibliography-mark> 8.1.11.

The `text:author` attribute has the data type `string` 18.2.

## 19.762 `text:bibliography-data-field`

The `text:bibliography-data-field` attribute specifies which part of a bibliography data field will be displayed.

The defined values for the `text:bibliography-data-field` attribute are:

- `address`: 19.756.
- `annotate`: 19.760.
- `author`: 19.761.
- `bibliography-type`: 19.763.
- `booktitle`: 19.764.
- `chapter`: 19.772.
- `custom1`: 19.792.
- `custom2`: 19.793.
- `custom3`: 19.794.
- `custom4`: 19.795.
- `custom5`: 19.796.
- `edition`: 19.806.
- `editor`: 19.807.
- `howpublished`: 19.813.
- `identifier`: 19.816.
- `institution`: 19.820.
- `isbn`: 19.823.
- `issn`: 19.824.
- `journal`: 19.825.
- `month`: 19.840.
- `note`: 19.842.
- `number`: 19.844.
- `organizations`: 19.849.
- `pages`: 19.852.
- `publisher`: 19.858.
- `report-type`: 19.862.
- `school`: 19.866.
- `series`: 19.868.
- `title`: 19.888.
- `url`: 19.890.
- `volume`: 19.908.
- `year`: 19.909.

The `text:bibliography-data-field` attribute is usable with the following element:  
`<text:index-entry-bibliography>` 8.13.5.

The values of the `text:bibliography-data-field` attribute are address, annote, author, bibliography-type, booktitle, chapter, custom1, custom2, custom3, custom4, custom5, edition, editor, howpublished, identifier, institution, isbn, issn, journal, month, note, number, organizations, pages, publisher, report-type, school, series, title, url, volume or year.

## 19.763 `text:bibliography-type`

The `text:bibliography-type` attribute specifies to which type of bibliographical entry a template applies.

The defined values for the `text:bibliography-type` attribute are:

- `article`: an article in a journal or magazine.
- `book`: a book with a named publisher.
- `booklet`: a work that is printed and bound, but without a named publisher or sponsoring institution.
- `conference`: an article in conference proceedings.
- `custom1`: the first of five user-defined fields for a bibliographic entry.
- `custom2`: the second of five user-defined fields for a bibliographic entry.
- `custom3`: the third of five user-defined fields for a bibliographic entry.
- `custom4`: the fourth of five user-defined fields for a bibliographic entry.
- `custom5`: the fifth of five user-defined fields for a bibliographic entry.
- `email`: an email address.
- `inbook`: a part of a book that can be identified.
- `incollection`: a part of a book that has its own title.
- `inproceedings`: a paper in conference proceedings.
- `journal`: a journal name.
- `manual`: technical documentation.
- `mastersthesis`: a masters thesis.
- `misc`: a type classification when more specific values are not appropriate.
- `phdthesis`: a PhD thesis.
- `proceedings`: the proceedings of a conference.
- `techreport`: a report published by a school or other institution.
- `unpublished`: a document with an author and title but not formally published.
- `www`: the IRI for a document.

**Note:** The definitions for `article`, `book`, `booklet`, `inbook`, `incollection`, `inproceedings`, `journal`, `manual`, `mastersthesis`, `misc`, `phdthesis`, `proceedings`, `techreport`, and `unpublished` are based on [BibTeXing].

The `text:bibliography-type` attribute is usable with the following elements:  
`<text:bibliography-entry-template>` 8.9.3 and `<text:bibliography-mark>` 8.1.11.

The values of the `text:bibliography-type` attribute are `article`, `book`, `booklet`, `conference`, `custom1`, `custom2`, `custom3`, `custom4`, `custom5`, `email`, `inbook`, `incollection`, `inproceedings`, `journal`, `manual`, `mastersthesis`, `misc`, `phdthesis`, `proceedings`, `techreport`, `unpublished` or `www`.

## 19.764 `text:booktitle`

The `text:booktitle` attribute specifies the book title for a bibliography index entry.

The `text:booktitle` attribute is usable with the following element:

`<text:bibliography-mark>` 8.1.11.

The `text:booktitle` attribute has the data type `string` 18.2.

## 19.765 `text:bullet-char`

The `text:bullet-char` attribute specifies the [UNICODE] character to use as the bullet in a bulleted list style.

The defined values for the `text:bullet-char` attribute are shown in Table 19:

*Table 19 - Typical bullet characters*

<b>UNICODE Character Code</b>	<b>Typical Shape</b>	<b>UNICODE Character Name</b>
U+2022	•	BULLET
U+25CF	●	BLACK CIRCLE
U+2714	✓	HEAVY CHECK MARK
U+2717	✗	BALLOT X
U+2794	➔	HEAVY WIDE-HEADED RIGHTWARDS ARROW
U+27A2	➤	THREE-D TOP-LIGHTED RIGHTWARDS ARROWHEAD

**Note:** These characters need not be available in all fonts.

The `text:bullet-char` attribute is usable with the following element: `<text:list-level-style-bullet>` 16.33.

The `text:bullet-char` attribute has the data type `character` 18.3.7.

## 19.766 `text:bullet-relative-size`

The `text:bullet-relative-size` attribute specifies a percentage value for the bullet size relative to the font size of the paragraphs in the bullet list.

The `text:bullet-relative-size` attribute is usable with the following element: `<text:list-level-style-bullet>` 16.33.

The `text:bullet-relative-size` attribute has the data type `percent` 18.3.24.

## 19.767 `text:capitalize-entries`

The `text:capitalize-entries` attribute specifies whether the entries in an index are capitalized.

The defined values for the `text:capitalize-entries` attribute are:

- `false`: the case of entries in an index should not change.
- `true`: the entries in an index should be capitalized.

The `text:capitalize-entries` attribute controls capitalization of index entries during the generation of indexes. Index entries may be entered in either upper or lower case, without regard to the `text:capitalize-entries` attribute.

The default value for this attribute is `false`.

The `text:capitalize-entries` attribute is usable with the following element:  
`<text:alphabetical-index-source>` 8.8.2.

The `text:capitalize-entries` attribute has the data type `boolean` 18.3.3.

## 19.768 `text:c`

The `text:c` attribute specifies the number of “ ” (U+0020, SPACE) characters that a `<text:s>` element represents. A missing `text:c` attribute is interpreted as a single “ ” (U+0020, SPACE) character.

The `text:c` attribute is usable with the following element: `<text:s>` 6.1.3.

The `text:c` attribute has the data type `nonNegativeInteger` 18.2.

## 19.769 `text:caption-sequence-format`

The `text:caption-sequence-format` attribute specifies the format of entries for an index of illustrations or tables.

The defined values for the `text:caption-sequence-format` attribute are:

- `caption`: text of the caption in which the referenced sequence field is included (without sequence number and value).
- `category-and-value`: displays the name and value of the referenced sequence field.
- `text`: displays the value of the referenced sequence field.

The `text:caption-sequence-format` attribute is usable with the following elements:  
`<text:illustration-index-source>` 8.4.2 and `<text:table-index-source>` 8.5.2.

The values of the `text:caption-sequence-format` attribute are `text`, `category-and-value` or `caption`.

## 19.770 `text:caption-sequence-name`

The `text:caption-sequence-name` attribute specifies the sequence by which captions are identified.

If the `text:use-caption` attribute is set to `true`, this attribute shall be present.

**Note:** Captions are regular paragraphs. For index generation, a paragraph is considered to be a caption if it contains the sequence variable with the name specified by the `text:caption-sequence-name` attribute.

The `text:caption-sequence-name` attribute is usable with the following elements:  
`<text:illustration-index-source>` 8.4.2 and `<text:table-index-source>` 8.5.2.

The `text:caption-sequence-name` attribute has the data type `string` 18.2.

## 19.771 text:change-id

The `text:change-id` attribute specifies the id that links `<text:change-start>`, `<text:change-end>`, and `<text:change>` elements to the `<text:changed-region>` element that contains change-tracking information.

The `text:change-id` attribute is usable with the following elements: `<text:change>` 5.5.8.4, `<text:change-end>` 5.5.8.3 and `<text:change-start>` 5.5.8.2.

The `text:change-id` attribute has the data type `IDREF` 18.2.

## 19.772 text:chapter

The `text:chapter` attribute specifies the chapter reference for a bibliography index entry.

The `text:chapter` attribute is usable with the following element: `<text:bibliography-mark>` 8.1.11.

The `text:chapter` attribute has the data type `string` 18.2.

## 19.773 text:citation-body-style-name

The `text:citation-body-style-name` attribute specifies the text style to use for a note citation in the text flow.

The `text:citation-body-style-name` attribute is usable with the following element: `<text:notes-configuration>` 16.31.3.

The `text:citation-body-style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.774 text:citation-style-name

The `text:citation-style-name` attribute specifies the text style to use for a note citation within the footnote.

The `text:citation-style-name` attribute is usable with the following element: `<text:notes-configuration>` 16.31.3.

The `text:citation-style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.775 text:class-names

### 19.775.1 General

A `text:class-names` attribute specifies a white-space-separated list of style names. The referenced styles are applied in the order they are contained in the list.

If both `text:style-name` and `text:class-names` are present, the style referenced by the `text:style-name` attribute is applied before the styles referenced by `text:class-names` attribute. If a conditional style is specified together with a `text:class-names` attribute, but without a `text:style-name` attribute, the `text:style-name` 19.880 attribute is assumed to have the value of the first style name in the list defined by the `text:class-name` attribute.

### 19.775.2 <text:h>

A `text:class-names` attribute specifies a white-space-separated list of paragraph style names.

The `text:class-names` attribute is usable with the following element: `<text:h>` 5.1.2.

The `text:class-names` attribute has the data type `styleNameRefs` 18.3.34.

### 19.775.3 <text:p>

A `text:class-names` attribute specifies a white-space-separated list of paragraph style names.

The `text:class-names` attribute is usable with the following element: `<text:p>` 5.1.3.

The `text:class-names` attribute has the data type `styleNameRefs` 18.3.34.

### 19.775.4 <text:span>

A `text:class-names` attribute specifies a white-space-separated list of text style names.

The `text:class-names` attribute is usable with the following element: `<text:span>` 6.1.7.

The `text:class-names` attribute has the data type `styleNameRefs` 18.3.34.

### 19.776 text:column-name

The `text:column-name` attribute specifies the name of the column from which data is displayed. The value of this attribute shall be a column name contained in the current selection as specified by the data source.

The `text:column-name` attribute is usable with the following element: `<text:database-display>` 7.6.3.

The `text:column-name` attribute has the data type `string` 18.2.

### 19.777 text:combine-entries

The `text:combine-entries` attribute specifies whether multiple entries for a single word should be combined into a single entry.

The defined values for the `text:combine-entries` attribute are:

- `false`: multiple entries for a single word should not be combined into a single entry.
- `true`: multiple entries for a single word should be combined into a single entry.

The default value for this attribute is `true`.

The `text:combine-entries` attribute is usable with the following element: `<text:alphabetical-index-source>` 8.8.2.

The `text:combine-entries` attribute has the data type `boolean` 18.3.3.

### 19.778 text:combine-entries-with-dash

The `text:combine-entries-with-dash` attribute specifies that multiple entries for a single word that should be displayed as a single entry, as per a `text:combine-entries` attribute, should be displayed as a range of numbers separated by a dash.

The defined values for the `text:combine-entries-with-dash` attribute are:

- `false`: multiple entries for a single word should not be displayed as a range of numbers separated by a dash.
- `true`: multiple entries for a single word should be displayed as a range of numbers separated by a dash.

The default value for this attribute is `false`.

The `text:combine-entries-with-dash` attribute is usable with the following element: `<text:alphabetical-index-source>` 8.8.2.

The `text:combine-entries-with-dash` attribute has the data type `boolean` 18.3.3.

## 19.779 text:combine-entries-with-pp

The `text:combine-entries-with-pp` attribute specifies that multiple entries for a single word that should be displayed as a single entry, as per a `text:combine-entries` attribute, should be displayed as the start page number, followed by a "ff", or the appropriate label for the chosen language.

- `false`: multiple entries for a single word should not be displayed as the start page number, followed by "ff", or the appropriate label for the chosen language.
- `true`: multiple entries for a single word should be displayed as the start page number, followed by "ff", or the appropriate label for the chosen language.

The default value for this attribute is `true`.

The `text:combine-entries-with-pp` attribute is usable with the following element:  
`<text:alphabetical-index-source>` 8.8.2.

The `text:combine-entries-with-pp` attribute has the data type `boolean` 18.3.3.

## 19.780 text:comma-separated

The `text:comma-separated` attribute specifies how to treat multiple index entries. Instead of listing each index entry on a separate line, multiple entries can be listed on a single line separated by a comma "," (U+002C, COMMA). If the value of this attribute is `true`, multiple entries are listed on a single line separated by a comma. By default, the value of this attribute is `false` and each index entry is displayed on a separate line.

The defined values for the `text:comma-separated` attribute are:

- `false`: each index entry for a word is displayed on a separate line.
- `true`: each index entry for a word is listed on a single line separated by a comma "," (U+002C, COMMA).

The default value for this attribute is `false`.

The `text:comma-separated` attribute is usable with the following element:  
`<text:alphabetical-index-source>` 8.8.2.

The `text:comma-separated` attribute has the data type `boolean` 18.3.3.

## 19.781 text:cond-style-name

The `text:cond-style-name` attribute specifies a conditional-style, that is, a style that contains conditions and maps to other styles. 16.3 The style referenced shall be a paragraph style.

If a conditional style is applied to a paragraph, the `text:style-name` attribute should contain the name of the style that was the result of the conditional style evaluation, while the conditional style name itself is the value of the `text:cond-style-name` attribute. The referenced style can be a common style or an automatic style.

The `text:cond-style-name` attribute is usable with the following elements: `<text:h>` 5.1.2 and `<text:p>` 5.1.3.

The `text:cond-style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.782 text:condition

### 19.782.1 General

The `text:condition` attribute specifies a condition. Conditions do not have a predefined syntax, but the attribute value should begin with a namespace prefix, followed by a ":"

(U+003A, COLON) separator, followed by the text of a formula. The namespace bound to the prefix determines the syntax and semantics of the formula.

If a namespace prefix is not specified, the namespace defaults to the "urn:oasis:names:tc:opendocument:xmlns:of:1.2" namespace.

### 19.782.2 <text:conditional-text>

The `text:condition` attribute specifies the condition that determines which of the two text strings of an `<text:conditional-text>` text field is displayed.

The `text:condition` attribute is usable with the following element:  
`<text:conditional-text>` 7.7.3.

The `text:condition` attribute has the data type `string 18.2`.

### 19.782.3 <text:database-next>

The `text:condition` attribute specifies a condition expression. The expression is evaluated and if the result interpreted as a Boolean value is true, the next row is used as the new current row.

If the `<text:database-next>` element has no `text:condition` attribute, the next available row is selected.

The `text:condition` attribute is usable with the following element: `<text:database-next>` 7.6.4.

The `text:condition` attribute has the data type `string 18.2`.

### 19.782.4 <text:database-row-select>

The `text:condition` attribute specifies a condition for selecting a row. If the attribute is present, a new row will be selected only if the condition evaluates to a value interpreted as true.

If a `<text:database-row-select>` element has no `text:condition` attribute, the next row is selected.

The `text:condition` attribute is usable with the following element: `<text:database-row-select>` 7.6.5.

The `text:condition` attribute has the data type `string 18.2`.

### 19.782.5 <text:hidden-paragraph>

The `text:condition` attribute specifies a Boolean expression. If the expression evaluates to true, the paragraph is hidden. If the expression evaluates to false, the paragraph is displayed.

The `text:condition` attribute is usable with the following element: `<text:hidden-paragraph>` 7.7.11.

The `text:condition` attribute has the data type `string 18.2`.

### 19.782.6 <text:hidden-text>

The `text:condition` attribute specifies a Boolean expression. If the expression evaluates to true, the text is hidden. If the expression evaluates to false, the text is displayed.

The `text:condition` attribute is usable with the following element: `<text:hidden-text>` 7.7.4.

The `text:condition` attribute has the data type `string` 18.2.

### 19.782.7 `<text:section>`

The `text:condition` attribute specifies the condition under which a section is hidden. The condition is encoded as a string. If the value of `text:display` 19.802 is `condition`, the `text:condition` attribute shall be present.

The `text:condition` attribute is usable with the following element: `<text:section>` 5.4.

The `text:condition` attribute has the data type `string` 18.2.

### 19.783 `text:connection-name`

The `text:connection-name` attribute specifies the name of the DDE connection to which a field refers.

The `text:connection-name` attribute is usable with the following element: `<text:dde-connection>` 7.7.12.

The `text:connection-name` attribute has the data type `string` 18.2.

### 19.784 `text:consecutive-numbering`

The `text:consecutive-numbering` attribute specifies whether the style uses consecutive numbering for all list levels or whether each list level restarts the numbering.

The defined values for the `text:consecutive-numbering` attribute are:

- `false`: each list level restarts numbering.
- `true`: consecutive numbering is used for all list levels.

The default value for this attribute is `false`.

The `text:consecutive-numbering` attribute is usable with the following element: `<text:list-style>` 16.32.

The `text:consecutive-numbering` attribute has the data type `boolean` 18.3.3.

### 19.785 `text:continue-list`

The `text:continue-list` attribute specifies the `xml:id` 19.920 value of a list that is to be continued.

The `text:continue-list` attribute is usable with the following element: `<text:list>` 5.3.1.

The `text:continue-list` attribute has the data type `IDREF` 18.2.

### 19.786 `text:continue-numbering`

The `text:continue-numbering` attribute specifies, if the numbering of the preceding list is continued or not. This attribute is ignored, if attribute `text:continue-list` 19.785 is present.

The defined values for the `text:continue-numbering` attribute are:

- `true`: if `text:continue-list` attribute is not present and the numbering style of the preceding list is the same as the current list, the number of the first list item in the current list is the number of the last item in the preceding list incremented by one.
- `false`: if the `text:continue-list` attribute is not present, the numbering of the preceding list is not continued.

The use of the `text:continue-numbering` attribute with the `<text:numbered-paragraph>` element has been deprecated.

The `text:continue-numbering` attribute is usable with the following elements: `<text:list>` 5.3.1 and `<text:numbered-paragraph>` 5.3.6.

The `text:continue-numbering` attribute has the data type boolean 18.3.3.

## 19.787 `text:copy-outline-levels`

The `text:copy-outline-levels` attribute specifies whether index entries are indented according to the outline level of their source.

The defined values for the `text:copy-outline-levels` attribute are:

- `false`: no indentation is added.
- `true`: index entries are indented according to the outline level of their source.

The default value for this attribute is `false`.

The `text:copy-outline-levels` attribute is usable with the following element: `<text:user-index-source>` 8.7.2.

The `text:copy-outline-levels` attribute has the data type boolean 18.3.3.

## 19.788 `text:count-empty-lines`

The `text:count-empty-lines` attribute specifies whether empty lines are included in the line count.

The defined values for the `text:count-empty-lines` attribute are:

- `false`: empty lines are not included in a line count.
- `true`: empty lines are included in a line count.

The default value for this attribute is `true`.

The `text:count-empty-lines` attribute is usable with the following element: `<text:linenumbering-configuration>` 16.31.1.

The `text:count-empty-lines` attribute has the data type boolean 18.3.3.

## 19.789 `text:count-in-text-boxes`

The `text:count-in-text-boxes` attribute specifies whether text in text boxes is included in the line count. If the value of this attribute is `true`, text within text boxes is included in the line count.

The defined values for the `text:count-in-text-boxes` attribute are:

- `false`: text within text boxes is not included in a line count.
- `true`: text within text boxes is included in a line count.

The default value for this attribute is `false`.

The `text:count-in-text-boxes` attribute is usable with the following element: `<text:linenumbering-configuration>` 16.31.1.

The `text:count-in-text-boxes` attribute has the data type boolean 18.3.3.

## 19.790 text:current-selected

The `text:current-selected` attribute specifies whether a `<text:label>` 7.4.17 element is currently selected.

The defined values for the `text:current-selected` attribute are:

- `false`: the `<text:label>` element is not currently selected.
- `true`: the `<text:label>` element is currently selected.

The default value for this attribute is `false`.

The `text:current-selected` attribute is usable with the following element:  
`<text:label>` 7.4.17.

The `text:current-selected` attribute has the data type `boolean` 18.3.3.

## 19.791 text:current-value

The `text:current-value` attribute specifies the evaluation result of a condition given by the expression in a `text:condition` 19.782 attribute. The value of this attribute is updated whenever the expression is evaluated.

The `text:current-value` attribute is usable with the following element:  
`<text:conditional-text>` 7.7.3.

The `text:current-value` attribute has the data type `boolean` 18.3.3.

## 19.792 text:custom1

The `text:custom1` attribute specifies the first of up to five user-defined fields for a bibliography index entry.

The `text:custom1` attribute is usable with the following element: `<text:bibliography-mark>` 8.1.11.

The `text:custom1` attribute has the data type `string` 18.2.

## 19.793 text:custom2

The `text:custom2` attribute specifies the second of up to five user-defined fields for a bibliography index entry.

The `text:custom2` attribute is usable with the following element: `<text:bibliography-mark>` 8.1.11.

The `text:custom2` attribute has the data type `string` 18.2.

## 19.794 text:custom3

The `text:custom3` attribute specifies the third of up to five user-defined fields for a bibliography index entry.

The `text:custom3` attribute is usable with the following element: `<text:bibliography-mark>` 8.1.11.

The `text:custom3` attribute has the data type `string` 18.2.

## 19.795 text:custom4

The `text:custom4` attribute specifies the fourth of up to five user-defined fields for a bibliography index entry.

The `text:custom4` attribute is usable with the following element: `<text:bibliography-mark>` 8.1.11.

The `text:custom4` attribute has the data type `string` 18.2.

## 19.796 `text:custom5`

The `text:custom5` attribute specifies the fifth of up to five user-defined fields for a bibliography index entry.

The `text:custom5` attribute is usable with the following element: `<text:bibliography-mark>` 8.1.11.

The `text:custom5` attribute has the data type `string` 18.2.

## 19.797 `text:database-name`

The `text:database-name` attribute specifies a source database by its name.

The `text:database-name` attribute is usable with the following elements: `<text:database-display>` 7.6.3, `<text:database-name>` 7.6.7, `<text:database-next>` 7.6.4, `<text:database-row-number>` 7.6.6 and `<text:database-row-select>` 7.6.5.

The `text:database-name` attribute has the data type `string` 18.2.

## 19.798 `text:date-adjust`

The `text:date-adjust` attribute specifies an adjustment to the value of a date field. If the time period is negative, it is subtracted from the value of the date field.

The `text:date-adjust` attribute is usable with the following element: `<text:date>` 7.3.2.

The `text:date-adjust` attribute has the data type `duration` 18.2.

## 19.799 `text:date-value`

The `text:date-value` attribute specifies a date value. If no value is specified, whether the field is marked as fixed or not, the current date is assumed.

**Note:** This attribute can also be used to specify a future date.

The `text:date-value` attribute is usable with the following elements: `<text:creation-date>` 7.5.3, `<text:date>` 7.3.2, `<text:modification-date>` 7.5.16 and `<text:print-date>` 7.5.8.

The `text:date-value` attribute has the data type `dateOrDateTime` 18.3.14.

## 19.800 `text:default-style-name`

The `text:default-style-name` attribute specifies a default footnote paragraph style for new footnotes that are inserted into a document. It is not evaluated for footnotes that already exist.

The `text:default-style-name` attribute is usable with the following element: `<text:notes-configuration>` 16.31.3.

The `text:default-style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.801 text:description

The `text:description` attribute specifies a message that is displayed when users are prompted for input.

The `text:description` attribute is usable with the following elements:

`<text:placeholder>` 7.7.2, `<text:text-input>` 7.4.15, `<text:user-field-input>` 7.4.10 and `<text:variable-input>` 7.4.6.

The `text:description` attribute has the data type `string` 18.2.

## 19.802 text:display

### 19.802.1 General

The `text:display` attribute specifies options for the display of text.

### 19.802.2 <text:chapter>

The `text:display` attribute specifies the information that a chapter field should display.

The defined values for the `text:display` attribute are:

- `name`: name of a chapter.
- `number`: number of a chapter.
- `number-and-name`: name and number of a chapter.
- `plain-number`: number of a chapter without text defined by `style:num-prefix 19.506` and `style:num-suffix 19.507`, both of which are attributes on the `<text:outline-level-style>` element that defines the format for a chapter number.16.37.
- `plain-number-and-name`: name of a chapter with the number of the same chapter without text defined by `style:num-prefix 19.506` and `style:num-suffix 19.507`, both of which are attributes on the `<text:outline-level-style>` element that defines the format for a chapter number.16.37.

The `text:display` attribute is usable with the following element: `<text:chapter>` 7.3.8.

The values of the `text:display` attribute are `name`, `number`, `number-and-name`, `plain-number-and-name` or `plain-number`.

### 19.802.3 <text:expression>

The `text:display` attribute specifies what information to display for an expression.

The defined values for the `text:display` attribute are:

- `formula`: displays the formula instead of the value of a field.
- `value`: displays the value of a field.

The `text:display` attribute is usable with the following element: `<text:expression>` 7.4.14.

The values of the `text:display` attribute are `value` or `formula`.

### 19.802.4 <text:file-name>

The `text:display` attribute specifies how much of the file name and path to display.

The defined values for the `text:display` attribute are:

- **full**: full file name, extension and file path.
- **path**: file path only.
- **name**: file name without extension.
- **name-and-extension**: file name and extension.

The filename may be an IRI, either because an IRI has been used to retrieve the file, or the consumer internally uses IRIs and therefore converts system-specific paths into an IRI. If this is the case, and if the path, the name or the extension cannot be evaluated from the IRI, then the IRI should be displayed unmodified.

The `text:display` attribute is usable with the following element: `<text:file-name>` 7.3.9.

The values of the `text:display` attribute are `full`, `path`, `name` or `name-and-extension`.

## 19.802.5 `<text:index-entry-chapter>`

The `text:display` attribute specifies what information to display for a chapter field.

The defined values for the `text:display` attribute are:

- **name**: name of a chapter.
- **number**: number of a chapter.
- **number-and-name**: name and number of a chapter.
- **plain-number**: number of a chapter without text defined by `style:num-prefix 19.506` and `style:num-suffix 19.507`, both of which are attributes on the `<text:outline-level-style>` element that defines the format for a chapter number.16.37.
- **plain-number-and-name**: name of a chapter with the number of the same chapter without text defined by `style:num-prefix 19.506` and `style:num-suffix 19.507`, both of which are attributes on the `<text:outline-level-style>` element that defines the format for a chapter number.16.37.

For a `<text:index-entry-chapter>` 8.13.1 element the default value for this attribute is `number`.

The `text:display` attribute is usable with the following element: `<text:index-entry-chapter>` 8.13.1.

The values of the `text:display` attribute are `name`, `number`, `number-and-name`, `plain-number` or `plain-number-and-name`.

## 19.802.6 `<text:section>`

The `text:display` attribute specifies whether the section is hidden.

The defined values for the `text:display` attribute are:

- **condition**: the section is hidden or displayed according to the condition specified by the `text:condition` 19.782 attribute.
- **none**: the section is hidden unconditionally.
- **true**: the section is displayed. This is the default setting.

The `text:display` attribute is usable with the following element: `<text:section>` 5.4.

The values of the `text:display` attribute are `true`, `none` or `condition`.

## 19.802.7 <text:table-formula>

The `text:display` attribute specifies what information to display for a table formula field.

The defined values for the `text:display` attribute are:

- `formula`: the formula used to calculate the value of the field.
- `value`: the value of the field.

The `text:display` attribute is usable with the following element: `<text:table-formula>` 7.7.14.

The values of the `text:display` attribute are `value` or `formula`.

## 19.802.8 <text:template-name>

The `text:display` attribute specifies what information to display about a document template.

The defined values for the `text:display` attribute are:

- `area`: The category of the template. Assignments of categories to templates is implementation-dependent.
- `full`: full file name, extension and file path.
- `name`: file name only.
- `name-and-extension`: file name and extension.
- `path`: file path only.
- `title`: The title.

The `text:display` attribute is usable with the following element: `<text:template-name>` 7.3.10.

The values of the `text:display` attribute are `full`, `path`, `name`, `name-and-extension`, `area` or `title`.

## 19.802.9 <text:user-field-get>

The `text:display` attribute specifies what information to display for a user field.

The defined values for the `text:display` attribute are:

- `formula`: displays the formula instead of the value of a field.
- `none`: hides the content of a field.
- `value`: displays the value of a field.

The `text:display` attribute is usable with the following element: `<text:user-field-get>` 7.4.9.

The values of the `text:display` attribute are `value`, `formula` or `none`.

## 19.802.10 <text:variable-get>

The `text:display` attribute specifies what information to display for a variable-get field.

The defined values for the `text:display` attribute are:

- `formula`: displays the formula instead of the value of a field.
- `value`: displays the value of a field.

The `text:display` attribute is usable with the following element: `<text:variable-get>` 7.4.5.

The values of the `text:display` attribute are `value` or `formula`.

### 19.802.11 `<text:variable-input>`

The `text:display` attribute specifies what information to display for a variable-input field.

The defined values for the `text:display` attribute are:

- `none`: hides the content of a field.
- `value`: displays the value of a field.

The `text:display` attribute is usable with the following element: `<text:variable-input>` 7.4.6.

The values of the `text:display` attribute are `value` or `none`.

### 19.802.12 `<text:variable-set>`

The `text:display` attribute specifies what information to display for a variable-set field.

The defined values for the `text:display` attribute are:

- `none`: hides the content of a field.
- `value`: displays the value of a field.

The `text:display` attribute is usable with the following element: `<text:variable-set>` 7.4.4.

The values of the `text:display` attribute are `value` or `none`.

## 19.803 `text:display-levels`

The `text:display-levels` attribute specifies the number of levels whose numbers are displayed at the current level.

The default value for this attribute is 1.

The `text:display-levels` attribute is usable with the following elements: `<text:list-level-style-number>` 16.34 and `<text:outline-level-style>` 16.37.

The `text:display-levels` attribute has the data type `positiveInteger` 18.2.

## 19.804 `text:display-outline-level`

The `text:display-outline-level` attribute specifies the numbering of a sequence by chapter. This attribute specifies an outline level that determines which chapters to reference for the chapter-specific numbering. All chapters that are at or below the specified outline level reset the value of the sequence to zero, the default value. The chapter number of the last chapter at or below the specified outline level is prefixed to the sequence number. Choosing an outline level of zero results in a continuous numbering of all sequence elements for that sequence variable.

The `text:display-outline-level` attribute is usable with the following element: `<text:sequence-decl>` 7.4.12.

The `text:display-outline-level` attribute has the data type `nonNegativeInteger` 18.2.

## 19.805 text:duration

The `text:duration` attribute contains the value of a `<text:editing-duration>` field element if its value is fixed.

The `text:duration` attribute is usable with the following element: `<text:editing-duration>` 7.5.14.

The `text:duration` attribute has the data type `duration` 18.2.

## 19.806 text:edition

The `text:edition` attribute specifies the edition for a bibliography index entry.

The `text:edition` attribute is usable with the following element: `<text:bibliography-mark>` 8.1.11.

The `text:edition` attribute has the data type `string` 18.2.

## 19.807 text:editor

The `text:editor` attribute specifies the editor for a bibliography index entry.

The `text:editor` attribute is usable with the following element: `<text:bibliography-mark>` 8.1.11.

The `text:editor` attribute has the data type `string` 18.2.

## 19.808 text:filter-name

The `text:filter-name` attribute specifies the filter type to be used to import the link target. The value of this attribute is implementation-dependent.

The `text:filter-name` attribute is usable with the following element: `<text:section-source>` 5.4.2.

The `text:filter-name` attribute has the data type `string` 18.2.

## 19.809 text:fixed

The `text:fixed` attribute specifies whether the value of a field element is fixed.

The defined values for the `text:fixed` attribute are:

- `false`: value of the field element where this attribute appears may be changed.
- `true`: value of the field element where this attribute appears is preserved.

The `text:fixed` attribute is usable with the following elements: `<text:author-initials>` 7.3.7.2, `<text:author-name>` 7.3.7.1, `<text:creation-date>` 7.5.3, `<text:creation-time>` 7.5.4, `<text:creator>` 7.5.17, `<text:date>` 7.3.2, `<text:description>` 7.5.5, `<text:editing-cycles>` 7.5.13, `<text:editing-duration>` 7.5.14, `<text:file-name>` 7.3.9, `<text:initial-creator>` 7.5.2, `<text:keywords>` 7.5.12, `<text:modification-date>` 7.5.16, `<text:modification-time>` 7.5.15, `<text:page-number>` 7.3.4, `<text:print-date>` 7.5.8, `<text:print-time>` 7.5.7, `<text:printed-by>` 7.5.9, `<text:sender-city>` 7.3.6.13, `<text:sender-company>` 7.3.6.10, `<text:sender-country>` 7.3.6.15, `<text:sender-email>` 7.3.6.7, `<text:sender-fax>` 7.3.6.9, `<text:sender-firstname>` 7.3.6.2, `<text:sender-initials>` 7.3.6.4, `<text:sender-lastname>` 7.3.6.3, `<text:sender-phone-private>` 7.3.6.8, `<text:sender-phone-work>` 7.3.6.11, `<text:sender-position>` 7.3.6.6, `<text:sender-postal-code>` 7.3.6.14, `<text:sender-state-or-province>` 7.3.6.16, `<text:sender-street>` 7.3.6.12,

```
<text:sender-title> 7.3.6.5, <text:subject> 7.5.11, <text:time> 7.3.3,  
<text:title> 7.5.10 and <text:user-defined> 7.5.6.
```

The `text:fixed` attribute has the data type `boolean` 18.3.3.

## 19.810 `text:footnotes-position`

The `text:footnotes-position` attribute specifies a position for a footnote.

The defined values for the `text:footnotes-position` attribute are:

- `document`: end of the document.
- `page`: bottom of the page where the footnote citation is located.
- `section`: end of the section.
- `text`: at the page where the footnote citation is located, immediately below the page's text.

The `text:footnotes-position` attribute is usable with the following element:

```
<text:notes-configuration> 16.31.3.
```

The values of the `text:footnotes-position` attribute are `text`, `page`, `section` or `document`.

## 19.811 `text:formula`

The `text:formula` attribute specifies the formula or expression used to compute the value of a field.

The attribute value should begin with a namespace prefix followed by ":" (U+003A, COLON) followed by the text of the formula. The namespace bound to the prefix determines the syntax and semantics of the formula.

The `text:formula` attribute is usable with the following elements: `<text:expression>` 7.4.14, `<text:sequence>` 7.4.13, `<text:table-formula>` 7.7.14, `<text:user-field-decl>` 7.4.8 and `<text:variable-set>` 7.4.4.

The `text:formula` attribute has the data type `string` 18.2.

## 19.812 `text:global (deprecated)`

The `text:global` attribute specifies whether the user interface of a consumer should consider the documents to which linked text sections are linked as individually editable parts of the document.

The defined values for the `text:global` attribute are:

- `false`: the user interface need not make a distinction between linked text sections and non-linked text sections.
- `true`: the user interface considers the content of linked text sections as parts of the document which may be edited by editing the documents to which the linked text sections are linked.

The `text:global` attribute is deprecated in favor of the `application/vnd.oasis.opendocument.text-master` MIME type. Using this MIME type for a document has the same effect as the value `true` for a `text:global` attribute.

The default value for this attribute is `false`.

The `text:global` attribute is usable with the following element: `<office:text>` 3.4.

The `text:global` attribute has the data type `boolean` 18.3.3.

## 19.813 text:howpublished

The `text:howpublished` attribute specifies the publication type for a bibliography index entry.

The `text:howpublished` attribute is usable with the following element:

`<text:bibliography-mark>` 8.1.11.

The `text:howpublished` attribute has the data type `string` 18.2.

## 19.814 text:increment

The `text:increment` attribute causes line numbers to be numbered as multiples of a specified increment.

The `text:increment` attribute is usable with the following elements:

`<text:linenumbering-configuration>` 16.31.1 and `<text:linenumbering-separator>` 16.31.2.

The `text:increment` attribute has the data type `nonNegativeInteger` 18.2.

## 19.815 text:id

### 19.815.1 General

The `text:id` attribute specifies an identifier or a name for an element.

### 19.815.2 <draw:text-box> (deprecated)

The `text:id` attribute specifies an identifier for for a text box.

OpenDocument consumers shall ignore a `text:id` attribute if it occurs on a `<draw:text-box>` element with an `xml:id` attribute value.

OpenDocument producers may write `text:id` attributes for `<draw:text-box>` elements in addition to an `xml:id` attribute.

The value of a `text:id` attribute shall equal the value of an `xml:id` attribute on the same `<draw:text-box>` element.

The `text:id` attribute is deprecated in favor of `xml:id`. 19.920

The `text:id` attribute is usable with the following element: `<draw:text-box>` 10.4.3.

The `text:id` attribute has the data type `NCName` 18.2.

### 19.815.3 <text:alphabetical-index-mark-end>

The `text:id` attribute specifies a name by which start- and end-mark elements are connected to each other. The name specified by the `text:id` attribute shall be unique except for a matching `<text:alphabetical-index-mark-start>` 8.1.8 element.

The `text:id` attribute is usable with the following element: `<text:alphabetical-index-mark-end>` 8.1.9.

The `text:id` attribute has the data type `string` 18.2.

### 19.815.4 <text:alphabetical-index-mark-start>

The `text:id` attribute specifies a name by which start- and end-mark elements are connected to each other. The name specified by the `text:id` attribute shall be unique except for a matching `<text:alphabetical-index-mark-end>` 8.1.9 element.

The `text:id` attribute is usable with the following element: `<text:alphabetical-index-mark-start>` 8.1.8.

The `text:id` attribute has the data type `string` 18.2.

### 19.815.5 `<text:changed-region>` (deprecated)

The `text:id` attribute specifies the an identifier for achanged region.

The ID value is used in `text:change-id` 19.771 IDREF values to refer to a particular `<text:changed-region>` element.

OpenDocument consumers shall ignore a `text:id` attribute if it occurs on an element with an `xml:id` attribute value.

OpenDocument producers may write `text:id` attributes for elements in addition to an `xml:id` attribute.

The value of a `text:id` attribute shall equal the value of an `xml:id` attribute on the same element.

The `text:id` attribute is deprecated in favor of `xml:id`. 19.920

The `text:id` attribute is usable with the following element: `<text:changed-region>` 5.5.3.

The `text:id` attribute has the data type `NCName` 18.2.

### 19.815.6 `<text:h>` (deprecated)

The `text:id` attribute specifies an identifier for a heading.

OpenDocument consumers shall ignore a `text:id` attribute if it occurs on an element with an `xml:id` attribute value.

OpenDocument producers may write `text:id` attributes for elements in addition to an `xml:id` attribute.

The value of a `text:id` attribute shall equal the value of an `xml:id` attribute on the same element.

The `text:id` attribute is deprecated in favor of `xml:id`. 19.920

The `text:id` attribute is usable with the following element: `<text:h>` 5.1.2.

The `text:id` attribute has the data type `NCName` 18.2.

### 19.815.7 `<text:note>`

The `text:id` attribute specifies specifies a name by which notes can be referenced from `<text:note-ref>` elements. See 7.7.7.

The `text:id` attribute is usable with the following element: `<text:note>` 6.3.2.

The `text:id` attribute has the data type `string` 18.2.

### 19.815.8 `<text:p>` (deprecated)

The `text:id` attribute specifies an identifier for a paragraph.

OpenDocument consumers shall ignore a `text:id` attribute if it occurs on a `<text:p>` element with an `xml:id` attribute value. If there is no `xml:id` attribute value, then a `text:id` attribute should be processed as it were an `xml:id` attribute.

OpenDocument producers may write `text:id` attributes for `<text:p>` elements in addition to an `xml:id` attribute.

A `<text:p>` element shall not have an `text:id` attribute if it has no `xml:id` attribute value. The value of a `text:id` attribute shall equal the value of an `xml:id` attribute on the same `<text:p>` element.

The `text:id` attribute is deprecated in favor of `xml:id`. 19.920

The `text:id` attribute is usable with the following element: `<text:p>` 5.1.3.

The `text:id` attribute has the data type `NCName` 18.2.

### 19.815.9 `<text:toc-mark-end>`

The `text:id` attribute specifies a name by which start- and end-mark elements are connected to each other. The name specified by the `text:id` attribute shall be unique except for a matching `<text:toc-mark-start>` 8.1.2 element.

The `text:id` attribute is usable with the following element: `<text:toc-mark-end>` 8.1.3.

The `text:id` attribute has the data type `string` 18.2.

### 19.815.10 `<text:toc-mark-start>`

The `text:id` attribute specifies a name by which start- and end-mark elements are connected to each other. The name specified by the `text:id` attribute shall be unique except for a matching `<text:toc-mark-end>` 8.1.3 element.

The `text:id` attribute is usable with the following element: `<text:toc-mark-start>` 8.1.2.

The `text:id` attribute has the data type `string` 18.2.

### 19.815.11 `<text:user-index-mark-end>`

The `text:id` attribute specifies a string by which start- and end-mark elements are connected to each other. The string shall be unique except for matching its corresponding `<text:user-index-mark-start>` 8.1.5 element.

The `text:id` attribute is usable with the following element: `<text:user-index-mark-end>` 8.1.6.

The `text:id` attribute has the data type `string` 18.2.

### 19.815.12 `<text:user-index-mark-start>`

The `text:id` attribute specifies a string by which start- and end-mark elements are connected to each other. The string shall be unique except for matching its corresponding `<text:user-index-mark-end>` 8.1.6 element.

The `text:id` attribute is usable with the following element: `<text:user-index-mark-start>` 8.1.5.

The `text:id` attribute has the data type `string` 18.2.

## 19.816 `text:identifier`

The `text:identifier` attribute specifies an identifier for a bibliography index entry.

The `text:identifier` attribute is usable with the following element:  
`<text:bibliography-mark>` 8.1.11.

The `text:identifier` attribute has the data type `string` 18.2.

## 19.817 `text:index-name`

The `text:index-name` attribute specifies the name of a user-defined index.

**Note:** In order to support multiple user-defined indexes with different contents, user index marks have a `text:index-name` attribute. The same attribute can be used with a `<text:user-index-source>` element to specify which index marks apply to the current index.

The `text:index-name` attribute is usable with the following elements: `<text:user-index-mark>` 8.1.7, `<text:user-index-mark-start>` 8.1.5 and `<text:user-index-source>` 8.7.2.

The `text:index-name` attribute has the data type `string` 18.2.

## 19.818 `text:index-scope`

The `text:index-scope` attribute specifies whether an index is constructed for an entire document or for a chapter.

The defined values for the `text:index-scope` attribute are:

- `chapter`: the index is constructed for the chapter that contains the index. A chapter consists of all those paragraphs and headings which have the same immediately preceding heading with outline level 1.
- `document`: the index is constructed for the entire document.

The default value for this attribute is `document`.

For `<text:alphabetical-index-source>` 8.8.2, `<text:illustration-index-source>` 8.4.2, `<text:object-index-source>` 8.6.2, `<text:table-index-source>` 8.5.2 and `<text:user-index-source>` 8.7.2 elements the default value for this attribute is `document`.

The `text:index-scope` attribute is usable with the following elements: `<text:alphabetical-index-source>` 8.8.2, `<text:illustration-index-source>` 8.4.2, `<text:object-index-source>` 8.6.2, `<text:table-index-source>` 8.5.2, `<text:table-of-content-source>` 8.3.2 and `<text:user-index-source>` 8.7.2.

The values of the `text:index-scope` attribute are `document` or `chapter`.

## 19.819 `text:ignore-case`

The `text:ignore-case` attribute specifies whether the capitalization of words is ignored. If the value is `true`, the capitalization is ignored and entries that are identical except for character case are listed as the same entries. If the value is `false`, the capitalization of words is not ignored.

The defined values for the `text:ignore-case` attribute are:

- `false`: index entries that are identical except for character case should be listed as the same entries.
- `true`: index entries that are identical but differ in character case should not be listed as separate entries.

The default value for this attribute is `false`.

The `text:ignore-case` attribute is usable with the following element:

`<text:alphabetical-index-source>` 8.8.2.

The `text:ignore-case` attribute has the data type `boolean` 18.3.3.

## 19.820 `text:institution`

The `text:institution` attribute specifies an institution for a bibliography index entry.

The `text:institution` attribute is usable with the following element:

`<text:bibliography-mark>` 8.1.11.

The `text:institution` attribute has the data type `string` 18.2.

## 19.821 `text:is-hidden`

The `text:is-hidden` attribute specifies whether a field or paragraph is currently visible.

**Note:** The value of this attribute is overwritten with a new value as soon as the consumer evaluates the expression.

The defined values for the `text:is-hidden` attribute are:

- `false`: condition for display has evaluated to false, field or paragraph is visible.
- `true`: condition for display has evaluated to true, field or paragraph is not visible.

The `text:is-hidden` attribute is usable with the following elements: `<text:hidden-paragraph>` 7.7.11 and `<text:hidden-text>` 7.7.4.

The `text:is-hidden` attribute has the data type `boolean` 18.3.3.

## 19.822 `text:is-list-header`

The `text:is-list-header` attribute specifies the appearance of a specific heading without numbering.

The defined values for the `text:is-list-header` attribute are:

- `false`: heading will be numbered.
- `true`: heading will be not numbered, even if the header has a list-style..

The default value for this attribute is `false`.

The `text:is-list-header` attribute is usable with the following element: `<text:h>` 5.1.2.

The `text:is-list-header` attribute has the data type `boolean` 18.3.3.

## 19.823 `text:isbn`

The `text:isbn` attribute specifies an ISBN for a bibliography index entry.

The `text:isbn` attribute is usable with the following element: `<text:bibliography-mark>` 8.1.11.

The `text:isbn` attribute has the data type `string` 18.2.

## 19.824 `text:issn`

The `text:issn` attribute specifies an ISSN for a bibliography index entry.

The `text:issn` attribute is usable with the following element: `<text:bibliography-mark>` 8.1.11.

The `text:issn` attribute has the data type `string` 18.2.

## 19.825 `text:journal`

The `text:journal` attribute specifies a journal for a bibliography index entry.

The `text:journal` attribute is usable with the following element: `<text:bibliography-mark>` 8.1.11.

The `text:journal` attribute has the data type `string` 18.2.

## 19.826 `text:key`

The `text:key` attribute specifies a type for the index entries that should be used for sorting. See 8.1.11.

The defined values for the `text:key` attribute are:

- `address`: 19.756.
- `annote`: 19.760.
- `author`: 19.761.
- `bibliography-type`: 19.763.
- `booktitle`: 19.764.
- `chapter`: 19.772.
- `custom1`: 19.792.
- `custom2`: 19.793.
- `custom3`: 19.794.
- `custom4`: 19.795.
- `custom5`: 19.796.
- `edition`: 19.806.
- `editor`: 19.807.
- `howpublished`: 19.813.
- `identifier`: 19.816.
- `institution`: 19.820.
- `isbn`: 19.823.
- `issn`: 19.824.
- `journal`: 19.825.
- `month`: 19.840.
- `note`: 19.842.
- `number`: 19.844.
- `organizations`: 19.849.
- `pages`: 19.852.
- `publisher`: 19.858.
- `report-type`: 19.862.
- `school`: 19.866.

- `series`: 19.868.
- `title`: 19.888.
- `url`: 19.890.
- `volume`: 19.908.
- `year`: 19.909.

The `text:key` attribute is usable with the following element: `<text:sort-key>` 16.31.7.

The values of the `text:key` attribute are `address`, `annote`, `author`, `bibliography-type`, `booktitle`, `chapter`, `custom1`, `custom2`, `custom3`, `custom4`, `custom5`, `edition`, `editor`, `howpublished`, `identifier`, `institution`, `isbn`, `issn`, `journal`, `month`, `note`, `number`, `organizations`, `pages`, `publisher`, `report-type`, `school`, `series`, `title`, `url`, `volume` or `year`.

## 19.827 `text:key1`

The `text:key1` attribute specifies an additional key for an alphabetical index mark. If only one key is used, it shall be contained in a `text:key1` attribute.

The `text:key1` attribute is usable with the following elements: `<text:alphabetical-index-mark>` 8.1.10 and `<text:alphabetical-index-mark-start>` 8.1.8.

The `text:key1` attribute has the data type `string` 18.2.

## 19.828 `text:key2`

The `text:key2` attribute specifies an additional key for an alphabetical index mark.

The `text:key2` attribute is usable with the following elements: `<text:alphabetical-index-mark>` 8.1.10 and `<text:alphabetical-index-mark-start>` 8.1.8.

The `text:key2` attribute has the data type `string` 18.2.

## 19.829 `text:key1-phonetic`

The `text:key1-phonetic` attribute specifies a phonetic key for `<text:alphabetical-index-mark>` and `<text:alphabetical-index-mark-start>` elements. Phonetic keys should be used for sorting.

The `text:key1-phonetic` attribute is usable with the following elements: `<text:alphabetical-index-mark>` 8.1.10 and `<text:alphabetical-index-mark-start>` 8.1.8.

The `text:key1-phonetic` attribute has the data type `string` 18.2.

## 19.830 `text:key2-phonetic`

The `text:key2-phonetic` attribute specifies an additional phonetic key for `<text:alphabetical-index-mark>` and `<text:alphabetical-index-mark-start>` elements.

The `text:key2-phonetic` attribute is usable with the following elements: `<text:alphabetical-index-mark>` 8.1.10 and `<text:alphabetical-index-mark-start>` 8.1.8.

The `text:key2-phonetic` attribute has the data type `string` 18.2.

## 19.831 text:kind

The `text:kind` attribute specifies which part of a measure is displayed.

The defined values for the `text:kind` attribute are:

- `gap`: Adds space to text if the measure text's writing direction is perpendicular to the measure line.
- `unit`: The measure's unit is displayed.
- `value`: The measure's value is displayed.

**Note:** The purpose of the `gap` value is add space between the measure line and the text if the text is displayed perpendicular to the measure line.

The `text:kind` attribute is usable with the following element: `<text:measure>` 7.7.13.

The values of the `text:kind` attribute are `value`, `unit` or `gap`.

## 19.832 text:label

The `text:label` attribute specifies a label for user -inserted notes in a document. If this attribute is omitted, the element content is used.

The `text:label` attribute is usable with the following element: `<text:note-citation>` 6.3.3.

The `text:label` attribute has the data type `string` 18.2.

## 19.833 text:label-followed-by

The `text:label-followed-by` attribute specifies a character that is inserted following a list label.

The defined values for the `text:label-followed-by` attribute are:

- `listtab`: a tab character is inserted after a list label before the text starts.
- `nothing`: text starts directly after a list label.
- `space`: a “ ” (U+0020, SPACE) character is inserted after a list label before the text starts.

The `text:label-followed-by` attribute is usable with the following element:

`<style:list-level-label-alignment>` 17.20.

The values of the `text:label-followed-by` attribute are `listtab`, `space` or `nothing`.

## 19.834 text:level

The `text:level` attribute specifies the level of an outline or number list style.

For a `<text:numbered-paragraph>` 5.3.6 element the default value for this attribute is 1.

The `text:level` attribute is usable with the following elements: `<number:num-list-label>` 16.29.31, `<text:list-level-style-bullet>` 16.33, `<text:list-level-style-image>` 16.35, `<text:list-level-style-number>` 16.34, `<text:numbered-paragraph>` 5.3.6 and `<text:outline-level-style>` 16.37.

The `text:level` attribute has the data type `positiveInteger` 18.2.

## 19.835 text:list-id (deprecated)

The `text:list-id` attribute is deprecated, because only the deprecated element `<text:numbered-paragraph>` uses it.

The `text:list-id` attribute specifies an id that is used to group numbered paragraphs form a list. All numbered paragraphs with the same `text:list-id` value belong to one list. This list defines the counter domain for the numbered paragraphs. In this context, a list is defined as a number of list items, spanning multiple levels. As with lists in `<text:list>` 5.3.1 representation the list style of the first numbered paragraph at every list level is used for the start value.

The `text:list-id` attribute did not exist in OpenDocument v1.0 and v1.1. For such text documents which do not contain the `text:list-id` attribute the following rules are applied to form a list from a group of numbered paragraphs:

- A series of numbered paragraphs each using the same list style belong to the same list.
- A numbered paragraph that does not have a `text:list-id` gets the `text:list-id` of the last numbered paragraph on the same list level using the same list style. If the previous numbered-paragraph has a different list style, then a new list is started.

The `text:list-id` attribute is usable with the following element: `<text:numbered-paragraph>` 5.3.6.

The `text:list-id` attribute has the data type `NCName` 18.2.

## 19.836 `text:list-tab-stop-position`

The `text:list-tab-stop-position` attribute specifies an additional tab stop which is inserted into the list of tab stops that are defined for a list item. By default the tab stop's position is following the list label. The text of the first line of list item starts at this tab stop. In left-to-right layout environments this tab stop is interpreted as a left tab. While in right-to-left layout environments it is interpreted as a right tab.

The `text:list-tab-stop-position` attribute is ignored unless the `text:label-followed-by` attribute has the value `listtab`.

There are two cases in which this additional tab stop does not specify the start the of the text of the first line of the list item:

- The position of the additional tab stop is before the end of the list label.
- Between the end of the list label and the position of the additional tab stop exists already a tab stop.

In these cases, the tab character that is inserted following the list label advances to the next tab stop, or a default tab stop if none exists. However, other tab stops contained in the list item's text may advance to the additional list tab stop. The additional tab stop is further applied to the full text of tab stop, that is, not only the first line.

The `text:list-tab-stop-position` attribute is usable with the following element: `<style:list-level-label-alignment>` 17.20.

The `text:list-tab-stop-position` attribute has the data type `length` 18.3.19.

## 19.837 `text:main-entry`

The `text:main-entry` attribute specifies which of multiple index marks for the same entry is the main entry.

The defined values for the `text:main-entry` attribute are:

- `false`: no index mark is specified as the main entry from multiple index marks for the same entry.
- `true`: one index mark out of multiple index marks for the same entry is specified as the main entry.

The default value for this attribute is `false`.

The `text:main-entry` attribute is usable with the following elements:  
`<text:alphabetical-index-mark>` 8.1.10 and `<text:alphabetical-index-mark-start>` 8.1.8.

The `text:main-entry` attribute has the data type `boolean` 18.3.3.

## 19.838 `text:main-entry-style-name`

The `text:main-entry-style-name` attribute specifies the text style to use for main entries. Sub entries are formatted using the default text style determined by the paragraph style of the entries.

The `text:main-entry-style-name` attribute is usable with the following element:  
`<text:alphabetical-index-source>` 8.8.2.

The `text:main-entry-style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.839 `text:master-page-name`

The `text:master-page-name` attribute specifies the name of a master page.

For `<text:page>` elements 5.2.2, the attribute specifies the name of the master page which is used to format the page.

For `<text:notes-configuration>` elements 16.31.3, the attribute specifies the master page which is used to format the pages on which the notes are displayed.

The `text:master-page-name` attribute is usable with the following elements:  
`<text:notes-configuration>` 16.31.3 and `<text:page>` 5.2.2.

The `text:master-page-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.840 `text:month`

The `text:month` attribute specifies a month for a bibliography index entry.

The `text:month` attribute is usable with the following element: `<text:bibliography-mark>` 8.1.11.

The `text:month` attribute has the data type `string` 18.2.

## 19.841 `text:name`

### 19.841.1 General

The `text:name` attribute specifies a name.

### 19.841.2 `<text:alphabetical-index>`

The `text:name` attribute specifies a unique name for an alphabetical index.

The `text:name` attribute is usable with the following element: `<text:alphabetical-index>` 8.8.

The `text:name` attribute has the data type `string` 18.2.

### 19.841.3 `<text:bibliography>`

The `text:name` attribute specifies a unique name for a bibliography.

The `text:name` attribute is usable with the following element: `<text:bibliography>` 8.9.

The `text:name` attribute has the data type `string 18.2`.

#### 19.841.4 `<text:bookmark>`

The `text:name` attribute specifies a name for a bookmark.

The `text:name` attribute is usable with the following element: `<text:bookmark>` 6.2.1.2.

The `text:name` attribute has the data type `string 18.2`.

#### 19.841.5 `<text:bookmark-end>`

The `text:name` attribute specifies a name that shall match the `text:name` attribute on only one `<text:bookmark-start>` 6.2.1.3 element in a document.

The `text:name` attribute is usable with the following element: `<text:bookmark-end>` 6.2.1.4.

The `text:name` attribute has the data type `string 18.2`.

#### 19.841.6 `<text:bookmark-start>`

The `text:name` attribute specifies a name that shall match the `text:name` attribute on only one `<text:bookmark-end>` 6.2.1.4 element in a document.

The `text:name` attribute is usable with the following element: `<text:bookmark-start>` 6.2.1.3.

The `text:name` attribute has the data type `string 18.2`.

#### 19.841.7 `<text:drop-down>`

The `text:name` attribute specifies a unique name for a drop-down list input field.

The `text:name` attribute is usable with the following element: `<text:drop-down>` 7.4.16.

The `text:name` attribute has the data type `string 18.2`.

#### 19.841.8 `<text:execute-macro>`

The `text:name` attribute specifies a macro to invoke when a field is activated.

The `text:name` attribute is usable with the following element: `<text:execute-macro>` 7.7.10.

The `text:name` attribute has the data type `string 18.2`.

#### 19.841.9 `<text:illustration-index>`

The `text:name` attribute specifies a unique name for an illustration index.

The `text:name` attribute is usable with the following element: `<text:illustration-index>` 8.4.

The `text:name` attribute has the data type `string 18.2`.

#### 19.841.10 `<text:index-title>`

The `text:name` attribute specifies a unique name for an index title.

The `text:name` attribute is usable with the following element: `<text:index-title>` 8.2.3.

The `text:name` attribute has the data type `string 18.2`.

### 19.841.11 <text:object-index>

The `text:name` attribute specifies a unique name for an object index.

The `text:name` attribute is usable with the following element: `<text:object-index>` 8.6.

The `text:name` attribute has the data type `string` 18.2.

### 19.841.12 <text:reference-mark>

The `text:name` attribute specifies a unique name for reference elements.

The `text:name` attribute is usable with the following element: `<text:reference-mark>` 6.2.2.2.

The `text:name` attribute has the data type `string` 18.2.

### 19.841.13 <text:reference-mark-end>

The `text:name` attribute specifies a name that shall match the `text:name` attribute on only one `<text:reference-mark-start>` 6.2.2.3 element in a document.

The `text:name` attribute is usable with the following element: `<text:reference-mark-end>` 6.2.2.4.

The `text:name` attribute has the data type `string` 18.2.

### 19.841.14 <text:reference-mark-start>

The `text:name` attribute specifies a name that shall match the `text:name` attribute on only one `<text:reference-mark-end>` 6.2.2.4 element in a document.

The `text:name` attribute is usable with the following element: `<text:reference-mark-start>` 6.2.2.3.

The `text:name` attribute has the data type `string` 18.2.

### 19.841.15 <text:section>

The `text:name` attribute specifies a unique name for a section.

The `text:name` attribute is usable with the following element: `<text:section>` 5.4.

The `text:name` attribute has the data type `string` 18.2.

### 19.841.16 <text:sequence>

The `text:name` attribute specifies name of a variable to display. It shall match the name of a sequence variable already declared by a preceding `<text:sequence-decl>` 7.4.12 element.

The `text:name` attribute is usable with the following element: `<text:sequence>` 7.4.13.

The `text:name` attribute has the data type `variableName` 18.3.40.

### 19.841.17 <text:sequence-decl>

The `text:name` attribute specifies a unique name of a variable to be declared.

The `text:name` attribute is usable with the following element: `<text:sequence-decl>` 7.4.12.

The `text:name` attribute has the data type `variableName` 18.3.40.

### 19.841.18 <text:table-of-content>

The `text:name` attribute specifies a unique name for a `<text:table-of-content>` element.

The `text:name` attribute is usable with the following element: `<text:table-of-content>` 8.3.

The `text:name` attribute has the data type `string` 18.2.

### 19.841.19 <text:table-index>

The `text:name` attribute specifies a unique name for a table index.

The `text:name` attribute is usable with the following element: `<text:table-index>` 8.5.

The `text:name` attribute has the data type `string` 18.2.

### 19.841.20 <text:user-defined>

The `text:name` attribute specifies a name that corresponds to the value of a `meta:name` attribute on a `<meta:user-defined>` element.

The `text:name` attribute is usable with the following element: `<text:user-defined>` 7.5.6.

The `text:name` attribute has the data type `string` 18.2.

### 19.841.21 <text:user-field-decl>

The `text:name` attribute specifies a unique name for a user-defined field declaration.

The `text:name` attribute is usable with the following element: `<text:user-field-decl>` 7.4.8.

The `text:name` attribute has the data type `variableName` 18.3.40.

### 19.841.22 <text:user-field-get>

The `text:name` attribute specifies the name of a user-defined field that shall be declared already by a preceding `<text:user-field-decl>` 7.4.8 element.

The `text:name` attribute is usable with the following element: `<text:user-field-get>` 7.4.9.

The `text:name` attribute has the data type `variableName` 18.3.40.

### 19.841.23 <text:user-field-input>

The `text:name` attribute specifies the name of a user-defined field that shall be declared already by a preceding `<text:user-field-decl>` 7.4.8 element.

The `text:name` attribute is usable with the following element: `<text:user-field-input>` 7.4.10.

The `text:name` attribute has the data type `variableName` 18.3.40.

### 19.841.24 <text:user-index>

The `text:name` attribute specifies a unique name for a user-specified index.

The `text:name` attribute is usable with the following element: `<text:user-index>` 8.7.

The `text:name` attribute has the data type `string` 18.2.

### 19.841.25 <text:variable-set>

The `text:name` attribute specifies a variable name that shall match a variable name already declared by a preceding `<text:variable-decl>` 7.4.3 element.

The `text:name` attribute is usable with the following element: `<text:variable-set>` 7.4.4.

The `text:name` attribute has the data type `variableName` 18.3.40.

### 19.841.26 <text:variable-decl>

The `text:name` attribute specifies a unique name for a variable declaration.

The `text:name` attribute is usable with the following element: `<text:variable-decl>` 7.4.3.

The `text:name` attribute has the data type `variableName` 18.3.40.

### 19.841.27 <text:variable-get>

The `text:name` attribute specifies a variable name that shall match a preceding `<text:variable-decl>` 7.4.3 element.

The `text:name` attribute is usable with the following element: `<text:variable-get>` 7.4.5.

The `text:name` attribute has the data type `variableName` 18.3.40.

### 19.841.28 <text:variable-input>

The `text:name` attribute specifies a variable name that shall match the name of variable already declared by a preceding `<text:variable-decl>` 7.4.3 element.

The `text:name` attribute is usable with the following element: `<text:variable-input>` 7.4.6.

The `text:name` attribute has the data type `variableName` 18.3.40.

## 19.842 text:note

The `text:note` attribute specifies a note for a bibliography index entry.

The `text:note` attribute is usable with the following element: `<text:bibliography-mark>` 8.1.11.

The `text:note` attribute has the data type `string` 18.2.

## 19.843 text:note-class

The `text:note-class` attribute specifies the class of a note.

The defined values for the `text:note-class` attribute are:

- `endnote`: note appears at the end of a chapter or document.
- `footnote`: note appears at the footer of a page.

The `text:note-class` attribute is usable with the following elements: `<text:note>` 6.3.2, `<text:note-ref>` 7.7.7 and `<text:notes-configuration>` 16.31.3.

The values of the `text:note-class` attribute are `footnote` or `endnote`.

## 19.844 text:number

The `text:number` attribute specifies a number for a bibliography index entry.

The `text:number` attribute is usable with the following element: `<text:bibliography-mark>` 8.1.11.

The `text:number` attribute has the data type `string` 18.2.

## 19.845 text:number-lines

The `text:number-lines` attribute specifies whether lines are numbered.

The defined values for the `text:number-lines` attribute are:

- `false`: lines are not numbered.
- `true`: lines are numbered.

For a `<text:linenumbering-configuration>` 16.31.1 element the default value for this attribute is `true`.

The `text:number-lines` attribute is usable with the following element:

`<text:linenumbering-configuration>` 16.31.1.

The `text:number-lines` attribute has the data type `boolean` 18.3.3.

## 19.846 text:number-position

The `text:number-position` attribute specifies in which margin of a page line numbers appear.

The defined values for the `text:number-position` attribute are:

- `inner`: line numbers appear on the inner margin of a page.
- `left`: line numbers appear on the left margin of a page.
- `outer`: line numbers appear on the outer margin of a page.
- `right`: line numbers appear on the right margin of a page

The default value for this attribute is `left`.

The `text:number-position` attribute is usable with the following element:

`<text:linenumbering-configuration>` 16.31.1.

The values of the `text:number-position` attribute are `left`, `right`, `inner` or `outer`.

## 19.847 text:numbered-entries

The `text:numbered-entries` attribute specifies whether a number is displayed for bibliography entries instead of their short name.

The defined values for the `text:numbered-entries` attribute are:

- `false`: short names for bibliographic entries displayed.
- `true`: number displayed for bibliographic entries.

The default value for this attribute is `false`.

The `text:numbered-entries` attribute is usable with the following element:

`<text:bibliography-configuration>` 16.31.6.

The `text:numbered-entries` attribute has the data type `boolean` 18.3.3.

## 19.848 text:offset

The `text:offset` attribute specifies the distance between a line number and the margin.

The `text:offset` attribute is usable with the following element: `<text:linenumbering-configuration>` 16.31.1.

The `text:offset` attribute has the data type `nonNegativeLength` 18.3.21.

## 19.849 text:organizations

The `text:organizations` attribute specifies one or more organizations for a bibliography index entry.

The `text:organizations` attribute is usable with the following element: `<text:bibliography-mark>` 8.1.11.

The `text:organizations` attribute has the data type `string` 18.2.

## 19.850 text:outline-level

### 19.850.1 General

The `text:outline-level` attribute specifies an outline level.

### 19.850.2 <text:alphabetical-index-entry-template>

The `text:outline-level` attribute specifies the level to which a template applies.

The defined values for the `text:outline-level` attribute are:

- 1: index mark with only text but no keys assigned is assigned to level one.
- 2: index mark with text and one key, key is assigned to level one and index mark value is assigned to level two.
- 3: index mark with two keys, first key is assigned to level one, second key is assigned to level two, and the index mark value is assigned to level three.
- separator: the template defines an alphabetical separator.

The `text:outline-level` attribute is usable with the following element: `<text:alphabetical-index-entry-template>` 8.8.4.

The values of the `text:outline-level` attribute are 1, 2, 3 or separator.

### 19.850.3 <text:chapter>

The `text:outline-level` attribute specifies an outline level to use. The `<text:chapter>` element displays a chapter number or title up to the specified outline level.

The `text:outline-level` attribute is usable with the following element: `<text:chapter>` 7.3.8.

The `text:outline-level` attribute has the data type `nonNegativeInteger` 18.2.

### 19.850.4 <text:h>

The `text:outline-level` attribute specifies the level of a heading, starting with 1. Headings without a level attribute are assumed to be at level 1.

The `text:outline-level` attribute is usable with the following element: `<text:h>` 5.1.2.

The `text:outline-level` attribute has the data type `positiveInteger` 18.2.

## 19.850.5 <text:index-entry-chapter>

The `text:outline-level` attribute specifies an outline level for an index entry.

The `text:outline-level` attribute is usable with the following element: `<text:index-entry-chapter>` 8.13.1.

The `text:outline-level` attribute has the data type `positiveInteger` 18.2.

## 19.850.6 <text:index-source-styles>

The `text:outline-level` attribute specifies an outline level for an index entry.

The `text:outline-level` attribute is usable with the following element: `<text:index-source-styles>` 8.10.

The `text:outline-level` attribute has the data type `positiveInteger` 18.2.

## 19.850.7 <text:user-index-mark>

The `text:outline-level` attribute specifies an outline level for an index entry.

The `text:outline-level` attribute is usable with the following element: `<text:user-index-mark>` 8.1.7.

The `text:outline-level` attribute has the data type `positiveInteger` 18.2.

## 19.850.8 <text:table-of-content-entry-template>

The `text:outline-level` attribute specifies to which outline level an entry configuration applies. Outline levels shall be unique for the template elements in one index source.

The `text:outline-level` attribute is usable with the following element: `<text:table-of-content-entry-template>` 8.3.3.

The `text:outline-level` attribute has the data type `positiveInteger` 18.2.

## 19.850.9 <text:table-of-content-source>

The `text:outline-level` attribute specifies which outline levels are used when generating a table of contents. The value of this attribute shall be an integer greater than zero. If this attribute is omitted, all outline levels are used by default.

The `text:outline-level` attribute is usable with the following element: `<text:table-of-content-source>` 8.3.2.

The `text:outline-level` attribute has the data type `positiveInteger` 18.2.

## 19.850.10 <text:toc-mark>

The `text:outline-level` attribute specifies an outline level for a table of contents index entry.

The `text:outline-level` attribute is usable with the following element: `<text:toc-mark>` 8.1.4.

The `text:outline-level` attribute has the data type `positiveInteger` 18.2.

## 19.850.11 <text:toc-mark-start>

The `text:outline-level` attribute specifies an outline level for a table of contents index entry.

The `text:outline-level` attribute is usable with the following element: `<text:toc-mark-start>` 8.1.2.

The `text:outline-level` attribute has the data type `positiveInteger` 18.2.

### 19.850.12 `<text:user-index-entry-template>`

The `text:outline-level` attribute specifies the template applied to entries at a specified outline level in a user-defined index. All `<text:user-index-entry-template>` elements that are contained in the same parent element shall specify different outline levels.

The `text:outline-level` attribute is usable with the following element: `<text:user-index-entry-template>` 8.7.3.

The `text:outline-level` attribute has the data type `positiveInteger` 18.2.

### 19.850.13 `<text:user-index-mark-start>`

The `text:outline-level` attribute specifies an outline level for a user index entry.

The `text:outline-level` attribute is usable with the following element: `<text:user-index-mark-start>` 8.1.5.

The `text:outline-level` attribute has the data type `positiveInteger` 18.2.

## 19.851 `text:page-adjust`

### 19.851.1 General

The `text:page-adjust` attribute specifies an adjustment to page numbering.

### 19.851.2 `<text:page-number>`

The `text:page-adjust` attribute specifies an adjustment of the value of a page number field, to display of page numbers of following or preceding pages. The specified value is added to the current page number. If a page with the resulting page number does not exist, no number is displayed.

The `text:page-adjust` attribute is usable with the following element: `<text:page-number>` 7.3.4.

The `text:page-adjust` attribute has the data type `integer` 18.2.

### 19.851.3 `<text:page-variable-set>`

The `text:page-adjust` attribute specifies a page adjustment. The value of an active page variable is the current page number plus any page adjustment value immediately prior to the occurrence of this variable in document order.

The `text:page-adjust` attribute is usable with the following element: `<text:page-variable-set>` 7.7.1.2.

The `text:page-adjust` attribute has the data type `integer` 18.2.

## 19.852 `text:pages`

The `text:pages` attribute specifies the page number or page numbers for a bibliography index entry.

The `text:pages` attribute is usable with the following element: `<text:bibliography-mark>` 8.1.11.

The `text:pages` attribute has the data type `string` 18.2.

## 19.853 `text:placeholder-type`

The `text:placeholder-type` attribute specifies the content type of a placeholder.

The defined values for the `text:placeholder-type` attribute are:

- `image`: placeholder can be replaced by an image.
- `object`: placeholder can be replaced by an object.
- `table`: placeholder can be replaced by a table.
- `text`: placeholder can be replaced by text.
- `text-box`: placeholder can be replaced by a text-box.

The `text:placeholder-type` attribute is usable with the following element:

`<text:placeholder> 7.7.2.`

The values of the `text:placeholder-type` attribute are `text`, `table`, `text-box`, `image` or `object`.

## 19.854 `text:prefix`

The `text:prefix` attribute specifies a string that is displayed before a bibliography entry's short name or number if it occurs outside the bibliography.

The `text:prefix` attribute is usable with the following element: `<text:bibliography-configuration> 16.31.6.`

The `text:prefix` attribute has the data type `string` 18.2.

## 19.855 `text:protected`

The `text:protected` attribute specifies whether a section is protected. A section can be protected, which means that a user cannot edit the section. The protection is enforced by the user interface only.

**Note:** If the section is protected and the `text:protection-key` 19.856 attribute is present, an authorization is required for resetting the protection to enable editing.

The defined values for the `text:protected` attribute are:

- `false`: section is not protected from editing.
- `true`: section is protected from editing.

The `text:protected` attribute is usable with the following elements:

`<text:alphabetical-index> 8.8, <text:bibliography> 8.9, <text:illustration-index> 8.4, <text:index-title> 8.2.3, <text:object-index> 8.6, <text:section> 5.4, <text:table-index> 8.5, <text:table-of-content> 8.3 and <text:user-index> 8.7.`

The `text:protected` attribute has the data type `boolean` 18.3.3.

## 19.856 `text:protection-key`

The `text:protection-key` attribute, when present, specifies that an authorization is required for removing the protection of a section. The authentication procedure is identified by the `text:protection-key-digest-algorithm` attribute 19.857. The attribute value is binary data that may be used by the authentication procedure.

The password shall be provided as a sequence of bytes in UTF-8 encoding.

The `text:protection-key` attribute is usable with the following elements:  
`<text:alphabetical-index>` 8.8, `<text:bibliography>` 8.9,  
`<text:illustration-index>` 8.4, `<text:index-title>` 8.2.3, `<text:object-index>` 8.6, `<text:section>` 5.4, `<text:table-index>` 8.5, `<text:table-of-content>` 8.3 and `<text:user-index>` 8.7.

The `text:protection-key` attribute has the data type `string` 18.2.

## 19.857 `text:protection-key-digest-algorithm`

The `text:protection-key-digest-algorithm` attribute value is an IRI that identifies an authentication procedure for removing a protection.

If the IRI identifies a message-digest algorithm specified in §5.7 of [xmlenc-core], the value of `text:protection-key` 19.856 attribute shall be the hash value of the password that is required to authorize removal of the protection. The password shall be provided as a sequence of bytes in UTF-8 encoding.

Any other procedures, their identifying IRIs, and their application of `text:protection-key` values are implementation-defined.

Consumers shall support <http://www.w3.org/2000/09/xmldsig#sha1>, which is the default, and <http://www.w3.org/2000/09/xmldsig#sha256>. They may support other algorithms described in §5.7 of [xmlenc-core] or alternative procedures. Producers should use <http://www.w3.org/2000/09/xmldsig#sha256>.

The default value for this attribute is <http://www.w3.org/2000/09/xmldsig#sha1>.

The `text:protection-key-digest-algorithm` attribute is usable with the following elements: `<text:alphabetical-index>` 8.8, `<text:bibliography>` 8.9, `<text:illustration-index>` 8.4, `<text:index-title>` 8.2.3, `<text:object-index>` 8.6, `<text:section>` 5.4, `<text:table-index>` 8.5, `<text:table-of-content>` 8.3 and `<text:user-index>` 8.7.

The `text:protection-key-digest-algorithm` attribute has the data type `anyIRI` 18.3.2.

## 19.858 `text:publisher`

The `text:publisher` attribute specifies a publisher for a bibliography index entry.

The `text:publisher` attribute is usable with the following element:  
`<text:bibliography-mark>` 8.1.11.

The `text:publisher` attribute has the data type `string` 18.2.

## 19.859 `text:ref-name`

### 19.859.1 General

The `text:ref-name` attribute specifies a name that is used as a reference.

### 19.859.2 `<text:bookmark-ref>`

The `text:ref-name` attribute references a `<text:bookmark>` 6.2.1.2 or `<text:bookmark-start>` 6.2.1.3 element by the value of that element's `text:name` 6.2.1.3 attribute.

The `text:ref-name` attribute is usable with the following element: `<text:bookmark-ref>` 7.7.6.

The `text:ref-name` attribute has the data type `string` 18.2.

### 19.859.3 <text:note-ref>

The `text:ref-name` attribute references a `<text:note>` 19.842 element by the value of that element's `text:id` 19.815 attribute.

The `text:ref-name` attribute is usable with the following element: `<text:note-ref>` 7.7.7.

The `text:ref-name` attribute has the data type `string` 18.2.

### 19.859.4 <text:reference-ref>

The `text:ref-name` attribute references a `<text:reference-mark>` 6.2.2.2 or `<text:reference-mark-start>` 6.2.2.3 element by the value of that element's `text:name` 19.841 attribute.

The `text:ref-name` attribute identifies the element referenced by this reference field element.

The `text:ref-name` attribute is usable with the following element: `<text:reference-ref>` 7.7.5.

The `text:ref-name` attribute has the data type `string` 18.2.

### 19.859.5 <text:sequence>

The `text:ref-name` attribute specifies the name for a sequence field that is the target of a reference field. No two sequence fields can have the same reference name.

If a sequence field is not the target of a reference, this attribute can be omitted.

The `text:ref-name` attribute is usable with the following element: `<text:sequence>` 7.4.13.

The `text:ref-name` attribute has the data type `string` 18.2.

### 19.859.6 <text:sequence-ref>

The `text:ref-name` attribute references a `<text:sequence>` 7.4.13 element by the value of that element's `text:ref-name` 7.4.13 attribute.

The `text:ref-name` attribute is usable with the following element: `<text:sequence-ref>` 7.7.8.

The `text:ref-name` attribute has the data type `string` 18.2.

## 19.860 text:reference-format

The `text:reference-format` attribute specifies what information about a reference is displayed. If the reference format is not specified, the page format is used as the default.

The defined values for the `text:reference-format` attribute supported by all reference fields are:

- `chapter`: displays the number of the chapter in which the referenced item appears.
- `direction`: displays whether the referenced item is above or below the reference field.
- `page`: displays the number of the page on which the referenced item appears.
- `text`: displays the text of the referenced item.

Additional defined values for the `text:reference-format` attribute supported by references to sequence fields are:

- `caption`: displays the caption in which the sequence is used.

- `category-and-value`: displays the name and value of the sequence.
- `value`: displays the value of the sequence.

References to bookmarks and other references support additional values, which display the *list label* of the referenced item. If the referenced item is contained in a list or a numbered paragraph, the list label is the formatted number of the paragraph which contains the referenced item. If the referenced item is not contained in a list or numbered paragraph, the list label is empty, and the referenced field therefore displays nothing. If the referenced bookmark or reference contains more than one paragraph, the list label of the paragraph at which the bookmark or reference starts is taken.

Additional defined values for the `text:reference-format` attribute supported by all references to bookmarks or other reference fields are:

- `number`: displays the list label of the referenced item. The list position of the referenced item plus all of its superior list levels are its reference.
  - `number-all-superior`: displays the list label of the referenced item and is prefixed by the list labels of all superior list levels.
- The list label of the referenced item may already contain numbers of superior levels. If this is the case, and if  $n$  is the level of the most superior level contained in the list label, then no list label content of superior levels smaller or equal than  $n$  are added.
- `number-no-superior`: displays the contents of the list label of the referenced item.

The `text:reference-format` attribute is usable with the following elements:

`<text:bookmark-ref> 7.7.6, <text:note-ref> 7.7.7, <text:reference-ref> 7.7.5 and <text:sequence-ref> 7.7.8.`

For `<text:reference-ref> 7.7.5` elements, the values of the `text:reference-format` attribute are , page, chapter, direction, text, `number-no-superior`, `number-all-superior` or `number`.

For `<text:bookmark-ref> 7.7.6` elements, the values of the `text:reference-format` attribute are , page, chapter, direction, text, `number-no-superior`, `number-all-superior` or `number`.

For `<text:note-ref> 7.7.7` elements, the values of the `text:reference-format` attribute are , page, chapter, direction or text.

For `<text:sequence-ref> 7.7.8` elements, the values of the `text:reference-format` attribute are , page, chapter, direction, text, `category-and-value`, caption or value.

## 19.861 `text:relative-tab-stop-position`

The `text:relative-tab-stop-position` attribute specifies whether the position of tab stops is relative to the left margin or to the left indent as determined by the paragraph style. It is used in the specification for the generation of index entries and table of content entries.

The defined values for the `text:relative-tab-stop-position` attribute are:

- `false`: paragraph style determines position of tab stops relative to the left margin.
- `true`: paragraph style determines the position of tab stops relative to the left indent.

The default value for this attribute is `true`.

The `text:relative-tab-stop-position` attribute is usable with the following elements:  
`<text:alphabetical-index-source> 8.8.2, <text:illustration-index-source> 8.4.2, <text:object-index-source> 8.6.2, <text:table-index-source>`

8.5.2, <text:table-of-content-source> 8.3.2 and <text:user-index-source> 8.7.2.

The `text:relative-tab-stop-position` attribute has the data type `boolean` 18.3.3.

## 19.862 `text:report-type`

The `text:report-type` attribute specifies a report type for a bibliography index entry.

The `text:report-type` attribute is usable with the following element:  
<text:bibliography-mark> 8.1.11.

The `text:report-type` attribute has the data type `string` 18.2.

## 19.863 `text:restart-numbering`

The `text:restart-numbering` attribute specifies whether to restart the numbering of headings.

The defined values for the `text:restart-numbering` attribute are:

- `false`: numbering of headings does not restart.
- `true`: numbering of headings does restart.

The default value for this attribute is `false`.

The `text:restart-numbering` attribute is usable with the following element: <text:h> 5.1.2.

The `text:restart-numbering` attribute has the data type `boolean` 18.3.3.

## 19.864 `text:restart-on-page`

The `text:restart-on-page` attribute specifies whether the line count is reset to 1 at the start of every page.

If the value of this attribute is `true`, the line count is reset to 1 at the beginning of every page, resulting in page-specific numbering of lines. The default value of this attribute is `false`, resulting in document-specific numbering of lines.

The defined values for the `text:restart-on-page` attribute are:

- `false`: line count is not reset, resulting in document-wide numbering of lines.
- `true`: line count is reset to 1 at the beginning of every page, resulting in page-specific line numbering.

The default value for this attribute is `false`.

The `text:restart-on-page` attribute is usable with the following element:  
<text:linenumbering-configuration> 16.31.1.

The `text:restart-on-page` attribute has the data type `boolean` 18.3.3.

## 19.865 `text:row-number`

The `text:row-number` attribute specifies a row number to select when a condition is `true`.

The `text:row-number` attribute is usable with the following element: <text:database-row-select> 7.6.5.

The `text:row-number` attribute has the data type `nonNegativeInteger` 18.2.

## 19.866 text:school

The `text:school` attribute specifies a school for a bibliography index entry.

The `text:school` attribute is usable with the following element: `<text:bibliography-mark>` 8.1.11.

The `text:school` attribute has the data type `string` 18.2.

## 19.867 text:section-name

The `text:section-name` attribute specifies a section to which a section is linked by its name. The referenced section occurs either in the document referenced by the `xlink:href` 19.916 attribute on the same `<text:section-source>` element, or in the same document if the `xlink:href` 19.916 attribute is not present.

If the `text:section-name` attribute is not present, the section is linked to the entire document referenced by the `xlink:href` attribute, unless the `xlink:href` attribute contains a fragment identifier. If neither the `xlink:href` attribute nor the `text:section-name` attribute is present, the `<text:section-source>` element shall be ignored.

The `text:section-name` attribute is usable with the following element: `<text:section-source>` 5.4.2.

The `text:section-name` attribute has the data type `string` 18.2.

## 19.868 text:series

The `text:series` attribute specifies the series for a bibliography index entry.

The `text:series` attribute is usable with the following element: `<text:bibliography-mark>` 8.1.11.

The `text:series` attribute has the data type `string` 18.2.

## 19.869 text:select-page

### 19.869.1 General

The `text:select-page` attribute specifies conditional display of text.

### 19.869.2 <text:page-continuation>

The `text:select-page` attribute specifies whether to check for a previous or next page and if that page exists, continuation text is displayed.

The defined values for the `text:select-page` attribute are:

- `next`: check for next page and if it exists, displays continuation text.
- `previous`: check for previous page and if it exists, display continuation text.

The `text:select-page` attribute is usable with the following element: `<text:page-continuation>` 7.3.5.

The values of the `text:select-page` attribute are `previous` or `next`.

### 19.869.3 <text:page-number>

The `text:select-page` attribute specifies whether to display or not the number of a previous or following page rather than the number of the current page.

The defined values for the `text:select-page` attribute are:

- `current`: number of the current page.
- `next`: number of the page immediately following the current page.
- `previous`: number of the page immediately preceding the current page.

**Note:** To display the current page number on all pages except the first or last page, use a combination of the `text:select-page` and `text:page-adjust` 19.851 attributes.

The `text:select-page` attribute is usable with the following element: `<text:page-number>` 7.3.4.

The values of the `text:select-page` attribute are `previous`, `current` or `next`.

## 19.870 `text:separation-character`

The `text:separation-character` attribute specifies the character used to separate values representing levels in an outline.

If the value of the `text:display-outline-level` attribute is a non-zero value, a separation character may be specified. The default separation character is " ." (U+002E, FULL STOP). If the value of `text:display-outline-level` 19.851 is zero, this attribute shall be omitted.

The `text:separation-character` attribute is usable with the following element: `<text:sequence-decl>` 7.4.12.

The `text:separation-character` attribute has the data type `character` 18.3.7.

## 19.871 `text:sort-algorithm`

The `text:sort-algorithm` specifies a locale-specific sorting algorithm by name.

The `text:sort-algorithm` attribute is usable with the following elements: `<text:alphabetical-index-source>` 8.8.2 and `<text:bibliography-configuration>` 16.31.6.

The `text:sort-algorithm` attribute has the data type `string` 18.2.

## 19.872 `text:sort-ascending`

The `text:sort-ascending` attribute specifies whether locale-specific sorting takes place in ascending or descending order.

The defined values for the `text:sort-ascending` attribute are:

- `false`: sorting takes place in descending order.
- `true`: sorting takes place in ascending order.

The default value for this attribute is `true`.

The `text:sort-ascending` attribute is usable with the following element: `<text:sort-key>` 16.31.7.

The `text:sort-ascending` attribute has the data type `boolean` 18.3.3.

## 19.873 `text:sort-by-position`

The `text:sort-by-position` attribute specifies whether bibliography entries are displayed in the order of their positions in the document, or by selected fields.

The defined values for the `text:sort-by-position` attribute are:

- `false`: bibliographic entries are displayed in the order of their positions in a document.
- `true`: bibliographic entries are displayed as sorted by selected fields. The collation order for entries sorted by fields is determined by the attributes `fo:language` 19.236, `fo:country` 19.234, `fo:script` 19.242, `style:rfc-language-tag` 19.516 and `text:sort-algorithm` 19.516.

The default value for this attribute is `true`.

The `text:sort-by-position` attribute is usable with the following element:  
`<text:bibliography-configuration>` 16.31.6.

The `text:sort-by-position` attribute has the data type `boolean` 18.3.3.

## 19.874 `text:start-value`

### 19.874.1 General

The `text:start-value` attribute specifies the value for starting or restarting numbering..

### 19.874.2 `<text:h>`

The `text:start-value` attribute specifies a value that restarts numbering at the current heading.

The `text:start-value` attribute is usable with the following element: `<text:h>` 5.1.2.

The `text:start-value` attribute has the data type `nonNegativeInteger` 18.2.

### 19.874.3 `<text:list-item>`

The `text:start-value` attribute specifies a value that restarts numbering of a list at the current item. This attribute can only be applied to items in a list with a numbering list style.

The `text:start-value` attribute is usable with the following element: `<text:list-item>` 5.3.4.

The `text:start-value` attribute has the data type `nonNegativeInteger` 18.2.

### 19.874.4 `<text:list-level-style-number>`

The `text:start-value` attribute specifies a value that restarts numbering at the current list level. A value of zero is permitted.

For a `<text:list-level-style-number>` 16.34 element the default value for this attribute is 1.

The `text:start-value` attribute is usable with the following element: `<text:list-level-style-number>` 16.34.

The `text:start-value` attribute has the data type `nonNegativeInteger` 18.2.

### 19.874.5 `<text:notes-configuration>`

The `text:start-value` attribute specifies the value at which note numbering starts.

For a `<text:notes-configuration>` 16.31.3 element the default value for this attribute is 1.

The `text:start-value` attribute is usable with the following element: `<text:notes-configuration>` 16.31.3.

The `text:start-value` attribute has the data type `nonNegativeInteger` 18.2.

## 19.874.6 <text:numbered-paragraph> (deprecated)

The `text:start-value` attribute specifies a value that restarts numbering of a numbered paragraph at the current item.

The `text:start-value` attribute is usable with the following element: `<text:numbered-paragraph>` 5.3.6.

The `text:start-value` attribute has the data type `nonNegativeInteger` 18.2.

## 19.874.7 <text:outline-level-style>

The `text:start-value` attribute specifies the first number of a heading item at the current level. A value of zero is permitted.

For a `<text:outline-level-style>` 16.37 element the default value for this attribute is 1.

The `text:start-value` attribute is usable with the following element: `<text:outline-level-style>` 16.37.

The `text:start-value` attribute has the data type `positiveInteger` 18.2.

## 19.875 text:start-numbering-at

The `text:start-numbering-at` attribute specifies if footnote numbers start with a new number at the beginning of the document or at the beginning of each chapter or page.

The defined values for the `text:start-numbering-at` attribute are:

- `chapter`: footnote numbers start with a new number at the beginning of a chapter.
- `document`: footnote numbers start with a new number at the beginning of a document.
- `page`: footnote numbers start with a new number at the beginning of a page.

The `text:start-numbering-at` attribute is usable with the following element: `<text:notes-configuration>` 16.31.3.

The values of the `text:start-numbering-at` attribute are `document`, `chapter` or `page`.

## 19.876 text:string-value-phonetic

The `text:string-value-phonetic` attribute specifies a phonetic description of a word.

The `text:string-value-phonetic` attribute is usable with the following elements: `<text:alphabetical-index-mark>` 8.1.10 and `<text:alphabetical-index-mark-start>` 8.1.8.

The `text:string-value-phonetic` attribute has the data type `string` 18.2.

## 19.877 text:string-value

### 19.877.1 General

The `text:string-value` attribute specifies text for display.

### 19.877.2 <text:alphabetical-index-mark>

The `text:string-value` attribute specifies text to be displayed in an alphabetical index.

The `text:string-value` attribute is usable with the following element: `<text:alphabetical-index-mark>` 8.1.10.

The `text:string-value` attribute has the data type `string` 18.2.

### 19.877.3 `<text:hidden-text>`

The `text:string-value` attribute specifies the text to display if a `text:condition` attribute has the value of `false`. If this attribute is omitted, the element content is used.

The `text:string-value` attribute is usable with the following element: `<text:hidden-text>` 7.7.4.

The `text:string-value` attribute has the data type `string` 18.2.

### 19.877.4 `<text:page-continuation>`

The `text:string-value` attribute specifies continuation text to display. If this attribute is omitted, the element content is used.

The `text:string-value` attribute is usable with the following element: `<text:page-continuation>` 7.3.5.

The `text:string-value` attribute has the data type `string` 18.2.

### 19.877.5 `<text:toc-mark>`

The `text:string-value` attribute specifies text to be displayed in a table of contents.

The `text:string-value` attribute is usable with the following element: `<text:toc-mark>` 8.1.4.

The `text:string-value` attribute has the data type `string` 18.2.

### 19.877.6 `<text:user-index-mark>`

The `text:string-value` attribute specifies text to be displayed in an index.

The `text:string-value` attribute is usable with the following element: `<text:user-index-mark>` 8.1.7.

The `text:string-value` attribute has the data type `string` 18.2.

## 19.878 `text:string-value-if-false`

The `text:string-value-if-false` attribute specifies text string to display if a condition is `false`.

The `text:string-value-if-false` attribute is usable with the following element: `<text:conditional-text>` 7.7.3.

The `text:string-value-if-false` attribute has the data type `string` 18.2.

## 19.879 `text:string-value-if-true`

The `text:string-value-if-true` attribute specifies a text string to display if a condition is `true`.

The `text:string-value-if-true` attribute is usable with the following element: `<text:conditional-text>` 7.7.3.

The `text:string-value-if-true` attribute has the data type `string` 18.2.

## 19.880 text:style-name

### 19.880.1 General

The `text:style-name` attribute specifies a style by name, with the additional requirement that the style belongs to a family that is consistent with the style's use.

### 19.880.2 <text:a>

The `text:style-name` attribute specifies a text style for an unvisited hyperlink.

The `text:style-name` attribute is usable with the following element: `<text:a>` 6.1.8.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.880.3 <text:alphabetical-index>

The `text:style-name` attribute specifies a section family style.

The `text:style-name` attribute is usable with the following element:  
`<text:alphabetical-index>` 8.8.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.880.4 <text:alphabetical-index-entry-template>

The `text:style-name` attribute specifies a paragraph family style.

The `text:style-name` attribute is usable with the following element:  
`<text:alphabetical-index-entry-template>` 8.8.4.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.880.5 <text:bibliography>

The `text:style-name` attribute specifies a section family style.

The `text:style-name` attribute is usable with the following element:  
`<text:bibliography>` 8.9.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.880.6 <text:bibliography-entry-template>

The `text:style-name` attribute specifies a paragraph family style.

The `text:style-name` attribute is usable with the following element:  
`<text:bibliography-entry-template>` 8.9.3.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.880.7 <text:h>

The `text:style-name` attribute specifies a paragraph family style. If a conditional style is applied, this attribute has the name of the style that was the result of the conditional style evaluation.

If both `text:style-name` and `text:class-names` 19.775 are present, the style referenced by the `text:style-name` attribute is treated as the first style in the list in `text:class-names`. Consumers should support the `text:class-names` attribute and also should preserve it while editing.

The `text:style-name` attribute is usable with the following element: `<text:h>` 5.1.2.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.880.8 <text:illustration-index>

The `text:style-name` attribute specifies a section family style.

The `text:style-name` attribute is usable with the following element:  
<text:illustration-index> 8.4.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.880.9 <text:illustration-index-entry-template>

The `text:style-name` attribute specifies a paragraph family style.

The `text:style-name` attribute is usable with the following element:  
<text:illustration-index-entry-template> 8.4.3.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.880.10 <text:index-entry-bibliography>

The `text:style-name` attribute specifies a text style.

The `text:style-name` attribute is usable with the following element: <text:index-entry-bibliography> 8.13.5.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.880.11 <text:index-entry-chapter>

The `text:style-name` attribute specifies a text style.

The `text:style-name` attribute is usable with the following element: <text:index-entry-chapter> 8.13.1.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.880.12 <text:index-entry-link-end>

The `text:style-name` attribute specifies a text style.

The `text:style-name` attribute is usable with the following element: <text:index-entry-link-end> 8.13.8.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.880.13 <text:index-entry-link-start>

The `text:style-name` attribute specifies a text style.

The `text:style-name` attribute is usable with the following element: <text:index-entry-link-start> 8.13.7.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.880.14 <text:index-entry-page-number>

The `text:style-name` attribute specifies a text style.

The `text:style-name` attribute is usable with the following element: <text:index-entry-page-number> 8.13.3.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.880.15 `<text:index-entry-span>`

The `text:style-name` attribute specifies a text style.

The `text:style-name` attribute is usable with the following element: `<text:index-entry-span>` 8.13.4.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.880.16 `<text:index-entry-tab-stop>`

The `text:style-name` attribute specifies a text style.

The `text:style-name` attribute is usable with the following element: `<text:index-entry-tab-stop>` 8.13.6.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.880.17 `<text:index-entry-text>`

The `text:style-name` attribute specifies a text style.

The `text:style-name` attribute is usable with the following element: `<text:index-entry-text>` 8.13.2.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.880.18 `<text:index-source-style>`

The `text:style-name` attribute specifies a paragraph family style.

The `text:style-name` attribute is usable with the following element: `<text:index-source-style>` 8.11.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.880.19 `<text:index-title>`

The `text:style-name` attribute specifies a section family style.

The `text:style-name` attribute is usable with the following element: `<text:index-title>` 8.2.3.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.880.20 `<text:index-title-template>`

The `text:style-name` attribute specifies a text style.

The `text:style-name` attribute is usable with the following element: `<text:index-title-template>` 8.12.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.880.21 `<text:linenumbering-configuration>`

The `text:style-name` attribute specifies a text style.

The `text:style-name` attribute is usable with the following element: `<text:linenumbering-configuration>` 16.31.1.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.880.22 <text:list>

The `text:style-name` attribute specifies the name of a list style that is applied to a list.

If this attribute is not included and therefore no list style is specified, one of the following actions is taken:

- If a list is contained within another list, the list style defaults to the style of the surrounding list.
- If there is no list style specified for the surrounding list, but the list's list items contain paragraphs that have paragraph styles attached specifying a list style, that list style is used.
- An implementation-dependent default list style is used.

To determine which formatting properties are applied to a list, the list level and list style name are taken into account.

The `text:style-name` attribute is usable with the following element: <`text:list`> 5.3.1.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.880.23 <text:list-level-style-number>

The `text:style-name` attribute specifies a text style.

The `text:style-name` attribute is usable with the following element: <`text:list-level-style-number`> 16.34.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.880.24 <text:list-level-style-bullet>

The `text:style-name` attribute specifies a text style.

The `text:style-name` attribute is usable with the following element: <`text:list-level-style-bullet`> 16.33.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.880.25 <text:numbered-paragraph> (deprecated)

The `text:style-name` attribute specifies the name of a list style that is applied to a list.

If this attribute is not included and therefore no list style is specified, one of the following actions is taken:

- If the list is contained within another list, the list style defaults to the style of the surrounding list.
- If there is no list style specified for the surrounding list, but the list contains paragraphs that have paragraph styles attached specifying a list style, this list style is used for any of these paragraphs.
- An implementation-dependent default list style is applied to any other paragraphs.

To determine which formatting properties are applied to a list, the list level and list style name are taken into account.

The `text:style-name` attribute is usable with the following element: <`text:numbered-paragraph`> 5.3.6.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.880.26 <text:object-index>

The `text:style-name` attribute specifies a section family style.

The `text:style-name` attribute is usable with the following element: `<text:object-index>` 8.6.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.880.27 <text:object-index-entry-template>

The `text:style-name` attribute specifies a paragraph family style.

The `text:style-name` attribute is usable with the following element: `<text:object-index-entry-template>` 8.6.3.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.880.28 <text:outline-level-style>

The `text:style-name` attribute specifies a text style.

The `text:style-name` attribute is usable with the following element: `<text:outline-level-style>` 16.37.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.880.29 <text:p>

The `text:style-name` attribute specifies a paragraph family style. If a conditional style is applied, this attribute has the name of the style that was the result of the conditional style evaluation.

If both `text:style-name` and `text:class-names` are present, the style referenced by the `text:style-name` attribute is treated as the first style in the list in `text:class-names` 19.775. Consumers should support the `text:class-names` attribute and also should preserve it while editing.

The `text:style-name` attribute is usable with the following element: `<text:p>` 5.1.3.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.880.30 <text:ruby>

The `text:style-name` attribute specifies a ruby family style.

The `text:style-name` attribute is usable with the following element: `<text:ruby>` 6.4.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.880.31 <text:ruby-text>

The `text:style-name` attribute specifies a text style.

The `text:style-name` attribute is usable with the following element: `<text:ruby-text>` 6.4.3.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.880.32 <text:section>

The `text:style-name` attribute specifies a section family style.

The `text:style-name` attribute is usable with the following element: `<text:section>` 5.4.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.880.33 `<text:span>`

The `text:style-name` attribute specifies style for span which shall be a style with family of text.

If both `text:style-name` and `text:class-names` are present, the style referenced by the `text:style-name` attribute is treated as the first style in the list in `text:class-names` 19.775. Consumers should support the `text:class-names` attribute and also should preserve it while editing.

The `text:style-name` attribute is usable with the following element: `<text:span>` 6.1.7.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.880.34 `<text:table-index-entry-template>`

The `text:style-name` attribute specifies a paragraph family style.

The `text:style-name` attribute is usable with the following element: `<text:table-index-entry-template>` 8.5.3.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.880.35 `<text:table-of-content>`

The `text:style-name` attribute specifies a section family style.

The `text:style-name` attribute is usable with the following element: `<text:table-of-content>` 8.3.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.880.36 `<text:table-of-content-entry-template>`

The `text:style-name` attribute specifies a paragraph family style.

The `text:style-name` attribute is usable with the following element: `<text:table-of-content-entry-template>` 8.3.3.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.880.37 `<text:table-index>`

The `text:style-name` attribute specifies a section family style.

The `text:style-name` attribute is usable with the following element: `<text:table-index>` 8.5.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.880.38 `<text:user-index>`

The `text:style-name` attribute specifies a section family style.

The `text:style-name` attribute is usable with the following element: `<text:user-index>` 8.7.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.880.39 <text:user-index-entry-template>

The `text:style-name` attribute specifies a paragraph family style.

The `text:style-name` attribute is usable with the following element: `<text:user-index-entry-template>` 8.7.3.

The `text:style-name` attribute has the data type `styleNameRef` 18.3.33.

### 19.881 text:style-override

The `text:style-override` attribute specifies an override to the list style to be applied to a list item.

The `text:style-override` attribute is usable with the following element: `<text:list-item>` 5.3.4.

The `text:style-override` attribute has the data type `styleNameRef` 18.3.33.

### 19.882 text:suffix

The `text:suffix` attribute specifies a string that is displayed after a bibliography entry's short name or number if it occurs outside the bibliography.

The `text:suffix` attribute is usable with the following element: `<text:bibliography-configuration>` 16.31.6.

The `text:suffix` attribute has the data type `string` 18.2.

### 19.883 text:tab-ref

The `text:tab-ref` attribute contains the number of the tab-stop to which a tab character refers. The position 0 marks the start margin of a paragraph.

**Note:** The `text:tab-ref` attribute is only a hint to help non-layout-oriented consumers to determine the association of tab with tab-stop. Layout-oriented consumers should determine the tab positions based on the style information.

The `text:tab-ref` attribute is usable with the following element: `<text:tab>` 6.1.4.

The `text:tab-ref` attribute has the data type `nonNegativeInteger` 18.2.

### 19.884 text:table-name

The `text:table-name` attribute specifies a table within a source database.

The `text:table-name` attribute is usable with the following elements: `<text:database-display>` 7.6.3, `<text:database-name>` 7.6.7, `<text:database-next>` 7.6.4, `<text:database-row-number>` 7.6.6 and `<text:database-row-select>` 7.6.5.

The `text:table-name` attribute has the data type `string` 18.2.

### 19.885 text:table-type

The `text:table-type` attribute specifies the type of reference made to a database.

The defined values for the `text:table-type` attribute are:

- `table` : the value of the `text:table-name` 19.884 attribute is the name of a database table.
- `query` : the value of the `text:table-name` attribute is the name of a database query.
- `command` : the value of the `text:table-name` attribute is an SQL statement.

The `text:table-type` attribute is usable with the following elements: `<text:database-display>` 7.6.3, `<text:database-name>` 7.6.7, `<text:database-next>` 7.6.4, `<text:database-row-number>` 7.6.6 and `<text:database-row-select>` 7.6.5.

The values of the `text:table-type` attribute are `table`, `query` or `command`.

## 19.886 `text:time-adjust`

The `text:time-adjust` attribute specifies an adjustment of the value of a time by a specific time period. Positive values adjust the time to a time in the future, while negative values adjust the time to a time in the past. Duration values are truncated to full minutes.

**Note:** Truncation of time values is defined by `number:truncate-on-overflow`. 19.369

The `text:time-adjust` attribute is usable with the following element: `<text:time>` 7.3.3.

The `text:time-adjust` attribute has the data type `duration` 18.2.

## 19.887 `text:time-value`

The `text:time-value` attribute specifies the time at which a document was last edited.

If no value is specified, whether the field is marked as fixed or not, the current time is assumed.

The `text:time-value` attribute is usable with the following elements: `<text:creation-time>` 7.5.4, `<text:modification-time>` 7.5.15, `<text:print-time>` 7.5.7 and `<text:time>` 7.3.3.

The `text:time-value` attribute has the data type `timeOrDateTime` 18.3.37.

## 19.888 `text:title`

The `text:title` attribute specifies a title for a bibliography index entry.

The `text:title` attribute is usable with the following element: `<text:bibliography-mark>` 8.1.11.

The `text:title` attribute has the data type `string` 18.2.

## 19.889 `text:track-changes`

The `text:track-changes` attribute specifies whether changes to the document should be tracked and recorded.

The defined values for the `text:track-changes` attribute are:

- `false`: changes are not tracked.
- `true`: changes are tracked.

The default value for this attribute is `true`.

The `text:track-changes` attribute is usable with the following element: `<text:tracked-changes>` 5.5.2.

The `text:track-changes` attribute has the data type `boolean` 18.3.3.

## 19.890 `text:url`

The `text:url` attribute specifies an IRI for a bibliography index entry.

The `text:url` attribute is usable with the following element: `<text:bibliography-mark>` 8.1.11.

The `text:url` attribute has the data type `string` 18.2.

## 19.891 `text:use-caption`

The `text:use-caption` attribute specifies whether captions or names of illustrations or tables are used for an index.

The defined values of the `text:use-caption` attribute are:

- `false`: illustration or table names are used for an index.
- `true`: illustration or table captions are used for an index.

The default value for this attribute is `true`.

The `text:use-caption` attribute is usable with the following elements: `<text:illustration-index-source>` 8.4.2 and `<text:table-index-source>` 8.5.2.

The `text:use-caption` attribute has the data type `boolean` 18.3.3.

## 19.892 `text:use-chart-objects`

The `text:use-chart-objects` attribute specifies whether to include embedded chart objects in an index of objects.

The defined values for the `text:use-chart-objects` attribute are:

- `false`: object is not included in index of objects.
- `true`: object is included in index of objects.

The default value for this attribute is `false`.

The `text:use-chart-objects` attribute is usable with the following element: `<text:object-index-source>` 8.6.2.

The `text:use-chart-objects` attribute has the data type `boolean` 18.3.3.

## 19.893 `text:use-draw-objects`

The `text:use-draw-objects` attribute specifies whether embedded draw objects (defined by `<draw:object>` 10.4.6.2 elements) will be included in an index of objects.

The defined values for the `text:use-draw-objects` attribute are:

- `false`: draw objects not included in an index of objects.
- `true`: draw objects included in an index of objects.

The default value for this attribute is `false`.

The `text:use-draw-objects` attribute is usable with the following element: `<text:object-index-source>` 8.6.2.

The `text:use-draw-objects` attribute has the data type `boolean` 18.3.3.

## 19.894 `text:use-floating-frames`

The `text:use-floating-frames` attribute specifies whether text boxes will be included in user-defined indexes.

The defined values for the `text:use-floating-frames` attribute are:

- `false`: text boxes not included in user-defined indexes.
- `true`: text boxes included in user-defined indexes.

The default value for this attribute is `false`.

The `text:use-floating-frames` attribute is usable with the following element:  
`<text:user-index-source> 8.7.2.`

The `text:use-floating-frames` attribute has the data type boolean 18.3.3.

## 19.895 `text:use-graphics`

The `text:use-graphics` attribute specifies whether graphics are include in user-defined indexes.

The defined values for the `text:use-graphics` attribute are:

- `false`: graphics are not included in user-defined indexes.
- `true`: graphics are included in user-defined indexes.

The default value for this attribute is `false`.

The `text:use-graphics` attribute is usable with the following element: `<text:user-index-source> 8.7.2.`

The `text:use-graphics` attribute has the data type boolean 18.3.3.

## 19.896 `text:use-index-marks`

The `text:use-index-marks` attribute specifies whether table of contents index marks are used to generate index entries. The `text:outline-level 19.850` attribute specifies up to which level index marks are being included.

The defined values for the `text:use-index-marks` attribute are:

- `false`: table of contents does not include entries generated from table of contents index marks.
- `true`: table of contents does include entries generated from table of contents index marks.

For a `<text:user-index-source> 8.7.2` element the default value for this attribute is `false`.

The `text:use-index-marks` attribute is usable with the following elements:  
`<text:table-of-content-source> 8.3.2` and `<text:user-index-source> 8.7.2.`

The `text:use-index-marks` attribute has the data type boolean 18.3.3.

## 19.897 `text:use-index-source-styles`

The `text:use-index-source-styles` attribute specifies whether index entries are generated for paragraphs formatted using the paragraph styles listed in the `<text:index-source-styles>` element.

The `text:outline-level 19.850` attribute specifies up to which level index source styles are being included.

The defined values for the `text:use-index-source-styles` attribute are:

- `false`: index entries are not generated for paragraphs formatted using the paragraph styles listed in a `<text:index-source-styles>` element.

- `true`: index entries are generated for paragraphs formatted using the paragraph styles listed in a `<text:index-source-styles>` element.

For a `<text:user-index-source>` 8.7.2 element the default value for this attribute is `false`.

The `text:use-index-source-styles` attribute is usable with the following elements: `<text:table-of-content-source>` 8.3.2 and `<text:user-index-source>` 8.7.2.

The `text:use-index-source-styles` attribute has the data type `boolean` 18.3.3.

## 19.898 `text:use-keys-as-entries`

The `text:use-keys-as-entries` attribute specifies the use of the up to two keys for an index mark as entries in the index.

The defined values for the `text:use-keys-as-entries` attribute are:

- `false`: additional keys are used as sub-entries.
- `true`: additional keys are used as additional entries.

The default value for this attribute is `false`.

The `text:use-keys-as-entries` attribute is usable with the following element: `<text:alphabetical-index-source>` 8.8.2.

The `text:use-keys-as-entries` attribute has the data type `boolean` 18.3.3.

## 19.899 `text:use-math-objects`

The `text:use-math-objects` attribute specifies whether embedded math objects are included in an index of objects.

The defined values for the `text:use-math-objects` attribute are:

- `false`: math objects are not included in an index of objects.
- `true`: math objects are included in an index of objects.

The default value for this attribute is `false`.

The `text:use-math-objects` attribute is usable with the following element: `<text:object-index-source>` 8.6.2.

The `text:use-math-objects` attribute has the data type `boolean` 18.3.3.

## 19.900 `text:use-objects`

The `text:use-objects` attribute specifies whether an object is included in a user-defined index.

The defined values for the `text:use-objects` attribute are:

- `false`: objects are not included in a user define index.
- `true`: objects are included in a user-defined index.

The default value for this attribute is `false`.

The `text:use-objects` attribute is usable with the following element: `<text:user-index-source>` 8.7.2.

The `text:use-objects` attribute has the data type `boolean` 18.3.3.

## 19.901 text:use-other-objects

The `text:use-other-objects` attribute specifies whether other embedded objects are included in an index of objects. Other objects are those not covered by `text:use-chart-objects` 19.892, `text:use-draw-objects` 19.893, `text:use-math-objects` 19.899, and `text:use-spreadsheet-objects` 19.904 attributes.

The defined values for the `text:use-other-objects` attribute are:

- `false`: other objects are not included in an index of objects.
- `true`: other objects are included in an index of objects.

The default value for this attribute is `false`.

The `text:use-other-objects` attribute is usable with the following element:  
`<text:object-index-source>` 8.6.2.

The `text:use-other-objects` attribute has the data type `boolean` 18.3.3.

## 19.902 text:use-outline-level

The `text:use-outline-level` attribute specifies whether headings are used to generate index entries. The `text:outline-level` attribute specifies up to which level headings are being included. 19.850

The defined values for the `text:use-outline-level` attribute are:

- `false`: table of contents does not include entries generated from headings.
- `true`: table of contents does include entries generated from headings.

The default value for this attribute is `true`.

The `text:use-outline-level` attribute is usable with the following element:  
`<text:table-of-content-source>` 8.3.2.

The `text:use-outline-level` attribute has the data type `boolean` 18.3.3.

## 19.903 text:use-soft-page-breaks

The `text:use-soft-page-breaks` attribute specifies whether a document contains soft page breaks.

A soft page break is a page break that has been included in a document by a page-oriented consumer at a position where the document itself does not include a page break as defined by `fo:break-after` 20.184 or `fo:break-before` 20.185.

Soft page breaks are specified by the `<text:soft-page-break>` element.

OpenDocument producers need not produce `<text:soft-page-break>` elements. They may include them if they have computed a paginated layout. Consumers may handle the element while computing the layout, but it shall not depend on its existence. Soft page breaks are only defined for text documents.

Producers that store soft page breaks shall indicate this by setting the `text:use-soft-page-breaks` attribute to `true`. Producers that do not store soft page breaks shall indicate that by omitting this attribute, or by setting it to `false`.

A producer that does not support pagination and soft page-breaks and that modifies an OpenDocument file, which includes soft page-breaks, shall set the `text:use-soft-page-breaks` attribute to `false` (or remove it). It should also remove the `<text:soft-page-break>` elements from the document.

Producers that compute a paginated layout of a document should provide a facility for the export of soft page breaks for the purposes of consistent page breaks and for conversion to talking book formats ( [DAISY]).

For `<text:soft-page-break>` elements that appear within table rows, the maximum number of `<text:soft-page-break>` elements that appear within the single table cells determines the number of page breaks that appear within the table row. The `<text:soft-page-break>` elements contained in each cell determine the positions where these page breaks appear within the cell content.

When `<text:soft-page-break>` elements appear within text boxes and other content displayed outside the text flow, they do not start a new page, but only indicate where the text-box's content breaks between two pages.

The defined values for the `text:use-soft-page-breaks` attribute are:

- `false`: soft page breaks not supported.
- `true`: soft page breaks supported.

The default value for this attribute is `false`.

The `text:use-soft-page-breaks` attribute is usable with the following element:  
`<office:text> 3.4.`

The `text:use-soft-page-breaks` attribute has the data type `boolean` 18.3.3.

## 19.904 text:use-spreadsheet-objects

The `text:use-spreadsheet-objects` attribute specifies whether embedded spreadsheet objects will appear in an object index.

The defined values for the `text:use-spreadsheet-objects` attribute are:

- `false`: spreadsheet objects do not appear in an object index.
- `true`: spreadsheet objects do appear in an object index.

The default value for this attribute is `false`.

The `text:use-spreadsheet-objects` attribute is usable with the following element:  
`<text:object-index-source> 8.6.2.`

The `text:use-spreadsheet-objects` attribute has the data type `boolean` 18.3.3.

## 19.905 text:use-tables

The `text:use-tables` attribute specifies whether tables will appear in a user-defined index.

The defined values for the `text:use-tables` attribute are:

- `false`: tables will not appear in a user-defined index.
- `true`: tables will appear in a user-defined index.

The default value for this attribute is `false`.

The `text:use-tables` attribute is usable with the following element: `<text:user-index-source> 8.7.2.`

The `text:use-tables` attribute has the data type `boolean` 18.3.3.

## 19.906 text:value

### 19.906.1 General

The `text:value` attribute specifies a value.

### 19.906.2 <text:label>

The `text:value` attribute specifies the text of a `<text:label>` element.

The `text:value` attribute is usable with the following element: `<text:label>` 7.4.17.

The `text:value` attribute has the data type `string` 18.2.

### 19.906.3 <text:database-row-number>

The `text:value` attribute specifies the current row number for a `<text:database-row-number>` element. The number changes when new data is added to the current document.

The `text:value` attribute is usable with the following element: `<text:database-row-number>` 7.6.6.

The `text:value` attribute has the data type `nonNegativeInteger` 18.2.

## 19.907 text:visited-style-name

The `text:visited-style-name` attribute specifies a style for a hyperlink that has been visited.

The `text:visited-style-name` attribute is usable with the following element: `<text:a>` 6.1.8.

The `text:visited-style-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.908 text:volume

The `text:volume` attribute specifies a volume for a bibliography index entry.

The `text:volume` attribute is usable with the following element: `<text:bibliography-mark>` 8.1.11.

The `text:volume` attribute has the data type `string` 18.2.

## 19.909 text:year

The `text:year` attribute specifies a year for a bibliography index entry.

The `text:year` attribute is usable with the following element: `<text:bibliography-mark>` 8.1.11.

The `text:year` attribute has the data type `string` 18.2.

## 19.910 xforms:bind

The `xforms:bind` attribute specifies the binding of an OpenDocument control to an XForm form.

With OpenDocument form buttons, the `bind` attribute refers to a `form:xform-submission` 19.319 attribute with the given ID. Pushing the button causes the appropriate XForms submission action to be performed.

**Note:** OpenDocument form button controls are: `<form:button>` 13.5.15, and, `<form:image>` 13.5.16.

With OpenDocument form non-button controls, the `xforms:bind` attribute refers to an `<xforms:bind>` element with the given ID. Any such bound control reads and writes its data as determined by the appropriate bind element.

Note: OpenDocument form non-button controls are: `<form:checkbox>` 13.5.17, `<form:column>` 13.5.23, `<form:combobox>` 13.5.11, `<form:date>` 13.5.8, `<form:file>` 13.5.5, `<form:fixed-text>` 13.5.10, `<form:formatted-text>` 13.5.6, `<form:frame>` 13.5.19, `<form:generic-control>` 13.5.25, `<form:grid>` 13.5.22, `<form:hidden>` 13.5.21, `<form:image-frame>` 13.5.20, `<form:item>` 13.5.12, `<form:listbox>` 13.5.13, `<form:number>` 13.5.7, `<form:option>` 13.5.14, `<form:password>` 13.5.4, `<form:radio>` 13.5.18, `<form:text>` 13.5.2, `<form:textarea>` 13.5.3, `<form:time>` 13.5.9, and `<form:value-range>` 13.5.24.

The `xforms:bind` attribute is usable with the following elements: `<form:button>` 13.5.15, `<form:checkbox>` 13.5.17, `<form:combobox>` 13.5.11, `<form:date>` 13.5.8, `<form:file>` 13.5.5, `<form:fixed-text>` 13.5.10, `<form:formatted-text>` 13.5.6, `<form:frame>` 13.5.19, `<form:generic-control>` 13.5.25, `<form:grid>` 13.5.22, `<form:hidden>` 13.5.21, `<form:image>` 13.5.16, `<form:image-frame>` 13.5.20, `<form:listbox>` 13.5.13, `<form:number>` 13.5.7, `<form:password>` 13.5.4, `<form:radio>` 13.5.18, `<form:text>` 13.5.2, `<form:textarea>` 13.5.3, `<form:time>` 13.5.9 and `<form:value-range>` 13.5.24.

The `xforms:bind` attribute has the data type `string` 18.2.

## 19.911 `xhtml:about`

The `xhtml:about` attribute specifies the subject of an RDF statement for in-content metadata. Predicates for RDF statements are specified by the `xhtml:property` attribute. 19.914

The RDF statement's object is specified by the `xhtml:content` 19.912 attribute, if present, or it is

- the literal content between a pair of `<text:bookmark-start>` and `<text:bookmark-end>` elements that are paired by the values of their `text:name` attributes. 6.2.1.3, 6.2.1.4
- for elements other than `<text:bookmark-start>` and `<text:bookmark-end>`, the literal content of the element containing an `xhtml:about` attribute.

The `xhtml:datatype` attribute provides the data type of the RDF object. 19.913

The `xhtml:about` attribute is usable with the following elements: `<table:covered-table-cell>` 9.1.5, `<table:table-cell>` 9.1.4, `<text:bookmark-start>` 6.2.1.3, `<text:h>` 5.1.2, `<text:meta>` 6.1.9 and `<text:p>` 5.1.3.

The `xhtml:about` attribute has the data type `URIorSafeCURIE` 18.3.38.

## 19.912 `xhtml:content`

The `xhtml:content` attribute specifies an object of an RDF statement for in-content metadata. 19.911.

The `xhtml:content` attribute is usable with the following elements: `<table:covered-table-cell>` 9.1.5, `<table:table-cell>` 9.1.4, `<text:bookmark-start>` 6.2.1.3, `<text:h>` 5.1.2, `<text:meta>` 6.1.9 and `<text:p>` 5.1.3.

The `xhtml:content` attribute has the data type `string` 18.2.

## 19.913 xhtml:datatype

The `xhtml:datatype` attribute specifies the RDF data type (see §5 of [RDF-CONCEPTS]) of the object of an RDF statement.

The `xhtml:datatype` attribute is usable with the following elements: `<table:covered-table-cell> 9.1.5, <table:table-cell> 9.1.4, <text:bookmark-start> 6.2.1.3, <text:h> 5.1.2, <text:meta> 6.1.9 and <text:p> 5.1.3.`

The `xhtml:datatype` attribute has the data type CURIE [18.3.12](#).

## 19.914 xhtml:property

The `xhtml:property` attribute specifies a list of one or more CURIEs. Each CURIE is the predicate of an RDF statement for in content-metadata. [19.911](#).

The `xhtml:property` attribute is usable with the following elements: `<table:covered-table-cell> 9.1.5, <table:table-cell> 9.1.4, <text:bookmark-start> 6.2.1.3, <text:h> 5.1.2, <text:meta> 6.1.9 and <text:p> 5.1.3.`

The `xhtml:property` attribute has the data type CURIEs [18.3.13](#).

## 19.915 xlink:actuate

See §5.6.2 of [XLink].

The `xlink:actuate` attribute has the value `onRequest` for the following elements: `<db:component> 12.25.5, <db:connection-resource> 12.7, <draw:a> 10.4.12, <form:form> 13.3, <meta:template> 4.3.2.12, <presentation:event-listener> 10.9.2, <presentation:sound> 10.8.2, <script:event-listener> 14.5.2, <svg:definition-src> 16.27, <svg:font-face-uri> 16.26, <table:cell-range-source> 9.3.1, <table:table-source> 9.2.6 and <text:a> 6.1.8.`

The `xlink:actuate` attribute has the value `onLoad` for the following elements:

`<draw:applet> 10.4.7, <draw:fill-image> 16.42.6, <draw:floating-frame> 10.4.10, <draw:image> 10.4.4, <draw:object> 10.4.6.2, <draw:object-ole> 10.4.6.3, <draw:plugin> 10.4.8, <meta:auto-reload> 4.3.2.13, <style:background-image> 17.3 and <text:list-level-style-image> 16.35.`

For `<db:connection-resource> 12.7, <draw:a> 10.4.12, <form:form> 13.3, <meta:template> 4.3.2.12, <presentation:event-listener> 10.9.2, <presentation:sound> 10.8.2, <script:event-listener> 14.5.2, <svg:definition-src> 16.27, <svg:font-face-uri> 16.26, <table:cell-range-source> 9.3.1, <table:table-source> 9.2.6 and <text:a> 6.1.8` elements the default value for this attribute is `onRequest`.

For `<draw:applet> 10.4.7, <draw:fill-image> 16.42.6, <draw:floating-frame> 10.4.10, <draw:image> 10.4.4, <draw:object> 10.4.6.2, <draw:object-ole> 10.4.6.3, <draw:plugin> 10.4.8, <meta:auto-reload> 4.3.2.13, <style:background-image> 17.3 and <text:list-level-style-image> 16.35` elements the default value for this attribute is `onLoad`.

The `xlink:actuate` attribute is usable with the following elements: `<db:component> 12.25.5, <db:connection-resource> 12.7, <draw:a> 10.4.12, <draw:applet> 10.4.7, <draw:fill-image> 16.42.6, <draw:floating-frame> 10.4.10, <draw:image> 10.4.4, <draw:object> 10.4.6.2, <draw:object-ole> 10.4.6.3, <draw:plugin> 10.4.8, <form:form> 13.3, <meta:auto-reload> 4.3.2.13, <meta:template> 4.3.2.12, <presentation:event-listener> 10.9.2, <presentation:sound> 10.8.2, <script:event-listener> 14.5.2, <style:background-image> 17.3, <svg:definition-src> 16.27, <svg:font-face-uri> 16.26, <table:cell-range-`

source> 9.3.1, <table:table-source> 9.2.6, <text:a> 6.1.8 and <text:list-level-style-image> 16.35.

## 19.916 xlink:href

### 19.916.1 General

The `xlink:href` 19.916 attribute specifies a remote resource. Its data type is `anyIRI`. See §5.4 of [XLink].

### 19.916.2 <anim:audio>

The `xlink:href` attribute specifies an audio stream.

The `xlink:href` attribute is usable with the following element: <anim:audio> 15.5.

The `xlink:href` attribute has the data type `anyIRI` 18.3.2.

### 19.916.3 <chart:chart>

The `xlink:href` attribute at the <chart:chart> element specifies the document which provides the data for a chart. The following values are supported:

- ". ." (U+002E, FULL STOP): The data is provided by the chart document itself. It is taken from the <table:table> 9.1.2 element inside the <chart:chart> element.
- ". ." (U+002E, FULL STOP followed by U+002E, FULL STOP): The data is provided by the document into which the chart is embedded. This is only supported if the chart is embedded into another document.
- A relative IRI: The data is provided by an embedded object referenced by the IRI. This embedded object shall be within the same package as the chart itself.

All cell range addresses that are specified within a chart are relative to the document referenced by the `xlink:href` attribute.

For a <chart:chart> element, if the `xlink:href` attribute is omitted, its value is assumed to be ". ." (U+002E, FULL STOP followed by U+002E, FULL STOP).

The `xlink:href` attribute is usable with the following element: <chart:chart> 11.1.

The `xlink:href` attribute has the data type `anyIRI` 18.3.2.

### 19.916.4 <chart:symbol-image>

The `xlink:href` attribute specifies a file containing an image.

The `xlink:href` attribute is usable with the following element: <chart:symbol-image> 17.23.

The `xlink:href` attribute has the data type `anyIRI` 18.3.2.

### 19.916.5 <db:component>

The `xlink:href` attribute specifies a form or report document.

The `xlink:href` attribute is usable with the following element: <db:component> 12.25.5.

The `xlink:href` attribute has the data type `anyIRI` 18.3.2.

## 19.916.6 <db:connection-resource>

The `xlink:href` attribute specifies either a database, or database connection data encoded as an IRI.

The `xlink:href` attribute is usable with the following element: `<db:connection-resource>` 12.7.

The `xlink:href` attribute has the data type anyIRI 18.3.2.

## 19.916.7 <db:file-based-database>

The `xlink:href` attribute specifies the location of a file that defines a database. If a database consists of multiple files of different types which all are required for it to be operational, `xlink:href` points to one of those files which defines the database as a whole.

The `xlink:href` attribute is usable with the following element: `<db:file-based-database>` 12.5.

The `xlink:href` attribute has the data type anyIRI 18.3.2.

## 19.916.8 <draw:a>

The `xlink:href` attribute specifies a target location for the hyperlink.

The `xlink:href` attribute is usable with the following element: `<draw:a>` 10.4.12.

The `xlink:href` attribute has the data type anyIRI 18.3.2.

## 19.916.9 <draw:applet>

The `xlink:href` attribute specifies the base IRI for an applet.

For a `<draw:applet>` element the default value for this attribute is an empty string.

The `xlink:href` attribute is usable with the following element: `<draw:applet>` 10.4.7.

The `xlink:href` attribute has the data type anyIRI 18.3.2.

## 19.916.10 <draw:area-circle>

The `xlink:href` attribute specifies a target location for the hyperlink.

The `xlink:href` attribute is usable with the following element: `<draw:area-circle>` 10.4.13.4.

The `xlink:href` attribute has the data type anyIRI 18.3.2.

## 19.916.11 <draw:area-polygon>

The `xlink:href` attribute specifies a target location for the hyperlink.

The `xlink:href` attribute is usable with the following element: `<draw:area-polygon>` 10.4.13.5.

The `xlink:href` attribute has the data type anyIRI 18.3.2.

## 19.916.12 <draw:area-rectangle>

The `xlink:href` attribute specifies a target location for the hyperlink.

The `xlink:href` attribute is usable with the following element: `<draw:area-rectangle>` 10.4.13.3.

The `xlink:href` attribute has the data type anyIRI 18.3.2.

### 19.916.13 `<draw:fill-image>`

The `xlink:href` attribute specifies a file containing an image.

The `xlink:href` attribute is usable with the following element: `<draw:fill-image>` 16.42.6.

The `xlink:href` attribute has the data type anyIRI 18.3.2.

### 19.916.14 `<draw:floating-frame>`

The `xlink:href` attribute specifies the source of a floating frame.

The `xlink:href` attribute is usable with the following element: `<draw:floating-frame>` 10.4.10.

The `xlink:href` attribute has the data type anyIRI 18.3.2.

### 19.916.15 `<draw:image>`

The `xlink:href` attribute specifies a file containing an image.

The `xlink:href` attribute is usable with the following element: `<draw:image>` 10.4.4.

The `xlink:href` attribute has the data type anyIRI 18.3.2.

### 19.916.16 `<draw:object>`

The `xlink:href` attribute specifies the location of an embedded object.

The `xlink:href` attribute is usable with the following element: `<draw:object>` 10.4.6.2.

The `xlink:href` attribute has the data type anyIRI 18.3.2.

### 19.916.17 `<draw:object-ole>`

The `xlink:href` attribute specifies the location of an embedded object.

The `xlink:href` attribute is usable with the following element: `<draw:object-ole>` 10.4.6.3.

The `xlink:href` attribute has the data type anyIRI 18.3.2.

### 19.916.18 `<draw:plugin>`

The `xlink:href` attribute specifies the source of a plugin.

The `xlink:href` attribute is usable with the following element: `<draw:plugin>` 10.4.8.

The `xlink:href` attribute has the data type anyIRI 18.3.2.

### 19.916.19 `<form:button>`

The `xlink:href` attribute specifies a resource that is loaded if a button is operated.

The `xlink:href` attribute is usable with the following element: `<form:button>` 13.5.15.

The `xlink:href` attribute has the data type anyIRI 18.3.2.

## 19.916.20 `<form:connection-resource>`

The `xlink:href` attribute specifies either a database, or database connection encoded as an IRI.

The `xlink:href` attribute is usable with the following element: `<form:connection-resource>` 7.6.2.

The `xlink:href` attribute has the data type anyIRI 18.3.2.

## 19.916.21 `<form:image>`

The `xlink:href` attribute specifies a resource that is loaded if a button is operated.

The `xlink:href` attribute is usable with the following element: `<form:image>` 13.5.16.

The `xlink:href` attribute has the data type anyIRI 18.3.2.

## 19.916.22 `<form:form>`

The `xlink:href` attribute specifies the processing agent for a form.

The `xlink:href` attribute is usable with the following element: `<form:form>` 13.3.

The `xlink:href` attribute has the data type anyIRI 18.3.2.

## 19.916.23 `<meta:auto-reload>`

The `xlink:href` attribute specifies the IRI of a replacement document.

The `xlink:href` attribute is usable with the following element: `<meta:auto-reload>` 4.3.2.13.

The `xlink:href` attribute has the data type anyIRI 18.3.2.

## 19.916.24 `<meta:template>`

The `xlink:href` attribute specifies the location of a document template.

The `xlink:href` attribute is usable with the following element: `<meta:template>` 4.3.2.12.

The `xlink:href` attribute has the data type anyIRI 18.3.2.

## 19.916.25 `<presentation:event-listener>`

The `xlink:href` attribute specifies either a document bookmark or an application depending on the action selected by the `presentation:action` 19.391 attribute.

The `xlink:href` attribute is usable with the following element: `<presentation:event-listener>` 10.9.2.

The `xlink:href` attribute has the data type anyIRI 18.3.2.

## 19.916.26 `<presentation:sound>`

The `xlink:href` attribute specifies a sound that is played when an effect is executed.

The `xlink:href` attribute is usable with the following element: `<presentation:sound>` 10.8.2.

The `xlink:href` attribute has the data type anyIRI 18.3.2.

### 19.916.27 `<script:event-listener>`

The `xlink:href` attribute specifies macro code that should be called for an event. The IRI may have any protocol. The interpretation of the IRI is script-language-dependent.

The `xlink:href` attribute is usable with the following element: `<script:event-listener>` 14.5.2.

The `xlink:href` attribute has the data type `anyIRI` 18.3.2.

### 19.916.28 `<style:background-image>`

The `xlink:href` attribute specifies an image.

The `xlink:href` attribute is usable with the following element: `<style:background-image>` 17.3.

The `xlink:href` attribute has the data type `anyIRI` 18.3.2.

### 19.916.29 `<svg:definition-src>`

See § 20.8.3 of [SVG].

The `xlink:href` attribute is usable with the following element: `<svg:definition-src>` 16.27.

The `xlink:href` attribute has the data type `anyIRI` 18.3.2.

### 19.916.30 `<svg:font-face-uri>`

See § 20.8.3 of [SVG].

The `xlink:href` attribute is usable with the following element: `<svg:font-face-uri>` 16.26.

The `xlink:href` attribute has the data type `anyIRI` 18.3.2.

### 19.916.31 `<table:cell-range-source>`

The `xlink:href` attribute specifies an IRI for a document containing a source table.

The `xlink:href` attribute is usable with the following element: `<table:cell-range-source>` 9.3.1.

The `xlink:href` attribute has the data type `anyIRI` 18.3.2.

### 19.916.32 `<table:table-source>`

The `xlink:href` attribute specifies an IRI of a document containing a source table.

The `xlink:href` attribute is usable with the following element: `<table:table-source>` 9.2.6.

The `xlink:href` attribute has the data type `anyIRI` 18.3.2.

### 19.916.33 `<text:a>`

The `xlink:href` attribute specifies the target location of a hyperlink.

The `xlink:href` attribute is usable with the following element: `<text:a>` 6.1.8.

The `xlink:href` attribute has the data type `anyIRI` 18.3.2.

### 19.916.34 <text:alphabetical-index-auto-mark-file>

The `xlink:href` attribute specifies the location of an alphabetical index mark file that contains a list of terms.

The `xlink:href` attribute is usable with the following element: `<text:alphabetical-index-auto-mark-file>` 8.8.3.

The `xlink:href` attribute has the data type `anyIRI` 18.3.2.

### 19.916.35 <text:list-level-style-image>

The `xlink:href` attribute specifies an image.

The `xlink:href` attribute is usable with the following element: `<text:list-level-style-image>` 16.35.

The `xlink:href` attribute has the data type `anyIRI` 18.3.2.

### 19.916.36 <text:script>

The `xlink:href` attribute specifies the location script source code.

The `xlink:href` attribute is usable with the following element: `<text:script>` 7.7.9.

The `xlink:href` attribute has the data type `anyIRI` 18.3.2.

### 19.916.37 <text:section-source>

The `xlink:href` attribute specifies the document or document fragment to which a section is linked. If the IRI contains a fragment identifier, and if the referenced document is an OpenDocument document, the fragment identifier should be interpreted as the name of a section to which a section is linked. A fragment identifier should be ignored if a `text:section-name` 19.867 attribute is present.

The `xlink:href` attribute is usable with the following element: `<text:section-source>` 5.4.2.

The `xlink:href` attribute has the data type `anyIRI` 18.3.2.

## 19.917 `xlink:show`

See §5.6.1 of [XLink].

For a `<meta:auto-reload>` 4.3.2.13 element the value for this attribute is `replace`.

For `<draw:applet>` 10.4.7, `<draw:fill-image>` 16.42.6, `<draw:floating-frame>` 10.4.10, `<draw:image>` 10.4.4, `<draw:object>` 10.4.6.2, `<draw:object-ole>` 10.4.6.3, `<draw:plugin>` 10.4.8, `<presentation:event-listener>` 10.9.2, `<style:background-image>` 17.3, `<text:list-level-style-image>` 16.35 and `<text:section-source>` 5.4.2 elements the value for this attribute is `embed`.

For `<draw:a>` 10.4.12, `<draw:area-circle>` 10.4.13.4, `<draw:area-polygon>` 10.4.13.5, `<draw:area-rectangle>` 10.4.13.3, `<meta:hyperlink-behaviour>` 4.3.2.14, `<presentation:sound>` 10.8.2 and `<text:a>` 6.1.8 elements the supported values for this attribute are `new` or `replace`.

For a `<meta:auto-reload>` 4.3.2.13 element the default value for this attribute is `replace`.

For `<draw:applet>` 10.4.7, `<draw:fill-image>` 16.42.6, `<draw:floating-frame>` 10.4.10, `<draw:image>` 10.4.4, `<draw:object>` 10.4.6.2, `<draw:object-ole>` 10.4.6.3, `<draw:plugin>` 10.4.8, `<presentation:event-listener>` 10.9.2,

<style:background-image> 17.3, <text:list-level-style-image> 16.35 and <text:section-source> 5.4.2 elements the default value for this attribute is `embed`.

For <db:connection-resource> 12.7 the default value for this attribute is `none`.

The `xlink:show` attribute is usable with the following elements: <db:component> 12.25.5, <db:connection-resource> 12.7, <draw:a> 10.4.12, <draw:applet> 10.4.7, <draw:area-circle> 10.4.13.4, <draw:area-polygon> 10.4.13.5, <draw:area-rectangle> 10.4.13.3, <draw:fill-image> 16.42.6, <draw:floating-frame> 10.4.10, <draw:image> 10.4.4, <draw:object> 10.4.6.2, <draw:object-ole> 10.4.6.3, <draw:plugin> 10.4.8, <meta:auto-reload> 4.3.2.13, <meta:hyperlink-behaviour> 4.3.2.14, <presentation:event-listener> 10.9.2, <presentation:sound> 10.8.2, <style:background-image> 17.3, <text:a> 6.1.8, <text:list-level-style-image> 16.35 and <text:section-source> 5.4.2.

## 19.918 `xlink:title`

The `xlink:title` attribute specifies a title to be associated with a document template.

The `xlink:title` attribute is usable with the following element: <meta:template> 4.3.2.12.

The `xlink:title` attribute has the data type `string` 18.2.

## 19.919 `xlink:type`

See §3.2 of [XLink]. This attribute always has the value `simple` in OpenDocument document instances.

For <db:connection-resource> 12.7, <db:file-based-database> 12.5, <draw:a> 10.4.12, <draw:applet> 10.4.7, <draw:area-circle> 10.4.13.4, <draw:area-polygon> 10.4.13.5, <draw:area-rectangle> 10.4.13.3, <draw:fill-image> 16.42.6, <draw:floating-frame> 10.4.10, <draw:image> 10.4.4, <draw:object> 10.4.6.2, <draw:object-ole> 10.4.6.3, <draw:plugin> 10.4.8, <form:form> 13.3, <meta:auto-reload> 4.3.2.13, <meta:template> 4.3.2.12, <presentation:event-listener> 10.9.2, <presentation:sound> 10.8.2, <script:event-listener> 14.5.2, <style:background-image> 17.3, <svg:definition-src> 16.27, <svg:font-face-uri> 16.26, <table:cell-range-source> 9.3.1, <table:table-source> 9.2.6, <text:a> 6.1.8, <text:alphabetical-index-auto-mark-file> 8.8.3, <text:list-level-style-image> 16.35, <text:script> 7.7.9 and <text:section-source> 5.4.2 elements the default value for this attribute is `simple`.

The `xlink:type` attribute is usable with the following elements: <chart:chart> 11.1, <db:component> 12.25.5, <db:connection-resource> 12.7, <db:file-based-database> 12.5, <draw:a> 10.4.12, <draw:applet> 10.4.7, <draw:area-circle> 10.4.13.4, <draw:area-polygon> 10.4.13.5, <draw:area-rectangle> 10.4.13.3, <draw:fill-image> 16.42.6, <draw:floating-frame> 10.4.10, <draw:image> 10.4.4, <draw:object> 10.4.6.2, <draw:object-ole> 10.4.6.3, <draw:plugin> 10.4.8, <form:form> 13.3, <meta:auto-reload> 4.3.2.13, <meta:template> 4.3.2.12, <presentation:event-listener> 10.9.2, <presentation:sound> 10.8.2, <script:event-listener> 14.5.2, <style:background-image> 17.3, <svg:definition-src> 16.27, <svg:font-face-uri> 16.26, <table:cell-range-source> 9.3.1, <table:table-source> 9.2.6, <text:a> 6.1.8, <text:alphabetical-index-auto-mark-file> 8.8.3, <text:list-level-style-image> 16.35, <text:script> 7.7.9 and <text:section-source> 5.4.2.

## 19.920 `xml:id`

See [XML-ID].

The `xml:id` attribute is usable with the following elements: `<anim:audio> 15.5, <anim:command> 15.6.1, <anim:iterate> 15.4.4, <anim:par> 15.4.2, <anim:seq> 15.4.3, <chart:chart> 11.1, <chart:data-point> 11.14, <chart:plot-area> 11.5, <chart:series> 11.12, <dr3d:cube> 10.5.4, <dr3d:extrude> 10.5.6, <dr3d:rotate> 10.5.7, <dr3d:scene> 10.5.2, <dr3d:sphere> 10.5.5, <draw:a> 10.4.12, <draw:applet> 10.4.7, <draw:caption> 10.3.11, <draw:circle> 10.3.8, <draw:connector> 10.3.10, <draw:control> 10.3.13, <draw:custom-shape> 10.6.1, <draw:ellipse> 10.3.9, <draw:floating-frame> 10.4.10, <draw:frame> 10.4.2, <draw:g> 10.3.15, <draw:image> 10.4.4, <draw:line> 10.3.3, <draw:measure> 10.3.12, <draw:object> 10.4.6.2, <draw:object-ole> 10.4.6.3, <draw:page> 10.2.4, <draw:page-thumbnail> 10.3.14, <draw:path> 10.3.7, <draw:plugin> 10.4.8, <draw:polygon> 10.3.5, <draw:polyline> 10.3.4, <draw:rect> 10.3.2, <draw:regular-polygon> 10.3.6, <draw:text-box> 10.4.3, <form:button> 13.5.15, <form:checkbox> 13.5.17, <form:combobox> 13.5.11, <form:date> 13.5.8, <form:file> 13.5.5, <form:fixed-text> 13.5.10, <form:formatted-text> 13.5.6, <form:frame> 13.5.19, <form:generic-control> 13.5.25, <form:grid> 13.5.22, <form:hidden> 13.5.21, <form:image> 13.5.16, <form:image-frame> 13.5.20, <form:listbox> 13.5.13, <form:number> 13.5.7, <form:password> 13.5.4, <form:radio> 13.5.18, <form:text> 13.5.2, <form:textarea> 13.5.3, <form:time> 13.5.9, <form:value-range> 13.5.24, <office:annotation> 14.1, <presentation:sound> 10.8.2, <table:covered-table-cell> 9.1.5, <table:table> 9.1.2, <table:table-cell> 9.1.4, <table:table-column> 9.1.6, <table:table-row> 9.1.3, <text:alphabetical-index> 8.8, <text:bibliography> 8.9, <text:bookmark> 6.2.1.2, <text:bookmark-start> 6.2.1.3, <text:changed-region> 5.5.3, <text:h> 5.1.2, <text:illustration-index> 8.4, <text:index-title> 8.2.3, <text:list> 5.3.1, <text:list-header> 5.3.3, <text:list-item> 5.3.4, <text:meta> 6.1.9, <text:meta-field> 7.5.19, <text:numbered-paragraph> 5.3.6, <text:object-index> 8.6, <text:p> 5.1.3, <text:section> 5.4, <text:table-index> 8.5, <text:table-of-content> 8.3 and <text:user-index> 8.7.`

The `xml:id` attribute has the data type [ID](#) [18.2](#).

## 19.921 `form:input-required`

The `form:input-required` attribute specifies whether a control that is bound to a database field requires user input for the form to submit.

The attribute is not evaluated for a control that is not bound to a database field via `form:data-field` [19.263](#), or that has the `form:convert-empty-to-null` [19.259](#) attribute evaluate to "false".

The defined values for the `form:input-required` attribute are:

- `false`: a control that has no user input does not prevent submission of the form.
- `true`: a control that has no user input does prevent submission of the form.

The default value for this attribute is `false`.

The `form:input-required` attribute is usable with the following elements: `<form:checkbox> 13.5.17, <form:combobox> 13.5.11, <form:date> 13.5.8, <form:formatted-text> 13.5.6, <form:image-frame> 13.5.20, <form:listbox> 13.5.13, <form:password> 13.5.4, <form:radio> 13.5.18, <form:text> 13.5.2, <form:textarea> 13.5.3 and <form:time> 13.5.9.`

The `form:input-required` attribute has the data type [boolean](#) [18.3.3](#).

## 19.922 style:num-list-format-name

The `style:num-list-format-name` attribute references a format for the displayed number string of the current list level.

If the attribute is present, `text:display-levels` 19.803, `style:num-prefix` 19.506 and `style:num-suffix` 19.507 shall be ignored.

Producers should also generate `text:display-levels`, `style:num-prefix` and `style:num-suffix` attributes for the benefit of consumers that don't support this feature.

The `style:num-list-format-name` attribute is usable with the following elements: `<text:list-level-style-number>` 16.34 and `<text:outline-level-style>` 16.37.

The `style:num-list-format-name` attribute has the data type `styleNameRef` 18.3.33.

## 19.923 draw:extrusion-metal-type

The `draw:extrusion-metal-type` attribute specifies how rendering of an extruded custom shape is modified in case its `draw:extrusion-metal` attribute has the value `true`.

The value of attribute `draw:extrusion-metal-type` is a namespaced token.

The defined values for the `draw:extrusion-metal-type` attribute are:

- `draw:MetalODF`: The rendering is modified as specified in attribute `draw:extrusion-metal` (19.158)
- any implementation-defined `namespacedToken` (18.3.20): The rendering is modified as defined by the implementer. Such namespaced token shall not use a prefix, which is bound to a namespace listed in section 1.5 Namespaces.

The attribute `draw:extrusion-metal-type` is only evaluated, if the attribute `draw:extrusion-metal` at the same shape has the value `true`.

The default value for the attribute `draw:extrusion-metal-type` is `draw:MetalODF`.

The `draw:extrusion-metal-type` attribute is usable with the following element: `<draw:enhanced-geometry>` 10.6.2.

The `draw:extrusion-metal-type` attribute has the data type `namespacedToken` 18.3.20.

## 19.924 draw:handle-position-x

The `draw:handle-position-x` attribute specifies the horizontal position of an XY-handle, see `<draw:handle>` 10.6.3 element.

The value of the `draw:handle-position-x` attribute is a string with the following syntax:

`handle-position-x-value ::= handle-parameter`

where `handle-parameter` is defined in `<draw:handle>` 10.6.3.

The resulting value is interpreted as x-coordinate of the position of the XY-handle.

The `draw:handle-position-x` attribute is usable with the following element: `<draw:handle>` 10.6.3.

The `draw:handle-position-x` attribute has the data type `string` 18.2.

## 19.925 draw:handle-position-y

The `draw:handle-position-y` attribute specifies the vertical position of an XY-handle, see `<draw:handle>` 10.6.3 element.

The value of the `draw:handle-position-y` attribute is a string with the following syntax:

```
handle-position-y-value ::= handle-parameter
```

where handle-parameter is defined in `<draw:handle>` 10.6.3.

The resulting value is interpreted as y-coordinate of the position of the XY-handle.

The `draw:handle-position-y` attribute is usable with the following element:  
`<draw:handle>` 10.6.3.

The `draw:handle-position-y` attribute has the data type string 18.2.

## 19.926 `draw:handle-polar-pole-x`

The `draw:handle-polar-pole-x` attribute specifies the horizontal position of the pole of the polar coordinate system used for a polar handle, see `<draw:handle>` 10.6.3 element.

The value of the `draw:handle-polar-pole-x` attribute is a string with the following syntax:

```
handle-polar-pole-x-value ::= handle-parameter
```

where handle-parameter is defined in `<draw:handle>` 10.6.3.

The resulting value is interpreted as x-coordinate of the pole.

The `draw:handle-polar-pole-x` attribute is usable with the following element:  
`<draw:handle>` 10.6.3.

The `draw:handle-polar-pole-x` attribute has the data type string 18.2.

## 19.927 `draw:handle-polar-pole-y`

The `draw:handle-polar-pole-y` attribute specifies the vertical position of the pole of the polar coordinate system used for a polar handle, see `<draw:handle>` 10.6.3 element.

The value of the `draw:handle-polar-pole-y` attribute is a string with the following syntax:

```
handle-polar-pole-y-value ::= handle-parameter
```

where handle-parameter is defined in `<draw:handle>` 10.6.3.

The resulting value is interpreted as y-coordinate of the pole.

The `draw:handle-polar-pole-y` attribute is usable with the following element:  
`<draw:handle>` 10.6.3.

The `draw:handle-polar-pole-y` attribute has the data type string 18.2.

## 19.928 `draw:handle-polar-radius`

The `draw:handle-polar-radius` attribute specifies the radius component of the polar coordinates of the handle position of a polar-handle, see `<draw:handle>` 10.6.3 element.

The value of the `draw:handle-polar-radius` attribute is a string with the following syntax:

```
handle-polar-radius-value ::= handle-parameter
```

where handle-parameter is defined in `<draw:handle>` 10.6.3.

The resulting value is interpreted as distance from the pole of the polar coordinate system to the position of the handle.

If the resulting value is negative, the handle position is undefined.

If the resulting value is zero, the handle position is on the pole of the polar coordinate system regardless of the value of the `draw:handle-polar-angle` 19.929 attribute.

The default value of the attribute is the string 0.

The `draw:handle-polar-radius` attribute is usable with the following element:  
`<draw:handle> 10.6.3.`

The `draw:handle-polar-radius` attribute has the data type `string 18.2`.

## 19.929 `draw:handle-polar-angle`

The `draw:handle-polar-angle` attribute specifies the polar angle component of the polar coordinates of the handle position of a polar handle, see `<draw:handle> 10.6.3` element.

The value of the `draw:handle-polar-angle` attribute is a string with the following syntax:

`handle-polar-angle-value ::= handle-parameter`

where `handle-parameter` is defined in `<draw:handle> 10.6.3`.

The resulting value is interpreted as angle in degrees.

The default value of the attribute is the string 0.

The `draw:handle-polar-angle` attribute is usable with the following element:  
`<draw:handle> 10.6.3.`

The `draw:handle-polar-angle` attribute has the data type `string 18.2`.

---

## 20 Formatting Attributes

### 20.1 General

In an OpenDocument Format document, `<style:*-properties>` elements contain the definitions of formatting that is associated with document structure elements. Formatting definitions are expressed as attributes on the `<style:*-properties>` elements or as child elements of these elements. This chapter defines formatting properties that are represented by attributes. Formatting properties that are represented by elements are defined in chapter 17.

### 20.2 chart:angle-offset

The `chart:angle-offset` attribute specifies a counter-clockwise rotation of a polar coordinate in a circle, ring or polar chart. The attribute value is an angle. See 18.3.1. The default value is `90deg`.

This attribute is evaluated for chart styles that are applied to a `<chart:plot-area>` 11.5 element.

The `chart:angle-offset` attribute is usable with the following element: `<style:chart-properties>` 17.22.

The `chart:angle-offset` attribute has the data type `angle` 18.3.1.

### 20.3 chart:auto-position

The `chart:auto-position` attribute specifies if an object should be positioned automatically.

The default value is `true` if `svg:x 19.577` and `svg:y 19.581` attributes are not specified. It is `false` if the `svg:x` and `svg:y` attributes are specified.

The `chart:auto-position` attribute is evaluated for chart styles applied to elements that may have `svg:x` and `svg:y` attributes.

The defined values for the `chart:auto-position` attribute are:

- `false`: `svg:x` and `svg:y` attribute values are used to position an object.
- `true`: `svg:x` and `svg:y` attribute values are ignored, and the consumer chooses a position.

The `chart:auto-position` attribute is usable with the following element: `<style:chart-properties>` 17.22.

The `chart:auto-position` attribute has the data type `boolean` 18.3.3.

### 20.4 chart:auto-size

The `chart:auto-size` attribute specifies if the size of an object should be chosen automatically.

The default value is `true` if `svg:width 19.575` and `svg:height 19.543` attributes are not specified. It is `false` if the `svg:width` and `svg:height` attributes are specified.

The `chart:auto-size` attribute is evaluated for chart styles applied to elements that have `svg:width` and `svg:height` attributes.

The defined values for the `chart:auto-size` attribute are:

- `false`: `svg:width` and `svg:height` attribute values are used to size an object.
- `true`: `svg:width` and `svg:height` attribute values are ignored, and the consumer chooses a size.

The `chart:auto-size` attribute is usable with the following element: `<style:chart-properties> 17.22`.

The `chart:auto-size` attribute has the data type `boolean` 18.3.3.

## 20.5 chart:axis-label-position

The `chart:axis-label-position` attribute specifies the position of the axis labels.

The defined values for the `chart:axis-label-position` attribute are:

- `near-axis`: labels are placed adjacent to the axis line. On which side of the line the labels are placed depends on the `chart:axis-position` 20.6 attribute. If the `chart:axis-position` attribute has the value `start` or `end`, the labels are placed outside the coordinate system. Otherwise, the labels are placed adjacent to the axis on the side that belongs to the lower values on the crossing axis.
- `near-axis-other-side`: labels are placed adjacent to the axis on the opposite side as for `near-axis`.
- `outside-end`: labels are placed outside the coordinate region on that side where the crossing axis has its maximum value.
- `outside-start`: labels are placed outside the coordinate region on that side where the crossing axis has its minimum value.

If the `chart:axis-label-position` attribute is missing but a `chart:axis-position` attribute is given, the value of the `chart:axis-label-position` attribute is assumed to be `near-axis`.

This attribute is evaluated for a chart style that is applied to a `<chart:axis> 11.9` element.

The `chart:axis-label-position` attribute is usable with the following element: `<style:chart-properties> 17.22`.

The values of the `chart:axis-label-position` attribute are `near-axis`, `near-axis-other-side`, `outside-start` or `outside-end`.

## 20.6 chart:axis-position

The `chart:axis-position` attribute specifies the position of the axis line on the scale of the crossing axis.

If the attribute is set for a y-axis, the position indicates a value on the scale of the first x-axis.

If the attribute is set for an x-axis, the position indicates a value on the scale of the first y-axis.

This attribute is evaluated for a chart style that is applied to a `<chart:axis> 11.9` element.

The defined values for the `chart:axis-position` attribute are:

- `end`: axis line is placed at the end of the crossing axis.
- `start`: axis line is placed at the start of the crossing axis.
- a value of type `double`: axis line is placed at the given value on the crossing axis. If the crossing axis is an axis displaying categories rather than values, a value of `1` indicates that the axis should be placed at the first category, a value of `2` indicates that the axis should be placed at the second category and so forth.

The `chart:axis-position` attribute is usable with the following element:  
`<style:chart-properties> 17.22.`

The values of the `chart:axis-position` attribute are `start`, `end` or a value of type `double` `18.2`.

## 20.7 chart:connect-bars

The `chart:connect-bars` attribute specifies if bars in a bar chart that belong to the same series are connected by lines.

This attribute is evaluated for chart styles that are applied to a `<chart:plot-area> 11.5` element.

The defined values for the `chart:connect-bars` attribute are:

- `false`: bars belonging to the same series are not connected by lines.
- `true`: bars belonging to the same series are connected by lines.

The `chart:connect-bars` attribute is usable with the following element: `<style:chart-properties> 17.22.`

The `chart:connect-bars` attribute has the data type `boolean` `18.3.3`.

## 20.8 chart:data-label-number

The `chart:data-label-number` attribute specifies whether the value or the percentage of a data point or both, should be displayed within a data label. By default, neither the value nor a percentage is displayed.

The defined values for the `chart:data-label-number` attribute are:

- `none`: neither the value nor the percentage should be displayed.
- `percentage`: only the percentage value should be displayed.
- `value`: only the value should be displayed within the data label.
- `value-and-percentage`: both the value and the percentage should be displayed.

This attribute is evaluated for chart styles that are applied to a `<chart:data-label> 11.15`, `<chart:data-point> 11.14`, `<chart:series> 11.12` or `<chart:plot-area> 11.5` elements.

The `style:data-style-name` `19.473` attribute can be used together with this attribute to indicate which number format should be used to display the value.

The `style:percentage-data-style-name` `19.511` attribute can be used together with this attribute to indicate which number format should be used to display the percentage value.

The `chart:data-label-number` attribute is usable with the following element:  
`<style:chart-properties> 17.22.`

The values of the `chart:data-label-number` attribute are `none`, `value`, `percentage` or `value-and-percentage`.

## 20.9 chart:data-label-series

The `chart:data-label-series` attribute determines whether or not to display the series name of the corresponding series in the data label.

The defined values for the `chart:data-label-series` attribute are:

- `false`: no series name is displayed on data label.

- `true`: series name is displayed on data label.

This attribute is evaluated for chart styles that are applied to a `<chart: data-label> 11.15`, `<chart: data-point> 11.14`, `<chart: series> 11.12` or `<chart: plot-area> 11.5` elements.

The `chart: data-label-series` attribute is usable with the following element:  
`<style: chart-properties> 17.22`.

The `chart: data-label-series` attribute has the data type `boolean 18.3.3`.

## 20.10 chart: data-label-symbol

The `chart: data-label-symbol` attribute specifies whether to display a legend symbol within the data label of a data point.

This attribute is evaluated for chart styles that are applied to a `<chart: data-label> 11.15`, `<chart: data-point> 11.14`, `<chart: series> 11.12` or `<chart: plot-area> 11.5` elements.

The defined values for the `chart: data-label-symbol` attribute are:

- `false`: a legend symbol is not displayed within the data label of a data point.
- `true`: a legend symbol is displayed within the data label of a data point.

The `chart: data-label-symbol` attribute is usable with the following element:  
`<style: chart-properties> 17.22`.

The `chart: data-label-symbol` attribute has the data type `boolean 18.3.3`.

## 20.11 chart: data-label-text

The `chart: data-label-text` attribute specifies whether to display the category within the data label of a data point.

This attribute is evaluated for chart styles that are applied to `<chart: data-label> 11.15`, `<chart: data-point> 11.14`, a `<chart: series> 11.12` or a `<chart: plot-area> 11.5` elements.

The defined values for the `chart: data-label-text` attribute are:

- `false`: a category is not displayed within the data label of a data point.
- `true`: a category is displayed within the data label of a data point.

The `chart: data-label-text` attribute is usable with the following element:  
`<style: chart-properties> 17.22`.

The `chart: data-label-text` attribute has the data type `boolean 18.3.3`.

## 20.12 chart: deep

The `chart: deep` attribute specifies whether data series are displayed behind each other along the z-axis or not. The `chart: deep` attribute is ignored unless a `chart: three-dimensional 20.65` attribute is present and has the value `true`.

This attribute is evaluated for a chart style that is applied to a `<chart: plot-area> 11.5` element.

The defined values for the `chart: deep` attribute are:

- `false`: data series are not displayed behind each other along the z-axis.
- `true`: data series are displayed behind each other along the z-axis. The z-axis displays the name of the series.

The `chart:deep` attribute is usable with the following element: `<style:chart-properties>` 17.22.

The `chart:deep` attribute has the data type `boolean` 18.3.3.

## 20.13 chart:display-label

The `chart:display-label` attribute specifies whether labels are displayed on an axis or not.

This attribute is evaluated for a chart style that is applied to a `<chart:axis>` 11.9 element.

**Note:** The labels controlled by `chart:display-label` attribute appear on an axis as opposed to being labels for the axis.

The defined values for the `chart:display-label` attribute are:

- `false`: labels are not displayed on an axis.
- `true`: labels are displayed on an axis.

The `chart:display-label` attribute is usable with the following element: `<style:chart-properties>` 17.22.

The `chart:display-label` attribute has the data type `boolean` 18.3.3.

## 20.14 chart:error-category

The `chart:error-category` attribute is used to determine which function is used to calculate error indicators at data points. By default, no error indicators are shown.

The defined values for the `chart:error-category` attribute are:

- `cell-range`: Use values from cell ranges for positive and negative error indicators. The `chart:error-lower-range` 20.17 and `chart:error-upper-range` 20.22 attributes determine the cell ranges which contain the values to be used for negative and positive error indicators.
- `constant`: Use fixed absolute values for both directions: positive and negative. The absolute value for a positive direction is given with `chart:error-upper-limit` attribute. The absolute value for a negative direction is given with `chart:error-lower-limit` attribute.
- `error-margin`: Use a fixed percentage of the largest value – this is called error-margin. The percentage value is given within `chart:error-margin` 20.18 attribute.
- `none`: No error indicators are shown.
- `percentage`: Use a fixed percentage of each value. The percentage value is given within `chart:error-percentage` 20.19 attribute.
- `standard-deviation`: Standard deviation of the values of a series assuming an equal distribution.
- `standard-error`: Standard error (standard deviation of the mean) of the values of a series.
- `variance`: Variance of the values of a series assuming an equal distribution.

This attribute is evaluated for chart styles that are applied to `<chart:error-indicator>` 11.17, `<chart:series>` 11.12 or `<chart:plot-area>` 11.5 elements.

The `chart:error-category` attribute is usable with the following element: `<style:chart-properties>` 17.22.

The values of the `chart:error-category` attribute are `none`, `variance`, `standard-deviation`, `percentage`, `error-margin`, `constant`, `standard-error` or `cell-range`.

## 20.15 `chart:error-lower-indicator`

The `chart:error-lower-indicator` attribute specifies whether negative error indicators are displayed (the error value is subtracted from the data point value).

This attribute should be used together with the `chart:error-category` 20.14 attribute.

The defined values for the `chart:error-lower-indicator` attribute are:

- `false`: negative error indicators are not displayed.
- `true`: negative error indicators are displayed.

The `chart:error-lower-indicator` attribute is usable with the following element:  
`<style:chart-properties> 17.22`.

The `chart:error-lower-indicator` attribute has the data type `boolean` 18.3.3.

## 20.16 `chart:error-lower-limit`

The `chart:error-lower-limit` attribute specifies the absolute value in the negative direction that is used to display error indicators if a `chart:error-category` 20.14 attribute has the value `constant`.

The `chart:error-lower-limit` attribute is usable with the following element:  
`<style:chart-properties> 17.22`.

The `chart:error-lower-limit` attribute has the data type `double` 18.2.

## 20.17 `chart:error-lower-range`

The `chart:error-lower-range` attribute specifies the cell range used for negative error indicators when a `chart:error-category` 20.14 attribute has a value of `cell-range`.

The `chart:error-lower-range` attribute is usable with the following element:  
`<style:chart-properties> 17.22`.

The `chart:error-lower-range` attribute has the data type `cellRangeAddressList` 18.3.6.

## 20.18 `chart:error-margin`

The `chart:error-margin` attribute specifies the percentage of the largest value in a series used in the display of error indicators for each data point of a series.

This attribute should be used together with the `chart:error-category` 20.14 attribute if it has the value `error-margin`.

The `chart:error-margin` attribute is usable with the following element: `<style:chart-properties> 17.22`.

The `chart:error-margin` attribute has the data type `double` 18.2.

## 20.19 `chart:error-percentage`

The `chart:error-percentage` attribute specifies the percentage that is used in the display of error indicators for each data point of a series.

This attribute should be used together with a `chart:error-category` 20.14 attribute if the `chart:error-category` attribute has the value `percentage`.

The `chart:error-percentage` attribute is usable with the following element:  
`<style:chart-properties> 17.22`.

The `chart:error-percentage` attribute has the data type `double` 18.2.

## 20.20 chart:error-upper-indicator

The `chart:error-upper-indicator` attribute specifies whether positive error indicators should be displayed (the error value is added to the data point value).

This attribute should be used together with a `chart:error-category` 20.14 attribute.

The defined values for the `chart:error-upper-indicator` attribute are:

- `false`: positive error indicators are not displayed.
- `true`: positive error indicators are displayed.

The `chart:error-upper-indicator` attribute is usable with the following element:  
`<style:chart-properties> 17.22`.

The `chart:error-upper-indicator` attribute has the data type `boolean` 18.3.3.

## 20.21 chart:error-upper-limit

The `chart:error-upper-limit` attribute specifies the absolute value in the positive direction that is used to display error indicators if the `chart:error-category` 20.14 attribute has the value `constant`.

This attribute should be used together with `chart:error-category` attribute.

The `chart:error-upper-limit` attribute is usable with the following element:  
`<style:chart-properties> 17.22`.

The `chart:error-upper-limit` attribute has the data type `double` 18.2.

## 20.22 chart:error-upper-range

The `chart:error-upper-range` attribute specifies the cell range used for positive error indicators when the `chart:error-category` 20.14 attribute has the value `cell-range`.

The `chart:error-upper-range` attribute is usable with the following element:  
`<style:chart-properties> 17.22`.

The `chart:error-upper-range` attribute has the data type `cellRangeAddressList` 18.3.6.

## 20.23 chart:gap-width

The `chart:gap-width` attribute specifies a gap between neighboring groups of bars in a bar chart (that is the distance between the last bar in one category and the first bar in the following category). It is specified as an integer percentage relative to the width of a single bar.

This attribute is evaluated for chart styles that are applied to a `<chart:axis>` 11.9 element with `chart:dimension` 19.18 attribute set to `y`.

**Note:** Bars attached to different axis can be arranged differently as long as they are grouped per axis (`chart:group-bars-per-axis` 20.24 attribute has the value `true`).

The `chart:gap-width` attribute is usable with the following element: `<style:chart-properties>` 17.22.

The `chart:gap-width` attribute has the data type `integer` 18.2.

## 20.24 chart:group-bars-per-axis

The `chart:group-bars-per-axis` attribute specifies whether bars in a bar chart are displayed side by side or behind each other when they are attached to different y-axes.

This attribute is evaluated for chart styles that are applied to a `<chart:plot-area>` 11.5 element.

The defined values for the `chart:group-bars-per-axis` attribute are:

- `false`: all bars on the same x-axis are handled as one group and are displayed side by side.
- `true`: only bars attached to the same x- and y-axis are handled as one group.

The `chart:group-bars-per-axis` attribute is usable with the following element: `<style:chart-properties>` 17.22.

The `chart:group-bars-per-axis` attribute has the data type `boolean` 18.3.3.

## 20.25 chart:hole-size

The `chart:hole-size` attribute specifies the diameter of the inner hole of a ring chart as a percentage of the outer diameter of the outermost ring.

This attribute is evaluated for chart styles that are applied to a `<chart:plot-area>` 11.5 element.

The `chart:hole-size` attribute is usable with the following element: `<style:chart-properties>` 17.22.

The `chart:hole-size` attribute has the data type `percent` 18.3.24.

## 20.26 chart:include-hidden-cells

The `chart:include-hidden-cells` attribute specifies whether data points in hidden cells are plotted in a chart. The default is to plot data points in hidden cells.

This attribute can be used in styles that are applied to a `<chart:plot-area>` 11.5 element.

The defined values for the `chart:include-hidden-cells` attribute are:

- `false`: data points in hidden cells are not plotted in a chart.
- `true`: data points in hidden cells are plotted in a chart.

The `chart:include-hidden-cells` attribute is usable with the following element: `<style:chart-properties>` 17.22.

The `chart:include-hidden-cells` attribute has the data type `boolean` 18.3.3.

## 20.27 chart:interpolation

The `chart:interpolation` attribute specifies interpolations for line and scatter charts.

The defined values for the `chart:interpolation` attribute are:

- `b-spline` – B-Splines. If the `chart:spline-resolution` 20.58 attribute has value `1` this is identical to the `chart:interpolation` attribute value `none`. If the `chart:spline-resolution` attribute has value `k>1` and the `chart:spline-order`

20.57 attribute has value  $p$  then given a sequence of data points, a B-spline interpolation with polynomials of degree  $p$  is constructed as described here. First, we omit any consecutive repeated data points obtaining  $(x_0, y_0), \dots, (x_n, y_n)$  where for all  $i=1, \dots, n$ ,  $(x_{i-1}, y_{i-1}) \neq (x_i, y_i)$ .

We then construct a parameter list  $t_0, \dots, t_n$  using the centripetal method with a

$$\text{power of 0.5: } t_k = \frac{\sum_{i=1}^k \sqrt[4]{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2}}{\sum_{i=1}^n \sqrt[4]{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2}} \text{ for } k=1, \dots, n \text{ and } t_0=0.$$

Next we generate a knot vector  $u_0, \dots, u_m$  with  $m=n+p+1$ :

$$\begin{aligned} u_0 = u_1 = \dots = u_p &= 0 \\ u_{j+p} &= \frac{1}{p} \sum_{i=j}^{j+p-1} t_i \text{ for } j=1, 2, \dots, n-p \\ u_{m-p} = u_{m-p+1} = \dots = u_m &= 1 \end{aligned}$$

Now let  $N_{i,p}(u)$  be the B-spline basis function defined recursively by

$$\begin{aligned} N_{i,0}(u) &= \begin{cases} 1 & \text{if } u_i \leq u < u_{i+1} \\ 0 & \text{otherwise} \end{cases} \\ N_{i,s}(u) &= \frac{u - u_i}{u_{i+s} - u_i} \cdot N_{i,s-1}(u) + \frac{u_{i+s+1} - u}{u_{i+s+1} - u_{i+1}} \cdot N_{i+1,s-1}(u) \text{ for } s > 0 \end{aligned}$$

Let  $C = \begin{bmatrix} c_0 & d_0 \\ c_1 & d_1 \\ \vdots & \vdots \\ c_n & d_n \end{bmatrix}$  be the solution of the matrix equation  $X = N \cdot C$  with

$$N = \begin{bmatrix} N_{0,p}(t_0) & N_{1,p}(t_0) & \cdots & N_{n,p}(t_0) \\ N_{0,p}(t_1) & N_{1,p}(t_1) & \cdots & N_{n,p}(t_1) \\ \vdots & \vdots & \ddots & \vdots \\ N_{0,p}(t_n) & N_{1,p}(t_n) & \cdots & N_{n,p}(t_n) \end{bmatrix} \text{ and } X = \begin{bmatrix} x_0 & y_0 \\ x_1 & y_1 \\ \vdots & \vdots \\ x_n & y_n \end{bmatrix}.$$

$(c_0, d_0), (c_1, d_1), \dots, (c_n, d_n)$  are the control points of the desired B-spline of degree  $p$  passing through the data points  $(x_0, y_0), \dots, (x_n, y_n)$ .

For  $0 < u < 1$  let  $P_{i,r}(u)$  be recursively defined by

$$\begin{aligned} P_{i,0}(u) &= (c_i, d_i) \\ P_{i,r}(u) &= (1 - a_{i,r}) \cdot P_{i-1,r-1} + a_{i,r} \cdot P_{i,r-1} \text{ with } a_{i,r} = \frac{u - u_i}{u_{i+p-r+1} - u_i} \text{ for } r > 0 \end{aligned}$$

If  $0 < u < 1$  and  $u_s < u < u_{s+1}$ , then  $P_{s,p}(u)$  is the point on this B-spline corresponding to the parameter  $u$ . If  $0 < u < 1$  and  $u = u_s$ , then  $P_{s-1,p-1}(u)$  is the point on this B-spline corresponding to the parameter  $u$ .

Note:  $P_{s,p}(u)$  and  $P_{s-1,p-1}(u)$  are best calculated using de Boor's algorithm for a B-spline of degree  $p$  with control points  $(c_0, d_0), (c_1, d_1), \dots, (c_n, d_n)$  and knots  $u_0, \dots, u_m$ .

The approximation of this B-spline is then plotted by using  $P_{s,p}(u)$  and  $P_{s-1,p-1}(u)$  to calculate the coordinates of the points on the spline corresponding to the parameters:

$$\begin{aligned}
 0 &= t_0, & t_0 + 1 \cdot \frac{t_1 - t_0}{k}, & t_0 + 2 \cdot \frac{t_1 - t_0}{k}, & \dots, & t_0 + k \cdot \frac{t_1 - t_0}{k} = t_1, \\
 t_1 + 1 \cdot \frac{t_2 - t_1}{k}, & t_1 + 2 \cdot \frac{t_2 - t_1}{k}, & \dots, & t_1 + k \cdot \frac{t_2 - t_1}{k} = t_2, \\
 & \dots \\
 t_{n-1} + 1 \cdot \frac{t_n - t_{n-1}}{k}, & t_{n-1} + 2 \cdot \frac{t_n - t_{n-1}}{k}, & \dots, & t_{n-1} + k \cdot \frac{t_n - t_{n-1}}{k} = t_n = 1
 \end{aligned}$$

and connecting these points with straight line segments.

- **cubic-spline – Cubic Splines.** If the `chart:spline-resolution` attribute has value `1` this is identical to the `chart:interpolation` attribute value `none`. If the `chart:spline-resolution` attribute has value `k>1` then given a sequence of data points,  $(x_0, y_0), \dots, (x_n, y_n)$  a cubic spline interpolation is constructed as described here:

Find the unique cubic polynomials  $f_1, f_2, \dots, f_n$   $g_1, g_2, \dots, g_n$  and such that:

for all  $i=1, \dots, n$ ,  $f_i(i) = x_i$  and  $f_i(i-1) = x_{i-1}$ ,

for all  $i=1, \dots, (n-1)$ ,  $f'_i(i) = f'_{i+1}(i)$  with  $f'$  denoting the first derivative of  $f$ ,

for all  $i=1, \dots, (n-1)$ ,  $f''_i(i) = f''_{i+1}(i)$  with  $f''$  denoting the second derivative of  $f$ ,

for all  $i=1, \dots, n$ ,  $g_i(i) = y_i$  and  $g_i(i-1) = y_{i-1}$ ,

for all  $i=1, \dots, (n-1)$ ,  $g'_i(i) = g'_{i+1}(i)$  with  $g'$  denoting the first derivative of  $g$ ,

for all  $i=1, \dots, (n-1)$ ,  $g''_i(i) = g''_{i+1}(i)$  with  $g''$  denoting the second derivative of  $g$ ,

and

•  $f''_1(0) = 0$ ,  $f''_n(n) = 0$ ,  $g''_1(0) = 0$  and  $g''_n(n) = 0$  if  $(x_0, y_0) \neq (x_n, y_n)$  (ie. the curve will not be closed) or

•  $f'_1(0) = f'_n(n)$ ,  $g'_1(0) = g'_n(n)$ ,  $f''_1(0) = f''_n(n)$  and  $g''_1(0) = g''_n(n)$  if  $(x_0, y_0) = (x_n, y_n)$  (ie. the curve will be closed).

With these cubic polynomials we define the splines (piecewise polynomial functions)  $F(t) = f_i(t)$  for  $(i-1) \leq t \leq i$  and  $G(t) = g_i(t)$  for  $(i-1) \leq t \leq i$ . (In view of the first and fourth conditions above this yields well defined functions  $F$  and  $G$ .) The interpolation line is then obtained by connecting consecutive points in

$$(F(0), G(0)), \left(F\left(\frac{1}{k}\right), G\left(\frac{1}{k}\right)\right), \left(F\left(\frac{2}{k}\right), G\left(\frac{2}{k}\right)\right), \dots, \left(F\left(\frac{n-1}{k}\right), G\left(\frac{n-1}{k}\right)\right), \left(F\left(\frac{n}{k}\right), G\left(\frac{n}{k}\right)\right)$$

using straight line segments.

- `none` - Straight lines. Consecutive data points are connected by straight line segments.

- **step-center-x** – Step function with a single step in the horizontal center between consecutive data points. Given a sequence of data points,  $(x_0, y_0), \dots, (x_n, y_n)$  then the interpolation line between any two consecutive data points  $(x_i, y_i), (x_{i+1}, y_{i+1})$  is obtained by connecting consecutive points in  $(x_i, y_i), \left(\frac{x_i + x_{i+1}}{2}, y_i\right), \left(\frac{x_i + x_{i+1}}{2}, y_{i+1}\right), (x_{i+1}, y_{i+1})$  using straight line segments.
- **step-center-y** – Step function with steps to the mean of consecutive data points. Given a sequence of data points,  $(x_0, y_0), \dots, (x_n, y_n)$  then the interpolation line between any two consecutive data points  $(x_i, y_i), (x_{i+1}, y_{i+1})$  is obtained by connecting consecutive points in  $(x_i, y_i), \left(x_i, \frac{y_i + y_{i+1}}{2}\right), \left(x_{i+1}, \frac{y_i + y_{i+1}}{2}\right), (x_{i+1}, y_{i+1})$  using straight line segments.
- **step-start** – Step function with a single step at the beginning of each interval between consecutive data points. Given a sequence of data points,  $(x_0, y_0), \dots, (x_n, y_n)$  then the interpolation line between any two consecutive data points  $(x_i, y_i), (x_{i+1}, y_{i+1})$  is obtained by connecting consecutive points in  $(x_i, y_i), (x_i, y_{i+1}), (x_{i+1}, y_{i+1})$  using straight line segments.
- **step-end** – Step function with a single step at the end of each interval between consecutive data points. Given a sequence of data points,  $(x_0, y_0), \dots, (x_n, y_n)$  then the interpolation line between any two consecutive data points  $(x_i, y_i), (x_{i+1}, y_{i+1})$  is obtained by connecting consecutive points in  $(x_i, y_i), (x_{i+1}, y_i), (x_{i+1}, y_{i+1})$  using straight line segments.

This attribute is evaluated for chart styles that are applied to a `<chart:plot-area> 11.5` or `<chart:series> 11.12` element.

The `chart:interpolation` attribute is usable with the following element:  
`<style:chart-properties> 17.22`.

The values of the `chart:interpolation` attribute are `none`, `cubic-spline`, `b-spline`, `step-start`, `step-end`, `step-center-x` or `step-center-y`.

## 20.28 chart:interval-major

The `chart:interval-major` attribute specifies major intervals on an axis 11.9.

If the axis is linear and the value of this attribute is  $x$  with the value of `chart:major-origin` 20.436 being  $w$ , major ticks are placed at  $w + n \cdot x$  for  $n$  being any (*positive*, *negative* or *zero*) integer.

If the axis is logarithmic and the value of this attribute is  $x$  with the value of `chart:major-origin` being  $w$ , major ticks are placed at  $w \cdot (10^{(n \cdot x)})$  for  $n$  being any (*positive*, *negative* or *zero*) integer.

**Note:** if  $x = \log_{10}(y)$ , then  $10^{(n \cdot x)} = y^n$ .

This attribute is evaluated for a chart style that is applied to a `<chart:axis> 11.9` element.

The `chart:interval-major` attribute is usable with the following element:  
`<style:chart-properties> 17.22`.

The `chart:interval-major` attribute has the data type `double` 18.2.

## 20.29 chart:interval-minor-divisor

The `chart:interval-minor-divisor` attribute specifies a divisor for the `chart:interval-major` 20.28 value, the division of which determines the minor interval.

If `chart:minor-logarithmic` 20.437 attribute is `false` the ticks are placed as follows:

If  $a$  and  $b$  are the values of consecutive major ticks (see 20.28) and  $m$  is the value of `chart:interval-minor-divisor` then the minor ticks are placed at locations

corresponding to the values  $a + (b-a)/m \cdot k$  with  $k=1,2,\dots,(m-1)$  So the minor ticks are equally spaced before the logarithmic transformation.

If `chart:minor-logarithmic` attribute is `true` the ticks are placed as follows:

If  $a$  and  $b$  are the values of consecutive major ticks (see 20.28) and  $m$  is the value of `chart:interval-minor-divisor` then the minor ticks are placed at locations

corresponding to the values  $a * ((b/a)^{(k/m)})$  with  $k=1,2,\dots,(m-1)$  So the minor ticks are equally spaced after the logarithmic transformation.

This attribute is evaluated for a chart style that is applied to a `<chart:axis>` 11.9 element.

The `chart:interval-minor-divisor` attribute is usable with the following element:

`<style:chart-properties>` 17.22.

The `chart:interval-minor-divisor` attribute has the data type `positiveInteger` 18.2.

## 20.30 chart:japanese-candle-stick

The `chart:japanese-candle-stick` attribute specifies the display of opening and closing values in a stock chart.

The defined values for the `chart:japanese-candle-stick` attribute are:

- `false`: opening and closing values are displayed as left and right tick-lines
- `true`: opening and closing values are displayed as colored bars, where the color of the bar depends on whether the opening value is larger than the closing value

This attribute is evaluated for chart styles that are applied to a `<chart:plot-area>` 11.5 element.

The `chart:japanese-candle-stick` attribute is usable with the following element:

`<style:chart-properties>` 17.22.

The `chart:japanese-candle-stick` attribute has the data type `boolean` 18.3.3.

## 20.31 chart:label-arrangement

The `chart:label-arrangement` attribute specifies the arrangement of labels on an axis.

The defined values for the `chart:label-arrangement` attribute are:

- `side-by-side`: Labels are not staggered, they are aligned on one line.
- `stagger-even`: Even labels are aligned on the same line as used for side by side arrangement. Counting starts with one, so the first label is odd.
- `stagger-odd`: All odd labels are aligned on the line used for side by side arrangement. Counting starts with one, so the first label is odd.

This attribute is evaluated for a chart style that is applied to a `<chart:axis>` 11.9 element.

The `chart:label-arrangement` attribute is usable with the following element:

`<style:chart-properties>` 17.22.

The values of the `chart:label-arrangement` attribute are `side-by-side`, `stagger-even` or `stagger-odd`.

## 20.32 chart:label-position

The `chart:label-position` attribute specifies where data labels are placed.

The defined values for the `chart:label-position` attribute are:

- `avoid-overlap`: an algorithm should be used that tries to place the labels without overlapping each other.
- `bottom`: the data label is placed below the data point.
- `bottom-left`: the data label is placed to the bottom left of the data point.
- `bottom-right`: the data label is placed to the bottom right of the data point.
- `center`: the data label is centered on the data point.
- `inside`: the data label is placed inside the data point (Pie charts, the label is placed inside the piece on the bisecting line aligned to the outer radius. Bar charts, the label is inside the bar aligned to that end that is given by the data points value. Polar charts, the label is placed on that side of the data point that points to the polar chart's center ).
- `left`: the data label is placed left of the data point.
- `near-origin`: the data label is placed inside the data point on that side that is near to the origin - where the origin is the beginning of the bar in a bar chart or the baseline in an area chart ( for bar charts the label is placed inside the bar like with value inside but aligned to the opposite end of the bar ).
- `outside`: the data label is placed outside the data point (Pie charts, the label is placed outside the circle aligned to the outer radius. Bar charts, the label is outside the bar aligned to that end that is given by the data points value. Polar charts, the label is placed on that side of the data point that points away from the polar charts center).
- `right`: the data label is placed right of the data point.
- `top`: the data label is placed on top of the data point.
- `top-left`: the data label is placed to the top left of the data point.
- `top-right`: the data label is placed to the top right of the data point.

This attribute is evaluated for chart styles that are applied to `<chart:label> 11.15`, `<chart:point> 11.14`, `<chart:series> 11.12` or `<chart:plot-area> 11.5` elements.

The `chart:label-position` attribute is usable with the following element:  
`<style:chart-properties> 17.22`.

The values of the `chart:label-position` attribute are `avoid-overlap`, `center`, `top`, `top-right`, `right`, `bottom-right`, `bottom`, `bottom-left`, `left`, `top-left`, `inside`, `outside` or `near-origin`.

## 20.33 chart:label-position-negative

If the `chart:label-position-negative` attribute is set in addition to a `chart:label-position` 20.32 attribute, it is used for all labels that belong to data points with negative values. Otherwise, the position set in a `chart:label-position` attribute is used for positive and negative values. If a `chart:label-position-negative` attribute is used without a `chart:label-position` attribute it is ignored.

The defined values for the `chart:label-position-negative` attribute are:

- `avoid-overlap`: an algorithm should be used that tries to place the labels without overlapping each other.
- `bottom`: the data label is placed below the data point.
- `bottom-left`: the data label is placed to the bottom left of the data point.
- `bottom-right`: the data label is placed to the bottom right of the data point.
- `center`: the data label is centered on the data point.
- `inside`: the data label is placed inside the data point ( for pie charts the label is placed inside the piece on the bisecting line aligned to the outer radius, for bar charts the label is inside the bar aligned to that end that is given by the data points value, for polar charts the label is placed on that side of the data point that points to the polar charts center ).
- `left`: the data label is placed left of the data point.
- `near-origin`: the data label is placed inside the data point on that side that is near to the origin - where the origin is the beginning of the bar in a bar chart or the base line in an area chart ( for bar charts the label is placed inside the bar like with value inside but aligned to the opposite end of the bar ).
- `outside`: the data label is placed outside the data point ( for pie charts the label is placed outside the circle aligned to the outer radius, for bar charts the label is outside the bar aligned to that end that is given by the data points value, for polar charts the label is placed on that side of the data point that points away from the polar charts center ).
- `right`: the data label is placed right of the data point.
- `top`: the data label is placed on top of the data point.
- `top-left`: the data label is placed to the top left of the data point.
- `top-right`: the data label is placed to the top right of the data point.

The `chart:label-position-negative` attribute is usable with the following element:  
`<style:chart-properties>` 17.22.

The values of the `chart:label-position-negative` attribute are `avoid-overlap`, `center`, `top`, `top-right`, `right`, `bottom-right`, `bottom`, `bottom-left`, `left`, `top-left`, `inside`, `outside` or `near-origin`.

## 20.34 chart:lines (deprecated)

The `chart:lines` attribute specifies whether connecting lines between data points are shown.

**Note:** The display of lines can be specified by the `draw:stroke` 20.167 attribute.

The defined values for the `chart:lines` attribute are:

- `false`: connecting lines between data points are not shown.
- `true`: connecting lines between data points are shown.

The `chart:lines` attribute is usable with the following element: `<style:chart-properties>` 17.22.

The `chart:lines` attribute has the data type `boolean` 18.3.3.

## 20.35 chart:link-data-style-to-source

The `chart:link-data-style-to-source` attribute can only be used in chart documents that are part of a document that provides the data for the chart.

This attribute is evaluated for a chart style that is applied to a `<chart:axis>` 11.9 element.

The defined values for the `chart:link-data-style-to-source` attribute are:

- `false`: number format used for rendering axis labels is defined by the `style:data-style-name` 19.473 attribute.
- `true`: number format used for rendering the axis labels is inherited from the document providing the data.

The `chart:link-data-style-to-source` attribute is usable with the following element: `<style:chart-properties>` 17.22.

The `chart:link-data-style-to-source` attribute has the data type `boolean` 18.3.3.

## 20.36 chart:logarithmic

The `chart:logarithmic` attribute specifies whether logarithmic scaling will be used on an axis. By default, proportional scaling is used.

This attribute is evaluated for a chart style that is applied to a `<chart:axis>` 11.9 element.

The defined values for the `chart:logarithmic` attribute are:

- `false`: proportional scaling is used.
- `true`: logarithmic scaling is used.

The `chart:logarithmic` attribute is usable with the following element: `<style:chart-properties>` 17.22.

The `chart:logarithmic` attribute has the data type `boolean` 18.3.3.

## 20.37 chart:maximum

The `chart:maximum` attribute specifies the maximum value of an axis.

This attribute is evaluated for a chart style that is applied to a `<chart:axis>` 11.9 element.

The `chart:maximum` attribute is usable with the following element: `<style:chart-properties>` 17.22.

The `chart:maximum` attribute has the data type `double` 18.2.

## 20.38 chart:mean-value

The `chart:mean-value` attribute specifies whether to display a line that represents the statistical mean value of all data points of a series.

This attribute is evaluated for chart styles that are applied to a `<chart:plot-area>` 11.5 element or a `<chart:series>` 11.12 element.

The defined values for the `chart:mean-value` attribute are:

- `false`: line representing the statistical mean of all the data points in a series is not displayed.
- `true`: line representing the statistical mean of all the data points in a series is displayed.

The `chart:mean-value` attribute is usable with the following element: `<style:chart-properties>` 17.22.

The `chart:mean-value` attribute has the data type `boolean` 18.3.3.

## 20.39 chart:minimum

The `chart:minimum` attribute specifies the minimum value of an axis.

This attribute is evaluated for a chart style that is applied to a `<chart:axis>` 11.9 element.

The `chart:minimum` attribute is usable with the following element: `<style:chart-properties>` 17.22.

The `chart:minimum` attribute has the data type `double` 18.2.

## 20.40 chart:origin

The `chart:origin` attribute defines the origin of the graphical representation of a data series attached to an axis.

**Note:** For example, the beginning of the bars in a bar chart or the base line in an area chart can be set by the `chart:origin` attribute in the style of the y-axis to which the data series is attached.

If the first x-axis has a `chart:axis-position` 20.6 attribute, that position is taken to indicate the origin of the data points attached to the first y-axis. It overrides the `chart:origin` attribute of the first y-axis.

This attribute is evaluated for a chart style that is applied to a `<chart:axis>` 11.9 element.

The `chart:origin` attribute is usable with the following element: `<style:chart-properties>` 17.22.

The `chart:origin` attribute has the data type `double` 18.2.

## 20.41 chart:overlap

The `chart:overlap` attribute specifies how much bars within the same category in a bar chart overlap. The attribute value is an integer that is interpreted as a percentage relative to the width of a single bar. Negative values specify gaps between bars.

This attribute is evaluated for chart styles that are applied to a `<chart:axis>` 11.9 element when a `chart:dimension` 19.18 attribute set to y.

**Note:** This allows bars to be attached to different axis arranged differently as long as they are grouped per axis (`chart:group-bars-per-axis` 20.24 attribute is `true`).

The `chart:overlap` attribute is usable with the following element: `<style:chart-properties>` 17.22.

The `chart:overlap` attribute has the data type `integer` 18.2.

## 20.42 chart:pie-offset

The `chart:pie-offset` attribute specifies the distance of a segment from the center of the circle in case of circle charts. The offset is given as an integer which is interpreted as a percentage of the radius of the circle.

In the case of ring charts the `chart:pie-offset` attribute specifies an additional distance of a segment from the center of the circle. The distance is given as a percentage of the thickness of the ring.

This attribute is evaluated for chart styles that are applied to a `<chart:series>` 11.14 or `<chart:series>` 11.12 element.

The `chart:pie-offset` attribute is usable with the following element: `<style:chart-properties>` 17.22.

The `chart:pie-offset` attribute has the data type `nonNegativeInteger` 18.2.

## 20.43 chart:percentage

The `chart:percentage` attribute specifies a percentage accumulation of values per category.

This attribute is evaluated for chart styles that are applied to a `<chart:plot-area>` 11.5 element.

The defined values for the `chart:percentage` attribute are:

- `false`: the attribute has no effect
- `true`: values should be accumulated by category. The sum of the values per category is treated as 100 percent and the y-axis displays according to percentages.

The `chart:percentage` attribute is usable with the following element: `<style:chart-properties>` 17.22.

The `chart:percentage` attribute has the data type `boolean` 18.3.3.

## 20.44 chart:regression-force-intercept

The `chart:regression-force-intercept` attribute specifies whether a regression curve of type `linear`, `exponential` or `polynomial` intercepts the y-axis at a certain value.

The defined values for the `chart:regression-force-intercept` attribute are:

- `true`: The regression curve intercepts the y-axis at the value which is defined by the `chart:regression-intercept-value` 20.45 attribute.
- `false`: The interception of regression-curve and y-axis is calculated from the data series.

The default value for this attribute is `false`.

This attribute is only evaluated together with the attribute `chart:regression-type` 20.50 and if that has one of the following values: `linear`, `exponential` or `polynomial`.

The `chart:regression-force-intercept` attribute is usable with the following element: `<style:chart-properties>` 17.22.

The `chart:regression-force-intercept` attribute has the data type `boolean` 18.3.3.

## 20.45 chart:regression-intercept-value

The `chart:regression-intercept-value` attribute specifies where a regression curve intercepts the y-axis if attribute `chart:regression-force-intercept` 20.44 is `true`.

This attribute is only evaluated together with the `chart:regression-force-intercept` attribute.

The default value for this attribute is 0 in case of regression type `linear` or `polynomial`, and the default value is 1 in case of regression type `exponential`.

The `chart:regression-intercept-value` attribute is usable with the following element: `<style:chart-properties>` 17.22.

The `chart:regression-intercept-value` attribute has the data type `double` 18.2.

## 20.46 chart:regression-max-degree

The `chart:regression-max-degree` attribute specifies the maximum degree of a polynomial regression curve. This attribute is only evaluated together with the attribute

chart:regression-type 20.50 and if that has the value polynomial. The value of the chart:regression-max-degree attribute is greater than or equal to 2.

The default value for this attribute is 2.

The chart:regression-max-degree attribute is usable with the following element:  
<style:chart-properties> 17.22.

The chart:regression-max-degree attribute has the data type positiveInteger  
18.2.

## 20.47 chart:regression-moving-type

The chart:regression-moving-type attribute specifies the type of a regression curve of type moving-average. The defined values for the chart:regression-moving-type attribute are:

- prior,
- central,
- averaged-abscissa.

Let  $D_j(x_j|y_j)$  with  $j=1$  to  $N$  denote the points of the data series in the order they occur in the data series, where  $N$  is the total number of data points for that series. Let  $k$  denote the value of the chart:regression-period attribute. Let  $P_i(r_i|s_i)$  with  $i=1$  to  $N-k+1$  denote the points of the regression curve of type moving-average. This regression curve is piecewise linear and continuous through the points  $P_i(r_i|s_i)$  while being linear between  $P_i(r_i|s_i)$  and  $P_{i+1}(r_{i+1}|s_{i+1})$  for any  $1 \leq i \leq N-k$ .

The ordinates  $s_i$  are calculated as  $s_i = \frac{1}{k} \sum_{z=i}^{i+k-1} y_z$ .

Calculation of the associated abscissas  $r_i$  depends on the value of the chart:regression-moving-type attribute. In case it has the value

- prior:  $r_i = x_{i+k-1}$
- central: In case of even value  $k$  it is  $r_i = x_{k/2+i}$  in case of odd value  $k$  it is  $r_i = x_{(k-1)/2+i}$
- averaged-abscissa:  $r_i = \frac{1}{k} \sum_{z=i}^{i+k-1} x_z$

This attribute is only evaluated together with attribute chart:regression-type and if that has the value moving-average.

The default value for this attribute is prior.

The chart:regression-moving-type attribute is usable with the following element:  
<style:chart-properties> 17.22.

The values of the chart:regression-moving-type attribute are prior, central or averaged-abscissa.

## 20.48 chart:regression-name

The chart:regression-name attribute specifies the name of a regression curve.

The default value for this attribute is the empty string.

The `chart:regression-name` attribute is usable with the following element:  
`<style:chart-properties> 17.22.`

The `chart:regression-name` attribute has the data type `string` 18.2.

## 20.49 chart:regression-period

The `chart:regression-period` attribute specifies the number of points to be used to calculate a point of the regression curve.

This attribute is only evaluated together with attribute `chart:regression-type` 20.50 and if that has the value `moving-average`.

The minimum value for this attribute is 2, the default value for this attribute is 2.

The `chart:regression-period` attribute is usable with the following element:  
`<style:chart-properties> 17.22.`

The `chart:regression-period` attribute has the data type `positiveInteger` 18.2.

## 20.50 chart:regression-type

The `chart:regression-type` attribute specifies the regression function for a series. A regression function can be used to approximate the data points in a series.

The defined values for the `chart:regression-type` attribute are:

- `exponential` – Exponential regression – approximate the values of the series using the model:  $y = A \cdot e^{Bx}$ .
- `linear` – Linear regression – approximate the values of the series using the model:  $y = A \cdot x + B$ .
- `logarithmic` – Logarithmic regression – approximate the values of the series using the model:  $y = A \cdot \ln(x) + B$ .
- `moving-average` – Moving average – each point on the regression curve is calculated from a set of  $k$  data points of the series, where  $k$  is given by attribute `chart:regression-period` 20.49 and the calculation method is chosen according to attribute `chart:regression-moving-type` 20.47.
- `none` – no regression curve will be displayed (default value)
- `polynomial` – Polynomial regression – approximate the values of the series using the model  $y = A_0 + A_1 x + A_2 x^2 \dots + A_n x^n$  where  $n$  is given by the `chart:regression-max-degree` 20.46 attribute.
- `power` – Regression with a power function – approximate the values of the series using the model:  $y = A \cdot x^B$ .

This attribute is evaluated for chart styles that are applied to `<chart:regression-curve> 11.18`, `<chart:series> 11.12` or `<chart:plot-area> 11.5` elements.

The `chart:regression-type` attribute is usable with the following element:  
`<style:chart-properties> 17.22.`

The values of the `chart:regression-type` attribute are `none`, `linear`, `logarithmic`, `moving-average`, `exponential`, `power` or `polynomial`.

## 20.51 chart:reverse-direction

The `chart:reverse-direction` attribute specifies whether the direction of an axis follows the Cartesian coordinate system or the reverse. `False` is the default.

In a Cartesian coordinate system the `x`-axis points from left to right and the `y`-axis points from bottom to top. 3D Cartesian coordinate systems are left-handed and polar coordinate systems are counter-clockwise. If `x` and `y` axes are swapped as in horizontal bar charts (when attribute `chart:vertical` 20.72 value is `true`), the unswapped coordinate system follows the Cartesian coordinate system and is left-handed.

This attribute is evaluated for chart styles that are applied to a `<chart:axis>` 11.9 element.

The defined values for the `chart:reverse-direction` attribute are:

- `false`: direction of axis follows the Cartesian coordinate system.
- `true`: direction of axis follows the reverse of the Cartesian coordinate system.

The `chart:reverse-direction` attribute is usable with the following element:  
`<style:chart-properties>` 17.22.

The `chart:reverse-direction` attribute has the data type `boolean` 18.3.3.

## 20.52 chart:right-angled-axes

The `chart:right-angled-axes` attribute specifies a modification of the projection of a 3D chart. In a Cartesian coordinate system, the projections of `x` and `y`-axis are horizontal and vertical.

This attribute is evaluated for chart styles that are applied to a `<chart:plot-area>` 11.5 element. The `chart:right-angled-axes` attribute is ignored unless a `chart:three-dimensional` 20.65 attribute is present and has the value `true`.

The defined values for the `chart:right-angled-axes` attribute are:

- `false`: the attribute has no effect.
- `true`: a 3D chart is sheared instead of rotated. For Cartesian coordinate systems, the `x`- and `y`-axis do not form a right angle in the 2D-projection.

The `chart:right-angled-axes` attribute is usable with the following element:  
`<style:chart-properties>` 17.22.

The `chart:right-angled-axes` attribute has the data type `boolean` 18.3.3.

## 20.53 chart:scale-text

The `chart:scale-text` attribute specifies that the text contained in an object should be scaled whenever the size of the chart changes. This attribute can appear at all chart objects that contain text.

The defined values for the `chart:scale-text` attribute are:

- `false`: text contained in an object should not be scaled when the size of its containing chart changes.
- `true`: text contained in an object should be scaled when the size of its containing chart changes.

The `chart:scale-text` attribute is usable with the following element: `<style:chart-properties>` 17.22.

The `chart:scale-text` attribute has the data type `boolean` 18.3.3.

## 20.54 chart:series-source (deprecated)

The `chart:series-source` attribute specifies whether a tabular data for a chart contains all the data series in columns or rows.

The tabular data can be given by the `<table:table>` 9.1.2 element inside the `<chart:chart>` 11.1 element.

When data is assigned more specifically to a chart element, with a values-cell-range-address attribute on a `<chart:series>` 11.12 element, the more specific data assignment has priority.

The `chart:series-source` attribute is evaluated for a chart style that is applied to a `<chart:plot-area>` 11.5 element.

The defined values for the `chart:series-source` attribute are:

- `columns`: tabular data for a chart contains all the data series in columns.
- `rows`: tabular data for a chart contains all the data series in rows.

The `chart:series-source` attribute is usable with the following element:  
`<style:chart-properties>` 17.22.

The values of the `chart:series-source` attribute are `columns` or `rows`.

## 20.55 chart:sort-by-x-values

The `chart:sort-by-x-values` attribute specifies whether the data points of a data series should be displayed in ascending order of the x values, or in the order they are contained in the underlying data. It is evaluated for styles that are assigned to the `<chart:plot-area>` 11.5 element. The `chart:sort-by-x-values` attribute is used for chart types with data series that contain x-values (for example scatter charts, 19.15).

The defined values for the `chart:sort-by-x-values` attribute are:

- `false`: data points of a data series should be displayed in the order they are contained in the underlying data.
- `true`: data points of a data series should be displayed in ascending order of the x values.

The `chart:sort-by-x-values` attribute is usable with the following element:  
`<style:chart-properties>` 17.22.

The `chart:sort-by-x-values` attribute has the data type `boolean` 18.3.3.

## 20.56 chart:solid-type

The `chart:solid-type` attribute specifies the rendering of bars in three-dimensional bar charts.

The defined values for the `chart:solid-type` attribute are:

- `cone` – bars have the shape of cones
- `cuboid` – bars have the shape of cuboids
- `cylinder` – bars have the shape of cylinders
- `pyramid` – bars have the shape of pyramids

This attribute is evaluated for chart styles that are applied to `<chart:point>` 11.14, `<chart:series>` 11.12 or `<chart:plot-area>` 11.5 elements.

The `chart:solid-type` attribute is usable with the following element: `<style:chart-properties>` 17.22.

The values of the `chart:solid-type` attribute are `cuboid`, `cylinder`, `cone` or `pyramid`.

## 20.57 chart:spline-order

The `chart:spline-order` attribute specifies the degree of the polynomials used as part of the spline.

This attribute should be used together with `chart:interpolation` 20.27 attribute when it has the value `b-spline`.

The `chart:spline-order` attribute is usable with the following element: `<style:chart-properties>` 17.22.

The `chart:spline-order` attribute has the data type `positiveInteger` 18.2.

## 20.58 chart:spline-resolution

The `chart:spline-resolution` attribute specifies the number of straight line segments used between any two data points.

This attribute should be used together with `chart:interpolation` 20.27 attribute when it has a value other than `none`.

The `chart:spline-resolution` attribute is usable with the following element: `<style:chart-properties>` 17.22.

The `chart:spline-resolution` attribute has the data type `positiveInteger` 18.2.

## 20.59 chart:stacked

The `chart:stacked` attribute specifies the accumulation of the series values per category. Each value is in addition to the other values in the same category.

This attribute is evaluated for chart styles that are applied to a `<chart:plot-area>` 11.5 element.

The defined values for the `chart:stacked` attribute are:

- `false`: series values should not be accumulated by category.
- `true`: series values should be accumulated by category. Bars in bar charts are stacked on top of each other.

The `chart:stacked` attribute is usable with the following element: `<style:chart-properties>` 17.22.

The `chart:stacked` attribute has the data type `boolean` 18.3.3.

## 20.60 chart:symbol-height

The `chart:symbol-height` attribute specifies the height of a symbol to be used for a data point in a chart.

This attribute should be used together with `chart:symbol-type` 20.62 attribute when the `chart:symbol-type` attribute has a value other than `none`.

The `chart:symbol-height` attribute is usable with the following element: `<style:chart-properties>` 17.22.

The `chart:symbol-height` attribute has the data type `nonNegativeLength` 18.3.21.

## 20.61 chart:symbol-name

The `chart:symbol-name` attribute specifies a symbol to be used for a data point in a chart.

The glyphs associated with the values of the `chart:symbol-name` attribute are implementation-defined.

This attribute should be used only together with `chart:symbol-type` 20.62 attribute and only when the `chart:symbol-type` 20.62 attribute has the value `named-symbol`.

The `chart:symbol-name` attribute is usable with the following element: `<style:chart-properties>` 17.22.

The values of the `chart:symbol-name` attribute are square, diamond, arrow-down, arrow-up, arrow-right, arrow-left, bow-tie, hourglass, circle, star, x, plus, asterisk, horizontal-bar or vertical-bar.

## 20.62 chart:symbol-type

The `chart:symbol-type` attribute specifies if a symbol is used for a data point in a chart, and if so, which type of symbol is to be used.

The defined values for the `chart:symbol-type` attribute are:

- `automatic`: the consumer chooses from the list of available named symbols (see `chart:symbol-name` 20.61 attribute), choose a different symbol per series in round-robin fashion, in this order: square, diamond, arrow-down, arrow-up, arrow-right, arrow-left, bow-tie, hourglass, circle, star, x, plus, asterisk, horizontal-bar, vertical-bar.
- `named-symbol`: a symbol specified within `chart:symbol-name` attribute should be used.
- `none`: no symbol should be used.
- `image`: an image defined by a `<chart:symbol-image>` 17.23 element is used as symbol.

This attribute is evaluated for chart styles that are applied to `<chart:data-point>` 11.14, `<chart:series>` 11.12 or `<chart:plot-area>` 11.5 elements.

The `chart:symbol-type` attribute is usable with the following element: `<style:chart-properties>` 17.22.

## 20.63 chart:symbol-width

The `chart:symbol-width` attribute specifies the width of a symbol to be used for a data point in a chart.

This attribute should be used together with `chart:symbol-type` 20.62 attribute when the `chart:symbol-type` attribute has a value other than `none`.

The `chart:symbol-width` attribute is usable with the following element: `<style:chart-properties>` 17.22.

The `chart:symbol-width` attribute has the data type `nonNegativeLength` 18.3.21.

## 20.64 chart:text-overlap

The `chart:text-overlap` attribute specifies whether axis labels may overlap each other.

This attribute is evaluated for a chart style that is applied to a `<chart:axis>` 20.62 element.

The defined values for the `chart:text-overlap` attribute are:

- `false`: axis labels shall not overlap each other.
- `true`: axis labels may overlap each other.

The `chart:text-overlap` attribute is usable with the following element: `<style:chart-properties>` 17.22.

The `chart:text-overlap` attribute has the data type `boolean` 18.3.3.

## 20.65 chart:three-dimensional

The `chart:three-dimensional` attribute specifies whether a chart is displayed as a 3D scene.

This attribute is evaluated for chart styles that are applied to a `<chart:plot-area>` 11.5 element.

The defined values for the `chart:three-dimensional` attribute are:

- `false`: chart should not be displayed as a 3D scene.
- `true`: chart should be displayed as a 3D scene.

The `chart:three-dimensional` attribute is usable with the following element: `<style:chart-properties>` 17.22.

The `chart:three-dimensional` attribute has the data type `boolean` 18.3.3.

## 20.66 chart:tick-marks-major-inner

The `chart:tick-marks-major-inner` attribute specifies the existence of major inner tick marks on an axis. They are drawn with respect to the major interval that may be specified by the `chart:interval-major` 20.28 attribute, and are drawn towards the inside of the plot area. That is, they are drawn to the right for an axis displayed on the left hand side of the plot area, and to the left for an axis displayed on the right hand side of the plot area.

If a `chart:tick-marks-major-inner` attribute and a `chart:tick-marks-major-outer` 20.67 attribute are both set to `true`, one tick mark is drawn that crosses the axis.

This attribute is evaluated for a chart style that is applied to a `<chart:axis>` 20.28 element.

The defined values for the `chart:tick-marks-major-inner` attribute are:

- `false`: major inner tick marks should not appear on an axis.
- `true`: major inner tick marks should appear on an axis.

The `chart:tick-marks-major-inner` attribute is usable with the following element: `<style:chart-properties>` 17.22.

The `chart:tick-marks-major-inner` attribute has the data type `boolean` 18.3.3.

## 20.67 chart:tick-marks-major-outer

The `chart:tick-marks-major-outer` attribute specifies the existence of major outer tick marks on an axis. They are drawn with respect to the major interval that may be specified by the `chart:interval-major` 20.28 attribute, and are drawn towards the outside of the plot area. That is, they are drawn to the left for an axis displayed on the left hand side of the plot area, and to the right for an axis displayed on the right hand side of the plot area.

If a `chart:tick-marks-major-inner` 20.66 attribute and a `chart:tick-marks-major-outer` attribute are both set to `true`, one tick mark is drawn that crosses the axis.

This attribute is evaluated for a chart style that is applied to a `<chart:axis>` 11.9 element.

The defined values for the `chart:tick-marks-major-outer` attribute are:

- `false`: major outer tick marks should not appear on an axis.
- `true`: major outer tick marks should appear on an axis.

The `chart:tick-marks-major-outer` attribute is usable with the following element:

`<style:chart-properties> 17.22.`

The `chart:tick-marks-major-outer` attribute has the data type `boolean` 18.3.3.

## 20.68 chart:tick-marks-minor-inner

The `chart:tick-marks-minor-inner` attribute specifies the existence of minor inner tick marks on an axis. They are drawn with respect to the minor interval divisor that may be specified by the `chart:interval-minor-divisor` 20.29 attribute, and are drawn towards the inside of the plot area. That is, they are drawn to the right for an axis displayed on the left hand side of the plot area, and to the left for an axis displayed on the right hand side of the plot area.

If a `chart:tick-marks-minor-inner` attribute and a `chart:tick-marks-minor-outer` 20.69 attribute are both set to `true`, one tick mark is drawn that crosses the axis.

This attribute is evaluated for a chart style that is applied to a `<chart:axis>` 11.9 element.

The defined values for the `chart:tick-marks-minor-inner` attribute are:

- `false`: minor inner tick marks should not appear on an axis.
- `true`: minor inner tick marks should appear on an axis.

The `chart:tick-marks-minor-inner` attribute is usable with the following element:

`<style:chart-properties> 17.22.`

The `chart:tick-marks-minor-inner` attribute has the data type `boolean` 18.3.3.

## 20.69 chart:tick-marks-minor-outer

The `chart:tick-marks-minor-outer` attribute specifies the existence of minor outer tick marks on an axis. They are drawn with respect to the minor interval divisor that may be specified by the `chart:interval-minor-divisor` 20.29 attribute, and are drawn towards the outside of the plot area. That is, they are drawn to the left for an axis displayed on the left hand side of the plot area, and to the right for an axis displayed on the right hand side of the plot area.

If a `chart:tick-marks-minor-inner` 20.68 attribute and a `chart:tick-marks-minor-outer` attribute are set to `true`, one tick mark is drawn that crosses the axis.

This attribute is evaluated for a chart style that is applied to a `<chart:axis>` 11.9 element.

The defined values for the `chart:tick-marks-minor-outer` attribute are:

- `false`: minor outer tick marks should not appear on an axis.
- `true`: minor outer tick marks should appear on an axis.

The `chart:tick-marks-minor-outer` attribute is usable with the following element:

`<style:chart-properties> 17.22.`

The `chart:tick-marks-minor-outer` attribute has the data type `boolean` 18.3.3.

## 20.70 chart:tick-mark-position

The `chart:tick-mark-position` attribute specifies the position of the interval marks.

The defined values for the `chart:tick-mark-position` attribute are:

- `at-labels`: interval marks are placed adjacent to axis labels.
- `at-axis`: interval marks are placed at the axis line.
- `at-labels-and-axis`: interval marks are placed at axis labels and at the axis line.

This attribute makes a difference only if the labels are not placed at the axis line, that is, if the `chart:axis-label-position` 20.5 attribute has the value `outside-end` or `outside-start`.

If the `chart:tick-mark-position` attribute is missing but a `chart:axis-position` attribute is given, `chart:tick-mark-position` is assumed to be `at-labels-and-axis`.

This attribute is evaluated for a chart style that is applied to a `<chart:axis>` element.

The `chart:tick-mark-position` attribute is usable with the following element:  
`<style:chart-properties> 17.22`.

The values of the `chart:tick-mark-position` attribute are `at-labels`, `at-axis` or `at-labels-and-axis`.

## 20.71 chart:treat-empty-cells

The `chart:treat-empty-cells` attribute specifies how missing and invalid values are plotted in a chart.

The defined values for the `chart:treat-empty-cells` attribute are:

- `ignore`: nothing is plotted for a point with missing values and the plot continues.
- `leave-gap`: nothing is plotted for a point with missing values and a continuous plot stops and restarts at the next valid point.
- `use-zero`: missing values are replaced with zero.

The default behavior is `ignore` for scatter charts (`chart:class` 19.15 is `chart:scatter`). For all other chart types the default behavior is `leave-gap`.

This attribute is evaluated for chart styles that are applied to a `<chart:plot-area>` 11.5 element.

The `chart:treat-empty-cells` attribute is usable with the following element:  
`<style:chart-properties> 17.22`.

The values of the `chart:treat-empty-cells` attribute are `use-zero`, `leave-gap` or `ignore`.

## 20.72 chart:vertical

The `chart:vertical` attribute specifies whether the x-axis in a Cartesian coordinate system is oriented horizontally or vertically.

**Note:** This attribute is used to distinguish bar (`chart:vertical="true"`) and column (`chart:vertical="false"`) charts.

This attribute is evaluated for chart styles that are applied to a `<chart:plot-area>` 11.5 element.

The defined values for the `chart:vertical` attribute are:

- `false`: horizontal x-axis, vertical y-axis.
- `true`: vertical x-axis, horizontal y-axis.

The `chart:vertical` attribute is usable with the following element: `<style:chart-properties> 17.22`.

The `chart:vertical` attribute has the data type `boolean` 18.3.3.

## 20.73 chart:visible

The `chart:visible` attribute specifies if an object in a chart is visible or not. By default, objects are visible.

This attribute is evaluated for chart styles that are applied to a `<chart:axis>` 11.9 element or a `<chart:series>` 11.12 element.

The defined values for the `chart:visible` attribute are:

- `false`: object is not visible in chart.
- `true`: object is visible in chart.

The `chart:visible` attribute is usable with the following element: `<style:chart-properties>` 17.22.

The `chart:visible` attribute has the data type `boolean` 18.3.3.

## 20.74 dr3d:ambient-color

See: 19.92

The `dr3d:ambient-color` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The `dr3d:ambient-color` attribute has the data type `color` 18.3.9.

## 20.75 dr3d:back-scale

The `dr3d:back-scale` attribute specifies the proportion of the background geometry for 3D rotation and extrude objects.

The `dr3d:back-scale` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The `dr3d:back-scale` attribute has the data type `percent` 18.3.24.

## 20.76 dr3d:backface-culling

The `dr3d:backface-culling` attribute specifies whether backface culling is enabled.

The defined values for the `dr3d:backface-culling` attribute are:

- `disabled`: backface culling is not enabled.
- `enabled`: backface culling is enabled.

The `dr3d:backface-culling` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The values of the `dr3d:backface-culling` attribute are `enabled` or `disabled`.

## 20.77 dr3d:close-back

The `dr3d:close-back` attribute specifies if a back face is generated. The attribute can be used with 3D extrude and 3D rotation objects. It is only evaluated for geometries that are closed.

The defined values for the `dr3d:close-back` attribute are:

- `false`: a back face should not be generated.
- `true`: a back face should be generated.

The `dr3d:close-back` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The `dr3d:close-back` attribute has the data type `boolean` 18.3.3.

## 20.78 dr3d:close-front

The `dr3d:close-front` attribute specifies if a front face is generated. The attribute can be used with 3D extrude and 3D rotation objects. It is only evaluated for geometries that are closed.

The defined values for the `dr3d:close-front` attribute are:

- `false`: a front face should not be generated.
- `true`: a front face should be generated.

The `dr3d:close-front` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The `dr3d:close-front` attribute has the data type `boolean` 18.3.3.

## 20.79 dr3d:depth

The `dr3d:depth` attribute specifies an extrusion depth for extruded objects.

The `dr3d:depth` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The `dr3d:depth` attribute has the data type `length` 18.3.19.

## 20.80 dr3d:diffuse-color

The `dr3d:diffuse-color` attribute, along with `dr3d:ambient-color` 20.74 20.74, `dr3d:emissive-color` 20.83, and `dr3d:specular-color` 20.91 specifies the four colors that define the color of a material.

The `dr3d:diffuse-color` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The `dr3d:diffuse-color` attribute has the data type `color` 18.3.9.

## 20.81 dr3d:edge-rounding

The `dr3d:edge-rounding` attribute specifies the size of an area at the edges of a geometry that is used for rounding edges, if the geometry of a 3D object is generated during run-time.

The `dr3d:edge-rounding` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The `dr3d:edge-rounding` attribute has the data type `percent` 18.3.24.

## 20.82 dr3d:edge-rounding-mode

The `dr3d:edge-rounding-mode` attribute specifies how to generate rounded edges.

Creating rounded edges in a 3D environment requires an inner and outer representation of the outline on which rounded edges will appear.

The defined values for the `dr3d:edge-rounding-mode` attribute are:

- `attractive`: the edge begins at the inner plane by moving points in the direction of combined perpendicular vectors of both neighboring vectors.

- `correct`: the inner face is linearly scaled smaller relative to its center and the original polygon is used for the outer face.

The `dr3d:edge-rounding-mode` attribute is usable with the following element:

`<style:graphic-properties> 17.21.`

The values of the `dr3d:edge-rounding-mode` attribute are `correct` or `attractive`.

## 20.83 dr3d:emissive-color

The `dr3d:emissive-color` attribute, along with `dr3d:ambient-color` 20.74, `dr3d:diffuse-color` 20.80, and `dr3d:specular-color` 20.91 specifies the four colors that define the color of a material.

The `dr3d:emissive-color` attribute is usable with the following element:

`<style:graphic-properties> 17.21.`

The `dr3d:emissive-color` attribute has the data type `color` 18.3.9.

## 20.84 dr3d:end-angle

The `dr3d:end-angle` attribute specifies the rotation angle for 3D rotation objects. The attribute value is an angle.

A value of 360 degrees defines that the 3D rotation object is closed and completely rotated. This is also the default. Values smaller than 360 degrees define opened 3D rotation objects (segments). Values larger than 360 degrees define 3D rotation objects with more than one rotation.

**Note:** The visible effect of values larger than 360 degrees only differs from that of 360 degrees when the `dr3d:back-scale` 20.75 attribute has a value different than 100%.

The `dr3d:end-angle` attribute is usable with the following element: `<style:graphic-properties> 17.21.`

The `dr3d:end-angle` attribute has the data type `angle` 18.3.1.

## 20.85 dr3d:horizontal-segments

The `dr3d:horizontal-segments` attribute specifies the number of horizontal segments that are used to generate a geometry, if the geometry of a 3D object is generated during run-time.

**Note:** Typical consumers support values between 2 and 256.

The `dr3d:horizontal-segments` attribute is usable with the following element:

`<style:graphic-properties> 17.21.`

The `dr3d:horizontal-segments` attribute has the data type `nonNegativeInteger` 18.2.

## 20.86 dr3d:lighting-mode

The `dr3d:lighting-mode` attribute specifies whether lighting is used for a three-dimensional object.

The defined values for the `dr3d:lighting-mode` attribute are:

- `double-sided`: the front and back sides of all planes are displayed if they are visible from the viewer's perspective.

- **standard:** Depending on the value of the `dr3d:normals-direction` 20.87 attribute, only the front or only the back side sides of the plane are displayed. The opposite sides are not displayed, whether they are visible from the viewer's perspective or not.

**Note:** The `dr3d:lighting-mode` 19.99 attribute is also defined for non-3D object uses.

The `dr3d:lighting-mode` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The values of the `dr3d:lighting-mode` attribute are **standard** or **double-sided**.

## 20.87 dr3d:normals-direction

The `dr3d:normals-direction` attribute specifies which sides of a 3D object's planes are displayed if the `dr3d:lighting-mode` attribute has the value 'standard'.

The defined values for the `dr3d:normals-direction` attribute are:

- **inverse:** only the back sides of planes are displayed.
- **normal:** only the front sides of planes are displayed.

The front side of the plane is the one the normal points away from.

The `dr3d:normals-direction` attribute has no effect if the `dr3d:lighting-mode` 20.86 attribute has the value **double-sided**.

The `dr3d:normals-direction` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The values of the `dr3d:normals-direction` attribute are **normal** or **inverse**.

## 20.88 dr3d:normals-kind

The `dr3d:normals-kind` attribute specifies the normal settings for generated lighting.

The defined values for the `dr3d:normals-kind` attribute are:

- **flat:** one normal per flat part.
- **object:** apply object-specific normals.
- **sphere:** normals behave as the object is a sphere.

The `dr3d:normals-kind` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The values of the `dr3d:normals-kind` attribute are **object**, **flat** or **sphere**.

## 20.89 dr3d:shadow

The `dr3d:shadow` attribute enables or disables a three-dimensional shadow for a three-dimensional object.

The defined values for the `dr3d:shadow` attribute are:

- **hidden:** three-dimensional shadow should not be shown for a three-dimensional object.
- **visible:** three-dimensional shadow should be shown for a three-dimensional object.

The `dr3d:shadow` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The values of the `dr3d:shadow` attribute are **visible** or **hidden**.

## 20.90 dr3d:shininess

The dr3d:shininess attribute specifies the shine of a material.

The dr3d:shininess attribute is usable with the following element: <style:graphic-properties> 17.21.

The dr3d:shininess attribute has the data type percent 18.3.24.

## 20.91 dr3d:specular-color

The dr3d:specular-color attribute, along with dr3d:ambient-color 20.74, dr3d:diffuse-color 20.80, and dr3d:emissive-color 20.80 specifies the four colors that define the color of a material.

The dr3d:specular-color attribute is usable with the following element: <style:graphic-properties> 17.21.

The dr3d:specular-color attribute has the data type color 18.3.9.

## 20.92 dr3d:texture-filter

The dr3d:texture-filter attribute specifies whether texture filtering is enabled.

The defined values for the dr3d:texture-filter attribute are:

- disabled: texture filtering should not be enabled.
- enabled: texture filtering should be enabled.

The dr3d:texture-filter attribute is usable with the following element: <style:graphic-properties> 17.21.

The values of the dr3d:texture-filter attribute are enabled or disabled.

## 20.93 dr3d:texture-kind

The dr3d:texture-kind attribute specifies if texture changes the luminance, intensity, or color of a shape.

The defined values for the dr3d:texture-kind attribute are:

- color: color of a shape.
- intensity: intensity of a shape.
- luminance: luminance of a shape.

The dr3d:texture-kind attribute is usable with the following element: <style:graphic-properties> 17.21.

The values of the dr3d:texture-kind attribute are luminance, intensity or color.

## 20.94 dr3d:texture-mode

The dr3d:texture-mode attribute specifies how the texture is modulated.

The defined values for the dr3d:texture-mode attribute are:

- blend: blends the texture blend color with the object color.
- modulate: modulates the object color with the texture color.
- replace: replaces the object color with the texture color.

The `dr3d:texture-mode` attribute is usable with the following element:

`<style:graphic-properties>` 17.21.

The values of the `dr3d:texture-mode` attribute are `replace`, `modulate` or `blend`.

## 20.95 dr3d:texture-generation-mode-x

The `dr3d:texture-generation-mode-x` attribute, along with the `dr3d:texture-generation-mode-y` 20.96 attribute, specifies how textual coordinates are generated.

The defined values for the `dr3d:texture-generation-mode-x` attribute are:

- `object`: This value specifies that the standard object projection method is used.
- `parallel`: This value specifies a flat parallel projection in the specified degree of freedom (X or Y).
- `sphere`: This value forces projection to wrapping in X and/or Y direction.

The `dr3d:texture-generation-mode-x` attribute is usable with the following element:

`<style:graphic-properties>` 17.21.

The values of the `dr3d:texture-generation-mode-x` attribute are `object`, `parallel` or `sphere`.

## 20.96 dr3d:texture-generation-mode-y

The `dr3d:texture-generation-mode-y` attribute, along with the `dr3d:texture-generation-mode-x` 20.95 attribute, specifies how textual coordinates are generated.

The defined values for the `dr3d:texture-generation-mode-y` attribute are:

- `object`: This value specifies that the standard object projection method is used.
- `parallel`: This value specifies a flat parallel projection in the specified degree of freedom (X or Y).
- `sphere`: This value forces projection to wrapping in X and/or Y direction.

The `dr3d:texture-generation-mode-y` attribute is usable with the following element:

`<style:graphic-properties>` 17.21.

The values of the `dr3d:texture-generation-mode-y` attribute are `object`, `parallel` or `sphere`.

## 20.97 dr3d:vertical-segments

The `dr3d:vertical-segments` attribute specifies the number of vertical segments that are used to generate the geometry, if the geometry of a 3D object is generated during run-time.

**Note:** Typical consumers support values between 2 and 256.

The `dr3d:vertical-segments` attribute is usable with the following element:

`<style:graphic-properties>` 17.21.

The `dr3d:vertical-segments` attribute has the data type `nonNegativeInteger` 18.2.

## 20.98 draw:auto-grow-height

The `draw:auto-grow-height` attribute specifies whether to automatically increase the height of the drawing object if text is added to the drawing object. This attribute is evaluated only for text boxes.

If both `draw:auto-grow-width` 20.99 and `draw:auto-grow-height` are present, a consumer should first grow the size of the drawing object in the dimension of the major text flow (width for horizontal writing, and height for vertical writing). Only after that size component is filled, a consumer should adjust the other dimension to fit the text content.

The defined values for the `draw:auto-grow-height` attribute are:

- `false`: height of a drawing object should not automatically increase if text is added to the drawing object.
- `true`: height of a drawing object should automatically increase if text is added to the drawing object.

The `draw:auto-grow-height` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The `draw:auto-grow-height` attribute has the data type `boolean` 18.3.3.

## 20.99 `draw:auto-grow-width`

The `draw:auto-grow-width` attribute specifies whether to automatically increase the width of the drawing object if text is added to the drawing object. This attribute is evaluated only for text boxes.

The defined values for the `draw:auto-grow-width` attribute are:

- `false`: width of a drawing object should not automatically increase if text is added to the drawing object.
- `true`: width of a drawing object should automatically increase if text is added to the drawing object.

The `draw:auto-grow-width` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The `draw:auto-grow-width` attribute has the data type `boolean` 18.3.3.

## 20.100 `draw:background-size`

The `draw:background-size` attribute specifies whether the background of a page is rendered on a full page or only inside the borders of a page.

The defined values for the `draw:background-size` attribute are:

- `border`: background should be rendered only inside borders of a page.
- `full`: background should be rendered on a full page.

The `draw:background-size` attribute is usable with the following element:  
`<style:drawing-page-properties>` 17.25.

The values of the `draw:background-size` attribute are `full` or `border`.

## 20.101 `draw:blue`

The `draw:blue` attribute specifies together with the attributes `draw:green` 20.136 and `draw:red` 20.156 a non destructive filter for a linear transformation of the white balance of a pixel image.

If any of these three attributes is specified, an offset is applied to each pixel of an image while it is rendered. The offsets for each color channel is given as a percentage in the range of -100% to +100%.

These offsets are scaled to the range  $-2^{\text{bits}}$  to  $2^{\text{bits}}$ , where bits is the number of bits reserved for each color channel within the image. If the resulting value is less than 0 it is set to 0 and if it is greater than the maximum possible value it is set to the maximum.

**Note:** For example, if the `draw:blue` attribute has the value 50% and the blue color channel has 8 bits, then 128 is added to the blue color value of each pixel inside the image before it is rendered. If `draw:blue` has the value -50% then 128 is subtracted.

The `draw:blue` attribute specifies the offset for the blue color channel.

The `draw:blue` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The `draw:blue` attribute has the data type `signedZeroToHundredPercent` 18.3.31.

## 20.102 `draw:caption-angle`

The `draw:caption-angle` attribute specifies the leaving angle of the line of a caption. It is evaluated only if the `draw:caption-angle-type` 20.103 attribute has the value `fixed`.

The `draw:caption-angle` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The `draw:caption-angle` attribute has the data type `angle` 18.3.1.

## 20.103 `draw:caption-angle-type`

The `draw:caption-angle-type` attribute specifies if the leaving angle of the line of a caption is `fixed` or `free`.

The defined values for the `draw:caption-angle-type` attribute are:

- `fixed`: leaving angle of line from caption has been specified by a `draw:caption-angle` 20.102 attribute.
- `free`: consumer can choose the best possible leaving angle of the line from a caption.

The `draw:caption-angle-type` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The values of the `draw:caption-angle-type` attribute are `fixed` or `free`.

## 20.104 `draw:caption-escape`

The `draw:caption-escape` attribute specifies the point where a caption's connector is connected to the caption's text area, measured from the top left corner of the text area. The value can be an absolute length or a percentage.

The `draw:caption-escape` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The values of the `draw:caption-escape` attribute are a value of type `length` 18.3.19 or a value of type `percent` 18.3.24.

## 20.105 `draw:caption-escape-direction`

The `draw:caption-escape-direction` attribute specifies the leaving direction for the line of a caption. If this is set to `auto` the consumer can choose the best direction.

The defined values for the `draw:caption-escape-direction` attribute are:

- `auto`: consumer should choose best leaving direction for a line of a caption.

- **horizontal**: the leaving line of a caption should be drawn horizontally.
- **vertical**: the leaving line of a caption should be drawn vertically.

The `draw:caption-escape-direction` attribute is usable with the following element:  
`<style:graphic-properties> 17.21.`

The values of the `draw:caption-escape-direction` attribute are **horizontal**, **vertical** or **auto**.

## 20.106 `draw:caption-fit-line-length`

The `draw:caption-fit-line-length` attribute enables the consumer to determine the best possible length for a caption line.

The defined values for the `draw:caption-fit-line-length` attribute are:

- **false**: length for caption line is fixed.
- **true**: consumer should determine the best possible length for a caption line.

The `draw:caption-fit-line-length` attribute is usable with the following element:  
`<style:graphic-properties> 17.21.`

The `draw:caption-fit-line-length` attribute has the data type **boolean** 18.3.3.

## 20.107 `draw:caption-gap`

The `draw:caption-gap` attribute specifies the distance between the text area of the caption and the start of the line.

The `draw:caption-gap` attribute is usable with the following element: `<style:graphic-properties> 17.21.`

The `draw:caption-gap` attribute has the data type **distance** 18.3.15.

## 20.108 `draw:caption-line-length`

The `draw:caption-line-length` attribute specifies the length of the first line of a caption. The attribute is only evaluated if `draw:caption-fit-line-length` 20.106 has the value **false**.

The `draw:caption-line-length` attribute is usable with the following element:  
`<style:graphic-properties> 17.21.`

The `draw:caption-line-length` attribute has the data type **length** 18.3.19.

## 20.109 `draw:caption-type`

The `draw:caption-type` attribute specifies the geometry of the line of a caption.

The defined values for the `draw:caption-type` attribute are:

- **angled-connector-line**: a line in leaving direction is drawn, followed by a line to the caption point.
- **angled-line**: a line is drawn to the caption point.
- **straight-line**: a line perpendicular to the leaving direction is drawn to the caption point.

The `draw:caption-type` attribute is usable with the following element:  
`<style:graphic-properties> 17.21.`

The values of the `draw:caption-type` attribute are `straight-line`, `angled-line` or `angled-connector-line`.

## 20.110 `draw:color-inversion`

The `draw:color-inversion` attribute specifies whether the colors in the graphic shape should be inverted.

**Note:** Given a color "#rrggbb", where rr, gg and bb are 8-bit hexadecimal digits, the inverted color is "#RRGGBB" where RR is the 8 bit hexadecimal number obtained as the difference of hexadecimal FF and rr, GG is the 8-bit hexadecimal number obtained as the difference of hexadecimal FF (=255) and gg, and BB is the 8 bit hexadecimal number obtained as the difference of hexadecimal FF (=255) and bb.

The defined values for the `draw:color-inversion` attribute are:

- `false`: colors in a graphic shape should not be inverted.
- `true`: colors in a graphic shape should be inverted.

The `draw:color-inversion` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The `draw:color-inversion` attribute has the data type `boolean` 18.3.3.

## 20.111 `draw:color-mode`

The `draw:color-mode` attribute sets the output of colors from a source bitmap or raster graphic.

The defined values for the `draw:color-mode` attribute are:

- `greyscale`: image is displayed using intensity only.
- `mono`: image is displayed in black and white.
- `standard`: image is displayed without modification by the `draw:color-mode` attribute.
- `watermark`: colors are modified to make the resulting image transparent.

The `draw:color-mode` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The values of the `draw:color-mode` attribute are `greyscale`, `mono`, `watermark` or `standard`.

## 20.112 `draw:contrast`

The `draw:contrast` attribute specifies a signed percentage value that sets the output contrast of a bitmap or raster graphic.

The `draw:contrast` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The `draw:contrast` attribute has the data type `percent` 18.3.24.

## 20.113 `draw:decimal-places`

The `draw:decimal-places` attribute specifies the number of decimal places that are used for the measure text.

The `draw:decimal-places` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The `draw:decimal-places` attribute has the data type `nonNegativeInteger` 18.2.

## 20.114 `draw:draw-aspect`

The `draw:draw-aspect` attribute specifies the display embedded objects.

The defined values for the `draw:draw-aspect` attribute are:

- `content`: Specifies an object is displayed as an embedded object inside a frame.
- `icon`: Specifies an icon to display.
- `print-view`: Specifies an object is displayed as in a print preview.
- `thumbnail`: Specifies a thumbnail of an object is displayed.

The `draw:draw-aspect` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The values of the `draw:draw-aspect` attribute are `content`, `thumbnail`, `icon` or `print-view`.

## 20.115 `draw:end-guide`

The `draw:end-guide` attribute specifies a length that is added to the length of the end extension line. The extension line is extended by this length towards the end reference point.

The `draw:end-guide` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The `draw:end-guide` attribute has the data type `length` 18.3.19.

## 20.116 `draw:end-line-spacing-horizontal`

The `draw:end-line-spacing-horizontal` attribute increments the length of the leaving line from the end shape. For line connectors, it specifies the absolute length of the leaving line from the end shape. For other connector types, it is ignored.

The `draw:end-line-spacing-horizontal` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The `draw:end-line-spacing-horizontal` attribute has the data type `distance` 18.3.15.

## 20.117 `draw:end-line-spacing-vertical`

The `draw:end-line-spacing-vertical` attribute increments the length of the leaving line from the end shape. For line connectors, it specifies the absolute length of the leaving line from the end shape. For other connector types, it is ignored.

The `draw:end-line-spacing-vertical` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The `draw:end-line-spacing-vertical` attribute has the data type `distance` 18.3.15.

## 20.118 `draw:fill`

The `draw:fill` attribute specifies the fill style for an object. Graphic objects that are not closed will not be filled.

**Note:** A path without a closepath at the end is open and will not be filled.

The defined values for the `draw:fill` attribute are:

- **bitmap**: the drawing object is filled with the bitmap specified by the `draw:fill-image-name` 20.124 attribute.
- **gradient**: the drawing object is filled with the gradient specified by the `draw:fill-gradient-name` 20.120 attribute.
- **hatch**: the drawing object is filled with the hatch specified by the `draw:fill-hatch-name` 20.121 attribute.
- **none**: the drawing object is not filled.
- **solid**: the drawing object is filled with the color specified by the `draw:fill-color` 20.119 attribute.

The `draw:fill` attribute is usable with the following elements: `<style:drawing-page-properties>` 17.25 and `<style:graphic-properties>` 17.21.

The values of the `draw:fill` attribute are `none`, `solid`, `bitmap`, `gradient` or `hatch`.

## 20.119 `draw:fill-color`

The `draw:fill-color` attribute specifies the color of the fill for an object. It is used only if the `draw:fill` 20.118 attribute has one of the values `solid` or `hatch`.

The `draw:fill-color` attribute is usable with the following elements: `<style:drawing-page-properties>` 17.25 and `<style:graphic-properties>` 17.21.

The `draw:fill-color` attribute has the data type `color` 18.3.9.

## 20.120 `draw:fill-gradient-name`

The `draw:fill-gradient-name` attribute specifies a gradient style that is used for filling objects. It is used only if the `draw:fill` 20.118 attribute has the value `gradient`.

The `draw:fill-gradient-name` attribute is usable with the following elements: `<style:drawing-page-properties>` 17.25 and `<style:graphic-properties>` 17.21.

The `draw:fill-gradient-name` attribute has the data type `styleNameRef` 18.3.33.

## 20.121 `draw:fill-hatch-name`

The `draw:fill-hatch-name` attribute specifies a hatch style that is used for filling objects. It is used only if the `draw:fill` 20.118 attribute has the value `hatch`.

The `draw:fill-hatch-name` attribute is usable with the following elements: `<style:drawing-page-properties>` 17.25 and `<style:graphic-properties>` 17.21.

The `draw:fill-hatch-name` attribute has the data type `styleNameRef` 18.3.33.

## 20.122 `draw:fill-hatch-solid`

The `draw:fill-hatch-solid` attribute specifies whether the background of a hatch filling is solid or transparent.

The defined values for the `draw:fill-hatch-solid` attribute are:

- `false`: background of a hatch filling should be transparent.
- `true`: background of a hatch filling is filled with the color specified by the `draw:fill-color` 20.119 attribute.

The `draw:fill-hatch-solid` attribute is usable with the following elements: `<style:drawing-page-properties>` 17.25 and `<style:graphic-properties>` 17.21. The `draw:fill-hatch-solid` attribute has the data type `boolean` 18.3.3.

## 20.123 `draw:fill-image-height`

The `draw:fill-image-height` attribute overrides the logical height of a source image that is used for filling. If the value of the `style:repeat` 20.341 attribute is `stretch`, this attribute is ignored.

The `draw:fill-image-height` attribute is usable with the following elements: `<style:drawing-page-properties>` 17.25 and `<style:graphic-properties>` 17.21.

The values of the `draw:fill-image-height` attribute are a value of type `length` 18.3.19 or a value of type `percent` 18.3.24.

## 20.124 `draw:fill-image-name`

The `draw:fill-image-name` attribute specifies a fill image that is used for filling. It is used only if the `draw:fill` 20.118 attribute has the value `bitmap`.

The `draw:fill-image-name` attribute is usable with the following elements: `<style:drawing-page-properties>` 17.25 and `<style:graphic-properties>` 17.21.

The `draw:fill-image-name` attribute has the data type `styleNameRef` 18.3.33.

## 20.125 `draw:fill-image-ref-point`

The `draw:fill-image-ref-point` attribute specifies an alignment of an image in the filling area.

If an alignment (`draw:fill-image-ref-point`) and a movement (`draw:fill-image-ref-point-x` 20.126, `draw:fill-image-ref-point-y` 20.127) are specified at the same time, the image first is aligned and then moved.

This attribute is only interpreted if the value of the current `style:repeat` 20.341 attribute is `repeat`.

The defined values for the `draw:fill-image-ref-point` attribute are:

- `bottom`: image aligned with the bottom of the filling area.
- `bottom-left`: image aligned with the bottom-left of the filling area.
- `bottom-right`: image aligned with the bottom-right of the filling area.
- `center`: image aligned with the center of the filling area.
- `left`: image aligned with the left of the filling area.
- `right`: image aligned with the right of the filling area.
- `top`: image aligned with the top of the filling area.
- `top-left`: image aligned with the top-left of the filling area.
- `top-right`: image aligned with the top-right of the filling area.

The `draw:fill-image-ref-point` attribute is usable with the following elements: `<style:drawing-page-properties>` 17.25 and `<style:graphic-properties>` 17.21.

The values of the `draw:fill-image-ref-point` attribute are `top-left`, `top`, `top-right`, `left`, `center`, `right`, `bottom-left`, `bottom` or `bottom-right`.

## 20.126 `draw:fill-image-ref-point-x`

The `draw:fill-image-ref-point-x` attribute specifies the horizontal movement of an image as a percentage value, where the percentage value is relative to the image width.

If an alignment (`draw:fill-image-ref-point 20.125`) and a movement (`draw:fill-image-ref-point-x`, `draw:fill-image-ref-point-y 20.127`) is specified at the same time, the image first is aligned and then moved.

This attribute is only interpreted if the value of the current `style:repeat 20.341` attribute is `repeat`.

The `draw:fill-image-ref-point-x` attribute is usable with the following elements: `<style:drawing-page-properties> 17.25` and `<style:graphic-properties> 17.21`.

The `draw:fill-image-ref-point-x` attribute has the data type `percent 18.3.24`.

## 20.127 `draw:fill-image-ref-point-y`

The `draw:fill-image-ref-point-y` attribute specifies the vertical movement of an image as a percentage value, where the percentage value is relative to the image width.

If an alignment (`draw:fill-image-ref-point 20.125`) and a movement (`draw:fill-image-ref-point-x 20.126`, `draw:fill-image-ref-point-y`) is specified at the same time, the image first is aligned and then moved.

This attribute is only interpreted if the value of the current `style:repeat 20.341` attribute is `repeat`.

The `draw:fill-image-ref-point-y` attribute is usable with the following elements: `<style:drawing-page-properties> 17.25` and `<style:graphic-properties> 17.21`.

The `draw:fill-image-ref-point-y` attribute has the data type `percent 18.3.24`.

## 20.128 `draw:fill-image-width`

The `draw:fill-image-width` attribute overrides the logical width of a source image that is used for filling. If the value of the `style:repeat 20.341` attribute is `stretch`, this attribute is ignored.

The `draw:fill-image-width` attribute is usable with the following elements: `<style:drawing-page-properties> 17.25` and `<style:graphic-properties> 17.21`.

The values of the `draw:fill-image-width` attribute are a value of type `length 18.3.19` or a value of type `percent 18.3.24`.

## 20.129 `draw:fit-to-contour`

The `draw:fit-to-contour` attribute specifies whether to stretch the text content of a drawing object to fill the contour of an object.

The defined values for the `draw:fit-to-contour` attribute are:

- `false`: text content should not be stretched to fill the contour of an object.
- `true`: text content should be stretched to fill the contour of an object.

The `draw:fit-to-contour` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The `draw:fit-to-contour` attribute has the data type `boolean` 18.3.3.

## 20.130 `draw:fit-to-size`

The `draw:fit-to-size` attribute specifies whether to stretch the text content of a drawing object to fill an entire object.

The defined values for the `draw:fit-to-size` attribute are:

- `false`: text content should not be stretched to fill an entire object.
- `true`: text content should be stretched to fill an entire object.

The `draw:fit-to-size` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The `draw:fit-to-size` attribute has the data type `boolean` 18.3.3.

## 20.131 `draw:frame-display-border`

The `draw:frame-display-border` attribute specifies whether a border is displayed on a floating frame. This attribute can be used with automatic styles only.

The defined values for the `draw:frame-display-border` attribute are:

- `false`: a border should not be displayed on a floating frame.
- `true`: a border should be displayed on a floating frame.

The `draw:frame-display-border` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The `draw:frame-display-border` attribute has the data type `boolean` 18.3.3.

## 20.132 `draw:frame-margin-horizontal`

The `draw:frame-margin-horizontal` attribute specifies the horizontal margin between the border and the content of the floating frame. If this attribute is not specified, the default margin is used. This attribute can be used with automatic styles only. The value of this attribute shall be a length in pixels.

The `draw:frame-margin-horizontal` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The `draw:frame-margin-horizontal` attribute has the data type `nonNegativePixelLength` 18.3.22.

## 20.133 `draw:frame-display-scrollbar`

The `draw:frame-display-scrollbar` attribute specifies whether vertical and horizontal scrollbars are displayed. This attribute can be assigned to automatic styles only.

The defined values for the `draw:frame-display-scrollbar` attribute are:

- `false`: vertical and horizontal scrollbars should not be displayed.
- `true`: vertical and horizontal scrollbars should be displayed.

The `draw:frame-display-scrollbar` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The `draw:frame-display-scrollbar` attribute has the data type `boolean` 18.3.3.

## 20.134 draw:frame-margin-vertical

The `draw:frame-margin-vertical` attribute specifies the vertical margin between the border and the content of the floating frame. If this attribute is not specified, the default margin is used. This attribute can be used with automatic styles only. The value of this attribute shall be a length in pixels.

The `draw:frame-margin-vertical` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The `draw:frame-margin-vertical` attribute has the data type  
`nonNegativePixelLength` 18.3.22.

## 20.135 draw:gamma

The `draw:gamma` attribute specifies a value that sets the output gamma of a bitmap or raster graphic.

The `draw:gamma` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The `draw:gamma` attribute has the data type `percent` 18.3.24.

## 20.136 draw:green

The `draw:green` attribute specifies together with the attributes `draw:blue` 20.101 and `draw:red` 20.156 a non destructive filter for a linear transformation of the white balance of a pixel image.

The `draw:green` attribute specifies the offset for the green color channel.

The `draw:green` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The `draw:green` attribute has the data type `signedZeroToHundredPercent` 18.3.31.

## 20.137 draw:gradient-step-count

The `draw:gradient-step-count` attribute specifies the gradient step count of a color interpolation.

- Value 0 means that the step count is automatically calculated on the size and resolution of the filled area.
- Values 1 and 2 shall not be used.
- Values of 3 or greater mean that the step count is that fixed value.

A gradient step count of color interpolation may be above 256.

The `draw:gradient-step-count` attribute is usable with the following elements:  
`<style:drawing-page-properties>` 17.25 and `<style:graphic-properties>` 17.21.

The `draw:gradient-step-count` attribute has the data type `nonNegativeInteger` 18.2.

## 20.138 draw:guide-distance

The `draw:guide-distance` attribute specifies the distance between the reference points and the start point of extension lines.

**Note:** This distance may be further modified for the start and end extensions lines individually with the `draw:start-guide` 20.164 and `draw:end-guide` 20.115 attributes.

The `draw:guide-distance` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The `draw:guide-distance` attribute has the data type `distance` 18.3.15.

## 20.139 `draw:guide-overhang`

The `draw:guide-overhang` attribute specifies the length of extension lines after their intersection with a dimension line.

The `draw:guide-overhang` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The `draw:guide-overhang` attribute has the data type `length` 18.3.19.

## 20.140 `draw:image-opacity`

The `draw:image-opacity` attribute specifies the opacity of an image.

The `draw:image-opacity` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The `draw:image-opacity` attribute has the data type `zeroToHundredPercent` 18.3.42.

## 20.141 `draw:line-distance`

The `draw:line-distance` attribute specifies the distance between a reference line (as defined by the reference points) and a dimension line.

The `draw:line-distance` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The `draw:line-distance` attribute has the data type `distance` 18.3.15.

## 20.142 `draw:luminance`

The `draw:luminance` attribute specifies a signed percentage value that sets the output luminance of a bitmap or raster graphic.

The `draw:luminance` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The `draw:luminance` attribute has the data type `signedZeroToHundredPercent` 18.3.31.

## 20.143 `draw:marker-end`

The `draw:marker-end` attribute specifies a stroke end marker, which is a path that can be connected to the end of a stroke.

The `draw:marker-end` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The `draw:marker-end` attribute has the data type `styleNameRef` 18.3.33.

## 20.144 draw:marker-end-center

The `draw:marker-end-center` attribute specifies whether an end marker is centered at the end of a stroke.

The defined values for the `draw:marker-end-center` attribute are:

- `false`: an end marker should not be centered at the end of a stroke. It is the default value.
- `true`: an end marker should be centered at the end of a stroke.

Details are described in `<draw:marker>` 16.42.8.

The `draw:marker-end-center` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The `draw:marker-end-center` attribute has the data type `boolean` 18.3.3.

## 20.145 draw:marker-end-width

The `draw:marker-end-width` attribute specifies the width of the marker at the end of a stroke.

The `draw:marker-end-width` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The `draw:marker-end-width` attribute has the data type `length` 18.3.19.

## 20.146 draw:marker-start

The `draw:marker-start` attribute specifies a stroke start marker, which is a path that can be connected to the start of a stroke.

The `draw:marker-start` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The `draw:marker-start` attribute has the data type `styleNameRef` 18.3.33.

## 20.147 draw:marker-start-center

The `draw:marker-start-center` attribute specifies whether a start marker is centered at the start of a stroke.

The defined values for the `draw:marker-start-center` attribute are:

- `false`: a start marker should not be centered at the start of a stroke.
- `true`: a start marker should be centered at the start of a stroke.

Details are described in `<draw:marker>` 16.42.8.

The `draw:marker-start-center` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The `draw:marker-start-center` attribute has the data type `boolean` 18.3.3.

## 20.148 draw:marker-start-width

The `draw:marker-start-width` attribute specifies the width of the marker at the start of a stroke.

The `draw:marker-start-width` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The `draw:marker-start-width` attribute has the data type [length 18.3.19](#).

## 20.149 `draw:measure-align`

The `draw:measure-align` attribute specifies the horizontal alignment of a measure text relative to its measure line.

The defined values for the `draw:measure-align` attribute are:

- `automatic`: horizontal alignment of a measure text relative to its measure line is chosen by the consumer.
- `inside`: horizontal alignment of a measure text should be inside relative to its measure line.
- `left-outside`: horizontal alignment of a measure text should be left-outside relative to its measure line.
- `right-outside`: horizontal alignment of a measure text should be right-outside relative to its measure line.

The `draw:measure-align` attribute is usable with the following element:  
`<style:graphic-properties> 17.21`.

The values of the `draw:measure-align` attribute are `automatic`, `left-outside`, `inside` or `right-outside`.

## 20.150 `draw:measure-vertical-align`

The `draw:measure-vertical-align` attribute specifies the vertical alignment of a measure text relative to its measure line. If the value of this attribute is `automatic`, the consumer chooses the best position.

The defined values for the `draw:measure-vertical-align` attribute are:

- `above`: vertical alignment of a measure text should be above relative to its measure line.
- `automatic`: vertical alignment of a measure text relative to its measure line is chosen by the consumer.
- `below`: vertical alignment of a measure text should be below relative to its measure line.
- `center`: vertical alignment of a measure text should be centered relative to its measure line.

The `draw:measure-vertical-align` attribute is usable with the following element:  
`<style:graphic-properties> 17.21`.

The values of the `draw:measure-vertical-align` attribute are `automatic`, `above`, `below` or `center`.

## 20.151 `draw:ole-draw-aspect` (deprecated)

The `draw:ole-draw-aspect` attribute specifies the display of embedded objects. See [OLE].

The use of this attribute is deprecated. The `draw:draw-aspect` 20.114 attribute should be used instead.

The `draw:ole-draw-aspect` attribute is usable with the following element:  
`<style:graphic-properties> 17.21`.

The `draw:ole-draw-aspect` attribute has the data type `nonNegativeInteger` 18.2.

## 20.152 draw:opacity

The `draw:opacity` attribute specifies the opacity for an image or graphic object. The value is a percentage, where 0% is transparent and 100% is opaque.

The defined value range for the `draw:opacity` attribute is 0% to 100%, inclusive.

Use of the `draw:opacity` attribute disables any transparency effect and set the opacity for the fill area of an object.

The `draw:opacity` attribute is usable with the following elements: `<style:drawing-page-properties>` 17.25 and `<style:graphic-properties>` 17.21.

The `draw:opacity` attribute has the data type `zeroToHundredPercent` 18.3.42.

## 20.153 draw:opacity-name

The `draw:opacity-name` attribute specifies an opacity gradient that defines the opacity for the fill area of an object. When applying an opacity gradient, the opacity is interpolated as defined in the referenced opacity gradient style. Opacity is applied after other fill styles have been applied to an image or graphic object.

The value of this attribute overrides the `draw:opacity` 20.152 attribute.

The `draw:opacity-name` attribute is usable with the following elements: `<style:drawing-page-properties>` 17.25 and `<style:graphic-properties>` 17.21.

The `draw:opacity-name` attribute has the data type `styleNameRef` 18.3.33.

## 20.154 draw:parallel

The `draw:parallel` attribute specifies whether a measure text is displayed parallel to its measure line or perpendicular to it.

The defined values for the `draw:parallel` attribute are:

- `false`: measure text is displayed perpendicular to its measure line.
- `true`: measure text is displayed parallel to its measure line.

The `draw:parallel` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The `draw:parallel` attribute has the data type `boolean` 18.3.3.

## 20.155 draw:placing

The `draw:placing` attribute specifies whether the measure line is rendered below or above the edge defined by the two reference points.

The defined values for the `draw:placing` attribute are:

- `above`: measure line is rendered above the edge defined by two reference points.
- `below`: measure line is rendered below the edge defined by two reference points.

The `draw:placing` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The values of the `draw:placing` attribute are `below` or `above`.

## 20.156 draw:red

The `draw:red` attribute specifies together with the attributes `draw:blue` 20.101 and `draw:green` 20.136 a non destructive filter for a linear transformation of the white balance of a pixel image.

The `draw:red` attribute specifies the offset for the red color channel.

The `draw:red` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The `draw:red` attribute has the data type `signedZeroToHundredPercent` 18.3.31.

## 20.157 draw:secondary-fill-color

The `draw:secondary-fill-color` attribute specifies a secondary fill color. It may be used as fill color for the extrusion.

The `draw:secondary-fill-color` attribute is usable with the following elements: `<style:drawing-page-properties>` 17.25 and `<style:graphic-properties>` 17.21.

The `draw:secondary-fill-color` attribute has the data type `color` 18.3.9.

## 20.158 draw:shadow

The `draw:shadow` attribute enables or disables the visibility of a shadow.

The defined values for the `draw:shadow` attribute are:

- `hidden`: a shadow should not be visible.
- `visible`: a shadow should be visible.

The `draw:shadow` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The values of the `draw:shadow` attribute are `visible` or `hidden`.

## 20.159 draw:shadow-color

The `draw:shadow-color` attribute specifies the color in which a shadow is rendered.

The `draw:shadow-color` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The `draw:shadow-color` attribute has the data type `color` 18.3.9.

## 20.160 draw:shadow-offset-x

The `draw:shadow-offset-x` attribute along with the `draw:shadow-offset-y` 20.161 attribute are used to render a shadow.

A copy of the shape is rendered in the single shadow color (specified by `draw:shadow-color` 20.159) behind the shape. The offset attributes specify the offset between the top left edge of the shape and the top left edge of the border.

The `draw:shadow-offset-x` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The `draw:shadow-offset-x` attribute has the data type `length` 18.3.19.

## 20.161 draw:shadow-offset-y

The `draw:shadow-offset-y` attribute along with the `draw:shadow-offset-x` 20.160 attribute are used to render a shadow.

The `draw:shadow-offset-y` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The `draw:shadow-offset-y` attribute has the data type `length` 18.3.19.

## 20.162 draw:shadow-opacity

The `draw:shadow-opacity` attribute specifies the opacity in which the shadow is rendered.

The `draw:shadow-opacity` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The `draw:shadow-opacity` attribute has the data type `zeroToHundredPercent` 18.3.42.

## 20.163 draw:show-unit

The `draw:show-unit` attribute specifies the visibility of a unit in the textual presentation of a measure shape.

The defined values for the `draw:show-unit` attribute are:

- `false`: units should not be visible in the textual presentation of a measure shape.
- `true`: units should be visible in the textual presentation of a measure shape.

The `draw:show-unit` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The `draw:show-unit` attribute has the data type `boolean` 18.3.3.

## 20.164 draw:start-guide

The `draw:start-guide` attribute specifies a length that is added to the length of a start extension line. The extension line is extended by this length towards the start reference point.

The `draw:start-guide` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The `draw:start-guide` attribute has the data type `length` 18.3.19.

## 20.165 draw:start-line-spacing-horizontal

The `draw:start-line-spacing-horizontal` attribute increments the length of the leaving line from the start shape for standard connectors. For lines connectors, these attributes specify the absolute length of the leaving line from the start shape. For other connector types, they are ignored.

The `draw:start-line-spacing-horizontal` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The `draw:start-line-spacing-horizontal` attribute has the data type `distance` 18.3.15.

## 20.166 draw:start-line-spacing-vertical

The `draw:start-line-spacing-vertical` attribute increments the length of the leaving line from the start shape for standard connectors. For lines connectors, these attributes

specify the absolute length of the leaving line from the start shape. For other connector types, they are ignored.

The `draw:start-line-spacing-vertical` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The `draw:start-line-spacing-vertical` attribute has the data type `distance` 18.3.15.

## 20.167 `draw:stroke`

The `draw:stroke` attribute specifies the style of the stroke on the current object.

The defined values for the `draw:stroke` attribute are:

- `dash`: stroke referenced by a `draw:stroke-dash` 20.168 attribute of a style on the object is drawn.
- `none`: no stroke is drawn.
- `solid`: solid stroke is drawn.

The `draw:stroke` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The values of the `draw:stroke` attribute are `none`, `dash` or `solid`.

## 20.168 `draw:stroke-dash`

The `draw:stroke-dash` attribute specifies the dash style that is used for a stroke.

The `draw:stroke-dash` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The `draw:stroke-dash` attribute has the data type `styleNameRef` 18.3.33.

## 20.169 `draw:stroke-dash-names`

The `draw:stroke-dash-names` attribute specifies a list of dash styles that are used for the stroke in addition to the dash specified by the `draw:stroke-dash` 20.168 attribute.

For each dash style referenced by the attribute a stroke using that dash style is drawn on top of the already drawn strokes.

The `draw:stroke-dash-names` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The `draw:stroke-dash-names` attribute has the data type `styleNameRefs` 18.3.34.

## 20.170 `draw:stroke-linejoin`

The `draw:stroke-linejoin` attribute specifies the shape at the corners of paths or other vector shapes when they are stroked.

The defined values for the `draw:stroke-linejoin` attribute are:

- `bevel`: See §11.4 of [SVG].
- `middle`: mean value between joins is used (deprecated)
- `miter`: See §11.4 of [SVG].
- `none`: no shape specified.
- `round`: See §11.4 of [SVG].

The `draw:stroke-linejoin` attribute is usable with the following element:  
`<style:graphic-properties> 17.21.`

The values of the `draw:stroke-linejoin` attribute are `miter`, `round`, `bevel`, `middle` or `none`.

## 20.171 `svg:stroke-linecap`

The `svg:stroke-linecap` attribute specifies the shape of the end of open subpaths when they are stroked.

For a dashed line, caps are applied to each dash. The values of the `draw:dots1-length` `19.136`, `draw:dots2-length` `19.138` and `draw:distance` `19.134` attributes of the referenced `<draw:stroke-dash> 20.168` element refer to the dashes without cap.

If the referenced `<draw:stroke-dash>` element has an attribute `draw:style` `19.218`, the attribute `draw:style` is ignored.

The defined values for the `svg:stroke-linecap` attribute are:

- `butt`: See §11.4 of [SVG].
- `round`: See §11.4 of [SVG].
- `square`: See §11.4 of [SVG].

The `svg:stroke-linecap` attribute is usable with the following element:  
`<style:graphic-properties> 17.21.`

The values of the `svg:stroke-linecap` attribute are `butt`, `square` or `round`.

## 20.172 `draw:symbol-color`

The `draw:symbol-color` attribute defines the color to be used to draw symbols contained on the drawing object.

The `draw:symbol-color` attribute is usable with the following element:  
`<style:graphic-properties> 17.21.`

The `draw:symbol-color` attribute has the data type `color` 18.3.9.

## 20.173 `draw:textarea-horizontal-align`

The `draw:textarea-horizontal-align` attribute specifies the horizontal alignment of the text area inside a shape.

The defined values for the `draw:textarea-horizontal-align` attribute are:

- `center`: text area is centered horizontally inside a shape.
- `justify`: text area is justified horizontally inside a shape.
- `left`: text area is left aligned horizontally inside a shape.
- `right`: text area is right aligned horizontally inside a shape.

The `draw:textarea-horizontal-align` attribute is usable with the following element:  
`<style:graphic-properties> 17.21.`

The values of the `draw:textarea-horizontal-align` attribute are `left`, `center`, `right` or `justify`.

## 20.174 draw:textarea-vertical-align

The `draw:textarea-vertical-align` attribute specifies the vertical alignment of the text area inside a shape.

The defined values for the `draw:textarea-vertical-align` attribute are:

- `bottom`: vertical alignment of text area inside a shape should start at the bottom of the shape.
- `justify`: vertical alignment of text area inside a shape should be justified.
- `middle`: vertical alignment of text area inside a shape should start at the middle of the shape.
- `top`: vertical alignment of text area inside a shape should start at the top of the shape.

The `draw:textarea-vertical-align` attribute is usable with the following element:  
`<style:graphic-properties> 17.21.`

The values of the `draw:textarea-vertical-align` attribute are `top`, `middle`, `bottom` or `justify`.

## 20.175 draw:tile-repeat-offset

The `draw:tile-repeat-offset` attribute specifies for tiling the translation of each tile in relation to the previous tile. This attribute is only interpreted if the value of the current `style:repeat` 20.341 attribute is `repeat`. The value of this attribute is a percentage value representing the tiles repeat offset relative to the tiles height or width, followed by either the word `horizontal` or `vertical`.

The defined values for the `draw:tile-repeat-offset` attribute are:

- `horizontal`: images are tiled horizontally with regard to the previous tile.
- `vertical`: images are tiled vertically with regard to the previous tile.

The `draw:tile-repeat-offset` attribute is usable with the following elements:  
`<style:drawing-page-properties> 17.25` and `<style:graphic-properties> 17.21.`

The `draw:tile-repeat-offset` attribute has two white space separated values. The first value is of type `zeroToHundredPercent` 18.3.42. The second value is one of: `horizontal` or `vertical`.

## 20.176 draw:visible-area-height

The visible area of an object may be specified by the `draw:visible-area-height` attribute along with the `draw:visible-area-left` 20.177, `draw:visible-area-top` 20.178, `draw:visible-area-width` 20.179 attributes. It is implementation-defined whether these attributes are used to specify the visible area, or whether the producer of the object stores the visible area within the object itself.

When the entire object is visible, the values of the `draw:visible-area-left` and `draw:visible-area-top` attributes are 0 and the `draw:visible-area-width` and `draw:visible-area-height` attributes specify the size of the object. These attributes can be assigned to automatic styles only.

The `draw:visible-area-height` attribute is usable with the following element:  
`<style:graphic-properties> 17.21.`

The `draw:visible-area-height` attribute has the data type `positiveLength` 18.3.27.

## 20.177 draw:visible-area-left

The visible area of an object may be specified by the `draw:visible-area-left` attribute along with the `draw:visible-area-height` 20.176, `draw:visible-area-top` 20.178, `draw:visible-area-width` 20.179 attributes. It is implementation-defined whether these attributes are used to specify the visible area, or whether the producer of the object stores the visible area within the object itself.

When the entire object is visible, the values of the `draw:visible-area-left` and `draw:visible-area-top` attributes are 0 and the `draw:visible-area-width` and `draw:visible-area-height` attributes specify the size of the object. These attributes can be assigned to automatic styles only.

The `draw:visible-area-left` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The `draw:visible-area-left` attribute has the data type `nonNegativeLength` 18.3.21.

## 20.178 draw:visible-area-top

The visible area of an object may be specified by the `draw:visible-area-top` attribute along with the `draw:visible-area-left` 20.177, `draw:visible-area-height` 20.176, `draw:visible-area-width` 20.179 attributes. It is implementation-defined whether these attributes are used to specify the visible area, or whether the producer of the object stores the visible area within the object itself.

When the entire object is visible, the values of the `draw:visible-area-left` and `draw:visible-area-top` attributes are 0 and the `draw:visible-area-width` and `draw:visible-area-height` attributes specify the size of the object. These attributes can be assigned to automatic styles only.

The `draw:visible-area-top` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The `draw:visible-area-top` attribute has the data type `nonNegativeLength` 18.3.21.

## 20.179 draw:visible-area-width

The visible area of an object may be specified by the `draw:visible-area-width` attribute along with the `draw:visible-area-left` 20.177, `draw:visible-area-top` 20.178, `draw:visible-area-height` 20.176 attributes. It is implementation-defined whether these attributes are used to specify the visible area, or whether the producer of the object stores the visible area within the object itself.

When the entire object is visible, the values of the `draw:visible-area-left` and `draw:visible-area-top` attributes are 0 and the `draw:visible-area-width` and `draw:visible-area-height` attributes specify the size of the object. These attributes can be assigned to automatic styles only.

The `draw:visible-area-width` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The `draw:visible-area-width` attribute has the data type `positiveLength` 18.3.27.

## 20.180 draw:unit

The `draw:unit` attribute specifies the unit of measurement used in the textual presentation of a measure shape.

The defined values for the `draw:unit` attribute are:

- `automatic`: unit of measurement chosen by consumer.
- `mm`: millimeter.
- `cm`: centimeter.
- `m`: meter
- `km`: kilometer.
- `pt`: point.
- `pc`: pica.
- `inch`: inch.
- `ft`: foot.
- `mi`: mile.

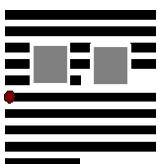
The `draw:unit` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The values of the `draw:unit` attribute are `automatic`, `mm`, `cm`, `m`, `km`, `pt`, `pc`, `inch`, `ft` or `mi`.

## 20.181 `draw:wrap-influence-on-position`

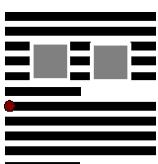
The `draw:wrap-influence-on-position` attribute specifies placement options for two or more frames or graphic objects with `style:wrap` 20.400 properties that influence the placement of the other object.

The situation in which this attribute makes a difference is when the anchor, position and wrapping mode of a frame influence each other. For example, consider a paragraph of text with two images positioned above the anchor. Without wrapping, the images overlie the text and can simply be placed at the given offset from the anchor.



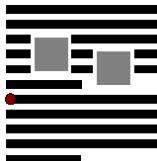
*Figure 7 - Image positions without wrapping*

If wrap-around is enabled, the text hidden behind the images now should flow around the images, making the first paragraph use more space than previously. This moves the anchor position further down. If the placement is only done once and concurrently for all objects, this is the final result. This corresponds to the option `once-concurrent`.



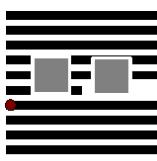
*Figure 8 - Image positions for “once-concurrently”*

If one image is processed at a time, the position shown Figure 10 results. This corresponds to the option `once-successive`.



*Figure 9 - Image positions for “once-successive”*

If the images are placed iteratively, until a position is found which corresponds to the given offset from the anchor, a placement that fully satisfies all the given layout formatting properties can be achieved. This corresponds to the option `iterative`.



*Figure 10 - Image positions for iterative*

The defined values for the `draw:wrap-influence-on-position` attribute are:

- `iterative`: images are placed iteratively until all specified layout formatting properties are satisfied.
- `once-concurrent`: images are aligned with one another.
- `once-successive`: subsequent image is placed after the preceding image.

The `draw:wrap-influence-on-position` attribute is usable with the following element:  
`<style:graphic-properties> 17.21.`

The values of the `draw:wrap-influence-on-position` attribute are `iterative`, `once-concurrent` or `once-successive`.

## 20.182 fo:background-color

The `fo:background-color` attribute specifies a background color for characters, paragraphs, text sections, frames, page bodies, headers, footers, table cells, table rows and tables. This can be transparent or a color. If the value is set to `transparent`, it switches off any background image that is specified by a `<style:background-image> 17.3.`

If a value for a `draw:fill` attribute is provided in a style, any background image that is specified by a `<style:background-image>` element and any background color that is specified with the `fo:background-color` attribute are switched off.

The `fo:background-color` attribute is usable with the following elements:  
`<style:graphic-properties> 17.21, <style:header-footer-properties> 17.5,`  
`<style:page-layout-properties> 17.2, <style:paragraph-properties> 17.6,`  
`<style:section-properties> 17.11, <style:table-cell-properties> 17.18,`  
`<style:table-properties> 17.15, <style:table-row-properties> 17.17 and`  
`<style:text-properties> 16.29.29.`

The values of the `fo:background-color` attribute are transparent or a value of type color 18.3.9.

## 20.183 fo Border Properties

### 20.183.1 General

The `fo:border`, `fo:border-top`, `fo:border-bottom`, `fo:border-left` and `fo:border-right` attributes specify border properties for these elements:

- `<style:graphic-properties>`
- `<style:header-footer-properties>`
- `<style:page-layout-properties>`
- `<style:paragraph-properties>`
- `<style:table-cell-properties>`

### 20.183.2 fo:border

See §7.29.3 of [XSL].

The `fo:border` attribute is usable with the following elements: `<style:graphic-properties>` 17.21, `<style:header-footer-properties>` 17.5, `<style:page-layout-properties>` 17.2, `<style:paragraph-properties>` 17.6 and `<style:table-cell-properties>` 17.18.

The `fo:border` attribute has the data type string 18.2.

### 20.183.3 fo:border-bottom

See §7.29.4 of [XSL].

The `fo:border-bottom` attribute is usable with the following elements: `<style:graphic-properties>` 17.21, `<style:header-footer-properties>` 17.5, `<style:page-layout-properties>` 17.2, `<style:paragraph-properties>` 17.6 and `<style:table-cell-properties>` 17.18.

The `fo:border-bottom` attribute has the data type string 18.2.

### 20.183.4 fo:border-left

See §7.29.6 of [XSL].

The `fo:border-left` attribute is usable with the following elements: `<style:graphic-properties>` 17.21, `<style:header-footer-properties>` 17.5, `<style:page-layout-properties>` 17.2, `<style:paragraph-properties>` 17.6 and `<style:table-cell-properties>` 17.18.

The `fo:border-left` attribute has the data type string 18.2.

### 20.183.5 fo:border-right

See §7.29.7 of [XSL].

The `fo:border-right` attribute is usable with the following elements: `<style:graphic-properties>` 17.21, `<style:header-footer-properties>` 17.5, `<style:page-layout-properties>` 17.2, `<style:paragraph-properties>` 17.6 and `<style:table-cell-properties>` 17.18.

The `fo:border-right` attribute has the data type string 18.2.

## 20.183.6 fo:border-top

See §7.29.10 of [XSL].

The `fo:border-top` attribute is usable with the following elements: `<style:graphic-properties> 17.21`, `<style:header-footer-properties> 17.5`, `<style:page-layout-properties> 17.2`, `<style:paragraph-properties> 17.6` and `<style:table-cell-properties> 17.18`.

The `fo:border-top` attribute has the data type `string` 18.2.

## 20.184 fo:break-after

See §7.19.1 of [XSL].

In the OpenDocument XSL-compatible namespace, the `fo:break-after` attribute does not support the value `inherit`.

The `fo:break-after` attribute is usable with the following elements: `<style:paragraph-properties> 17.6`, `<style:table-column-properties> 17.16`, `<style:table-properties> 17.15` and `<style:table-row-properties> 17.17`.

The values of the `fo:break-after` attribute are `auto`, `column`, `page`, `even-page` or `odd-page`.

## 20.185 fo:break-before

See §7.19.2 of [XSL].

In the OpenDocument XSL-compatible namespace, the `fo:break-before` attribute does not support the value `inherit`.

The `fo:break-before` attribute is usable with the following elements: `<style:paragraph-properties> 17.6`, `<style:table-column-properties> 17.16`, `<style:table-properties> 17.15` and `<style:table-row-properties> 17.17`.

The values of the `fo:break-before` attribute are `auto`, `column`, `page`, `even-page` or `odd-page`.

## 20.186 fo:clip

See §7.20.1 of [XSL].

In the OpenDocument XSL-compatible namespace, the `fo:clip` attribute does not support `em` and `px` values.

The defined value for the `fo:clip` attribute is a value of type `clipShape` 18.3.8.

The `fo:clip` attribute is usable with the following element: `<style:graphic-properties> 17.21`.

The values of the `fo:clip` attribute are `auto` or a value of type `clipShape` 18.3.8.

## 20.187 fo:color

See §7.17.1 of [XSL].

In the OpenDocument XSL-compatible namespace, the `fo:color` attribute does not support the `inherit` value.

The `fo:color` attribute is usable with the following element: `<style:text-properties> 16.29.29`.

The `fo:color` attribute has the data type `color` 18.3.9.

## 20.188 fo:country

See §7.9.1 of [XSL].

The attribute is evaluated for any [UNICODE] characters whose script type is latin. 20.358

It may be ignored if is not specified together with a `fo:language` 20.202 attribute.

In the OpenDocument XSL-compatible namespace, the `fo:country` attribute does not support `none` or `inherit` values.

The `fo:country` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The `fo:country` attribute has the data type `countryCode` 18.3.11.

## 20.189 fo:font-family

See §7.8.2 of [XSL].

The `fo:font-family` attribute is evaluated for any [UNICODE] character whose script type is latin. 20.358

Instead of this attribute, the `style:font-name` 20.277 attribute should be used to specify the properties of a font.

The `fo:font-family` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The `fo:font-family` attribute has the data type `string` 18.2.

## 20.190 fo:font-size

See §7.8.4 of [XSL].

This attribute is evaluated for any [UNICODE] character whose script type is latin. 20.358

The value of this attribute is either an absolute length or a percentage as described in §7.8.4 of [XSL]. In contrast to XSL, percentage values can be used within common styles only and are based on the font height of the parent style rather than to the font height of the attributes neighborhood. Absolute font heights and relative font heights are not supported.

**Note:** The `style:font-size-asian` attribute (20.284) is evaluated for [UNICODE] characters whose type is asian. 20.358 The `style:font-size-complex` attribute (20.285) is evaluated for [UNICODE] characters whose type is complex. 20.358

In the OpenDocument XSL-compatible namespace, the `fo:font-size` attribute does not support `absolute-size`, `inherit` and `relative-size` values.

The `fo:font-size` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The values of the `fo:font-size` attribute are a value of type `positiveLength` 18.3.27 or a value of type `percent` 18.3.24.

## 20.191 fo:font-style

See §7.8.7 of [XSL].

This attribute is evaluated for any [UNICODE] character whose script type is latin. 20.358

In the OpenDocument XSL-compatible namespace, the `fo:font-style` attribute does not support `backslant` and `inherit` values.

The `fo:font-style` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The values of the `fo:font-style` attribute are `normal`, `italic` or `oblique`.

## 20.192 `fo:font-variant`

See §7.8.8 of [XSL].

In the OpenDocument XSL-compatible namespace, the `fo:font-variant` attribute does not support the `inherit` value.

The `fo:font-variant` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The values of the `fo:font-variant` attribute are `normal` or `small-caps`.

## 20.193 `fo:font-weight`

See §7.8.9 of [XSL].

This attribute is evaluated for any [UNICODE] character whose script type is latin. 20.358

In the OpenDocument XSL-compatible namespace, the `fo:font-weight` attribute does not support `bolder`, `inherit` and `lighter` values.

The `fo:font-weight` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The values of the `fo:font-weight` attribute are `normal`, `bold`, `100`, `200`, `300`, `400`, `500`, `600`, `700`, `800` or `900`.

## 20.194 `fo:height`

See §7.14.4 of [XSL].

The `fo:height` attribute is used with `fo:width` 20.229 to set the size of a bullet image.

In the OpenDocument XSL-compatible namespace, the `fo:height` attribute does not support `auto`, `inherit` and `percentage` values.

The `fo:height` attribute is usable with the following element: `<style:list-level-properties>` 17.19.

The `fo:height` attribute has the data type `positiveLength` 18.3.27.

## 20.195 `fo:hyphenate`

See §7.9.4 of [XSL].

In the OpenDocument XSL-compatible namespace, the `fo:hyphenate` attribute does not support the `inherit` value.

The `fo:hyphenate` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The `fo:hyphenate` attribute has the data type `boolean` 18.3.3.

## 20.196 `fo:hyphenation-keep`

See §7.15.1 of [XSL].

The `fo:hyphenation-keep` attribute is usable with the following element: `<style:paragraph-properties>` 17.6.

The values of the `fo:hyphenation-keep` attribute are `auto` or `page`.

## 20.197 `fo:hyphenation-ladder-count`

See §7.15.2 of [XSL].

The defined values for the `fo:hyphenation-ladder-count` attribute are:

- `no-limit`:
- a value of type `positiveInteger`

The `fo:hyphenation-ladder-count` attribute is usable with the following element:  
`<style:paragraph-properties> 17.6.`

The values of the `fo:hyphenation-ladder-count` attribute are `no-limit` or a value of type `positiveInteger` 18.2.

## 20.198 `fo:hyphenation-push-char-count`

See §7.9.6 of [XSL].

The `fo:hyphenation-push-char-count` attribute is usable with the following element:  
`<style:text-properties> 16.29.29.`

The `fo:hyphenation-push-char-count` attribute has the data type `positiveInteger` 18.2.

## 20.199 `fo:hyphenation-remain-char-count`

See §7.9.7 of [XSL].

The `fo:hyphenation-remain-char-count` attribute is usable with the following element:  
`<style:text-properties> 16.29.29.`

The `fo:hyphenation-remain-char-count` attribute has the data type `positiveInteger` 18.2.

## 20.200 `fo:keep-together`

See §7.19.3 of [XSL].

In the OpenDocument XSL-compatible namespace, the `fo:keep-together` attribute does not support the `integer` value.

The `fo:keep-together` attribute is usable with the following elements:  
`<style:paragraph-properties> 17.6` and `<style:table-row-properties> 17.17.`

The values of the `fo:keep-together` attribute are `auto` or `always`.

## 20.201 `fo:keep-with-next`

See §7.19.4 of [XSL].

In the OpenDocument XSL-compatible namespace, the `fo:keep-with-next` attribute does not support the `integer` value.

The `fo:keep-with-next` attribute is usable with the following elements:  
`<style:paragraph-properties> 17.6` and `<style:table-properties> 17.15.`

The values of the `fo:keep-with-next` attribute are `auto` or `always`.

## 20.202 fo:language

See §7.9.2 of [XSL]. Compared to the XSL language attribute, the value range of this attribute is extended to additional parts of ISO 639. See 18.3.18.

This attribute is evaluated for any [UNICODE] character whose script type is latin. 20.358

This attribute may be ignored if it is not specified together with a fo:country 20.188 attribute.

In the OpenDocument XSL-compatible namespace, the fo:language attribute does not support the inherit and none values.

The fo:language attribute is usable with the following element: <style:text-properties> 16.29.29.

The fo:language attribute has the data type languageCode 18.3.18.

## 20.203 fo:letter-spacing

See §7.16.2 of [XSL].

In the OpenDocument XSL-compatible namespace, the fo:letter-spacing attribute does not support the inherit and space values.

The defined value for the fo:letter-spacing attribute is a value of type length 18.3.19.

The fo:letter-spacing attribute is usable with the following element: <style:text-properties> 16.29.29.

The values of the fo:letter-spacing attribute are a value of type length 18.3.19 or normal.

## 20.204 fo:line-height

See §7.15.4 of [XSL].

- The defined values for the fo:line-height attribute are, normal a value of type percent, or a value of type nonNegativeLength.

In the OpenDocument XSL compatible namespace, the fo:line-height attribute does not support the inherit, number, and space values.

In the OpenDocument XSL compatible namespace the values are interpreted as this:

- normal: activates the default line height calculation. The calculation method is not specified, but should be similar to a percentage in the range of 100% to 120%. It may consider font metrics. The value normal disables the effects of style:line-height-at-least 20.317 and style:line-spacing 20.318.
- a value of type percent: The font-size is multiplied with the percent value. In case the line contains content of descendants of the paragraph, for which a different font-size is specified, the maximum of the font-sizes is used. This calculated value is a minimum line-height. If content would vertically overflow the line area, e.g. an image with a larger specified height, the height of the line is increased till the content does not overflow. The vertical position of the baseline is implementation-dependent. In case a style:line-height-at-least 20.317 or a style:line-spacing 20.318 attribute occurs in addition in the same element, the attribute fo:line-height 20.204 has precedence.
- a value of type nonNegativeLength: The value specifies a fixed line-height. The content of the line might vertically overflow the line area. It is implementation-dependent whether the overflow is shown or hidden. In case a style:line-height-at-least 20.317 or a style:line-spacing 20.318 attribute occurs in addition in the same element, the attribute fo:line-height 20.204 has precedence.

**Note:** The effect of this attribute value in the OpenDocument XSL compatible namespace differs from the XSL and the associated CSS specifications.

The `fo:line-height` attribute is usable with the following element: `<style:paragraph-properties> 17.6`.

The values of the `fo:line-height` attribute are `normal`, a value of type `nonNegativeLength 18.3.21` or a value of type `percent 18.3.24`.

## 20.205 fo:margin

See §7.29.14 of [XSL].

In the OpenDocument XSL-compatible namespace, the `fo:margin` attribute does not support `auto` and `inherit` values.

The `fo:margin` attribute is usable with the following elements: `<style:graphic-properties> 17.21`, `<style:header-footer-properties> 17.5`, `<style:page-layout-properties> 17.2`, `<style:paragraph-properties> 17.6` and `<style:table-properties> 17.15`.

The values of the `fo:margin` attribute are a value of type `nonNegativeLength 18.3.21` or a value of type `percent 18.3.24`.

## 20.206 fo:margin-bottom

See §7.10.2 of [XSL].

If this attribute is contained in a `<style:paragraph-properties> 17.6` element, its value may be a percentage that refers to the corresponding margin of a parent style.

In the OpenDocument XSL-compatible namespace, the `fo:margin-bottom` attribute does not support the `auto` and `inherit` values.

The `fo:margin-bottom` attribute is usable with the following elements: `<style:graphic-properties> 17.21`, `<style:header-footer-properties> 17.5`, `<style:page-layout-properties> 17.2`, `<style:paragraph-properties> 17.6` and `<style:table-properties> 17.15`.

The values of the `fo:margin-bottom` attribute are a value of type `nonNegativeLength 18.3.21` or a value of type `percent 18.3.24`.

## 20.207 fo:margin-left

See §7.10.3 of [XSL].

If this attribute is contained in a `<style:paragraph-properties> 17.6` element, its value may be a percentage that refers to the corresponding margin of a parent style.

Tables that align to the left or to the center ignore right margins, and tables that align to the right or to the center ignore left margins.

The `fo:margin-left` attribute is usable with the following elements: `<style:graphic-properties> 17.21`, `<style:header-footer-properties> 17.5`, `<style:page-layout-properties> 17.2`, `<style:paragraph-properties> 17.6`, `<style:section-properties> 17.11` and `<style:table-properties> 17.15`.

The values of the `fo:margin-left` attribute are a value of type `length 18.3.19` or a value of type `percent 18.3.24`.

## 20.208 fo:margin-right

See §7.10.4 of [XSL].

If this attribute is contained in a `<style:paragraph-properties>` 17.6 element, its value may be a percentage that refers to the corresponding margin of a parent style.

Tables that align to the left or to the center ignore right margins, and tables that align to the right or to the center ignore left margins.

The `fo:margin-right` attribute is usable with the following elements: `<style:graphic-properties>` 17.21, `<style:header-footer-properties>` 17.5, `<style:page-layout-properties>` 17.2, `<style:paragraph-properties>` 17.6, `<style:section-properties>` 17.11 and `<style:table-properties>` 17.15.

The values of the `fo:margin-right` attribute are a value of type `length` 18.3.19 or a value of type `percent` 18.3.24.

## 20.209 fo:margin-top

See §7.10.1 of [XSL].

If this attribute is contained in a `<style:paragraph-properties>` 17.6 element, its value may be a percentage that refers to the corresponding margin of a parent style.

In the OpenDocument XSL-compatible namespace, the `fo:margin-top` attribute does not support the `inherit` value.

The `fo:margin-top` attribute is usable with the following elements: `<style:graphic-properties>` 17.21, `<style:header-footer-properties>` 17.5, `<style:page-layout-properties>` 17.2, `<style:paragraph-properties>` 17.6 and `<style:table-properties>` 17.15.

The values of the `fo:margin-top` attribute are a value of type `nonNegativeLength` 18.3.21 or a value of type `percent` 18.3.24.

## 20.210 fo:max-height

The `fo:max-height` attribute specifies a default maximum height for new frames that are created using a `graphics` style. See 19.238.

The `fo:max-height` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The values of the `fo:max-height` attribute are a value of type `length` 18.3.19 or a value of type `percent` 18.3.24.

## 20.211 fo:max-width

The `fo:max-width` attribute specifies a default maximum width for new frames that are created using a `graphics` style. See 19.239.

The `fo:max-width` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The values of the `fo:max-width` attribute are a value of type `length` 18.3.19 or a value of type `percent` 18.3.24.

## 20.212 fo:min-height

### 20.212.1 <style:graphic-properties>

The `fo:min-height` attribute specifies a default minimum height for new frames that are created using a graphics style. See 19.240.

The `fo:min-height` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The values of the `fo:min-height` attribute are a value of type `length` 18.3.19 or a value of type `percent` 18.3.24.

### 20.212.2 <style:header-footer-properties>

The `fo:min-height` attribute specifies a minimum height for a header or footer.

The `fo:min-height` attribute is usable with the following element: `<style:header-footer-properties>` 17.5.

The `fo:min-height` attribute has the data type `length` 18.3.19.

## 20.213 fo:min-width

The `fo:min-width` attribute specifies a default minimum width for new frames that are created using a graphics style. See 19.241.

The `fo:min-width` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The values of the `fo:min-width` attribute are a value of type `length` 18.3.19 or a value of type `percent` 18.3.24.

## 20.214 fo:orphans

See §7.19.6 of [XSL].

The `fo:orphans` attribute is usable with the following element: `<style:paragraph-properties>` 17.6.

The `fo:orphans` attribute has the data type `nonNegativeInteger` 18.2.

## 20.215 fo:page-height

See §7.25.13 of [XSL].

The `fo:page-height` attribute is usable with the following element: `<style:page-layout-properties>` 17.2.

The `fo:page-height` attribute has the data type `length` 18.3.19.

## 20.216 fo:page-width

See §7.25.15 of [XSL].

The `fo:page-width` attribute is usable with the following element: `<style:page-layout-properties>` 17.2.

The `fo:page-width` attribute has the data type `length` 18.3.19.

## 20.217 fo:padding

See §7.29.15 of [XSL].

The `fo:padding` attribute is usable with the following elements: `<style:graphic-properties>` 17.21, `<style:header-footer-properties>` 17.5, `<style:page-layout-properties>` 17.2, `<style:paragraph-properties>` 17.6 and `<style:table-cell-properties>` 17.18.

The `fo:padding` attribute has the data type `nonNegativeLength` 18.3.21.

## 20.218 fo:padding-bottom

See §7.7.36 of [XSL].

The `fo:padding-bottom` attribute is usable with the following elements: `<style:graphic-properties>` 17.21, `<style:header-footer-properties>` 17.5, `<style:page-layout-properties>` 17.2, `<style:paragraph-properties>` 17.6 and `<style:table-cell-properties>` 17.18.

The `fo:padding-bottom` attribute has the data type `nonNegativeLength` 18.3.21.

## 20.219 fo:padding-left

See §7.7.37 of [XSL].

The `fo:padding-left` attribute is usable with the following elements: `<style:graphic-properties>` 17.21, `<style:header-footer-properties>` 17.5, `<style:page-layout-properties>` 17.2, `<style:paragraph-properties>` 17.6 and `<style:table-cell-properties>` 17.18.

The `fo:padding-left` attribute has the data type `nonNegativeLength` 18.3.21.

## 20.220 fo:padding-right

See §7.7.38 of [XSL].

The `fo:padding-right` attribute is usable with the following elements: `<style:graphic-properties>` 17.21, `<style:header-footer-properties>` 17.5, `<style:page-layout-properties>` 17.2, `<style:paragraph-properties>` 17.6 and `<style:table-cell-properties>` 17.18.

The `fo:padding-right` attribute has the data type `nonNegativeLength` 18.3.21.

## 20.221 fo:padding-top

See §7.7.35 of [XSL].

The `fo:padding-top` attribute is usable with the following elements: `<style:graphic-properties>` 17.21, `<style:header-footer-properties>` 17.5, `<style:page-layout-properties>` 17.2, `<style:paragraph-properties>` 17.6 and `<style:table-cell-properties>` 17.18.

The `fo:padding-top` attribute has the data type `nonNegativeLength` 18.3.21.

## 20.222 fo:script

See §7.9.3 of [XSL]. The attribute should be used only if necessary according to the rules of §2.2.3 of [RFC5646], or its successors.

The attribute is evaluated for any [UNICODE] character whose script type is latin. 20.358

It may be ignored if is not specified together with a `fo:language` 20.202 attribute.

In the OpenDocument XSL-compatible namespace, the `fo:script` attribute does not support the `inherit` or `none` values.

The `fo:script` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The `fo:script` attribute has the data type `scriptCode` 18.3.30.

## 20.223 `fo:text-align`

See §7.15.9 of [XSL].

If there are no values specified for the `fo:text-align` and `style:justify-single-word` 20.301 attributes within the same `formatting-properties` element, the values of those attributes is set to `start` and `false` respectively.

In the OpenDocument XSL-compatible namespace, the `fo:text-align` attribute does not support the `inherit`, `inside`, `outside`, or `string` values.

The `fo:text-align` attribute is usable with the following elements: `<style:list-level-properties>` 17.19 and `<style:paragraph-properties>` 17.6.

The values of the `fo:text-align` attribute are `start`, `end`, `left`, `right`, `center` or `justify`.

### 20.223.1 `<style:paragraph-properties>`

The `fo:text-align` attribute when used with the `<style:paragraph-properties>` 17.6 element specifies the alignment of text in a paragraph.

The values of `start` and `end` are interpreted according to the writing direction of the text.

The `fo:text-align` attribute is usable with the following element: `<style:paragraph-properties>` 17.6.

The values of the `fo:text-align` attribute are `start`, `end`, `left`, `right`, `center` or `justify`.

### 20.223.2 `<style:list-level-properties>`

The `fo:text-align` attribute when used with the `<style:list-level-properties>` 17.20 element specifies the alignment of list labels.

If the `text:list-level-position-and-space-mode` 20.431 attribute on the same `<style:list-level-properties>` element has the value `label-width-and-position`, the `fo:text-align` attribute specifies the horizontal alignment of the list label within the width specified by the `text:min-label-width` 20.433 attribute. If the actual width of the list label is greater than the specified minimum width no alignment takes place.

If the `text:list-level-position-and-space-mode` attribute on the same `<style:list-level-properties>` element has the value `label-alignment`, the value of the `text:min-label-width` attribute is treated as 0 and the `fo:text-align` attribute specifies the horizontal alignment of the list label at the alignment position. The alignment position is specified by the `fo:margin-left` 20.207 and `fo:text-indent` 20.225 attributes of its child element `<style:list-level-label-alignment>` 17.20.

The defined values for the `fo:text-align` attribute are:

- `center`: center of the list label is positioned at the alignment position.
- `end`: interpreted as `fo:text-align="right"`

- `justify`: label is justified.
- `left`: list label starts at the alignment position.
- `right`: list label ends at the alignment position.
- `start`: interpreted as `fo:text-align="left"`.

The `fo:text-align` attribute is usable with the following element: `<style:list-level-properties>` 17.19.

The values of the `fo:text-align` attribute are `start`, `end`, `left`, `right`, `center` or `justify`.

## 20.224 fo:text-align-last

See §7.15.10 of [XSL].

This attribute is ignored if it not accompanied by an `fo:text-align` 20.223 attribute.

If no value is specified for this attribute, the value is set to `start`.

The `fo:text-align-last` attribute is usable with the following element: `<style:paragraph-properties>` 17.6.

The values of the `fo:text-align-last` attribute are `start`, `center` or `justify`.

## 20.225 fo:text-indent

The `fo:text-indent` attribute specifies a positive or negative indent for the first line of a paragraph. See §7.15.11 of [XSL]. The attribute value is a length. If the attribute is contained in a common style, the attribute value may be also a percentage that refers to the corresponding text indent of a parent style.

In the OpenDocument XSL-compatible namespace, a negative value of attribute `fo:text-indent` extends the paragraph content area to include the text of the first line of the paragraph.

The `fo:text-indent` attribute is usable with the following element: `<style:paragraph-properties>` 17.6.

The values of the `fo:text-indent` attribute are a value of type `length` 18.3.19 or a value of type `percent` 18.3.24.

## 20.226 fo:text-shadow

The `fo:text-shadow` attribute specifies the text shadow style to use.

The defined values for this attribute are those defined in §7.16.5 of [XSL], except the value `inherit`.

The `fo:text-shadow` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The values of the `fo:text-shadow` attribute are `none` or a value of type `string` 18.2.

## 20.227 fo:text-transform

See §7.16.6 of [XSL].

If `fo:text-transform` and `fo:font-variant` 20.192 attributes are used simultaneously and have different values than `normal` and `none`, the result is undefined.

**Note:** In consumers, the `fo:text-transform` and `fo:font-variant` attributes are mutually exclusive.

The `fo:text-transform` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The values of the `fo:text-transform` attribute are `none`, `lowercase`, `uppercase` or `capitalize`.

## 20.228 `fo:widows`

See §7.19.7 of [XSL].

The `fo:widows` attribute specifies the minimum number of lines that shall be displayed at the top of a page to avoid paragraph widows.

In the OpenDocument XSL-compatible namespace, the `fo:widows` attribute does not support the `inherit` value.

The `fo:widows` attribute is usable with the following element: `<style:paragraph-properties>` 17.6.

The `fo:widows` attribute has the data type `nonNegativeInteger` 18.2.

## 20.229 `fo:width`

See §7.14.12 of [XSL].

The `fo:width` attribute is used with `fo:height` 20.194 to set the size of a bullet image.

In the OpenDocument XSL-compatible namespace, the `fo:width` attribute does not support the `auto` or `inherit` values.

The `fo:width` attribute is usable with the following element: `<style:list-level-properties>` 17.19.

The `fo:width` attribute has the data type `positiveLength` 18.3.27.

## 20.230 `fo:wrap-option`

See §7.15.13 of [XSL].

If wrapping is disabled, it is implementation-defined whether the overflow text is visible or hidden. If the text is hidden consumers may support a scrolling to access the text.

The `fo:wrap-option` attribute is usable with the following elements: `<style:graphic-properties>` 17.21 and `<style:table-cell-properties>` 17.18.

The values of the `fo:wrap-option` attribute are `no-wrap` or `wrap`.

## 20.231 `presentation:background-objects-visible`

The `presentation:background-objects-visible` attribute specifies whether to display objects in the background of a master page when displaying a presentation page.

The defined values for the `presentation:background-objects-visible` attribute are:

- `false`: objects in background of a master page should not be displayed when displaying a presentation page.
- `true`: objects in background of a master page should be displayed when displaying a presentation page.

The `presentation:background-objects-visible` attribute is usable with the following element: `<style:drawing-page-properties>` 17.25.

The `presentation:background-objects-visible` attribute has the data type `boolean` 18.3.3.

## 20.232 presentation:background-visible

The `presentation:background-visible` attribute specifies whether to display the background of a master page when displaying a presentation page.

The defined values for the `presentation:background-visible` attribute are:

- `false`: background of a master page should not be displayed when displaying a presentation page.
- `true`: background of a master page should be displayed when displaying a presentation page.

The `presentation:background-visible` attribute is usable with the following element:  
`<style:drawing-page-properties>` 17.25.

The `presentation:background-visible` attribute has the data type `boolean` 18.3.3.

## 20.233 presentation:display-date-time

The `presentation:display-date-time` attribute specifies the visibility of a drawing shape from a `<style:master-page>` 16.9 element, where the shape has the presentation class `date-time`.

The defined values for the `presentation:display-date-time` attribute are:

- `false`: drawing shape from a `<style:master-page>` element with the presentation class `date-time` should not be visible.
- `true`: drawing shape from a `<style:master-page>` element with the presentation class `date-time` should be visible.

The `presentation:display-date-time` attribute is usable with the following element:  
`<style:drawing-page-properties>` 17.25.

The `presentation:display-date-time` attribute has the data type `boolean` 18.3.3.

## 20.234 presentation:display-footer

The `presentation:display-footer` attribute specifies the visibility of a drawing shape used as a footer in a `<style:master-page>` 16.9 element.

The defined values for the `presentation:display-footer` attribute are:

- `false`: drawing shape used as a footer of a `<style:master-page>` element should not be visible.
- `true`: drawing shape used as a footer of a `<style:master-page>` element should be visible.

The `presentation:display-footer` attribute is usable with the following element:  
`<style:drawing-page-properties>` 17.25.

The `presentation:display-footer` attribute has the data type `boolean` 18.3.3.

## 20.235 presentation:display-header

The `presentation:display-header` attribute specifies the visibility of a drawing shape used as a header in a `<style:master-page>` 16.9 element.

The defined values for the `presentation:display-header` attribute are:

- `false`: drawing shape used as a header of a `<style:master-page>` element should not be visible.

- **true**: drawing shape used as a header of a `<style:master-page>` element should be visible.

The `presentation:display-header` attribute is usable with the following element:  
`<style:drawing-page-properties>` 17.25.

The `presentation:display-header` attribute has the data type `boolean` 18.3.3.

## 20.236 presentation:display-page-number

The `presentation:display-page-number` attribute specifies the visibility of a drawing shape from a `<style:master-page>` 16.9 element, where the shape has the presentation class `page-number`.

The defined values for the `presentation:display-page-number` attribute are:

- **false**: drawing shape from a `<style:master-page>` element with the presentation class `page-number` should not be visible.
- **true**: drawing shape from a `<style:master-page>` element with the presentation class `page-number` should be visible.

Note: Presentation class 19.393 defines presentation class values.

The `presentation:display-page-number` attribute is usable with the following element:  
`<style:drawing-page-properties>` 17.25.

The `presentation:display-page-number` attribute has the data type `boolean` 18.3.3.

## 20.237 presentation:duration

The `presentation:duration` attribute specifies the amount of time that a presentation page is displayed.

The `presentation:duration` attribute is usable with the following element:  
`<style:drawing-page-properties>` 17.25.

The `presentation:duration` attribute has the data type `duration` 18.2.

## 20.238 presentation:transition-speed

The `presentation:transition-speed` attribute specifies the speed at which a presentation page is removed from display, and replaced by a new presentation page.

The defined values for the `presentation:transition-speed` attribute are:

- **fast**: An implementation-defined speed. The `fast` speed should be faster than the speed the consumer chooses for the values `medium` and `slow`.
- **medium**: An implementation-defined speed. The `medium` speed should be faster than the speed the consumer chooses for the value `slow`, and slower than the speed the consumer chooses for the value `fast`.
- **slow**: An implementation-defined speed. The `slow` speed should be slower than the speed the consumer chooses for the values `medium` and `fast`.

The `presentation:transition-speed` attribute is usable with the following element:  
`<style:drawing-page-properties>` 17.25.

The values of the `presentation:transition-speed` attribute are `slow`, `medium` or `fast`.

## 20.239 presentation:transition-style

The `presentation:transition-style` attribute specifies the way that each presentation page replaces the previous presentation page.

The defined values for the `presentation:transition-style` attribute are:

- `clockwise`: page is uncovered by the hand of a watch, moving clockwise.
- `close`: Combination of close-horizontal and close-vertical.
- `close-horizontal`: page is uncovered by drawing it line by line horizontally starting at the edge of the page.
- `close-vertical`: page is uncovered by drawing it line by line vertically starting at the edge of the page.
- `counterclockwise`: page is uncovered by the hand of a watch, moving counterclockwise.
- `dissolve`: page is faded in by drawing small blocks in a random fashion.
- `fade-from-bottom`: page fades from a visible or hidden state to a hidden or visible state to the bottom of the screen.
- `fade-from-left`: page fades from a visible or hidden state to a hidden or visible state to the left of the screen.
- `fade-from-right`: pages fade from a visible or hidden state to a hidden or visible state to the right of the screen.
- `fade-from-top`: page fades from a visible or hidden state to a hidden or visible state to the top of the screen.
- `fade-from-lowerleft`: page fades from a visible or hidden state to a hidden or visible state to the lower left of the screen.
- `fade-from-lowerright`: page fades from a visible or hidden state to a hidden or visible state to the lower right of the screen.
- `fade-from-upperleft`: page fades from a visible or hidden state to a hidden or visible state to the upper left of the screen.
- `fade-from-upperright`: page fades from a visible or hidden state to a hidden or visible state to the upper right of the screen.
- `fade-from-center`: page fades from a visible or hidden state to a hidden or visible state from the center of the screen.
- `fade-to-center`: page fades from a visible or hidden state to a hidden or visible state to the center of the screen.
- `fly-away`: page first reduces itself to a smaller size (while remaining centered in the screen), and then "flies away" (turns around and moves to the bottom-right corner of the screen). The next slide appears under it meanwhile.
- `horizontal-checkerboard`: page is uncovered by drawing checkerboard like blocks that increase in size horizontally.
- `horizontal-lines`: page is uncovered by drawing it line by line horizontally in a random fashion.
- `horizontal-stripes`: page is uncovered by drawing horizontal stripes that change their size during this effect.
- `interlocking-horizontal-left`: page appears in 4 horizontal stripes (the height is divided in 4, like in the horizontal-stripes effect) but those stripes come from left, right, left, and right, and cross each other in the middle of the screen.

- `interlocking-horizontal-right`: page appears in 4 horizontal stripes (the height is divided in 4, like in the horizontal-stripes effect) but those stripes come from right, left, right, and left, and cross each other in the middle of the screen.
- `interlocking-vertical-bottom`: same effect as `interlocking-horizontal-*` but with vertical stripes crossing each other.
- `interlocking-vertical-top`: same effect as `interlocking-horizontal-*` but with vertical stripes crossing each other.
- `melt`: Small vertical stripes move down at random speed, which gives the effect of the current page "melting down".
- `move-from-bottom`: page moves from the bottom of the screen to its final position.
- `move-from-left`: page moves from the left of the screen to its final position.
- `move-from-right`: page moves from the right of the screen to its final position.
- `move-from-top`: page moves from the top of the screen to its final position.
- `move-from-lowerleft`: page moves from the lower left of the screen to its final position.
- `move-from-lowerright`: page moves from the lower right of the screen to its final position.
- `move-from-upperleft`: page moves from the upper left of the screen to its final position.
- `move-from-upperright`: page moves from the upper right of the screen to its final position.
- `none`: no effect is used.
- `open`: Combination of `open-horizontal` and `open-vertical`.
- `open-horizontal`: page is uncovered by drawing it line by line horizontally, starting at the center of the page.
- `open-vertical`: page is uncovered by drawing it line by line vertically, starting at the center of the page.
- `random`: an effect is chosen at random to uncover a page.
- `roll-from-bottom`: page moves towards the bottom of the screen to its final position, pushing the previous page out.
- `roll-from-left`: pages move towards the left of the screen to its final position, pushing the previous page out.
- `roll-from-right`: page moves towards the right of the screen to its final position, pushing the previous page out.
- `roll-from-top`: page moves towards the top of the screen to its final position, pushing the previous page out.
- `spiralin-left`: page is uncovered by drawing blocks in a spiral fashion, starting from the left edge of the screen.
- `spiralin-right`: page is uncovered by drawing blocks in a spiral fashion, starting from the right edge of the screen.
- `spiralout-left`: page is uncovered by drawing blocks in a spiral fashion, starting from the center of the page.
- `spiralout-right`: page is uncovered by drawing blocks in a spiral fashion, starting from the center of the page.

- `stretch-from-bottom`: page is uncovered by changing its size from the bottom of the screen during this effect.
- `stretch-from-left`: page is uncovered by changing its size from the left of the screen during this effect.
- `stretch-from-right`: page is uncovered by changing its size from the right of the screen during this effect.
- `stretch-from-top`: page is uncovered by changing its size from the left of the screen during this effect.
- `uncover-to-bottom`: page is uncovered from the bottom of the screen.
- `uncover-to-left`: page is uncovered from the left of the screen.
- `uncover-to-right`: page is uncovered from the right of the screen.
- `uncover-to-top`: page is uncovered from the top of the screen.
- `uncover-to-lowerleft`: page is uncovered from the lower left of the screen.
- `uncover-to-lowerright`: page is uncovered from the lower right of the screen.
- `uncover-to-upperleft`: page is uncovered from the upper left of the screen.
- `uncover-to-upperright`: page is uncovered from the upper right of the screen.
- `vertical-checkerboard`: page is uncovered by drawing checkerboard like blocks that increase in size vertically.
- `vertical-lines`: page is uncovered by drawing it line by line vertically in a random fashion.
- `vertical-stripes`: page is uncovered by drawing vertical stripes that change their size during this effect.
- `wavyline-from-bottom`: page is uncovered by drawing small blocks in a snake like fashion from the bottom of the screen.
- `wavyline-from-left`: page is uncovered by drawing small blocks in a snake like fashion from the left of the screen.
- `wavyline-from-right`: page is uncovered by drawing small blocks in a snake like fashion from the right of the screen.
- `wavyline-from-top`: page is uncovered by drawing small blocks in a snake like fashion from the top of the screen.

The `presentation:transition-style` attribute is usable with the following element:  
`<style:drawing-page-properties>` 17.25.

The values of the `presentation:transition-style` attribute are `none`, `fade-from-left`, `fade-from-top`, `fade-from-right`, `fade-from-bottom`, `fade-from-upperleft`, `fade-from-upperright`, `fade-from-lowerleft`, `fade-from-lowerright`, `move-from-left`, `move-from-top`, `move-from-right`, `move-from-bottom`, `move-from-upperleft`, `move-from-upperright`, `move-from-lowerleft`, `move-from-lowerright`, `uncover-to-left`, `uncover-to-top`, `uncover-to-right`, `uncover-to-bottom`, `uncover-to-upperleft`, `uncover-to-upperright`, `uncover-to-lowerleft`, `uncover-to-lowerright`, `fade-to-center`, `fade-from-center`, `vertical-stripes`, `horizontal-stripes`, `clockwise`, `counterclockwise`, `open-vertical`, `open-horizontal`, `close-vertical`, `close-horizontal`, `wavyline-from-left`, `wavyline-from-top`, `wavyline-from-right`, `wavyline-from-bottom`, `spiralin-left`, `spiralin-right`, `spiralout-left`, `spiralout-right`, `roll-from-top`, `roll-from-left`, `roll-from-right`, `roll-from-bottom`, `stretch-from-left`, `stretch-from-top`, `stretch-from-right`, `stretch-from-bottom`,

```
vertical-lines, horizontal-lines, dissolve, random, vertical-checkerboard, horizontal-checkerboard, interlocking-horizontal-left, interlocking-horizontal-right, interlocking-vertical-top, interlocking-vertical-bottom, fly-away, open, close or melt.
```

## 20.240 presentation:transition-type

The `presentation:transition-type` attribute specifies the mode of a transition.

The defined values for the `presentation:transition-type` attribute are:

- `automatic`: slide transition and shape effects start automatically.
- `manual`: slide transition and shape effects are started separately by the user.
- `semi-automatic`: slide transition starts automatically, shape effects are started by the user.

The `smil:type` attribute should be used for presentations. The `presentation:transition-type` and `presentation:transition-style` 20.239 attributes are retained for legacy compatibility only.

The `presentation:transition-type` attribute is usable with the following element:  
`<style:drawing-page-properties>` 17.25.

The values of the `presentation:transition-type` attribute are `manual`, `automatic` or `semi-automatic`.

## 20.241 presentation:visibility

The `presentation:visibility` attribute specifies if the content of a `<draw:page>` element is visible during a presentation.

The defined values for the `presentation:visibility` attribute are:

- `hidden`: content is hidden during a presentation.
- `visible`: content is visible during a presentation.

The `presentation:visibility` attribute is usable with the following element:  
`<style:drawing-page-properties>` 17.25.

The values of the `presentation:visibility` attribute are `visible` or `hidden`.

## 20.242 smil:direction

See §12.4.1 of [SMIL20].

The `smil:direction` attribute is usable with the following element: `<style:drawing-page-properties>` 17.25.

The values of the `smil:direction` attribute are `forward` or `reverse`.

## 20.243 smil:fadeColor

See §12.4.1 of [SMIL20].

The `smil:fadeColor` attribute is usable with the following element: `<style:drawing-page-properties>` 17.25.

The `smil:fadeColor` attribute has the data type `color` 18.3.9.

## 20.244 smil:subtype

See §12.4.1 of [SMIL20].

**Note:** See §12.8 of [SMIL20] for a list of supported subtypes.

The `smil:subtype` attribute is usable with the following element: `<style:drawing-page-properties>` 17.25.

The `smil:subtype` attribute has the data type `string` 18.2.

## 20.245 smil:type

The [SMIL20] `smil:type` attribute specifies a transition type or family.

See §12.8 of [SMIL20] for a list of supported types.

If this attribute is present, the attributes `presentation:transition-type` 20.240 and `presentation:transition-style` 20.239 attributes should be ignored.

The `smil:type` attribute is usable with the following element: `<style:drawing-page-properties>` 17.25.

The `smil:type` attribute has the data type `string` 18.2.

## 20.246 style:auto-text-indent

The `style:auto-text-indent` attribute specifies that the first line of a paragraph is indented by a value that is based on the current font size.

If this attribute has a value of `true` and is used together with a `fo:text-indent` 20.225 attribute the `fo:text-indent` attribute is ignored.

The `style:auto-text-indent` attribute is usable with the following element: `<style:paragraph-properties>` 17.6.

The `style:auto-text-indent` attribute has the data type `boolean` 18.3.3.

## 20.247 style:background-transparency

The `style:background-transparency` attribute specifies the transparency of a paragraph's background color.

The `style:background-transparency` attribute is usable with the following elements: `<style:graphic-properties>` 17.21 and `<style:paragraph-properties>` 17.6.

The `style:background-transparency` attribute has the data type `zeroToOnePercent` 18.3.42.

## 20.248 style:border-line-width

The `style:border-line-width` attribute specifies the widths of borders defined by the FO border properties (see 20.183) for borders where the value of these properties is `double`.

The value of the `style:border-line-width` attribute is a list of three white space-separated lengths, as follows:

- The first value specifies the width of the inner line
- The second value specifies the distance between the two lines
- The third value specifies the width of the outer line

The `style:border-line-width` attribute is usable with the following elements: `<style:graphic-properties>` 17.21, `<style:header-footer-properties>` 17.5,

```
<style:page-layout-properties> 17.2, <style:paragraph-properties> 17.6  
and <style:table-cell-properties> 17.18.
```

The values of the `style:border-line-width` attribute are three white space separated values of type `positiveLength` 18.3.27.

## 20.249 `style:border-line-width-bottom`

The `style:border-line-width-bottom` attribute specifies the widths of the bottom border for borders defined by the FO border properties (see 20.183) if the property for the bottom border has the value `double`.

The value of the `style:border-line-width-bottom` attribute is a list of three white space-separated lengths, as follows:

- The first value specifies the width of the inner line
- The second value specifies the distance between the two lines
- The third value specifies the width of the outer line

The `style:border-line-width-bottom` attribute is usable with the following elements: `<style:graphic-properties> 17.21, <style:header-footer-properties> 17.5, <style:page-layout-properties> 17.2, <style:paragraph-properties> 17.6 and <style:table-cell-properties> 17.18.`

The values of the `style:border-line-width-bottom` attribute are three white space separated values of type `positiveLength` 18.3.27.

## 20.250 `style:border-line-width-left`

The `style:border-line-width-left` attribute specifies the widths of the left border for borders defined by the FO border properties (see 20.183) if the property for the left border has the value `double`.

The value of the `style:border-line-width-left` attribute is a list of three white space-separated lengths, as follows:

- The first value specifies the width of the inner line
- The second value specified the distance between the two lines
- The third value specifies the width of the outer line

The `style:border-line-width-left` attribute is usable with the following elements: `<style:graphic-properties> 17.21, <style:header-footer-properties> 17.5, <style:page-layout-properties> 17.2, <style:paragraph-properties> 17.6 and <style:table-cell-properties> 17.18.`

The values of the `style:border-line-width-left` attribute are three white space separated values of type `positiveLength` 18.3.27.

## 20.251 `style:border-line-width-right`

The `style:border-line-width-right` attribute specifies the widths of the right border for borders defined by the FO border properties (see 20.183) if the property for the right border has the value `double`.

The value of the `style:border-line-width-right` attribute is a list of three white space-separated lengths, as follows:

- The first value specifies the width of the inner line
- The second value specified the distance between the two lines

- The third value specifies the width of the outer line

The `style:border-line-width-right` attribute is usable with the following elements: `<style:graphic-properties> 17.21`, `<style:header-footer-properties> 17.5`, `<style:page-layout-properties> 17.2`, `<style:paragraph-properties> 17.6` and `<style:table-cell-properties> 17.18`.

The values of the `style:border-line-width-right` attribute are three white space separated values of type `positiveLength` 18.3.27.

## 20.252 `style:border-line-width-top`

The `style:border-line-width-top` attribute specifies the widths of the top border for borders defined by the FO border properties (see 20.183) if the property for the top border has the value `double`.

The value of the `style:border-line-width-top` attribute is a list of three white space-separated lengths, as follows:

- The first value specifies the width of the inner line
- The second value specified the distance between the two lines
- The third value specifies the width of the outer line

The `style:border-line-width-top` attribute is usable with the following elements: `<style:graphic-properties> 17.21`, `<style:header-footer-properties> 17.5`, `<style:page-layout-properties> 17.2`, `<style:paragraph-properties> 17.6` and `<style:table-cell-properties> 17.18`.

The values of the `style:border-line-width-top` attribute are three white space separated values of type `positiveLength` 18.3.27.

## 20.253 `style:cell-protect`

The `style:cell-protect` attribute specifies how a cell is protected.

This attribute is only evaluated if the current table is protected.

The defined values for the `style:cell-protect` attribute are:

- `formula-hidden`: if cell content is a formula, it is not displayed. It can be replaced by changing the cell content.

**Note:** Replacement of cell content includes replacement with another formula or other cell content.

- `hidden-and-protected`: cell content is not displayed and cannot be edited. If content is a formula, the formula result is not displayed.
- `none`: formula responsible for cell content is neither hidden nor protected.
- `protected`: cell content cannot be edited.
- `protected formula-hidden`: cell content cannot be edited. If content is a formula, it is not displayed. A formula result is displayed.

The `style:cell-protect` attribute is usable with the following element: `<style:table-cell-properties> 17.18`.

The values of the `style:cell-protect` attribute are `none`, `hidden-and-protected`, or white space separated non-empty lists of one of these values: `protected`, or `formula-hidden`.

## 20.254 style:column-width

The `style:column-width` attribute specifies a fixed width for a column.

The `style:column-width` attribute is usable with the following element: `<style:table-column-properties>` 17.16.

The `style:column-width` attribute has the data type `positiveLength` 18.3.27.

## 20.255 style:contextual-spacing

The `fo:margin-bottom` 20.206 attribute of a paragraph and the `fo:margin-top` 20.209 attribute of the next paragraph are ignored, so that the space between the paragraphs is zero, if all of the following conditions hold:

- The `style:contextual-spacing` attribute of both paragraphs has the value `true`.
- The paragraphs belong to the same content area.
- The `text:style-name` 19.880 attribute of the paragraphs refer to the same common paragraph style. In case a `text:style-name` attribute refers to an automatic style, the value of the `style:parent-style-name` 19.510 attribute of the automatic style is taken for the style comparison. If a paragraph has a conditional style, the value of its `text:cond-style-name` 19.781 attribute is taken for the style comparison.

The default value for this attribute is `false`.

The `style:contextual-spacing` attribute is usable with the following element: `<style:paragraph-properties>` 17.6.

The `style:contextual-spacing` attribute has the data type `boolean` 18.3.3.

## 20.256 style:country-asian

The `style:country-asian` attribute specifies the country of a text.

It is evaluated for [UNICODE] characters whose script type is `asian`. 20.358

It may be ignored if it is not specified together with a `style:language-asian` 20.302 attribute.

The `style:country-asian` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The `style:country-asian` attribute has the data type `countryCode` 18.3.11.

## 20.257 style:country-complex

The `style:country-complex` attribute specifies the country of a text. See §7.9.1 of [XSL].

It is evaluated for [UNICODE] characters whose script type is `complex`. 20.358

It may be ignored if it is not specified together with a `style:language-complex` 20.303 attribute.

The `style:country-complex` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The `style:country-complex` attribute has the data type `countryCode` 18.3.11.

## 20.258 style:decimal-places

The `style:decimal-places` attribute specifies the maximum number of decimal places that are displayed if numbers are formatted by a data style that has no setting for number of decimal places itself.

This attribute is only evaluated if it is contained in a default style.

The `style:decimal-places` attribute is usable with the following element:  
`<style:table-cell-properties>` 17.18.

The `style:decimal-places` attribute has the data type `nonNegativeInteger` 18.2.

## 20.259 `style:diagonal-bl-tr`

The `style:diagonal-bl-tr` attribute specifies the style of border to use for a bottom-left to top-right diagonal in a spreadsheet cell.

The `style:diagonal-bl-tr` attribute is usable with the following element:  
`<style:table-cell-properties>` 17.18.

The `style:diagonal-bl-tr` attribute has the data type `string` 18.2.

## 20.260 `style:diagonal-bl-tr-widths`

The `style:diagonal-bl-tr-widths` attribute specifies the width between a double line border to use for a bottom-left to top-right diagonal in a spreadsheet cell.

The `style:diagonal-bl-tr-widths` attribute is usable with the following element:  
`<style:table-cell-properties>` 17.18.

The values of the `style:diagonal-bl-tr-widths` attribute are three white space separated values of type `positiveLength` 18.3.27.

## 20.261 `style:diagonal-tl-br`

The `style:diagonal-tl-br` attribute specifies the style of border to use for a left-top to bottom-right diagonal in a spreadsheet cell.

The `style:diagonal-tl-br` attribute is usable with the following element:  
`<style:table-cell-properties>` 17.18.

The `style:diagonal-tl-br` attribute has the data type `string` 18.2.

## 20.262 `style:diagonal-tl-br-widths`

The `style:diagonal-tl-br-widths` attribute specifies the width between a double line border to use for a top-left to bottom-right diagonal in a spreadsheet cell.

The `style:diagonal-tl-br-widths` attribute is usable with the following element:  
`<style:table-cell-properties>` 17.18.

The values of the `style:diagonal-tl-br-widths` attribute are three white space separated values of type `positiveLength` 18.3.27.

## 20.263 `style:direction`

The `style:direction` attribute specifies the direction of characters.

The `style:direction` attribute modifies the direction of text rendering as specified by a `style:writing-mode` attribute. 20.404

The defined values for the `style:direction` attribute are:

- `ltr` – left to right, text is rendered in the direction specified by the `style:writing-mode` attribute
- `ttb` – top to bottom, characters are vertically stacked but not rotated

The `style:direction` attribute is usable with the following elements: `<style:chart-properties>` 17.22 and `<style:table-cell-properties>` 17.18.

The values of the `style:direction` attribute are `ltr` or `ttb`.

## 20.264 style:dynamic-spacing

The `style:dynamic-spacing` attribute specifies whether the header or footer grows into the space between the page body and the header or footer before the height of the page body becomes smaller.

The defined values for the `style:dynamic-spacing` attribute are:

- `false`: header or footers do not grow into the space between the header and footer and the page body.
- `true`: header or footers first grow into the space between the header and footer and the page body.

The `style:dynamic-spacing` attribute is usable with the following element: `<style:header-footer-properties>` 17.5.

The `style:dynamic-spacing` attribute has the data type `boolean` 18.3.3.

## 20.265 style:editable

The `style:editable` attribute specifies if a text section or box can be edited within a read-only document.

The defined values for the `style:editable` attribute are:

- `false`: text section or box in read-only document should not be editable.
- `true`: text section or box in read-only document should be editable.

The `style:editable` attribute is usable with the following elements: `<style:graphic-properties>` 17.21 and `<style:section-properties>` 17.11.

The `style:editable` attribute has the data type `boolean` 18.3.3.

## 20.266 style:first-page-number

The `style:first-page-number` attribute specifies the number of a document.

The value of this attribute can be an integer or `continue`. If the value is `continue`, the page number is the preceding page number incremented by 1. The default first page number is 1.

The `style:first-page-number` attribute is usable with the following element: `<style:page-layout-properties>` 17.2.

The values of the `style:first-page-number` attribute are a value of type `integer` 18.2 or `continue`.

## 20.267 style:flow-with-text

The `style:flow-with-text` attribute specifies whether a drawing shape flows with the text of its layout environment or not. The layout environment of a drawing shape is determined by the location of its anchor. The following layout environments are defined:

- endnote area - anchor is inside an endnote.
- footnote area - anchor is inside a footnote.
- frame area - anchor is inside a frame or is a frame.
- page footer area - anchor is inside the page footer.

- page header area - anchor is inside the page header.
- page text area - anchor is inside body text.
- table cell area - anchor is inside a table cell.

The innermost enclosing area of the anchor defines the layout environment of the drawing shape.

If the value of the attribute is `true`, the following conditions shall hold:

- The drawing shape flows with the text of its layout environment.
- The drawing shape does not leave its layout environment in the either text flow direction.

When a drawing shape leaves its layout environment due to its position and/or size in the forward text flow direction, it flows with the text flow to the next layout environment. If there is no next layout environment, the position of the layout environment is adjusted. If the drawing shape is larger than its layout environment, the size of the layout environment is adjusted.

If due to its position a drawing shape would leave its layout environment in the backward text flow direction, the position is adjusted to the beginning of its layout environment.

A drawing object may leave its layout environment in the other (non text flow) directions. Exceptions from this rule are the layout environments table cell area and frame area. For these layout environments it shall be ensured that the drawing shape is laid out inside its layout environment by adjusting the position and (if necessary) the size.

If the value of the attribute is `false`, the drawing shape does not have to flow with the text of its layout environment. It can be laid out anywhere on the page its anchor is on.

This attribute is not applicable to drawing shapes that are anchored with `text:anchor-type="as-char"` 20.418, because such drawing shapes are treated as characters. For drawing shapes, which are anchored to a specific page with `text:anchor-type="page"`, the attribute is also not applicable, because such drawing shapes are to be laid out on the page they are anchored to.

The `style:flow-with-text` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The `style:flow-with-text` attribute has the data type `boolean` 18.3.3.

## 20.268 style:font-charset

The `style:font-charset` attribute specifies whether a font defines glyphs according to the semantics of [UNICODE] or not.

The value of this attributes can be `x-symbol` or a character encoding in the notation described in the §4.3.3 of [XML1.0]. If the value is `x-symbol`, the font does not define glyphs according to the semantics of [UNICODE]. If the value is one of the encodings or transformations of [UNICODE], the font does define glyphs according to the semantics of [UNICODE]. The use of other values is deprecated.

**Note:** Fonts for which the attribute has the value `x-symbol` may define glyphs for code points in the private use area of [UNICODE]

This attribute is evaluated for any [UNICODE] character whose script type is `latin`. 20.358

This attribute is ignored if there is no `fo:font-family` attribute attached to the same formatting property element.

Instead of this attribute, the `style:font-name` attribute should be used to specify the properties of a font.

The `style:font-charset` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The `style:font-charset` attribute has the data type `textEncoding` 18.3.36.

## 20.269 `style:font-charset-asian`

The `style:font-charset-asian` attribute specifies whether a font defines glyphs according to the semantics of [UNICODE] or not.

The value of this attributes can be `x-symbol` or a character encoding in the notation described in the §4.3.3 of [XML1.0]. If the value is `x-symbol`, the font does not define glyphs according to the semantics of [UNICODE]. If the value is one of the encodings or transformations of [UNICODE], the font does define glyphs according to the semantics of [UNICODE]. The use of other values is deprecated.

**Note:** Fonts for which the attribute has the value `x-symbol` may define glyphs for code points in the private use area of [UNICODE]

This attribute is evaluated for [UNICODE] characters whose script type is `asian`. 20.358

This attribute is ignored if there is no `style:font-family-asian` 20.271 attribute attached to the same formatting property element.

Instead of this attribute, the `style:font-name-asian` 20.278 attribute should be used to specify the properties of a font.

The `style:font-charset-asian` attribute is usable with the following element:  
`<style:text-properties>` 16.29.29.

The `style:font-charset-asian` attribute has the data type `textEncoding` 18.3.36.

## 20.270 `style:font-charset-complex`

The `style:font-charset-complex` attribute specifies whether a font defines glyphs according to the semantics of [UNICODE] or not.

The value of this attributes can be `x-symbol` or a character encoding in the notation described in the §4.3.3 of [XML1.0]. If the value is `x-symbol`, the font does not define glyphs according to the semantics of [UNICODE]. If the value is one of the encodings or transformations of [UNICODE], the font does define glyphs according to the semantics of [UNICODE]. The use of other values is deprecated.

**Note:** Fonts for which the attribute has the value `x-symbol` may define glyphs for code points in the private use area of [UNICODE]

The `style:font-charset-complex` attribute is evaluated for [UNICODE] characters whose script type is `complex`. 20.358

This attribute is ignored if there is no `style:font-family-complex` 20.272 attribute attached to the same formatting properties element.

Instead of this attribute, the `style:font-name-complex` 20.279 attribute should be used to specify the properties of a font.

The `style:font-charset-complex` attribute is usable with the following element:  
`<style:text-properties>` 16.29.29.

The `style:font-charset-complex` attribute has the data type `textEncoding` 18.3.36.

## 20.271 `style:font-family-asian`

The `style:font-family-asian` attribute specifies the font family for a text. See §7.8.2 of [XSL].

This is evaluated for [UNICODE] characters whose script type is `asian`. 20.285

Instead of this attribute, the `style:font-name-asian` 20.278 attribute should be used to specify the properties of a font.

The `style:font-family-asian` attribute is usable with the following element:  
`<style:text-properties>` 16.29.29.

The `style:font-family-asian` attribute has the data type `string` 18.2.

## 20.272 `style:font-family-complex`

The `style:font-family-complex` attribute specifies the font family for a text. See §7.8.2 of [XSL].

This attribute is evaluated for [UNICODE] characters whose script type is `complex`. 20.358

Instead of this attribute, the `style:font-name-complex` 20.279 attribute should be used to specify the properties of a font.

The `style:font-family-complex` attribute is usable with the following element:  
`<style:text-properties>` 16.29.29.

The `style:font-family-complex` attribute has the data type `string` 18.2.

## 20.273 `style:font-family-generic`

The `style:font-family-generic` attribute specifies a generic font family name.

This attribute is evaluated for any [UNICODE] character whose script type is `latin`. 20.358

This attribute is ignored if there is no corresponding `fo:font-family` 20.189 attribute attached to the same formatting property element.

Instead of this attribute, the `style:font-name` 20.277 attribute should be used to specify the properties of a font.

The defined values for the `style:font-family-generic` attribute are:

- `decorative`: the family of decorative fonts.
- `modern`: the family of modern fonts.
- `roman`: the family roman fonts (with serifs).
- `script`: the family of script fonts.
- `swiss`: the family roman fonts (without serifs).
- `system`: the family system fonts.

The `style:font-family-generic` attribute is usable with the following element:  
`<style:text-properties>` 16.29.29.

The values of the `style:font-family-generic` attribute are `roman`, `swiss`, `modern`, `decorative`, `script` or `system`.

## 20.274 `style:font-family-generic-asian`

The `style:font-family-generic-asian` attribute specifies a generic Asian font family name.

The `style:font-family-generic-asian` attribute is evaluated for [UNICODE] characters whose script type is `asian`. 20.285

This attribute is ignored if there is no corresponding `fo:font-family` 20.189 attribute attached to the same formatting property element.

Instead of this attribute, the `style:font-name-asian` 20.278 attribute should be used to specify the properties of a font.

The defined values for the `style:font-family-generic-asian` attribute are:

- `decorative`: the family of decorative fonts.
- `modern`: the family of modern fonts.
- `roman`: the family roman fonts (with serifs).
- `script`: the family of script fonts.
- `swiss`: the family roman fonts (without serifs).
- `system`: the family system fonts.

The `style:font-family-generic-asian` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The values of the `style:font-family-generic-asian` attribute are `roman`, `swiss`, `modern`, `decorative`, `script` or `system`.

## 20.275 style:font-family-generic-complex

The `style:font-family-generic-complex` attribute specifies a generic font family name.

This attribute is ignored if there is no corresponding `fo:font-family` 20.189 attribute attached to the same formatting property element.

Instead of this attribute, the `style:font-name-complex` 20.279 attribute should be used to specify the properties of a font.

The defined values for the `style:font-family-generic-complex` attribute are:

- `decorative`: the family of decorative fonts.
- `modern`: the family of modern fonts.
- `roman`: the family roman fonts (with serifs).
- `script`: the family of script fonts.
- `swiss`: the family roman fonts (without serifs).
- `system`: the family system fonts.

The `style:font-family-generic-complex` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The values of the `style:font-family-generic-complex` attribute are `roman`, `swiss`, `modern`, `decorative`, `script` or `system`.

## 20.276 style:font-independent-line-spacing

The `style:font-independent-line-spacing` attribute specifies if font independent line spacing is used.

The defined values for the `style:font-independent-line-spacing` attribute are:

- `false`: font metric of the font is taken into account.
- `true`: line height is calculated only from the font height as specified by the font size attributes `fo:font-size` 20.190, `style:font-size-asian` 20.284 and `style:font-size-complex` 20.285.

The `style:font-independent-line-spacing` attribute is usable with the following element: `<style:paragraph-properties>` 17.6.

The `style:font-independent-line-spacing` attribute has the data type `boolean` 18.3.3.

## 20.277 `style:font-name`

The `style:font-name` attribute specifies a font that is declared by a `<style:font-face>` 16.23 element with a `style:name` 19.502 attribute whose name is the same as that of the `style:font-name` attribute value.

This attribute is evaluated for any [UNICODE] character whose script type is `latin`. 20.358

The `style:font-name` attribute is usable with the following elements: `<style:list-level-properties>` 17.19 and `<style:text-properties>` 16.29.29.

The `style:font-name` attribute has the data type `string` 18.2.

## 20.278 `style:font-name-asian`

The `style:font-name-asian` attribute specifies a font that is declared by a `<style:font-face>` 16.23 element with a `style:name` 19.502 attribute whose name is the same as that of the `style:font-name-asian` attribute value.

This attribute is evaluated for [UNICODE] characters whose script type is `asian`. 20.285

The `style:font-name-asian` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The `style:font-name-asian` attribute has the data type `string` 18.2.

## 20.279 `style:font-name-complex`

The `style:font-name-complex` attribute specifies a font that is declared by a `<style:font-face>` 16.23 element with a `style:name` 19.502 attribute whose name is the same as that of the `style:font-name-complex` attribute value.

The `style:font-name-complex` attribute is evaluated for [UNICODE] characters whose script type is `complex`. 20.358

The `style:font-name-complex` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The `style:font-name-complex` attribute has the data type `string` 18.2.

## 20.280 `style:font-pitch`

The `style:font-pitch` attribute specifies whether a font has a fixed or variable width.

This attribute is evaluated for any [UNICODE] character whose script type is `latin`. 20.358

This attribute is ignored if there is no corresponding `fo:font-family` attribute attached to the same formatting property element.

Instead of this attribute, the `style:font-name` 20.277 attribute should be used to specify the properties of a font.

The defined values for the `style:font-pitch` attribute are:

- `fixed`: font pitch is fixed.
- `variable`: font pitch is variable.

The `style:font-pitch` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The values of the `style:font-pitch` attribute are **fixed** or **variable**.

## 20.281 `style:font-pitch-asian`

The `style:font-pitch-asian` attribute specifies whether a font has a fixed or variable width.

This attribute is evaluated for [UNICODE] characters whose script type is `asian`. 20.358

This attribute is ignored if there is no corresponding `fo:font-family` 20.189 attribute attached to the same formatting property element.

Instead of this attribute, the `style:font-name-asian` 20.189 attribute should be used to specify the properties of a font.

The defined values for the `style:font-pitch-asian` attribute are:

- `fixed`: font width is fixed.
- `variable`: font width is variable.

The `style:font-pitch-asian` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The values of the `style:font-pitch-asian` attribute are **fixed** or **variable**.

## 20.282 `style:font-pitch-complex`

The `style:font-pitch-complex` attribute specifies whether a font has a fixed or variable width.

This attribute is evaluated for [UNICODE] characters whose script type is `complex`. 20.358

This attribute is ignored if there is no corresponding `fo:font-family` 20.189 attribute attached to the same formatting property element.

Instead of this attribute, the `style:font-name-complex` 20.279 attribute should be used to specify the properties of a font.

The defined values for the `style:font-pitch-complex` attribute are:

- `fixed`: font width is fixed.
- `variable`: font width is variable.

The `style:font-pitch-complex` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The values of the `style:font-pitch-complex` attribute are **fixed** or **variable**.

## 20.283 `style:font-relief`

The `style:font-relief` attribute specifies whether a font should be embossed, engraved, or neither.

The defined values for the `style:font-relief` attribute are:

- `embossed`: characters are embossed.
- `engraved`: characters are engraved.
- `none`: characters are neither embossed or engraved.

The `style:font-relief` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The values of the `style:font-relief` attribute are `none`, `embossed` or `engraved`.

## 20.284 `style:font-size-asian`

The `style:font-size-asian` attribute specifies the size of a font.

This attribute is evaluated for [UNICODE] characters whose script type is `asian`. 20.358

The value of this attribute is either an absolute length or a percentage as described in §7.8.4 of [XSL]. In contrast to XSL, percentage values can be used within common styles only and are based on the font height of the parent style rather than to the font height of the attributes neighborhood. Absolute font heights and relative font heights are not supported.

The `style:font-size-asian` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The values of the `style:font-size-asian` attribute are a value of type `positiveLength` 18.3.27 or a value of type `percent` 18.3.24.

## 20.285 `style:font-size-complex`

The `style:font-size-complex` attribute specifies the size of a font.

This attribute is evaluated for [UNICODE] characters whose script type is `complex`. 20.358

The value of this attribute is either an absolute length or a percentage as described in §7.8.4 of [XSL]. In contrast to XSL, percentage values can be used within common styles only and are based on the font height of the parent style rather than to the font height of the attributes neighborhood. Absolute font heights and relative font heights are not supported.

The `style:font-size-complex` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The values of the `style:font-size-complex` attribute are a value of type `positiveLength` 18.3.27 or a value of type `percent` 18.3.24.

## 20.286 `style:font-size-rel`

The `style:font-size-rel` attribute specifies a relative font size change.

This attribute is evaluated for any [UNICODE] character whose script type is `latin`. 20.358

This attribute specifies a relative font size change as a length. It cannot be used within automatic styles. This attribute changes the font size based on the font size of the parent style.

The `style:font-size-rel` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The `style:font-size-rel` attribute has the data type `length` 18.3.19.

## 20.287 `style:font-size-rel-asian`

The `style:font-size-rel-asian` attributes specifies a relative font size change.

This attribute is evaluated for [UNICODE] characters whose script type is `asian`. 20.358

This attribute specifies a relative font size change as a length. It cannot be used within automatic styles. This attribute changes the font size based on the font size of the parent style.

The `style:font-size-rel-asian` attribute is usable with the following element:  
`<style:text-properties>` 16.29.29.

The `style:font-size-rel-asian` attribute has the data type `length` 18.3.19.

## 20.288 `style:font-size-rel-complex`

The `style:font-size-rel-complex` attribute specifies a relative font size change.

The `style:font-size-rel-complex` attribute is evaluated for [UNICODE] characters whose script type is `complex`. 20.358

This attribute specifies a relative font size change as a length. It cannot be used within automatic styles. This attribute changes the font size based on the font size of the parent style.

The `style:font-size-rel-complex` attribute is usable with the following element:  
`<style:text-properties>` 16.29.29.

The `style:font-size-rel-complex` attribute has the data type `length` 18.3.19.

## 20.289 `style:font-style-asian`

The `style:font-style-asian` attribute specifies whether to use normal or italic font face. See §7.8.7 of [XSL].

This attribute is evaluated for [UNICODE] characters whose script type is `asian`. 20.358

The defined values for the `style:font-style-asian` attribute are:

- `italic`: characters displayed in an italic font face.
- `normal`: characters displayed in normal font face.
- `oblique`: characters displayed in an oblique font face.

The `style:font-style-asian` attribute is usable with the following element:  
`<style:text-properties>` 16.29.29.

The values of the `style:font-style-asian` attribute are `normal`, `italic` or `oblique`.

## 20.290 `style:font-style-complex`

The `style:font-style-complex` attribute specifies whether to use normal or italic font face. See §7.8.7 of [XSL].

This attribute is evaluated for [UNICODE] characters whose script type is `complex`. 20.358

The `fo:font-style` attribute is evaluated for any other [UNICODE] character.

The defined values for the `style:font-style-complex` attribute are:

- `italic`: characters displayed in an italic font face.
- `normal`: characters displayed in normal font face.
- `oblique`: characters displayed in an oblique font face.

The `style:font-style-complex` attribute is usable with the following element:  
`<style:text-properties>` 16.29.29.

The values of the `style:font-style-complex` attribute are `normal`, `italic` or `oblique`.

## 20.291 style:font-style-name

The `style:font-style-name` attribute specifies a font style name.

This attribute is evaluated for any [UNICODE] character whose script type is latin. 20.358

This attribute is ignored if there is no corresponding `fo:font-family` 20.189 attribute attached to the same formatting property element.

Instead of this attribute, the `style:font-name` 20.277 attribute should be used to specify the properties of a font.

The `style:font-style-name` attribute is usable with the following element:

`<style:text-properties>` 16.29.29.

The `style:font-style-name` attribute has the data type `string` 18.2.

## 20.292 style:font-style-name-asian

The `style:font-style-name-asian` attribute specifies a font style name.

The `style:font-style-name-asian` attribute is evaluated for [UNICODE] characters whose script type is asian. 20.358

This attribute is ignored if there is no corresponding `fo:font-family` 20.189 attribute attached to the same formatting property element.

Instead of this attribute, the `style:font-name-asian` 20.278 attribute should be used to specify the properties of a font.

The `style:font-style-name-asian` attribute is usable with the following element:

`<style:text-properties>` 16.29.29.

The `style:font-style-name-asian` attribute has the data type `string` 18.2.

## 20.293 style:font-style-name-complex

The `style:font-style-name-complex` attribute specifies a font style name.

The `style:font-style-name-complex` attribute is evaluated for [UNICODE] characters whose script type is complex. 20.358

This attribute is ignored if there is no corresponding `fo:font-family` 20.189 attribute attached to the same formatting property element.

Instead of this attribute, the `style:font-name-complex` 20.279 attribute should be used to specify the properties of a font.

The `style:font-style-name-complex` attribute is usable with the following element:

`<style:text-properties>` 16.29.29.

The `style:font-style-name-complex` attribute has the data type `string` 18.2.

## 20.294 style:font-weight-asian

The `style:font-weight-asian` attribute specifies the weight of a font. See §7.8.9 of [XSL].

This attribute is evaluated for [UNICODE] characters whose script type is asian. 20.358

The `style:font-weight-asian` attribute is usable with the following element:

`<style:text-properties>` 16.29.29.

The values of the `style:font-weight-asian` attribute are `normal`, `bold`, `100`, `200`, `300`, `400`, `500`, `600`, `700`, `800` or `900`.

## 20.295 style:font-weight-complex

The `style:font-weight-complex` attributes specify the weight of a font. See §7.8.9 of [XSL].

This attribute is evaluated for [UNICODE] characters whose script type is `complex`. 20.358

The `fo:font-weight` 20.193 attribute is evaluated for any other [UNICODE] character.

The `style:font-weight-complex` attribute is usable with the following element:  
`<style:text-properties>` 16.29.29.

The values of the `style:font-weight-complex` attribute are `normal`, `bold`, `100`, `200`, `300`, `400`, `500`, `600`, `700`, `800` or `900`.

## 20.296 style:footnote-max-height

The `style:footnote-max-height` attribute specifies the maximum amount of space on a page that a footnote can occupy. The value of the attribute is a length, which determines the maximum height of a footnote area.

If the value of this attribute is set to `0cm`, there is no limit to the amount of space that the footnote can occupy.

The `style:footnote-max-height` attribute is usable with the following element:  
`<style:page-layout-properties>` 17.2.

The `style:footnote-max-height` attribute has the data type `length` 18.3.19.

## 20.297 style:glyph-orientation-vertical

The `style:glyph-orientation-vertical` attribute specifies a vertical glyph orientation. See §10.7.3 of [SVG]. The attribute specifies an angle or automatic mode. The only defined angle is 0 degrees, which disables this feature.

**Note:** OpenDocument v1.1 did not support angle specifications that contain an angle unit identifier. Angle unit identifiers should be omitted for compatibility with OpenDocument v1.1.

The `style:glyph-orientation-vertical` attribute is usable with the following element:  
`<style:table-cell-properties>` 17.18.

The values of the `style:glyph-orientation-vertical` attribute are `auto`, `0`, `0deg`, `0rad` or `0grad`.

## 20.298 style:horizontal-pos

The `style:horizontal-pos` attribute specifies the horizontal alignment of a frame in relation to an area specified by the `style:horizontal-rel` 20.299 attribute.

The defined values for the `style:horizontal-pos` attribute are:

- `center`: horizontal alignment of a frame should be centered relative to the specified area.
- `from-inside`: on pages with an odd page number the left edge of the specific area is taken as the horizontal alignment of a frame. On pages with an even page number the right edge of the specified area is taken. Attribute `svg:x` 19.577 associated with the frame element specifies the horizontal position of the frame from the edge which is taken.
- `from-left`: the `svg:x` attribute associated with the frame element specifies the horizontal position of the frame from the left edge of the specified area.

- `inside`: on pages with an odd page number the horizontal alignment of a frame is the same as for the attribute value `left`. On pages with an even page number the horizontal alignment of a frame is the same as for the attribute value `right`.
- `left`: horizontal alignment of a frame should be left aligned relative to the specified area.
- `outside`: on pages with an odd page number the horizontal alignment of a frame is the same as for the attribute value `right`. On pages with an even page number the horizontal alignment of a frame is the same as for the attribute value `left`.
- `right`: horizontal alignment of a frame should be right aligned relative to the specified area.

If the attribute value is not `from-left` and not `from-inside`, the `svg:x` attribute associated with the frame element is ignored for text documents.

Tables 20 and 21 display the defined combinations of values of the attributes `style:horizontal-pos` and `style:horizontal-rel` 20.299. The values of these alignment attributes are listed in the first column on the left, and an alignment attribute value/anchor type value match is indicated by an X.

*Table 20 - Horizontal position values*

<b>Value of <code>style:horizontal-pos</code></b>	<b>Value of <code>text:anchor-type</code></b>				
	<b><code>page</code></b>	<b><code>frame</code></b>	<b><code>paragraph</code></b>	<b><code>char</code></b>	<b><code>as-char</code></b>
any	X	X	X	X	

*Table 21 - Horizontal relation values*

<b>Value of <code>style:horizontal-rel</code></b>	<b>Value of <code>text:anchor-type</code></b>				
	<b><code>page</code></b>	<b><code>frame</code></b>	<b><code>paragraph</code></b>	<b><code>char</code></b>	<b><code>as-char</code></b>
char				X	
frame		X			
frame-content		X			
frame-end-margin		X			
frame-start-margin		X			
page	X		X	X	
page-content	X		X	X	
page-end-margin	X		X	X	
page-start-margin	X		X	X	
paragraph			X	X	
paragraph-content			X	X	

Value of <del>style:horizontal-pos</del> paragraph-start-margin	Value of <i>text:anchor-type</i>			
			X	X
paragraph-start-margin			X	X

The *style:horizontal-pos* attribute is usable with the following element:  
`<style:graphic-properties> 17.21.`

The values of the *style:horizontal-pos* attribute are `left`, `center`, `right`, `from-left`, `inside`, `outside` or `from-inside`.

## 20.299 *style:horizontal-rel*

The *style:horizontal-rel* attribute specifies the area against which the horizontal position of a frame is positioned.

The value `start-margin` determines the left margin, except when the horizontal position is `from-inside`, `inside` or `outside` and the anchor for the frame is on a page with an even page number, in which case it determines the right margin. The value `end-margin` determines the opposite margin to the `start-margin` values.

The defined values for the *style:horizontal-rel* attribute are:

- `char`: horizontal position of a frame is positioned relative to a character.
- `page`: horizontal position of a frame is positioned relative to a page.
- `page-content`: horizontal position of a frame is positioned relative to page-content.
- `page-start-margin`: horizontal position of a frame is positioned relative to a page start margin.
- `page-end-margin`: horizontal position of a frame is positioned relative to a page end margin.
- `frame`: horizontal position of a frame is positioned relative to another frame.
- `frame-content`: horizontal position of a frame is positioned relative to frame content.
- `frame-end-margin`: horizontal position of a frame is positioned relative to a frame end margin.
- `frame-start-margin`: horizontal position of a frame is positioned relative to a frame start margin
- `paragraph`: horizontal position of a frame is positioned relative to a paragraph.
- `paragraph-content`: horizontal position of a frame is positioned relative to paragraph content.
- `paragraph-end-margin`: horizontal position of a frame is positioned relative to a paragraph end margin.
- `paragraph-start-margin`: horizontal position of a frame is positioned relative to a paragraph start margin.

The *style:horizontal-rel* attribute is usable with the following element:  
`<style:graphic-properties> 17.21.`

The values of the *style:horizontal-rel* attribute are `page`, `page-content`, `page-start-margin`, `page-end-margin`, `frame`, `frame-content`, `frame-start-margin`, `frame-end-margin`, `paragraph`, `paragraph-content`, `paragraph-start-margin`, `paragraph-end-margin` or `char`.

## 20.300 style:join-border

The `style:join-border` property specifies whether a border for one paragraph is to be extended around the following paragraph.

In addition to the value of this attribute, joining of borders requires meeting these conditions:

- 1) Values of attributes `fo:border-top` 20.183.6, `fo:border-bottom` 20.183.3, `fo:border-left` 20.183.4 and `fo:border-right` 20.183.5 are the same. These values can also be given by the `fo:border` 20.183.2 attribute.
- 2) Values of attributes `style:border-line-width-top` 20.252, `style:border-line-width-bottom` 20.249, `style:border-line-width-left` 20.250 and `style:border-line-width-right` 20.251 are the same. These values can also be given by the `style:border-line-width` 20.248 attribute.
- 3) Values of attributes `fo:padding-left` 20.219 and `fo:padding-right` 20.220 are the same. These values can also be given by the `fo:padding` 20.217 attribute.
- 4) Values of the `fo:margin-right` 20.208 attributes are the same. These values can also be given by the `fo:margin` 20.205 attribute.
- 5) Values of the `fo:margin-left` 20.207 attribute, which can also be given by the `fo:margin`, and `fo:text-indent` 19.246 attributes, that meet one of these conditions:
  1. All values are the same.
  2. Values of the `fo:margin-left` attributes are the same and values of the `fo:text-indent` attributes are non-negative.
  3. Value of the `fo:margin-left` attribute of one paragraph whose value of the `fo:text-indent` attribute is non-negative is the same as the sum of values of the `fo:margin-left` and `fo:text-indent` attributes of the other paragraph whose value of the `fo:text-indent` attribute is negative.
  4. Both values of the `fo:text-indent` attributes are negative and the sums of values of the `fo:margin-left` and `fo:text-indent` attributes are equal.

The default value of this attribute is `true`.

The defined values for the `style:join-border` attribute are:

- `false`: borders should not be joined.
- `true`: borders should be joined.

The `style:join-border` attribute is usable with the following element:  
`<style:paragraph-properties>` 17.6.

The `style:join-border` attribute has the data type `boolean` 18.3.3.

## 20.301 style:justify-single-word

The `style:justify-single-word` attribute specifies whether a single word should be justified when the last line in a paragraph is justified.

Specifying a `style:justify-single-word` attribute without specifying a `fo:text-align` 20.223 and `fo:text-align-last` 20.224 attribute has no effect. Unspecified, both `fo:text-align` and `fo:text-align-last` have the value `start`.

The defined values for the `style:justify-single-word` attribute are:

- `false`: single word should not be justified when the last line in a paragraph is justified.
- `true`: single word should be justified when last line in a paragraph is justified.

The `style:justify-single-word` attribute is usable with the following element:  
`<style:paragraph-properties>` 17.6.

The `style:justify-single-word` attribute has the data type `boolean` 18.3.3.

## 20.302 style:language-asian

The `style:language-asian` attribute specifies the language of a text.

This attribute is evaluated for [UNICODE] characters whose script type is `asian`. 20.358

This attribute may be ignored if it is not specified together with a `style:country-asian` 20.256 attribute.

The `style:language-asian` attribute is usable with the following element:  
`<style:text-properties>` 16.29.29.

The `style:language-asian` attribute has the data type `languageCode` 18.3.18.

## 20.303 style:language-complex

The `style:language-complex` attribute specifies the language of a text.

The `style:language-asian` attribute is evaluated for [UNICODE] characters whose script type is `asian`. 20.358

This attribute is evaluated for [UNICODE] characters whose script type is `complex`. 20.358

This attribute may be ignored if it is not specified together with a `style:country-complex` 20.257 attribute.

The `style:language-complex` attribute is usable with the following element:  
`<style:text-properties>` 16.29.29.

The `style:language-complex` attribute has the data type `languageCode` 18.3.18.

## 20.304 style:layout-grid-base-height

The `style:layout-grid-base-height` attribute specifies the height reserved in layout grid lines for non-ruby text.

The `style:layout-grid-base-height` attribute is usable with the following element:  
`<style:page-layout-properties>` 17.2.

The `style:layout-grid-base-height` attribute has the data type `length` 18.3.19.

## 20.305 style:layout-grid-base-width

The `style:layout-grid-base-width` attribute specifies a grid's base width. This attribute is only evaluated if `style:layout-grid-standard-mode` 20.314 attribute has the value `true`.

The `style:layout-grid-base-width` attribute is usable with the following element:  
`<style:page-layout-properties>` 17.2.

The `style:layout-grid-base-width` attribute has the data type `length` 18.3.19.

## 20.306 style:layout-grid-color

The `style:layout-grid-color` attribute specifies the color of layout grid border lines.

The `style:layout-grid-color` attribute is usable with the following element:  
`<style:page-layout-properties>` 17.2.

The `style:layout-grid-color` attribute has the data type [color 18.3.9](#).

## 20.307 style:layout-grid-display

The `style:layout-grid-display` attribute specifies whether layout grid border lines are displayed.

The defined values for the `style:layout-grid-display` attribute are:

- `false`: layout grid border lines should not be displayed.
- `true`: layout grid border lines should be displayed.

The `style:layout-grid-display` attribute is usable with the following element:

`<style:page-layout-properties> 17.2`.

The `style:layout-grid-display` attribute has the data type [boolean 18.3.3](#).

## 20.308 style:layout-grid-lines

The `style:layout-grid-lines` attribute specifies the number of layout grid lines per page. The number of lines displayed may be smaller than specified if the page does not have space to display the specified number of lines with the specified line height.

The `style:layout-grid-lines` attribute is usable with the following element:

`<style:page-layout-properties> 17.2`.

The `style:layout-grid-lines` attribute has the data type [positiveInteger 18.2](#).

## 20.309 style:layout-grid-mode

The `style:layout-grid-mode` attribute enables Asian layout grids.

The defined values for the `style:layout-grid-mode` attribute are:

- `both`: like lines, except that the lines are divided into rectangular layout cells. The calculation of layout cells depends on the `style:layout-grid-standard-mode` attribute. Within a layout cell, no more than one Asian [UNICODE] character is displayed. An Asian character that does not fit into a single layout cell is displayed centered into as many layout cells as required. Non Asian text is centered within as many cells as required.
- `lines`: enables a line layout, this is, the page is divided in a fixed number of lines. The exact number of lines depends on the other layout grid formatting properties. There is no space between the layout grid lines. The layout grid itself is centered on the page.
- `none`: disables the layout grid.

The `style:layout-grid-mode` attribute is usable with the following element:

`<style:page-layout-properties> 17.2`.

The values of the `style:layout-grid-mode` attribute are `none`, `line` or `both`.

## 20.310 style:layout-grid-print

The `style:layout-grid-print` attribute specifies whether layout grid border lines are printed.

The defined values for the `style:layout-grid-print` attribute are:

- `false`: layout grid border lines should not be printed.
- `true`: layout grid border lines should be printed.

The `style:layout-grid-print` attribute is usable with the following element:

`<style:page-layout-properties> 17.2`.

The `style:layout-grid-print` attribute has the data type [boolean 18.3.3](#).

## 20.311 style:layout-grid-ruby-below

The `style:layout-grid-ruby-below` attribute specifies whether ruby text is displayed above or below the base text.

The defined values for the `style:layout-grid-ruby-below` attribute are:

- `false`: ruby text should be displayed above the base text.
- `true`: ruby text should be displayed below the base text.

The `style:layout-grid-ruby-below` attribute is usable with the following element:  
`<style:page-layout-properties> 17.2`.

The `style:layout-grid-ruby-below` attribute has the data type [boolean 18.3.3](#).

## 20.312 style:layout-grid-ruby-height

The `style:layout-grid-ruby-height` attribute specifies the height reserved in layout grid lines for ruby text.

The `style:layout-grid-ruby-height` attribute appears on a `<style:page-layout-properties> 17.2` element only if a `style:layout-grid-standard-mode 20.314` attribute, with the value `false`, appears on the same `<style:page-layout-properties>` element.

The `style:layout-grid-ruby-height` attribute is usable with the following element:  
`<style:page-layout-properties> 17.2`.

The `style:layout-grid-ruby-height` attribute has the data type [length 18.3.19](#).

## 20.313 style:layout-grid-snap-to

The `style:layout-grid-snap-to` attribute specifies whether the text is snapped to the grid or not. It is evaluated only if the `style:layout-grid-mode 20.309` attribute has the value `both` and the `style:layout-grid-standard-mode 20.314` attribute has the value `true`.

The defined values for the `style:layout-grid-snap-to` attribute are:

- `false`: text should not be snapped to grid.
- `true`: text should be snapped to grid.

The `style:layout-grid-snap-to` attribute is usable with the following element:  
`<style:page-layout-properties> 17.2`.

The `style:layout-grid-snap-to` attribute has the data type [boolean 18.3.3](#).

## 20.314 style:layout-grid-standard-mode

The `style:layout-grid-standard-mode` attribute specifies how the rectangular grid cells are calculated if the `style:layout-grid-mode 20.309` attribute has the value `both`.

The `style:layout-grid-standard-mode` attribute can only be used in the default page layout. If the `style:layout-grid-standard-mode` attribute appears inside a `<style:page-layout> 16.5` element it shall be ignored.

The defined values for the `style:layout-grid-standard-mode` attribute are:

- `false`: the page is divided in a fixed number of lines, and the lines are divided into square cells. The number of cells per grid line depends on the line height, where the line height is

the sum of the base height and the ruby height as specified by the `style:layout-grid-base-height` 20.304 and `style:layout-grid-ruby-height` 20.312 attributes.

- `true`: the page is divided into a fixed numbers of lines, and the lines are divided into rectangular cells. The line height is specified by the `style:layout-grid-base-height` attribute, and the cell width is specified by `style:layout-grid-base-width` 20.305 attribute. The number of cells per line depends on the grid base width. This mode is also called *standard paper mode*. The `style:layout-grid-ruby-height` attribute is ignored in this mode.

The `style:layout-grid-standard-mode` attribute is usable with the following element:  
`<style:page-layout-properties>` 17.2.

The `style:layout-grid-standard-mode` attribute has the data type `boolean` 18.3.3.

## 20.315 style:line-break

The `style:line-break` attribute specifies line breaking rules.

The defined values for the `style:line-break` attribute are:

- `normal`: line breaks may occur between any characters.
- `strict`: line breaks shall not occur before or after implementation-defined characters.

The `style:line-break` attribute is usable with the following element:  
`<style:paragraph-properties>` 17.6.

The values of the `style:line-break` attribute are `normal` or `strict`.

## 20.316 style:letter-kerning

The `style:letter-kerning` attribute specifies whether kerning between characters is enabled or disabled.

The defined values for the `style:letter-kerning` attribute are:

- `false`: kerning between letters should be disabled.
- `true`: kerning between letters should be enabled.

The `style:letter-kerning` attribute is usable with the following element:  
`<style:text-properties>` 16.29.29.

The `style:letter-kerning` attribute has the data type `boolean` 18.3.3.

## 20.317 style:line-height-at-least

The `style:line-height-at-least` attribute specifies a minimum line height. The value of this attribute is a length.

The effect of this attribute is disabled when `fo:line-height` 20.204 has the value of `normal`.

The `style:line-height-at-least` attribute is usable with the following element:  
`<style:paragraph-properties>` 17.6.

The `style:line-height-at-least` attribute has the data type `nonNegativeLength` 18.3.21.

## 20.318 style:line-spacing

The `style:line-spacing` attribute specifies a fixed distance between two lines.

The effect of this attribute is disabled when `fo:line-height` 20.204 has the value of `normal`.

The `style:line-spacing` attribute is usable with the following element:  
`<style:paragraph-properties>` 17.6.

The `style:line-spacing` attribute has the data type `length` 18.3.19.

## 20.319 `style:may-break-between-rows`

The `style:may-break-between-rows` attribute specifies that a page break may occur inside a table.

The defined values for the `style:may-break-between-rows` attribute are:

- `false`: page break shall not occur inside a table.
- `true`: page break may occur inside a table.

The `style:may-break-between-rows` attribute is usable with the following element:  
`<style:table-properties>` 17.15.

The `style:may-break-between-rows` attribute has the data type `boolean` 18.3.3.

## 20.320 `style:min-row-height`

The `style:min-row-height` attribute specifies a fixed minimum height for a row.

The `style:min-row-height` attribute is usable with the following element:  
`<style:table-row-properties>` 17.17.

The `style:min-row-height` attribute has the data type `nonNegativeLength` 18.3.21.

## 20.321 `style:mirror`

The `style:mirror` attribute specifies whether an image is mirrored before it is displayed. The mirroring can be vertical or horizontal or both.

The defined values for the `style:mirror` attribute are:

- `none`: image should not be mirrored before being displayed.
- `horizontal`: image should be mirrored horizontally before being displayed.
- `horizontal-on-even`: image should be mirrored horizontally on even numbered pages before being displayed.
- `horizontal-on-odd`: image should be mirrored horizontally on odd numbered pages before being displayed.
- `vertical`: image should be mirrored vertically before being displayed.

The value `vertical` and the horizontal values can be specified together, separated by a white space.

The `style:mirror` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The values of the `style:mirror` attribute are `none`, `vertical`, or two white space separated values, that may appear in any order. One of these values is always `vertical`. The other value is one of: `horizontal`, `horizontal-on-odd` or `horizontal-on-even`.

## 20.322 style:num-format

The `style:num-format` attribute specifies a numbering sequence.

If no value is given, no number sequence is displayed.

The defined values for the `style:num-format` attribute are:

- 1: number sequence starts with "1".
- a: number sequence starts with "a".
- A: number sequence starts with "A".
- empty string: no number sequence displayed.
- i: number sequence starts with "i".
- I: number sequence start with "I".
- a value of type `string` 18.2

The `style:num-format` attribute is usable with the following element: `<style:page-layout-properties>` 17.2.

The values of the `style:num-format` attribute are 1, i, I, a value of type `string` 18.2, an empty string, a or A.

## 20.323 style:num-letter-sync

The `style:num-letter-sync` attribute specifies whether letter synchronization shall take place. If letters are used in alphabetical order for numbering, there are two ways to process overflows within a digit, as follows:

- `false`: A new digit is inserted. Its start value is "a" or "A", and it is incremented every time an overflow occurs in the following digit. The numbering sequence (for lower case numberings) in that case is a,b,c, ..., z, aa, ab, ac, ...,az, ba, ..., and so on.
- `true`: A new digit is inserted that always has the same value as the following digit. The numbering sequence (for lower case numberings) in that case is a, b, c, ..., z, aa, bb, cc, ..., zz, aaa, ..., and so on. This is called letter synchronization.

The `style:num-letter-sync` attribute is usable with the following element: `<style:page-layout-properties>` 17.2.

The `style:num-letter-sync` attribute has the data type `boolean` 18.3.3.

## 20.324 style:num-prefix

The `style:num-prefix` attribute specifies what to display before a number.

If the `style:num-prefix` and `style:num-suffix` 19.507 values do not contain any character that has a [UNICODE] category of Nd, Nl, No, Lu, Li, Lt, Lm or Lo, an [XSLT] format attribute can be created from the OpenDocument attributes by concatenating the values of the `style:num-prefix`, `style:num-format` 19.504, and `style:num-suffix` 19.507 attributes.

The `style:num-prefix` attribute can also specify a character before the value of a `text:bullet-char` 19.765 attribute.

The `style:num-prefix` attribute is usable with the following element: `<style:page-layout-properties>` 17.2.

The `style:num-prefix` attribute has the data type `string` 18.2.

## 20.325 style:num-suffix

The `style:num-prefix` and `style:num-suffix` `style:num-suffix` attributes specify what to display before and after a number.

If the `style:num-prefix` and `style:num-suffix` values do not contain any character that has a [UNICODE] category of Nd, Nl, No, Lu, Ll, Lt, Lm or Lo, an [XSLT] `format` attribute can be created from the OpenDocument attributes by concatenating the values of the `style:num-prefix`, `style:num-format` 19.504, and `style:num-suffix` 19.507 attributes.

The `style:num-suffix` attribute can also specify a character after the value of a `text:bullet-char` 19.765 attribute.

The `style:num-suffix` attribute is usable with the following element: `<style:page-layout-properties> 17.2`.

The `style:num-suffix` attribute has the data type `string` 18.2.

## 20.326 style:number-wrapped-paragraphs

The `style:number-wrapped-paragraphs` attribute specifies the number of paragraphs that can wrap around a frame if the anchor position of a frame or drawing shape is a paragraph or a character, and the wrap mode specified by the `style:wrap` 20.400 attribute is `left`, `right`, `parallel`, or `dynamic`.

This attribute is only recognized in frames or styles that have a `style:wrap` attribute attached with a value of `left`, `right`, `parallel`, or `dynamic`.

The defined values for the `style:number-wrapped-paragraphs` attribute are:

- `no-limit`: there is no limit on the number of paragraphs that may wrap around a frame.
- a value of type `positiveInteger`

The `style:number-wrapped-paragraphs` attribute is usable with the following element: `<style:graphic-properties> 17.21`.

The values of the `style:number-wrapped-paragraphs` attribute are `no-limit` or a value of type `positiveInteger` 18.2.

## 20.327 style:overflow-behavior

The `style:overflow-behavior` attribute specifies the behavior of text boxes where the containing text does not fit into a text box.

If the `style:overflow-behavior` attribute value is `auto-create-new-frame` and the text box has a minimum width or height specified the text box will grow as long as there is space left on the page. If no space is left, a new frame will be created on the next page, with the same position and dimensions of the original frame.

The defined values for the `style:overflow-behavior` attribute are:

- `clip`: text that does not fit into a text box is not displayed.
- `auto-create-new-frame`: a new frame will be created on the next page, with the same position and dimensions of the original frame.

The `style:overflow-behavior` attribute is usable with the following element: `<style:graphic-properties> 17.21`.

The values of the `style:overflow-behavior` attribute are `clip` or `auto-create-new-frame`.

## 20.328 style:page-number

The `style:page-number` attribute specifies the page number that should be used for a new page when either a paragraph or table style specifies a master page that should be applied beginning from the start of a paragraph or table.

The defined values for the `style:page-number` attribute are:

- `auto`: a page has the page number of the previous page, incremented by one.
- a value of type `nonNegativeInteger` 18.2: specifies a page number.

The `style:page-number` attribute is usable with the following elements:

`<style:paragraph-properties>` 17.6 and `<style:table-properties>` 17.15.

The values of the `style:page-number` attribute are a value of type `nonNegativeInteger` 18.2 or `auto`.

## 20.329 style:paper-tray-name

The `style:paper-tray-name` attribute specifies the paper tray to use when printing a document. The names assigned to the paper trays depends upon the printer.

The defined values for the `style:paper-tray-name` attribute are:

- `default`: the default tray specified by printer configuration settings.
- a value of type `string` 18.2

The `style:paper-tray-name` attribute is usable with the following element:

`<style:page-layout-properties>` 17.2.

The values of the `style:paper-tray-name` attribute are `default` or a value of type `string` 18.2.

## 20.330 style:print

The `style:print` attribute specifies the components in a spreadsheet document to print.

The value of the `style:print` attribute is a white space separated list of one or more of these values:

- `annotations`: annotations should be printed.
- `charts`: charts should be printed.
- `drawings`: drawings should be printed.
- `formulas`: formulas should be printed.
- `grid`: grid lines should be printed.
- `headers`: headers should be printed.
- `objects`: (including graphics): objects should be printed.
- `zero-values`: zero-values should be printed.

The `style:print` attribute is usable with the following element: `<style:page-layout-properties>` 17.2.

The value of the `style:print` attribute is a white space separated lists of one of these values: `headers`, `grid`, `annotations`, `objects`, `charts`, `drawings`, `formulas`, or `zero-values`, including the empty list.

## 20.331 style:print-content

### 20.331.1 General

The `style:print-content` attribute specifies if content is printed.

### 20.331.2 <style:graphic-properties>

The `style:print-content` attribute specifies if frame content is printed.

The `style:print-content` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The `style:print-content` attribute has the data type `boolean` 18.3.3.

### 20.331.3 <style:table-cell-properties>

The `style:print-content` attribute specifies if cell content is printed.

The `style:print-content` attribute is usable with the following element:  
`<style:table-cell-properties>` 17.18.

The `style:print-content` attribute has the data type `boolean` 18.3.3.

## 20.332 style:print-page-order

The `style:print-page-order` attribute specifies the order in which data in a spreadsheet is numbered and printed when the data does not fit on one printed page.

The defined values for the `style:print-page-order` attribute are:

- `ltr`: create pages from the first column to the last column before continuing with the next set of rows.
- `ttb`: create pages from the top row to the bottom row before continuing with the next set of columns.

The `style:print-page-order` attribute is usable with the following element:  
`<style:page-layout-properties>` 17.2.

The values of the `style:print-page-order` attribute are `ttb` or `ltr`.

## 20.333 style:print-orientation

The `style:print-orientation` attribute specifies the orientation of the printed page. The value of this attribute can be `portrait` or `landscape`.

The defined values for the `style:print-orientation` attribute are:

- `landscape`: a page is printed in landscape orientation.
- `portrait`: a page is printed in portrait orientation.

The `style:print-orientation` attribute is usable with the following element:  
`<style:page-layout-properties>` 17.2.

The values of the `style:print-orientation` attribute are `portrait` or `landscape`.

## 20.334 style:protect

### 20.334.1 General

The `style:protect` attribute specifies the protection of content.

## 20.334.2 <style:graphic-properties>

The `style:protect` attribute specifies whether the content, size, or position of a frame is protected. The value of this attribute can be either `none` or a white space separated list that consists of any of the values `content`, `position`, or `size`.

The defined values for the `style:protect` attribute are:

- `content`: content of a frame should be protected from editing.
- `none`: the content, size and position of a frame should be subject to editing.
- `position`: the position of a frame should be protected from editing.
- `size`: the size of a frame should be protected from editing.

The `style:protect` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The values of the `style:protect` attribute are `none`, or white space separated non-empty lists of one of these values: `content`, `position`, or `size`.

## 20.334.3 <style:section-properties>

The `style:protect` attribute specifies that sections should not be changed. The user interface should prevent the user from manually making any changes. The `style:protect` attribute should be set to `true` by default for linked sections or indexes. Removing the protection from linked sections or indexes leaves them unprotected from editing, but any edits will be over-written by updates to those sections.

The defined values for the `style:protect` attribute are:

- `false`: sections should not be protected from editing.
- `true`: sections should be protected from editing.

The `style:protect` attribute is usable with the following element: `<style:section-properties>` 17.11.

The `style:protect` attribute has the data type `boolean` 18.3.3.

## 20.335 style:punctuation-wrap

The `style:punctuation-wrap` attribute specifies whether a punctuation mark, if one is present, can be hanging, that is, whether it can be placed in the margin area at the end of a full line of text.

The defined values for the `style:punctuation-wrap` attribute are:

- `hanging`: a punctuation mark can be placed in the margin area at the end of a full line of text.
- `simple`: a punctuation mark cannot be placed in the margin area at the end of a full line of text.

The `style:punctuation-wrap` attribute is usable with the following element: `<style:paragraph-properties>` 17.6.

The values of the `style:punctuation-wrap` attribute are `simple` or `hanging`.

## 20.336 style:register-true

The `style:register-true` attribute specifies whether the lines on both sides of a printed page align. The text baselines of text in page columns or text box columns also align.

The defined values for the `style:register-true` attribute are:

- `false`: lines on both sides of a printed text need not align.
- `true`: lines on both sides of a printed text should align.

The `style:register-true` attribute is usable with the following element:  
`<style:paragraph-properties>` 17.6.

The `style:register-true` attribute has the data type `boolean` 18.3.3.

## 20.337 style:register-truth-ref-style-name

The `style:register-truth-ref-style-name` attribute specifies a paragraph style. The line distance specified of the paragraph style is used as the reference line distance for all paragraphs that have the register-truth feature enabled.

The `style:register-truth-ref-style-name` attribute is usable with the following element: `<style:page-layout-properties>` 17.2.

The `style:register-truth-ref-style-name` attribute has the data type `styleNameRef` 18.3.33.

## 20.338 style:rel-column-width

The `style:rel-column-width` attribute specifies a relative width of a column with a number value, followed by a "\*" (U+002A, ASTERISK) character. If  $r_c$  is the relative width of the column,  $r_s$  the sum of all relative columns widths, and  $w_s$  the absolute width that is available for these columns the absolute width  $w_c$  of the column is  $w_c = r_c w_s / r_s$ .

The `style:rel-column-width` attribute is usable with the following element:  
`<style:table-column-properties>` 17.16.

The `style:rel-column-width` attribute has the data type `relativeLength` 18.3.28.

## 20.339 style:rel-height

The `style:rel-height` attribute specifies the height of a drawing object as a relative value within a frame.

The defined values for the `style:rel-height` attribute are:

- `scale`: the height should be calculated depending on the width, so that the ratio of width and height of the original image or object size is preserved.
- `scale-min`: the height should be calculated as for value `scale`, but the calculated height is a minimum height rather than an absolute one.
- a value of type `percent` 18.3.24.

The interpretation of relative values depends on the anchor of the drawing object. If the anchor for the drawing object is in a table cell, the percentage value is relative to the surrounding table box. If the anchor for the drawing object is in a text box, the percentage value is relative to the surrounding text box. In other cases, the percentage values is relative to the width of the page or window.

To support consumers that do not support relative width and heights, producers should also provide the height in `svg:height` 19.543 and `fo:min-height` 19.240 attributes.

The `style:rel-height` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The values of the `style:rel-height` attribute are a value of type `percent` 18.3.24, `scale` or `scale-min`.

## 20.340 style:rel-width

### 20.340.1 <style:graphic-properties>

The `style:rel-width` attribute specifies the relative width of a drawing object.

The defined values for the `style:rel-width` attribute are:

- `scale`: the width should be calculated depending on the height, so that the ratio of width and height of the original image or object size is preserved.
- `scale-min`: the width should be calculated as for value `scale`, but the calculated width is a minimum width rather than an absolute one.
- a value of type `percent` 18.3.24.

The interpretation of the percent value depends on the anchor of the drawing object. If the anchor for the drawing object is in a table cell, the percent value of the surrounding table box. If the anchor for the drawing object is in a text box, the percentage value of the surrounding text box. In all other cases, the percent value of the containing page or window

To support consumers that do not support relative width, producers should also provide the width in a `svg:width` 19.575 attribute.

The `style:rel-width` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The values of the `style:rel-width` attribute are a value of type `percent` 18.3.24, `scale` or `scale-min`.

### 20.340.2 <style:table-properties>

The `style:rel-width` attribute specifies the width of a table relative to the width of the area that the table is in.

The `style:rel-width` attribute is usable with the following element: `<style:table-properties>` 17.15.

The `style:rel-width` attribute has the data type `percent` 18.3.24.

## 20.341 style:repeat

The `style:repeat` attribute specifies if an image can be repeated or stretched over an area.

The defined values for the `style:repeat` attribute are:

- `no-repeat`: image should not be repeated.
- `repeat`: image should be repeated.
- `stretch`: image should be scaled anisotropically to fill the background completely.
- `scale`: image should be scaled isotropically to fill the background along one axis. By default, the image is centered on the other axis, unless the `draw:fill-image-ref-point` 20.125 attribute exists and specifies the alignment on the other axis.

The `style:repeat` attribute is usable with the following elements: `<style:drawing-page-properties>` 17.25 and `<style:graphic-properties>` 17.21.

The values of the `style:repeat` attribute are `no-repeat`, `repeat`, `stretch` or `scale`.

## 20.342 style:repeat-content

The `style:repeat-content` attribute specifies whether text content of a cell is displayed as many times as there is space left in the cell's writing direction. The attribute has no effect for cell content that contains a line break.

The defined values for the `style:repeat-content` attribute are:

- `false`: text content of a cell should not be displayed as many times as there is space left in the cell's writing direction.
- `true`: text content of a cell should be displayed as many times as there is space left in the cell's writing direction.

The `style:repeat-content` attribute is usable with the following element:  
`<style:table-cell-properties>` 17.18.

The `style:repeat-content` attribute has the data type `boolean` 18.3.3.

## 20.343 style:rfc-language-tag

The `style:rfc-language-tag` attribute specifies a language identifier according to the rules of [RFC5646], or its successors.

It shall only be used if its value cannot be expressed as a valid combination of the `fo:language` 20.202, `fo:script` 20.222 and `fo:country` 20.188 attributes.

Producers may add support for consumers that don't support the `style:rfc-language-tag` attribute by specifying `fo:language`, `fo:script` and `fo:country` attributes with values that are implementation-dependent.

The attribute is evaluated for any [UNICODE] characters that are not of type asian or complex. 20.358

The `style:rfc-language-tag` attribute is usable with the following element:  
`<style:text-properties>` 16.29.29.

The `style:rfc-language-tag` attribute has the data type `language` 18.3.17.

## 20.344 style:rfc-language-tag-asian

The `style:rfc-language-tag-asian` attribute specifies a language identifier according to the rules of [RFC5646], or its successors.

It shall only be used if its value cannot be expressed as a valid combination of the `style:language-asian` 20.302, `style:script-asian` 20.356 and `style:country-asian` 20.256 attributes.

Producers may add support for consumers that don't support the `style:rfc-language-tag-asian` attribute by specifying `style:language-asian`, `style:script-asian` and `style:country-asian` attributes with values that are implementation-dependent.

This attribute is evaluated for [UNICODE] characters that are of type asian. 20.358.

The `style:rfc-language-tag-asian` attribute is usable with the following element:  
`<style:text-properties>` 16.29.29.

The `style:rfc-language-tag-asian` attribute has the data type `language` 18.3.17.

## 20.345 style:rfc-language-tag-complex

The `style:rfc-language-tag-complex` attribute specifies a language identifier according to the rules of [RFC5646], or its successors.

It shall only be used if its value cannot be expressed as a valid combination of the `style:language-complex 20.303`, `style:script-complex 20.357` and `style:country-complex 20.257` attributes.

Producers may add support for consumers that don't support the `style:rfc-language-tag-complex` attribute by specifying `style:language-complex`, `style:script-complex` and `style:country-complex` attributes with values that are implementation-dependent.

This attribute is evaluated for [UNICODE] characters that are of type complex..

The `style:rfc-language-tag-complex` attribute is usable with the following element: `<style:text-properties> 16.29.29`.

The `style:rfc-language-tag-complex` attribute has the data type `language 18.3.17`.

## 20.346 style:rotation-align

The `style:rotation-align` attribute specifies how the edge of the text in a cell is aligned after a rotation.

The defined values for the `style:rotation-align` attribute are shown in Table 22:

*Table 22 - Rotation align values*

<b>Alignment</b>	<b>Text is...</b>	<b>Borders and background are...</b>
None.	Rotated and aligned within the cell.	Unchanged.
Bottom	Rotated and may overlap with other cells if the text is longer than the length of the cell.	Positioned parallel to the text, whereby the edge that is named by the attribute value aligns with the corresponding edge of the cell's original position.
Center		
Top		

The `style:rotation-align` attribute is usable with the following element: `<style:table-cell-properties> 17.18`.

The values of the `style:rotation-align` attribute are `none`, `bottom`, `top` or `center`.

## 20.347 style:rotation-angle

The `style:rotation-angle` attribute specifies the rotation angle of content. The attribute value is an angle.

If used with a chart style applied to a `<chart:axis> 11.9` element the attribute specifies the rotation of the axis labels, each tick label is rotated, the axis title is not affected in this case.

If used with a chart style applied to a `<chart:label> 11.15`, `<chart:label> 11.14`, `<chart:series> 11.12` or `<chart:plot-area> 11.5` element the attribute specifies the rotation angle of the according data labels.

If used with a chart style applied to a `<chart:title> 11.2.1` element the attribute specifies the rotation angle of the title.

The `style:rotation-angle` attribute is usable with the following elements: `<style:chart-properties> 17.22` and `<style:table-cell-properties> 17.18`.

The `style:rotation-angle` attribute has the data type `angle 18.3.1`.

## 20.348 style:row-height

The `style:row-height` attribute specifies a fixed row height.

The `style:row-height` attribute is usable with the following element: `<style:table-row-properties>` 17.17.

The `style:row-height` attribute has the data type `positiveLength` 18.3.27.

## 20.349 `style:ruby-align`

The `style:ruby-align` attribute specifies the horizontal alignment of the ruby text in relationship to the ruby base.

The defined values for the `style:ruby-align` attribute are:

- `center`: ruby text centered on ruby base text.
- `distribute-letter`: If the width of the ruby text is smaller than that of the ruby base text, then the ruby text contents are evenly distributed across the width of the ruby base text, with the first and last ruby text glyphs lining up with the corresponding first and last base glyphs. If the width of the ruby text is at least the width of the ruby base text, then the letters of the ruby base text are evenly distributed across the width of the ruby text.
- `distribute-space`: If the width of the ruby text is smaller than that of the ruby base text, then the ruby text contents are evenly distributed across the width of the ruby base text, with a amount of white space preceding the first and following the last character in the ruby text. That amount of white space is normally equal to half the amount of inter-character space of the ruby text. If the width of the ruby text is at least the width of the ruby base text, then the same type of space distribution applies to the ruby base text. If the base is shorter than the ruby text, the base is `distribute-space` aligned.
- `left`: ruby text to the left of ruby base text.
- `right`: ruby text to the right of ruby base text.

The `style:ruby-align` attribute is usable with the following element: `<style:ruby-properties>` 17.10.

The values of the `style:ruby-align` attribute are `left`, `center`, `right`, `distribute-letter` or `distribute-space`.

## 20.350 `style:ruby-position`

The `style:ruby-position` attribute specifies the vertical position of the ruby text in relationship to the ruby base.

The defined values for the `style:ruby-position` attribute are:

- `above`: ruby text should be above ruby base text.
- `below`: ruby text should be below ruby base text.

The `style:ruby-position` attribute is usable with the following element: `<style:ruby-properties>` 17.10.

The values of the `style:ruby-position` attribute are `above` or `below`.

## 20.351 `style:run-through`

The `style:run-through` attribute specifies whether content of a frame is displayed in the background or foreground.

The defined values for the `style:run-through` attribute are:

- `background`: frame content is displayed behind a text.
- `foreground`: frame content is displayed in front of a text.

The default value for this attribute is `foreground`.

Use of this attribute depends upon the presence of a `style:wrap` 20.400 attribute with the value `run-through`.

The `style:run-through` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The values of the `style:run-through` attribute are `foreground` or `background`.

## 20.352 `style:scale-to`

The `style:scale-to` attribute specifies that a document is to be scaled to a percentage value. A value of 100% means no scaling.

If `style:scale-to`, `style:scale-to-pages` 20.353, `style:scale-to-X` 20.354 and `style:scale-to-Y` 20.355 attributes are absent, a document is not scaled.

The `style:scale-to` attribute is usable with the following element: `<style:page-layout-properties>` 17.2.

The `style:scale-to` attribute has the data type `percent` 18.3.24.

## 20.353 `style:scale-to-pages`

The `style:scale-to-pages` attribute specifies the number of pages on which a document should be printed. The document is scaled to fit a specified number of pages.

If `style:scale-to-pages`, `style:scale-to` 20.352, `style:scale-to-X` 20.354 and `style:scale-to-Y` 20.355 attributes are absent, a document is not scaled.

The `style:scale-to-pages` attribute is usable with the following element:  
`<style:page-layout-properties>` 17.2.

The `style:scale-to-pages` attribute has the data type `positiveInteger` 18.2.

## 20.354 `style:scale-to-X`

The `style:scale-to-X` attribute specifies the number of pages on which a document should be printed. The document is scaled to fit up to `style:scale-to-X` pages horizontally. If `style:scale-to-X`, `style:scale-to-Y` 20.355, `style:scale-to-pages` 20.353 and `style:scale-to` 20.352 are absent, a document is not scaled.

The `style:scale-to-X` attribute is usable with the following element: `<style:page-layout-properties>` 17.2.

The `style:scale-to-X` attribute has the data type `positiveInteger` 18.2.

## 20.355 `style:scale-to-Y`

The `style:scale-to-Y` attribute specifies the number of pages on which a document should be printed. The document is scaled to fit up to `style:scale-to-Y` pages vertically. If `style:scale-to-Y`, `style:scale-to-X` 20.354, `style:scale-to-pages` 20.353 and `style:scale-to` 20.352 are absent, a document is not scaled.

The `style:scale-to-Y` attribute is usable with the following element: `<style:page-layout-properties>` 17.2.

The `style:scale-to-Y` attribute has the data type `positiveInteger` 18.2.

## 20.356 `style:script-asian`

The `style:script-asian` attribute specifies the script code of a text. The attribute should be used only if necessary according to the rules of §2.2.3 of [RFC5646], or its successors.

This attribute is evaluated for [UNICODE] characters whose script type is `asian`. 20.358

It may be ignored if is not specified together with a `style:language-asian` 20.302 attribute.

The `style:script-asian` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The `style:script-asian` attribute has the data type `scriptCode` 18.3.30.

## 20.357 `style:script-complex`

The `style:script-complex` attribute specifies the script code of a text. The attribute should be used only if necessary according to the rules of §2.2.3 of [RFC5646], or its successors.

This attribute is evaluated for [UNICODE] characters whose script type is `complex`. 20.358

It may be ignored if is not specified together with a `style:language-complex` 20.303 attribute.

The `style:script-complex` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The `style:script-complex` attribute has the data type `scriptCode` 18.3.30.

## 20.358 `style:script-type`

The `style:script-type` attribute specifies which script type dependent attributes ( `fo:font-family` 20.189, `style:font-family-asian` 20.302, `style:font-family-complex` 20.303) are currently active for a portion of text. The attribute may be evaluated by consumers that cannot determine the script types of Unicode characters to select the correct script type dependent formatting properties. Consumers that can determine script types of Unicode characters may also evaluate the attribute and overwrite the script type they determine for a specific character with the value of the attribute.

**Note:** The usage of this attribute simplifies transformations from and to [CSS2]/[XSL] and other formats that do not have script-dependent attributes, and also can be used to assign script-types to weak [UNICODE] characters, where consumers may choose different script types.

The value `ignore` can be used only within default styles. If it is set, all script-dependent attributes are applied to all script types. This would mean that a `fo:font-family` would be applied to all script types as well as a `style:font-family-asian` or `style:font-family-complex`.

**Note:** Use of the `ignore` value simplifies saving documents with producers that do not support a script type.

The defined values for the `style:script-type` attribute are:

- `asian`: all asian script type dependent attributes are active.
- `complex`: all complex script type dependent attributes are active.
- `ignore`: all script type dependent attributes are applied to all script types. This is available on default styles only.
- `latin`: all latin script type dependent attributes are active.

The mapping of Unicode code points to script types is defined by table 23. Consumers should apply this mapping. For Unicode code points for which no mapping is defined, the mapping is implementation-dependent.

Table 23 - Unicode Code Point to Script Type Mapping

Unicode Code Point Ranges	Script Type
U+0003..U+001F, U+0021..U+009F, U+00A1..U+04FF, U+0530..U+058F, U+10A0..U+10FF, U+13A0..U+16FF, U+1E00..U+1FFF, U+2C60..U+2C7F, U+2C80..U+2CE3, U+A720..U+A7FF	latin
U+0590..U+074F, U+0780..U+07BF, U+0900..U+109F, U+1200..U+137F, U+1780..U+18AF, U+FB50..U+FDFF, U+FE70..U+FEFF	complex
U+1100..U+11FF, U+2E80..U+31BF, U+31C0..U+31EF, U+3200..U+4DBF, U+4E00..U+A4CF, U+AC00..U+D7AF, U+F900..U+FAFF, U+FE30..U+FE4F, U+FF00..U+FFEF, U+20000..U+2A6DF, U+2F800..U+2FA1F	asian

The `style:script-type` attribute specifies which script type dependent attributes are currently active for a portion of text. The attribute may be evaluated by consumers that do not support script types to select the correct script dependent formatting properties. Consumers that support script types may also evaluate the attribute and overwrite the script type they would evaluate for a specific character.

The `style:script-type` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The values of the `style:script-type` attribute are `latin`, `asian`, `complex` or `ignore`.

## 20.359 style:shadow

The `style:shadow` attribute specifies a shadow effect.

The defined values for this attribute are those defined in §7.16.5 of [XSL], except the value `inherit`.

The shadow effect is not applied to the text content of an element, but depending on the element where the attribute appears, to a paragraph, a text box, a page body, a header, a footer, a table or a table cell.

The `style:shadow` attribute is usable with the following elements: `<style:graphic-properties>` 17.21, `<style:header-footer-properties>` 17.5, `<style:page-layout-properties>` 17.2, `<style:paragraph-properties>` 17.6, `<style:table-cell-properties>` 17.18 and `<style:table-properties>` 17.15.

The values of the `style:shadow` attribute are `none` or a value of type `string` 18.2.

## 20.360 style:shrink-to-fit

The `style:shrink-to-fit` attribute specifies whether content is reduced in size to fit within a cell or drawing object. Shrinking means that the font size of the content is decreased to fit the content into a cell or drawing object. The attribute has no effect on cells where the cell content already fits into the cell.

The defined values for the `style:shrink-to-fit` attribute are:

- `false`: content should not be reduced in size to fit within a cell or drawing object.
- `true`: content should be reduced in size to fit within a cell or drawing object.

The `style:shrink-to-fit` attribute is usable with the following elements: `<style:graphic-properties>` 17.21 and `<style:table-cell-properties>` 17.18.

The `style:shrink-to-fit` attribute has the data type `boolean` 18.3.3.

## 20.361 style:snap-to-layout-grid

The `style:snap-to-layout-grid` attribute specifies whether the layout of a paragraph should consider the layout grid settings of the page where it appears.

The defined values for the `style:snap-to-layout-grid` attribute are:

- `false`: layout of a paragraph should not consider the layout grid settings of the page where it appears.
- `true`: layout of a paragraph should consider the layout grid settings of the page where it appears.

The `style:snap-to-layout-grid` attribute is usable with the following element:  
`<style:paragraph-properties>` 17.6.

The `style:snap-to-layout-grid` attribute has the data type `boolean` 18.3.3.

## 20.362 style:tab-stop-distance

The `style:tab-stop-distance` attribute specifies the distance between default tab stops. A default tab stop is repeated automatically after the specified distance. Default tab stops are only evaluated if they are specified within a default style.

The `style:tab-stop-distance` attribute is usable with the following element:  
`<style:paragraph-properties>` 17.6.

The `style:tab-stop-distance` attribute has the data type `nonNegativeLength` 18.3.21.

## 20.363 style:table-centering

The `style:table-centering` attribute specifies whether tables are centered horizontally and/or vertically on the page. This attribute only applies to spreadsheet documents.

The default is to align the table to the top-left or top-right corner of the page, depending of its writing direction.

The defined values for the `style:table-centering` attribute are:

- `both`: tables should be centered both horizontally and vertically on the pages where they appear.
- `horizontal`: tables should be centered horizontally on the pages where they appear.
- `none`: tables should not be centered both horizontally or vertically on the pages where they appear.
- `vertical`: tables should be centered vertically on the pages where they appear.

The `style:table-centering` attribute is usable with the following element:  
`<style:page-layout-properties>` 17.2.

The values of the `style:table-centering` attribute are `horizontal`, `vertical`, `both` or `none`.

## 20.364 style:text-align-source

The `style:text-align-source` attribute specifies the source of a `text-align` attribute.

The defined values for the `style:text-align-source` attribute are:

- `fix`: content alignment uses the value of the `fo:text-align` 20.223 attribute.
- `value-type`: content alignment uses the value-type of the cell.

The default alignment for a cell value-type `string` is left, for other value-types it is right.

The `style:text-align-source` attribute is usable with the following element:  
`<style:table-cell-properties>` 17.18.

The values of the `style:text-align-source` attribute are `fix` or `value-type`.

## 20.365 `style:text-autospace`

The `style:text-autospace` attribute specifies whether to add space between portions of Asian, Western, and complex texts.

The defined values for the `style:text-autospace` attribute are:

- `ideograph-alpha`: space should be added between portions of Asian, Western and complex texts.
- `none`: space should not be added between portions of Asian, Western and complex texts.

The `style:text-autospace` attribute is usable with the following element:  
`<style:paragraph-properties>` 17.6.

The values of the `style:text-autospace` attribute are `none` or `ideograph-alpha`.

## 20.366 `style:text-blinking`

The `style:text-blinking` attribute specifies whether text blinks.

The defined values for the `style:text-blinking` attribute are:

- `false`: text should not blink.
- `true`: text should blink.

The `style:text-blinking` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The `style:text-blinking` attribute has the data type `boolean` 18.3.3.

## 20.367 `style:text-combine`

The `style:text-combine` attribute specifies whether to combine characters so that they are displayed within two lines.

The defined values for the `style:text-combine` attribute are:

- `letters`: Display text in Kumimoji. Up to five (5) characters are combined within two lines and are displayed with a reduced size in a single wide-cell character. Additional characters are displayed as normal text.
- `lines`: Displays text in Warichu. All characters with the `style:text-combine` attribute that immediately follow each other are displayed within two lines of approximately the same length. A line break may occur between any two characters to meet this constraint.
- `none`: characters should not be combined.

The `style:text-combine` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The values of the `style:text-combine` attribute are `none`, `letters` or `lines`.

## 20.368 style:text-combine-start-char

The `style:text-combine-start-char` attribute specifies the start character that is displayed before a portion of text whose `style:text-combine` 20.367 attribute has a value of lines.

The `style:text-combine-start-char` attribute is usable with the following element:  
`<style:text-properties>` 16.29.29.

The `style:text-combine-start-char` attribute has the data type `character` 18.3.7.

## 20.369 style:text-combine-end-char

The `style:text-combine-end-char` attribute specifies the end character that is displayed after a portion of text whose `style:text-combine` 20.367 attribute has a value of lines.

The `style:text-combine-end-char` attribute is usable with the following element:  
`<style:text-properties>` 16.29.29.

The `style:text-combine-end-char` attribute has the data type `character` 18.3.7.

## 20.370 style:text-emphasize

The `style:text-emphasize` attribute specifies emphasis in a text composed of [UNICODE] characters whose script type is `asian`. 20.358

The value of this attribute consists of two white space-separated values.

The first value represents the style to use for emphasis.

The second value represents the position of the emphasis and it can be `above` or `below`. If the first value is `none`, the second value can be omitted.

The defined values for the `style:text-emphasize` attribute are:

- `accent`: calligraphic accent strokes.
- `circle`: hollow circles.
- `disc`: filled circles.
- `dot`: calligraphic dot.
- `none`: no emphasis marks.

The `style:text-emphasize` attribute is usable with the following element:  
`<style:text-properties>` 16.29.29.

The values of the `style:text-emphasize` attribute are `none`, or two white space separated values. The first of these values is one of: `none`, `accent`, `dot`, `circle` or `disc`. The second of these values is one of: `above` or `below`.

## 20.371 style:text-line-through-color

The `style:text-line-through-color` attribute specifies the color that is used for line-through text.

The defined values for the `style:text-line-through-color` attribute are:

- `font-color`: current text color is used for underlining.
- a value of type `color` 18.3.9

The `style:text-line-through-color` attribute is usable with the following element:  
`<style:text-properties>` 16.29.29.

The values of the `style:text-line-through-color` attribute are `font-color` or a value of type `color` 18.3.9.

## 20.372 `style:text-line-through-mode`

The `style:text-line-through-mode` attribute specifies whether lining through is applied to words only or to portions of text.

The defined values for the `style:text-line-through-mode` attribute are:

- `continuous`: lining is applied to words and separating spaces.
- `skip-white-space`: lining is not applied to spaces between words.

The `style:text-line-through-mode` attribute is usable with the following element:  
`<style:text-properties>` 16.29.29.

The values of the `style:text-line-through-mode` attribute are `continuous` or `skip-white-space`.

## 20.373 `style:text-line-through-style`

The `style:text-line-through-style` attribute specifies a style for rendering a line-through text.

The defined values for the `style:text-line-through-style` attribute are:

- `none`: text has no line through it.
- `dash`: text has a dashed line through it.
- `dot-dash`: text has a line whose repeating pattern is a dot followed by a dash through it.
- `dot-dot-dash`: text has a line whose repeating pattern is two dots followed by a dash through it.
- `dotted`: text has a dotted line through it.
- `long-dash`: text has a dashed line whose dashes are longer than the ones from the dashed line for value dash through it.
- `solid`: text has a solid line through it.
- `wave`: text has a wavy line through it.

**Note:** The definitions of the values of the `style:text-line-through-style` attribute are based on the text decoration style 'text-line-through-style' from [CSS3Text], §9.2.

The `style:text-line-through-style` attribute is usable with the following element:  
`<style:text-properties>` 16.29.29.

The values of the `style:text-line-through-style` attribute are `none`, `solid`, `dotted`, `dash`, `long-dash`, `dot-dash`, `dot-dot-dash` or `wave`.

## 20.374 `style:text-line-through-text`

The `style:text-line-through-text` attribute specifies a text that is used for line-through. The attribute will be evaluated only if the value of `style:text-line-through-style` 20.373 attribute is different than `none`.

If the attribute value is not empty, the attribute value string is used for line-through instead of the line style that has been specified by the `style:text-line-through-style` attribute. Consumers that do not support line-through with text should ignore the attribute, and should use the line style specified by the `style:text-line-through-style` attribute.

Consumers that support line-through with single characters only, should use the first character of the value for line-through, if the `style:text-line-through-text` attribute value has more than one character. Consumers that support line-through with specific characters only (like "x" or "/" (U+002F, SOLIDUS) should use one of these characters if the attribute specifies characters that are not supported.

The `style:text-line-through-text` attribute is usable with the following element:  
`<style:text-properties>` 16.29.29.

The `style:text-line-through-text` attribute has the data type `string` 18.2.

## 20.375 `style:text-line-through-text-style`

The `style:text-line-through-text-style` specifies a text style that is applied to text-line-through characters. It is not applied to line-through lines. If the attribute appears in an automatic style, it may reference either an automatic text style or a common style. If the attribute appears in a common style, it may reference a common style only.

The `style:text-line-through-text-style` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The `style:text-line-through-text-style` attribute has the data type `styleNameRef` 18.3.33.

## 20.376 `style:text-line-through-type`

The `style:text-line-through-type` attribute specifies whether text is lined through, and if so, whether a single or double line will be used.

The defined values for the `style:text-line-through-type` attribute are:

- `double`: a double line should be used for a line-through text.
- `none`: deprecated.
- `single`: a single line should be used for a line-through text.

Every occurrence of the `style:text-line-through-type` attribute should be accompanied by an occurrence of the `style:text-line-through-style` 20.373 attribute on the same element. There should not be an occurrence of the `style:text-line-through-type` attribute if the value of the `style:text-line-through-style` attribute is `none`.

The `style:text-line-through-type` attribute is usable with the following element:  
`<style:text-properties>` 16.29.29.

The values of the `style:text-line-through-type` attribute are `none`, `single` or `double`.

## 20.377 `style:text-line-through-width`

The `style:text-line-through-width` attribute specifies the width of a line-through line. The value `bold` specifies a line width that is calculated from the font sizes like an `auto` width, but is wider than an `auto` width.

The defined values for the `style:text-line-through-width` attribute are:

- `auto`: the width of a line-through should be calculated from the font size of the text where the line-through will appear.
- `bold`: the width of a line-through should be calculated from the font size of the text where the line-through will appear but is wider than for the value of `auto`.
- a value of type `percent` 18.3.24

- a value of type `positiveInteger` 18.2
- a value of type `positiveLength` 18.3.27

The line-through text styles referenced by the values `dash`, `medium`, `thick` and `thin`, are implementation-defined. `Thin` shall be smaller width than `medium` and `medium` shall be a smaller width than `thick`.

The `style:text-line-through-width` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The values of the `style:text-line-through-width` attribute are `auto`, `normal`, `bold`, `thin`, `medium`, `thick`, a value of type `positiveInteger` 18.2, a value of type `percent` 18.3.24 or a value of type `positiveLength` 18.3.27.

## 20.378 `style:text-outline`

The `style:text-outline` attribute specifies whether to display an outline of text or the text itself.

The defined values for the `style:text-outline` attribute are:

- `false`: text itself should be displayed.
- `true`: outline of text should be displayed.

The `style:text-outline` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The `style:text-outline` attribute has the data type `boolean` 18.3.3.

## 20.379 `style:text-overline-color`

The `style:text-overline-color` attribute specifies a color that is used to overline text.

The defined values for the `style:text-overline-color` attribute are:

- `font-color`: the current text color is used for overlining.
- a value of type `color`

The `style:text-overline-color` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The values of the `style:text-overline-color` attribute are `font-color` or a value of type `color` 18.3.9.

## 20.380 `style:text-overline-mode`

The `style:text-overline-mode` attribute specifies whether overlining is applied to words only or to portions of text.

The defined values for the `style:text-line-through-mode` attribute are:

- `continuous`: overlining is applied to words and separating spaces.
- `skip-white-space`: overlining is not applied to spaces between words.

The `style:text-overline-mode` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The values of the `style:text-overline-mode` attribute are `continuous` or `skip-white-space`.

## 20.381 style:text-overline-style

The `style:text-overline-style` attribute specifies a style for rendering a line over text.

The defined values for the `style:text-overline-style` attribute are:

- `none`: text has no overlining.
- `dash`: text has a dashed line overlining it.
- `dot-dash`: text has a line whose repeating pattern is a dot followed by a dash overlining it.
- `dot-dot-dash`: text has a line whose repeating pattern is two dots followed by a dash overlining it.
- `dotted`: text has a dotted line overlining it.
- `long-dash`: text has a dashed line whose dashes are longer than the ones from the dashed line for value `dash` overlining it.
- `solid`: text has a solid line overlining it.
- `wave`: text has a wavy line overlining it.

**Note:** The definitions of the values of the `style:text-overline-style` attribute are based on the text decoration style 'text-overline-style' from [CSS3Text], §9.2.

The `style:text-overline-style` attribute is usable with the following element:  
`<style:text-properties>` 16.29.29.

The values of the `style:text-overline-style` attribute are `none`, `solid`, `dotted`, `dash`, `long-dash`, `dot-dash`, `dot-dot-dash` or `wave`.

## 20.382 style:text-overline-type

The `style:text-overline-type` attribute specifies the type of overlining applied to a text.

The defined values for the `style:text-overline-type` attribute are:

- `double`: a double line should be used for overlining text.
- `none`: deprecated.
- `single`: a single line should be used for overlining text.

Every occurrence of the `style:text-overline-type` attribute should be accompanied by an occurrence of the `style:text-overline-style` 20.381 attribute on the same element. There should not be an occurrence of the `style:text-overline-type` attribute if the value of the `style:text-overline-style` attribute is `none`.

The `style:text-overline-type` attribute is usable with the following element:  
`<style:text-properties>` 16.29.29.

The values of the `style:text-overline-type` attribute are `none`, `single` or `double`.

## 20.383 style:text-overline-width

The `style:text-overline-width` attribute specifies the width of an overline. The value `bold` specifies a line width that is calculated from the font sizes like an `auto` width, but is wider than an `auto` width.

The defined values for the `style:text-overline-width` attribute are:

- `auto`: the width of an overline should be calculated from the font size of the text where the overline will appear.

- `bold`: the width of an overline should be calculated from the font size of the text where the overline will appear but is wider than for the value of `auto`.
- a value of type `percent` 18.3.24
- a value of type `positiveInteger` 18.2
- a value of type `positiveLength` 18.3.27

The overline text styles referenced by the values `dash`, `medium`, `thick` and `thin`, are implementation-defined. `Thin` shall be smaller width than `medium` and `medium` shall be a smaller width than `thick`.

The `style:text-overline-width` attribute is usable with the following element:  
`<style:text-properties>` 16.29.29.

The values of the `style:text-overline-width` attribute are `auto`, `normal`, `bold`, `thin`, `medium`, `thick`, a value of type `positiveInteger` 18.2, a value of type `percent` 18.3.24 or a value of type `positiveLength` 18.3.27.

## 20.384 style:text-position

The `style:text-position` attribute specifies whether text is positioned above or below the baseline and to specify the relative font height that is used for this text.

This attribute can have one or two values.

The first value shall be present and specifies the vertical text position as a percentage of the current font height or it takes one of the values `sub` or `super`. Negative percentages or the `sub` value place the text below the baseline. Positive percentages or the `super` value place the text above the baseline. If `sub` or `super` is specified, the consumer chooses an appropriate text position.

The second value may be present and specifies the font height as a percentage of the current font-height. If this value is not specified, an appropriate font height is used.

The `style:text-position` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The `style:text-position` attribute has one or two white space separated values. The first values is of type `percent` 18.3.24, or is one of: `super` or `sub`. The second value is of type `percent` 18.3.24.

## 20.385 style:text-rotation-angle

The `style:text-rotation-angle` attribute specifies an angle to which text is rotated. The value of this attribute can be any integer, which specifies the rotation angle in degrees, or an angle as specified in `angle`. 18.3.1

If more than one character is selected, the entire selection is rotated as a block.

The `style:text-rotation-angle` attribute is usable with the following element:  
`<style:text-properties>` 16.29.29.

The `style:text-rotation-angle` attribute has the data type `angle` 18.3.1.

## 20.386 style:text-rotation-scale

The `style:text-rotation-scale` attribute specifies whether for rotated text the width of the text should be scaled to fit into the current line height or the width of the text should remain fixed, therefore changing the current line height.

The defined values for the `style:text-rotation-scale` attribute are:

- `fixed`: width of text should remain fixed.

- `line-height`: width of text should be scaled to fit the current line height.

The `style:text-rotation-scale` attribute is usable with the following element:  
`<style:text-properties>` 16.29.29.

The values of the `style:text-rotation-scale` attribute are `fixed` or `line-height`.

## 20.387 `style:text-scale`

The `style:text-scale` attribute specifies whether to decrease or increase the width of text by scaling the font width.

The `style:text-scale` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The `style:text-scale` attribute has the data type `percent` 18.3.24.

## 20.388 `style:text-underline-color`

The `style:text-underline-color` attribute specifies a color that is used to underline text.

The defined values for the `style:text-underline-color` attribute are:

- `font-color`: the current text color is used for underlining.
- a value of type `color`: the color to be used for underlining.

The `style:text-underline-color` attribute is usable with the following element:  
`<style:text-properties>` 16.29.29.

The values of the `style:text-underline-color` attribute are `font-color` or a value of type `color` 18.3.9.

## 20.389 `style:text-underline-mode`

The `style:text-underline-mode` attribute specifies whether underlining is applied to words only or to portions of text. If underlining is applied to text portions, the spaces between words and the words are underlined.

The defined values for the `style:text-underline-mode` attribute are:

- `continuous`: underlining is applied to words and separating spaces.
- `skip-white-space`: underlining is not applied to spaces between words.

The `style:text-underline-mode` attribute is usable with the following element:  
`<style:text-properties>` 16.29.29.

The values of the `style:text-underline-mode` attribute are `continuous` or `skip-white-space`.

## 20.390 `style:text-underline-style`

The `style:text-underline-style` attribute specifies a style for underlining text.

The defined values for the `style:text-underline-style` attribute are:

- `none`: text has no underlining.
- `dash`: text has a dashed line underlining it.
- `dot-dash`: text has a line whose repeating pattern is a dot followed by a dash underlining it.

- `dot-dot-dash`: text has a line whose repeating pattern is two dots followed by a dash underlining it.
- `dotted`: text has a dotted line underlining it.
- `long-dash`: text has a dashed line whose dashes are longer than the ones from the dashed line for value `dash` underlining it.
- `solid`: text has a solid line underlining it.
- `wave`: text has a wavy line underlining it.

**Note:** The definitions of the values of the `style:text-underline-style` attribute are based on the text decoration style 'text-decoration-style' from [CSS3Text], §9.2.

The `style:text-underline-style` attribute is usable with the following element:  
`<style:text-properties>` 16.29.29.

The values of the `style:text-underline-style` attribute are `none`, `solid`, `dotted`, `dash`, `long-dash`, `dot-dash`, `dot-dot-dash` or `wave`.

## 20.391 style:text-underline-type

The `style:text-underline-type` attribute specifies the type of underlining applied to a text.

The defined values for the `style:text-underline-type` attribute are:

- `double`: a double line should be used for underlining applied to a text.
- `none`: deprecated.
- `single`: a single line should be used for underlining applied to a text.

Every occurrence of the `style:text-underline-type` attribute should be accompanied by an occurrence of the `style:text-underline-style` 20.390 attribute on the same element. There should not be an occurrence of the `style:text-underline-type` attribute if the value of the `style:text-underline-style` attribute is `none`.

The `style:text-underline-type` attribute is usable with the following element:  
`<style:text-properties>` 16.29.29.

The values of the `style:text-underline-type` attribute are `none`, `single` or `double`.

## 20.392 style:text-underline-width

The `style:text-underline-width` attribute specifies the width of an underline. The value `bold` specifies a line width that is calculated from the font sizes like an `auto` width, but is wider than an `auto` width.

The defined values for the `style:text-underline-width` attribute are:

- `auto`: the width of an underline should be calculated from the font size of the text where the underline will appear.
- `bold`: the width of an underline should be calculated from the font size of the text where the underline will appear but is wider than for the value of `auto`.
- a value of type `percent` 18.3.24.
- a value of type `positiveInteger` 18.2.
- a value of type `positiveLength` 18.3.27.

The underline text styles referenced by the values `dash`, `medium`, `thick` and `thin`, are implementation-defined. `Thin` shall be smaller width than `medium` and `medium` shall be a smaller width than `thick`.

The `style:text-underline-width` attribute is usable with the following element:  
`<style:text-properties>` 16.29.29.

The values of the `style:text-underline-width` attribute are `auto`, `normal`, `bold`, `thin`, `medium`, `thick`, a value of type `positiveInteger` 18.2, a value of type `percent` 18.3.24 or a value of type `positiveLength` 18.3.27.

## 20.393 style:use-optimal-column-width

The `style:use-optimal-column-width` attribute specifies that a column width should be recalculated automatically if content in the column changes.

The defined values for the `style:use-optimal-column-width` attribute are:

- `false`: column width should not be recalculated automatically if content in the column changes.
- `true`: column width should be recalculated automatically if content in the column changes.

The `style:use-optimal-column-width` attribute is usable with the following element:  
`<style:table-column-properties>` 17.16.

The `style:use-optimal-column-width` attribute has the data type `boolean` 18.3.3.

## 20.394 style:use-optimal-row-height

The `style:use-optimal-row-height` attribute specifies that a row height should be recalculated automatically if content in the row changes.

The defined values for the `style:use-optimal-row-height` attribute are:

- `false`: row height should not be recalculated automatically if content in the row changes.
- `true`: row height should be recalculated automatically if content in the row changes.

The `style:use-optimal-row-height` attribute is usable with the following element:  
`<style:table-row-properties>` 17.17.

The `style:use-optimal-row-height` attribute has the data type `boolean` 18.3.3.

## 20.395 style:use-window-font-color

The `style:use-window-font-color` attribute specifies whether the window foreground color should be used as the foreground color for a light background color and white for a dark background color. The determination of light or dark color is implementation-defined.

The defined values for the `style:use-window-font-color` attribute are:

- `false`: the foreground color is specified by the `fo:color` attribute.
- `true`: windows foreground color should be used as the foreground color for a light background color and white for a dark background color.

The `style:use-window-font-color` attribute is usable with the following element:  
`<style:text-properties>` 16.29.29.

The `style:use-window-font-color` attribute has the data type `boolean` 18.3.3.

## 20.396 style:vertical-align

### 20.396.1 <style:paragraph-properties>

The `style:vertical-align` attribute specifies the vertical position of a character. By default characters are aligned according to their baseline.

The defined values for the `style:vertical-align` attribute are:

- `auto`: automatically, which sets the vertical alignment to suit the text rotation. Text that is rotated 0 or 90 degrees is aligned to the baseline, while text that is rotated 270 degrees is aligned to the center of the line.
- `baseline`: to the baseline of the character.
- `bottom`: to the bottom of the line.
- `middle`: to the center of the line.
- `top`: to the top of the line.

The `style:vertical-align` attribute is usable with the following element:  
`<style:paragraph-properties>` 17.6.

The values of the `style:vertical-align` attribute are `top`, `middle`, `bottom`, `auto` or `baseline`.

### 20.396.2 <style:table-cell-properties>

The `style:vertical-align` attribute specifies the vertical alignment of text in a table cell. The options for the vertical alignment attribute are as follows:

The defined values for the `style:vertical-align` attribute are:

- `automatic`: consumer determines how to align the text.
- `bottom`: aligns text vertically with the bottom of the cell.
- `middle`: aligns text vertically with the middle of the cell.
- `top`: aligns text vertically with the top of the cell.

The `style:vertical-align` attribute is usable with the following element:  
`<style:table-cell-properties>` 17.18.

The values of the `style:vertical-align` attribute are `top`, `middle`, `bottom` or `automatic`.

## 20.397 style:vertical-pos

The `style:vertical-pos` attribute specifies the vertical alignment of a frame in relation to an area or line specified by the `style:vertical-rel` [20.398] attribute.

The defined values for the `style:vertical-pos` attribute are:

- `below`: the top corner of the frame is positioned below the reference area or line.
- `bottom`: the bottom of the frame is aligned with the bottom edge of the reference area or line.
- `from-top`: the `svg:y` attribute associated with the frame element specifies the vertical position of the top edge of the frame relative to a reference line. Otherwise, the `svg:y` attribute is ignored for text documents.

In case the attribute `style:vertical-rel` specifies an area, the top edge of the area is used as reference line.

- `middle`: the middle of the frame is aligned with the middle of the reference area or line.
- `top`: the top of the frame is aligned with the top edge of the reference area or line.

The `style:vertical-pos` attribute is usable with the following elements:  
`<style:graphic-properties>` 17.21 and `<style:list-level-properties>` 17.19.  
 The values of the `style:vertical-pos` attribute are `top`, `middle`, `bottom`, `from-top` or `below`.

## 20.398 `style:vertical-rel`

The `style:vertical-rel` attribute specifies the area or line against which the vertical position of a frame is positioned.

The defined values for the `style:vertical-rel` attribute are:

`baseline`: vertical position of frame positioned relative to a baseline.

`char`: vertical position of frame positioned relative to a character.

`frame`: vertical position of frame positioned relative to a frame area.

`frame-content`: vertical position of frame positioned relative to frame content area.

`line`: vertical position of frame positioned relative to a line.

`page`: vertical position of frame positioned relative to a page area.

`page-content`: vertical position of frame positioned relative to page content area.

`page-content-bottom`: vertical position of frame positioned relative to the bottom edge of page content area.

`page-content-top`: vertical position of frame positioned relative to the top edge of page content area.

`paragraph`: vertical position of frame positioned relative to a paragraph area.

`paragraph-content`: vertical position of frame positioned relative to paragraph content area.

`text`: vertical position of frame positioned relative to text.

Table 24 displays the defined values of `style:vertical-rel` and `text:anchor-type` attributes. Value combinations that can appear in a document are marked with an X.

*Table 24 - Vertical Relation Values*

<b>Value of <code>style:vertical-rel</code></b>	<b>Value of <code>text:anchor-type</code></b>				
	<b><code>page</code></b>	<b><code>frame</code></b>	<b><code>paragraph</code></b>	<b><code>char</code></b>	<b><code>as-char</code></b>
<code>baseline</code>					X
<code>char</code>				X	X
<code>frame</code>		X			
<code>frame-content</code>		X			
<code>line</code>					X
<code>page</code>	X		X	X	
<code>page-content</code>	X		X	X	
<code>page-content-bottom</code>	X		X	X	

<b>Value of style:vertical-rel</b>	<b>Value of text:anchor-type</b>				
	X		X	X	
page-content-top	X		X	X	
paragraph			X	X	
paragraph-content			X	X	
text					X

The `style:vertical-rel` attribute is usable with the following elements: `<style:graphic-properties>` 17.21 and `<style:list-level-properties>` 17.19.

The values of the `style:vertical-rel` attribute are page, page-content, page-content-bottom, page-content-top, frame, frame-content, paragraph, paragraph-content, char, line, baseline or text.

## 20.399 style:width

The `style:width` attribute specifies the fixed width of a table. Every table shall have a fixed width.

The `style:width` attribute is usable with the following element: `<style:table-properties>` 17.15.

The `style:width` attribute has the data type `positiveLength` 18.3.27.

## 20.400 style:wrap

The `style:wrap` attribute specifies how text is displayed around a frame or graphic object.

The defined values for the `style:wrap` attribute are:

- `biggest`: text may wrap around the shape where the difference to the left or right page or column border is largest.
- `dynamic`: text may wrap around both sides of the shape. The space for wrapping is set by the `style:wrap-dynamic-threshold` 20.403 attribute.
- `left`: text wraps around the left side of the shape.
- `none`: text does not wrap around the shape.
- `parallel`: text wraps around both sides of the shape.
- `right`: text wraps around the right side of the shape.
- `run-through`: text runs through the shape.

The `style:wrap` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The values of the `style:wrap` attribute are `none`, `left`, `right`, `parallel`, `dynamic`, `run-through` or `biggest`.

## 20.401 style:wrap-contour

The `style:wrap-contour` attribute specifies whether text wraps around an object or the bounding box of the object.

The defined values for the `style:wrap-contour` attribute are:

- `false`: text wraps around the bounding box.
- `true`: text wraps around the object.

This is called contour wrapping.

The `style:wrap-contour` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The `style:wrap-contour` attribute has the data type `boolean` 18.3.3.

## 20.402 style:wrap-contour-mode

The `style:wrap-contour-mode` attribute specifies the nature of the wrapping of text around the contour of a shape.

This attribute shall be evaluated only by frames/drawing shapes or styles with non-empty `style:wrap` 20.400 and `style:wrap-contour` 20.401 attributes.

The defined values for the `style:wrap-contour-mode` attribute are:

- `full`: text wraps around the shape and fills any spaces and indentations in the shape.
- `outside`: text wraps around the general area to the left and right of the shape.

The `style:wrap-contour-mode` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The values of the `style:wrap-contour-mode` attribute are `full` or `outside`.

## 20.403 style:wrap-dynamic-threshold

The `style:wrap-dynamic-threshold` attribute specifies the minimum distance between the page or column border and the object for which wrapping will be enabled. It is evaluated only if a `style:wrap` 20.400 attribute, with the value `dynamic`, appears on the same `<style:graphic-properties>` 17.21 element.

The `style:wrap-dynamic-threshold` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The `style:wrap-dynamic-threshold` attribute has the data type `nonNegativeLength` 18.3.21.

## 20.404 style:writing-mode

### 20.404.1 General

The `style:writing-mode` attribute controls the direction in which characters are ordered within a line (inline direction), the direction in which lines are stacked (block direction) and the glyph orientation of individual characters. Furthermore it controls the order and display direction of columns in layout objects like section, table or page and in text boxes (called here layout direction).

**Note:** In context of these sections, the term “character” is to be interpreted to mean a text character as perceived by the user, or a glyph as defined in Unicode.

The orientation of images and shapes is not affected by the writing mode.

The defined values for the `style:writing-mode` attribute are `lr-tb`, `lr`, `rl-tb`, `rl`, `tb-rl`, `tb-lr`, `sideways-rl`, `sideways-lr` and `page`.

- `lr-tb`: see §7.27.7 of [\[XSL\]](#).

Inline direction is left-to-right, block direction is top-to-bottom, all characters are rendered upright.

Columns are stacked left-to-right, origin of a table is top-left.

**Note:** This value is common for `latin` scripts, such as used for English or Spanish texts, for example.

- `lr`: short name for value `lr-tb`.
- `rl-tb`: see §7.27.7 of [\[XSL\]](#).

Inline direction is right-to-left, block direction is top-to-bottom, all characters are rendered upright.

Columns are stacked right-to-left, origin of a table is top-right.

**Note:** This value is common for `complex` scripts, such as used for Arabic or Hebrew texts, for example.

- `rl`: short name for value `rl-tb`.
- `tb-rl`: see §7.27.7 of [\[XSL\]](#).

Inline direction is top-to-bottom, block direction is right-to-left. Characters of `asian` scripts are rendered upright, characters of other scripts are rotated 90deg clockwise.

Columns are stacked top-to-bottom, origin of a table is top-right.

**Note:** This value is used for vertical Korean or Japanese texts, for example.

- `tb`: short name for value `tb-rl`.
- `tb-lr`: see Appendix A.1. of [\[\[XSL\]\]](#).

Inline direction is top-to-bottom, block direction is left-to-right. Characters of `asian` scripts are rendered upright, characters of other scripts are rotated 90deg clockwise.

Columns are stacked top-to-bottom, origin of a table is top-left.

**Note:** This value is used for the traditional Mongolian script, for example.

- `sideways-rl`: see §3.2 of [\[CSS4WritingModes\]](#)
- Inline direction is top-to-bottom, block direction is right-to-left. All characters are rotated 90deg clockwise.

Columns are stacked top-to-bottom, origin of a table is top-right.

**Note:** This writing mode is not intended for specific languages, but can be used for table headers in Arabic or Hebrew texts or for 90deg clockwise rotation of tables, for example.

- `sideways-lr`: see §3.2 of [\[CSS4WritingModes\]](#)
- Inline direction is bottom-to-top, block direction is left-to-right, all characters are rotated 90deg counter-clockwise.

Columns are stacked bottom-to-top, origin of a table is bottom-left.

**Note:** This writing mode is not intended for specific languages, but can be used for table headers in western languages or for 90deg counter-clockwise rotation of tables, for example.

- `page`: This value specifies that the writing-mode is inherited from the closest layout object in which the element is contained or from the anchor object of the element. Details are specified in the following sections.

**Note:** The `style:writing-mode` attribute can also be used in chart styles. However, it does not belong to the `<style:chart-properties>` element, but to the `<style:graphic-properties>` element or to the `<style:paragraph-properties>` element and therefore 'chart-properties' does not have its own section here.

#### 20.404.2 `<style:graphic-properties>`

The `style:writing-mode` attribute of a `<style:graphic-properties>` element specifies the layout direction of columns within text boxes that have the graphic style assigned.

If the attribute value is `page` the layout direction of columns within text boxes that are anchored to a page is inherited from the layout direction of the page on which the text box is displayed. For text boxes that have a different anchor type, the layout direction is inherited from the paragraph direction of the paragraph or object that contains the text box.

If a graphic style is evaluated for the `style:writing-mode` property of a paragraph as specified in section 16.2, not only the `<style:paragraph-properties>` element but also the `<style:graphic-properties>` element is evaluated. If the `style:writing-mode` attribute is present in both elements, it shall have the same value.

The `style:writing-mode` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The values of the `style:writing-mode` attribute are `lr-tb`, `lr`, `rl-tb`, `rl`, `tb-rl`, `tb`, `tb-lr`, `sideways-rl`, `sideways-lr` or `page`.

#### 20.404.3 `<style:page-layout-properties>`

The `style:writing-mode` attribute of a `<style:page-layout-properties>` element specifies the layout direction of page columns for pages that are formatted using the `page` style. It further determines the binding edge and progress direction of books, see `style:page-usage` 19.509 and it determines the writing mode in the inheritance chain of objects, which have a `style:writing-mode` attribute with value `page` applied.

The location of the header area at top of a page and the footer area at bottom of a page is not affected by the writing mode of the page.

The value `page` shall not be used for a `style:writing-mode` attribute of a `<style:page-layout-properties>` element..

The `style:writing-mode` attribute is usable with the following element: `<style:page-layout-properties>` 17.2.

The values of the `style:writing-mode` attribute are `lr-tb`, `lr`, `rl-tb`, `rl`, `tb-rl`, `tb`, `tb-lr`, `sideways-rl`, `sideways-lr` or `page`.

#### 20.404.4 `<style:paragraph-properties>`

The `style:writing-mode` attribute of a `<style:paragraph-properties>` element specifies the inline direction, block direction and character orientation for all paragraphs that have the paragraph style assigned.

The `style:writing-mode` attribute specifies for the paragraph the base direction required by the BiDi Algorithm, see BD5 of [\[\[UAX9\]\]](#).

**Note:** The inline direction determines the location of "start" and "end" of a paragraph. For paragraphs that are contained in lists, it thus specifies where

the list numbers and bullets are displayed, for writing mode `rl-tb` on the right side, for example.

If a `<style:paragraph-property>` element is child element of a graphic style that is assigned to a text box, then it specifies the layout direction of columns within the text box as well. In case the style `writing-mode` attribute is present in the `<style:paragraph-properties>` child element and in the `<style:graphic-properties>` child element of the same graphic style, their values shall be identical.

The value `page` specifies that the actual value is inherited from the layout direction of the closest layout object in which the paragraph is contained, and which has a layout direction other than `page`. For a paragraph element, which is direct child element of a shape, the closest layout object is the shape itself.

**Note:** If a paragraph style does not contain a `style:writing-mode` attribute, then the hierarchical style model as specified in section 16.2 is used to determine a value for the writing mode of the element that has the paragraph style applied. That is different from an attribute with value `page`.

The `style:writing-mode` attribute is usable with the following element:  
`<style:paragraph-properties>` 17.6.

The values of the `style:writing-mode` attribute are `lr-tb`, `lr`, `rl-tb`, `rl`, `tb-rl`, `tb`, `tb-lr`, `sideways-rl`, `sideways-lr` or `page`.

## 20.404.5 `<style:section-properties>`

The `style:writing-mode` attribute of a `<style:section-properties>` element specifies the layout direction of section columns for text sections that have the section style assigned.

The `style:writing-mode` attribute is usable with the following element:  
`<style:section-properties>` 17.11.

The values of the `style:writing-mode` attribute are `lr-tb`, `lr`, `rl-tb`, `rl`, `tb-rl`, `tb`, `tb-lr`, `sideways-rl`, `sideways-lr` or `page`.

## 20.404.6 `<style:table-cell-properties>`

The `style:writing-mode` attribute of a `<style:table-cell-properties>` element specifies a layout direction for any descendant element of the cell that has a style assigned with writing mode `page` and for which the table cell is the closest containing layout object on which the layout direction is specified.

**Note:** OpenDocument further has the table cell style attribute `style:direction` 20.263 to specify that characters are vertically stacked but not rotated, and it further has the table cell style attribute `style:glyph-orientation-vertical` 20.297 to specify, that no characters in a vertical writing mode are rotated.

The `style:writing-mode` attribute is usable with the following element: `<style:table-cell-properties>` 17.18.

The values of the `style:writing-mode` attribute are `lr-tb`, `lr`, `rl-tb`, `rl`, `tb-rl`, `tb`, `tb-lr`, `sideways-rl`, `sideways-lr` or `page`.

## 20.404.7 `<style:table-properties>`

The `style:writing-mode` attribute of a `<style:table-properties>` element specifies the layout direction of table columns for tables that have the table style assigned.

If the attribute value is `page` the layout direction is inherited from the layout direction of the closest layout object in which the table is contained, and which has a layout direction other than `page`.

The `style:writing-mode` attribute is usable with the following element: `<style:table-properties>` 17.15.

The values of the `style:writing-mode` attribute are `lr-tb`, `lr`, `rl-tb`, `rl`, `tb-rl`, `tb`, `tb-lr`, `sideways-rl`, `sideways-lr` or `page`.

## 20.405 `style:writing-mode-automatic`

The `style:writing-mode-automatic` attribute specifies whether a consumer may recalculate the writing mode of a paragraph based on its content whenever the content is edited.

The writing-mode should be specified in a `style:writing-mode` attribute.

If the `fo:text-align` with value `start`, text alignment can be adapted to the writing mode.

The defined values for the `style:writing-mode-automatic` attribute are:

- `false`: consumers should not recalculate writing mode of a paragraph whenever its content is edited.
- `true`: consumers should recalculate writing mode of a paragraph whenever its content is edited.

The `style:writing-mode-automatic` attribute is usable with the following element: `<style:paragraph-properties>` 17.6.

The `style:writing-mode-automatic` attribute has the data type `boolean` 18.3.3.

## 20.406 `svg:fill-rule`

See §11.3 of [SVG].

OpenDocument does not support the value `inherit`.

The `svg:fill-rule` attribute is usable with the following elements: `<style:drawing-page-properties>` 17.25 and `<style:graphic-properties>` 17.21.

The values of the `svg:fill-rule` attribute are `nonzero` or `evenodd`.

## 20.407 `svg:height`

### 20.407.1 `<style:graphic-properties>`

The `svg:height` attribute specifies a default height for new frames that are created using the `graphics` style. See 19.543.

The `svg:height` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The `svg:height` attribute has the data type `length` 18.3.19.

### 20.407.2 `<style:header-footer-properties>`

The `svg:height` attribute specifies the height of a header or footer.

The `svg:height` attribute is usable with the following element: `<style:header-footer-properties>` 17.5.

The `svg:height` attribute has the data type `length` 18.3.19.

## 20.408 `svg:stroke-color`

The `svg:stroke-color` attribute specifies the color of a stroke.

The `svg:stroke-color` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The `svg:stroke-color` attribute has the data type `color` 18.3.9.

## 20.409 `svg:stroke-opacity`

The `svg:stroke-opacity` attribute specifies the opacity of a stroke. The value of this attribute can be a number between 0 (transparent) and 1 (opaque) or a percentage value in the range 0% to 100%.

The `svg:stroke-opacity` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The values of the `svg:stroke-opacity` attribute are a value of type `double` 18.2 in the range [0,1] or a value of type `zeroToHundredPercent` 18.3.42.

## 20.410 `svg:stroke-width`

The `svg:stroke-width` attribute specifies the width of a stroke.

The `svg:stroke-width` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The `svg:stroke-width` attribute has the data type `length` 18.3.19.

## 20.411 `svg:x`

The `svg:x` attribute specifies a default horizontal position for new frames that are created using the `graphics` style. See 19.577.

The `svg:x` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The `svg:x` attribute has the data type `coordinate` 18.3.10.

## 20.412 `svg:y`

### 20.412.1 `<style:graphic-properties>`

The `svg:y` attribute specifies a default vertical position for new frames that are created using the `graphics` style. See 19.581.

The `svg:y` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The `svg:y` attribute has the data type `coordinate` 18.3.10.

### 20.412.2 `<style:list-level-properties>`

The `svg:y` attribute specifies the vertical position of a bullet image.

The `svg:y` attribute is usable with the following element: `<style:list-level-properties>` 17.19.

The `svg:y` attribute has the data type `coordinate` 18.3.10.

## 20.413 `svg:width`

The `svg:width` attribute specifies a default width for new frames that are created using the `graphics` style. See 19.575.

The `svg:width` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The `svg:width` attribute has the data type `length` 18.3.19.

## 20.414 `table:align`

The `table:align` attribute specifies the horizontal alignment of a table.

The defined values for the `table:align` attribute are:

- `center`: table aligns to the center between left and right margins.
- `left`: table aligns to the left margin.
- `margins`: table fills all the space between the left and right margins.
- `right`: table aligns to the right margin.

Consumers that do not support the `margins` value, may treat this value as `left`.

The `table:align` attribute is usable with the following element: `<style:table-properties>` 17.15.

The values of the `table:align` attribute are `left`, `center`, `right` or `margins`.

## 20.415 `table:border-model`

The `table:border-model` attribute specifies what border model to use when creating a table with a border.

The defined values for the `table:border-model` attribute are:

- `collapsing`: when two adjacent cells have different borders, the wider border appears as the border between the cells. Each cell receives half of the width of the border.
- `separating`: borders appear within the cell that specifies the border.

In OpenDocument, a row height or column width includes any space required to display borders or padding. This means that, while the width and height of the content area is less than the column width and row height, the sum of the widths of all columns is equal to the total width of the table.

The `table:border-model` attribute is usable with the following element: `<style:table-properties>` 17.15.

The values of the `table:border-model` attribute are `collapsing` or `separating`.

## 20.416 `table:display`

The `table:display` attribute specifies whether a table is displayed.

The defined values for the `table:display` attribute are:

- `false`: table should not be displayed.
- `true`: table should be displayed.

The `table:display` attribute is usable with the following element: `<style:table-properties>` 17.15.

The `table:display` attribute has the data type `boolean` 18.3.3.

## 20.417 text:anchor-page-number

The `text:anchor-page-number` attribute specifies the physical page number of an anchor if the drawing object is bound to a page within a text document.

The `text:anchor-page-number` attribute is usable with the following element:

`<style:graphic-properties>` 17.21.

The `text:anchor-page-number` attribute has the data type `positiveInteger` 18.2.

## 20.418 text:anchor-type

The `text:anchor-type` attribute specifies how a drawing shape is bound to a text document. The anchor position is the point at which a drawing shape is bound to a text document.

The same attribute is available as an element attribute `text:anchor-type` 19.759.

The defined values for the `text:anchor-type` attribute are shown in Table 25.

*Table 25 - Text anchor positions*

<b><i>If the value of the <code>text:anchor-type</code> attribute is ...</i></b>	<b><i>The anchor position is...</i></b>	<b><i>The drawing shape element shall be located in the file text ...</i></b>	<b><i>Notes</i></b>
as-char	There is no anchor position. The drawing shape behaves like a character.	At the position where the character appears in the document.	
char	The character after the drawing shape element.	Just before the character.	
frame	The parent text box that the current drawing shape element is contained in.	In the element representing the text box to which the drawing object is bound.  If an image is bound to a text box, the image element is located in the text box element.	
page	The page that has the same physical page number as the value of the <code>text:anchor-page-number</code> attribute that is attached to the drawing shape element. If no <code>text:anchor-page-number</code> attribute is given, the anchor position is the page at which the character behind the drawing object element appears.	Either <ul style="list-style-type: none"><li>At the start of the document body, outside any paragraph or frame, provided a <code>text:anchor-page-number</code> attribute is given.</li></ul> Or <ul style="list-style-type: none"><li>Inside any paragraph element that is not contained in a header, footer, footnote, or text box, if a <code>text:anchor-page-number</code> attribute is not given.</li></ul>	The physical page number is the number assigned to the page if all pages in the document are counted starting with page 1.

If the value of the <code>text:anchor-type</code> attribute is ...	The anchor position is...	The drawing shape element shall be located in the file text ...	Notes
paragraph	The paragraph that the current drawing shape element is contained in.	At the start of the paragraph element.	

The `text:anchor-type` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The values of the `text:anchor-type` attribute are `page`, `frame`, `paragraph`, `char` or `as-char`.

## 20.419 `text:animation`

The `text:animation` attribute specifies the type of animation that is used for a text.

The defined values for the `text:animation` attribute are:

- `alternate`: scrolls the text from one side to another and back.
- `none`: disables the text animation.
- `scroll`: scrolls the text from one side to another.
- `slide`: scrolls the text from one side to the original text position and stops there.

The `text:animation` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The values of the `text:animation` attribute are `none`, `scroll`, `alternate` or `slide`.

## 20.420 `text:animation-delay`

The `text:animation-delay` attribute specifies a delay before an animation is started.

The `text:animation-delay` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The `text:animation-delay` attribute has the data type `duration` 18.2.

## 20.421 `text:animation-direction`

The `text:animation-direction` attribute specifies the scroll direction of animated text.

The defined values for the `text:animation-direction` attribute are:

- `down`: animated text scrolls down.
- `left`: animated text scrolls left.
- `right`: animated text scrolls right.
- `up`: animated text scrolls up.

The `text:animation-direction` attribute is usable with the following element: `<style:graphic-properties>` 17.21.

The values of the `text:animation-direction` attribute are `left`, `right`, `up` or `down`.

## 20.422 text:animation-repeat

The `text:animation-repeat` attribute specifies the number of times an animation is repeated. If the value of the attribute is 0, the animation is repeated indefinitely.

The `text:animation-repeat` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The `text:animation-repeat` attribute has the data type `nonNegativeInteger` 18.2.

## 20.423 text:animation-start-inside

The `text:animation-start-inside` attribute specifies if a text animation starts inside or outside a shape.

The defined values for the `text:animation-start-inside` attribute are:

- `false`: text starts its animation just outside the shape's bounding rectangle.
- `true`: text starts its animation inside the shape.

The `text:animation-start-inside` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The `text:animation-start-inside` attribute has the data type `boolean` 18.3.3.

## 20.424 text:animation-steps

The `text:animation-steps` attribute specifies the distance by which text is moved within a scrolling step.

The `text:animation-steps` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The `text:animation-steps` attribute has the data type `length` 18.3.19.

## 20.425 text:animation-stop-inside

The `text:animation-stop-inside` attribute specifies if a text animation stops inside or outside a shape.

The defined values for the `text:animation-stop-inside` attribute are:

- `false`: text stops its animation outside the shape's bounding rectangle.
- `true`: text stops its animation when it is inside the shape.

The `text:animation-stop-inside` attribute is usable with the following element:  
`<style:graphic-properties>` 17.21.

The `text:animation-stop-inside` attribute has the data type `boolean` 18.3.3.

## 20.426 text:condition

The `text:condition` attribute specifies the display of text.

The defined value of the `text:condition` attribute is `none`, which means text is hidden.

The `text:condition` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

The only value of the `text:condition` attribute is `none`.

## 20.427 text:display

The `text:display` attribute specifies whether text is hidden.

The defined values for the `text:display` attribute are:

- `condition`: text is hidden under the condition specified in the `text:condition` 20.426 attribute.
- `none`: text is hidden unconditionally.
- `true`: text is displayed. This is the default setting.

The `text:display` attribute is usable with the following element: `<style:text-properties>` 16.29.29.

## 20.428 text:dont-balance-text-columns

The `text:dont-balance-text-columns` attribute specifies whether the text column content should be evenly distributed over all text columns or not.

The defined values for the `text:dont-balance-text-columns` attribute are:

- `false`: text column content should not be evenly distributed over all text columns in a set of columns.
- `true`: text column content should be evenly distributed over all text columns in a set of columns.

The `text:dont-balance-text-columns` attribute is usable with the following element: `<style:section-properties>` 17.11.

The `text:dont-balance-text-columns` attribute has the data type `boolean` 18.3.3.

## 20.429 text:line-break

The `text:line-break` attribute specifies whether word wrapping is allowed for axis labels.

This attribute is evaluated for a chart style that is applied to a `<chart:axis>` 11.9 element.

The defined values for the `text:line-break` attribute are:

- `false`: word wrapping should not be allowed for axis labels.
- `true`: word wrapping should be allowed for axis labels.

The `text:line-break` attribute is usable with the following element: `<style:chart-properties>` 17.22.

The `text:line-break` attribute has the data type `boolean` 18.3.3.

## 20.430 text:line-number

The `text:line-number` attribute specifies a new start value for line numbering, if a `text:number-lines` 20.434 attribute, with the value `true`, appears on the same `<style:paragraph-properties>` 17.6 element. Otherwise, this attribute shall be ignored.

The `text:line-number` attribute is usable with the following element: `<style:paragraph-properties>` 17.6.

The `text:line-number` attribute has the data type `nonNegativeInteger` 18.2.

## 20.431 text:list-level-position-and-space-mode

The `text:list-level-position-and-space-mode` attribute specifies how the position and spacing of the list labels (numbers or bullets) is defined.

The defined values for the `text:list-level-position-and-space-mode` attribute are:

- `label-alignment`: The `<style:list-level-label-alignment>` 17.20 element and the `fo:text-align` 20.223 attribute are used to define the position and spacing of the list label and the list item. The values of the attributes for `text:space-before` 20.435, `text:min-label-width` 20.433 and `text:min-label-distance` 20.432 are assumed to be 0.
- `label-width-and-position`: The `text:space-before`, `text:min-label-width`, `text:min-label-distance` and `fo:text-align` attributes are used to define the position and spacing of the list label and the list item.

The default value of the `text:list-level-position-and-space-mode` attribute is `label-width-and-position`.

The `text:list-level-position-and-space-mode` attribute is usable with the following element: `<style:list-level-properties>` 17.19.

The values of the `text:list-level-position-and-space-mode` attribute are `label-width-and-position` or `label-alignment`.

## 20.432 text:min-label-distance

The `text:min-label-distance` attribute specifies the minimum distance between a number and a list item.

The content of a list label is rendered inside a specified minimum width - `text:min-label-width` 20.433. The text of a list item is rendered following the label width area. If the distance between a list label and the text of a list item is smaller than the value of the `text:min-label-distance` attribute, the following actions are performed. First, the content of the list label is moved inside the specified minimum label width up to the value of the `text:min-label-distance` attribute. If that does not result in a value equal to the `text:min-label-distance` attribute, the text of the list item is moved until the distance between the list label and text of list item equals the value of the `text:min-label-distance` attribute.

This attribute can be associated with a formatting properties element that is contained in a `<text:list-level-style-*>` element.

The attribute will be evaluated only if the `text:list-level-position-and-space-mode` attribute has the value `label-width-and-position` or is not present.

The `text:min-label-distance` attribute is usable with the following element: `<style:list-level-properties>` 17.19.

The `text:min-label-distance` attribute has the data type `nonNegativeLength` 18.3.21.

## 20.433 text:min-label-width

The `text:min-label-width` attribute specifies the minimum width of a list label.

This attribute can be associated with a formatting properties element that is contained in a `<text:list-level-style-*>` element.

The list label can be aligned horizontally within the specified minimum width using the `fo:text-align` 20.223 attribute. If the actual width of the list label is greater than the specified minimum width no alignment takes place.

The attribute will be evaluated only if the `text:list-level-position-and-space-mode` 20.431 attribute has the value `label-width-and-position` or is not present.

The `text:min-label-width` attribute is usable with the following element:  
`<style:list-level-properties>` 17.19.

The `text:min-label-width` attribute has the data type `nonNegativeLength` 18.3.21.

## 20.434 `text:number-lines`

The `text:number-lines` attribute specifies whether lines are numbered.

The defined values for the `text:number-lines` attribute are:

- `false`: lines should not be numbered.
- `true`: lines should be numbered.

The `text:number-lines` attribute is usable with the following element:  
`<style:paragraph-properties>` 17.6.

The `text:number-lines` attribute has the data type `boolean` 18.3.3.

## 20.435 `text:space-before`

The `text:space-before` attribute specifies the space to include before a number for all paragraphs at a specified level. If a paragraph has a left margin greater than 0, the position of the list label box is the left margin width plus the start indent value.

This attribute can be associated with a formatting properties element that is contained in a `<text:list-level-style-*>` element.

This attribute is evaluated only if the `text:list-level-position-and-space-mode` 20.431 attribute has the value `label-width-and-position` or is not present.

The `text:space-before` attribute is usable with the following element: `<style:list-level-properties>` 17.19.

The `text:space-before` attribute has the data type `length` 18.3.19.

## 20.436 `chart:major-origin`

The `chart:major-origin` attribute specifies the location of one major tick.

The `chart:major-origin` attribute is usable with the following element: `<style:chart-properties>` 17.22.

The `chart:major-origin` attribute has the data type `double` 18.2.

## 20.437 `chart:minor-logarithmic`

The `chart:minor-logarithmic` attribute specifies whether the minor ticks are spaced equally after the logarithmic transformation. If this value is `false` they are spaced equally before the transformation.

The `chart:minor-logarithmic` attribute is usable with the following element:  
`<style:chart-properties>` 17.22.

The `chart:minor-logarithmic` attribute has the data type `boolean` 18.3.3.

## 20.438 draw:allow-overlap

The `draw:allow-overlap` attribute specifies whether a frame or graphic object may overlap other frame or graphic objects. The defined values for the `draw:allow-overlap` attribute are:

- `true`: The object may overlap other frame or graphic objects.
- `false`: The consumer should change the displayed layout so, that the object does not overlap with other frame or graphic objects. How the displayed layout is modified to prevent overlap is implementation-dependent.

The attribute is ignored in case the attribute `style:wrap` is evaluated to value `run-through` on the same object.

The default value for the attribute `draw:allow-overlap` is `true`.

The `draw:allow-overlap` attribute is usable with the following element:  
`<style:graphic-properties> 17.21`.

The `draw:allow-overlap` attribute has the data type `boolean` 18.3.3.

## 20.439 style:margin-gutter

The `style:margin-gutter` attribute introduces a distance information, which area at the “binding edge” (see `style:page-usage` 19.509) of a page should be treated as gutter, that means that it might be consumed from binding. The distance is measured from binding edge towards text content area.

The default value for this attribute is `0`.

The `style:margin-gutter` attribute is usable with the following element: `<style:page-layout-properties> 17.2`.

The `style:margin-gutter` attribute has the data type `nonNegativeLength` 18.3.21.

## 20.440 text:relative-tab-stop-position

The `text:relative-tab-stop-position` attribute specifies whether the position of tab stops is relative to the outer or inner edge of the margin area of the paragraph.

The defined values for the `text:relative-tab-stop-position` attribute are:

- `false`: paragraph style determines position of tab stops relative to the outer edge of the margin that is specified by `fo:margin` or `fo:margin-left` of the paragraph.
- `true`: paragraph style determines the position of tab stops relative to the inner edge of the margin that is specified by `fo:margin` or `fo:margin-left` of the paragraph.  
Exception: the tab stop set by applied list style is determined by the paragraph style relative to the outer edge of the margin that is specified by `fo:margin` or `fo:margin-left` of the paragraph.

**Note:** The paragraph's margin may be specified by its paragraph style, an applied list style or paragraph default style.

**Note:** In right-to-left layout environments, the paragraph's right margin is specified by `fo:margin-left`, and its tab stop positions are relative to the right edge.

The default value for this attribute is `true`.

This attribute is only evaluated if it is specified within a default style of family paragraph.

The `text:relative-tab-stop-position` attribute is usable with the following element: `<style:paragraph-properties> 17.6.`

The `text:relative-tab-stop-position` attribute has the data type `boolean` 18.3.3.

## 20.441 draw:decorative

The `draw:decorative` attribute specifies that the object is purely decorative and may be ignored by assistive technologies.

The default value for this attribute is `false`.

The `draw:decorative` attribute is usable with the following element: `<style:graphic-properties> 17.21.`

The `draw:decorative` attribute has the data type `boolean` 18.3.3.

## 20.442 table:tab-color

The `table:tab-color` attribute specifies the color of the tab associated with a sheet.

When this attribute is missing, the application should use the default color used for sheet tabs.

The `table:tab-color` attribute is usable with the following element: `<style:table-properties> 17.15.`

The `table:tab-color` attribute has the data type `color` 18.3.9.

---

## Appendix A OpenDocument Relax NG Schema

The OpenDocument Relax-NG (see [RNG]) schema is defined by a separate document, whose location can be found in the Additional artifacts section on the introductory pages.

---

## Appendix B OpenDocument Metadata Manifest Ontology

The OpenDocument metadata manifest ontology is defined by a separate document, whose location can be found in the Additional artifacts section on the introductory pages.

---

## Appendix C MIME Types and File Name Extensions (Non Normative)

The following table contains a list of MIME types and file name extensions for OpenDocument documents, that, at the time this specification is published, have been registered according to [RFC6838]. Please see [MIMETYPES] for a current list of registered MIME types.

Registered MIME types are for use with documents contained in OpenDocument packages.

*Table 26 - Registered MIME types*

<b>MIME type</b>	<b>Ext.</b>	<b>Description</b>
application/vnd.oasis.opendocument.text	odt	Text document
application/vnd.oasis.opendocument.text-template	ott	Text document used as template
application/vnd.oasis.opendocument.graphics	odg	Graphics document (Drawing)
application/vnd.oasis.opendocument.graphics-template	otg	Drawing document used as template
application/vnd.oasis.opendocument.presentation	odp	Presentation document
application/vnd.oasis.opendocument.presentation-template	otp	Presentation document used as template
application/vnd.oasis.opendocument.spreadsheet	ods	Spreadsheet document
application/vnd.oasis.opendocument.spreadsheet-template	ots	Spreadsheet document used as template
application/vnd.oasis.opendocument.chart	odc	Chart document
application/vnd.oasis.opendocument.chart-template	otc	Chart document used as template
application/vnd.oasis.opendocument.image (deprecated)	odi	Image document
application/vnd.oasis.opendocument.image-template (deprecated)	oti	Image document used as template
application/vnd.oasis.opendocument.formula	odf	Formula document
application/vnd.oasis.opendocument.formula-template	otf	Formula document used as template
application/vnd.oasis.opendocument.text-master	odm	Global Text document. See section 19.812.
application/vnd.oasis.opendocument.text-master-template	otm	Global text document used as template
application/vnd.oasis.opendocument.text-web	oth	Text document used as template for HTML documents

The following table contains a list of MIME types and file name extensions for office documents that conform to this specification where a registration according to [RFC6838] is in progress at the time this specification is published.

Please check [MIMETYPES] before using these MIME types. If a MIME type is not listed there, the MIME type that is the result of inserting "x-" behind the "/" (U+002F, SOLIDUS) character (application/x-vnd.oasis.opendocument.text) should be used.

*Table 27 - Recommended MIME types*

<b>MIME type</b>	<b>Ext.</b>	<b>Description</b>
application/vnd.oasis.opendocument.base	odb	Database front end document

---

## Appendix D Accessibility Guidelines (Non Normative)

### D.1 Title, Description and Caption of Graphical Elements

OpenDocument provides conventions to support accessible names, descriptions and captions for graphical elements.

Accessible names, if available, are represented by the `<svg:title>` 10.3.17 element. If the `<svg:title>` element is not available, any text referenced by a `draw:caption-id` 19.115 attribute is used as the title.

When transforming from another document format to OpenDocument the short names, like HTML's alt text on the `<img>` elements should be mapped to the `<svg:title>` element.

User agents should not manufacture names for the `<svg:title>` element, such as using the drawing object name followed by a cardinal number in a string as it is used for accessibility. Such name assignments provide no semantic meaning to the user.

Accessible descriptions, if available, are represented by the `<svg:desc>` 10.3.18 element.

If a user agent supports relationships between graphic elements and their captions, the value of the `draw:caption-id` attribute for a graphical element should be used to for that relationship. 19.115

#### D.1.1 Guidance for authors

Authors should not assign names to objects having no semantic value. If no name is assigned the caption text will be used in its place. The `<svg:title>` element should take precedence over the caption text for accessible name assignment by the user agent.

Assignment of the long description should only be necessary when a drawing object is significantly complex and the user needs more information to describe it. Long descriptions would be more applicable to drawing groupings than basic drawing shapes.

#### D.1.2 Authoring tool responsibility for presenting and prompting for the `<svg:title>` and `<svg:desc>` elements

Authoring tools should provide an option from an objects context menu to allow the user to enter the text for either of these elements as a minimum. More proactive authoring tools should have a facility for prompting the author for this text. Since the `<svg:desc>` element is a long description, a text area vs. a text field should be used to prompt the user accordingly in GUI-based authoring tools like office applications.

Navigation tools used to list the objects in the view should provide the type of object followed by the contents of `<svg:title>` element.

For `<draw:g>` 10.3.15 elements the drawing objects which are members of the group should visible only when the group is expanded.

### D.2 Hyperlink Titles

When transforming from another document format to OpenDocument the alt text of hyperlinks, should be mapped to the `office:title` attribute of `<text:a>` 6.1.8 elements or `<draw:a>` 10.4.12 elements. When exporting OpenDocument documents to HTML, the contents of title text should be mapped to title attribute text on HTML anchor tags. As a minimum, authoring tools should provide a mechanism to provide the hint text.

The title text should be made accessible to the assistive technology and user. The user agent should allow for programmatic access through standard accessibility APIs such as the accessible description. Users should experience visible access to the hint text via the keyboard or mouse.

### **D.3 Tables in Presentations**

Users importing non-OpenDocument slides that contain tables need access to the table structure via their assistive technology. Therefore tables imported into an OpenDocument application from another file format should have their structure preserved, and when saved as OpenDocument should be saved as as embedded spreadsheets.

### **D.4 Further Guidelines**

Additional, non-normative Accessibility Guidelines are available at: [https://docs.oasis-open.org/office/office-accessibility/v1.0/ODF\\_Accessibility\\_Guidelines-v1.0.html](https://docs.oasis-open.org/office/office-accessibility/v1.0/ODF_Accessibility_Guidelines-v1.0.html). That more comprehensive document will be the up-to-date set of recommendations for what all OpenDocument applications should do in order to fully support accessibility.

---

# Appendix E Bidirectional (BiDi) Scripts, Numeric Digits Presentation and Calendars (Non Normative)

This appendix specifies how bidirectional (BiDi) scripts and related information are represented in OpenDocument.

## E.1 Paragraph and Layout Direction

In OpenDocument, the direction of text runs inside a paragraph is calculated using the Unicode BiDi Algorithm. See [UAX9]. The paragraph direction, as required by the BiDi Algorithm (see BD5 of [UAX9]), and the display direction of layout objects like table or page columns (in the following called layout direction) is controlled by a writing mode attribute (`style:writing-mode`) that can be used within styles.

The writing mode attribute can be applied individually to paragraph styles, page styles, section styles, table styles, table cell styles, graphic styles and chart styles. If present within a paragraph style, it controls the paragraph direction of those paragraphs, to which the style is applied. If present within a page style, section style, table style, table cell style, graphic style or chart style, it controls the layout direction of those pages, text sections, tables, table cells, text-boxes and chart objects to which the styles is applied.

The paragraph direction specifies the default bidirectional orientation of the text in that paragraph. The result of the BiDi Algorithm can be manually changed by inserting BiDi embedding control characters (U+202A ... U+202E) and implicit directional marks (U+200E ...U+200F) into the text. See [UTR20].

## E.2 Numeric Digits Presentation and Calendars

All digits that have a Unicode code point can be included in an OpenDocument document.

**Note 1:** office application have a feature that allows the user to specify whether the ASCII digits U+0030 ... U+0039 should be displayed as Latin-Indic digits or as Arabic-Indic digits (U+0660 ... U+0669). Since this feature effects only what digits are displayed and does not influence the representation of digits in the document itself, OpenDocument only allows storing this setting as an implementation-dependent setting, not as document or style content.

For list numbers, that are calculated automatically, OpenDocument provides a generic mechanism to specify the applicable numbering systems. See 19.504.

**Note 2:** The specification currently mentions only "1, 2, 3...", "I, II, III...", and "i, ii, iii" explicitly, but the schema also allows a generic string value.

OpenDocument further supports data styles, which describe how different types of data are displayed, for example, a number or a date. Data styles are described in section 16.29. The presentation of numeric digits can be controlled by the transliteration attributes described in sections 19.365-19.368. The presentation of date information can be controlled by the `number:calendar` attribute specified in section 19.348.

---

## Appendix F Recommended Usage of SMIL

The following sections describe the usage of SMIL animation elements that enables an office application to present the animation elements in a simple and easy to use UI to the user. This UI may contain a single main sequence of effects, and in addition to this, multiple sequences of effects that are started as interactions on drawing shapes. An effect is a combination of one or more animation elements that animate a single shape and or a shape's paragraphs.

In user interfaces, effects should be creatable by using presets that have localized and meaningful names. This way, the user will not work on a hierarchy of SMIL animation elements, but on one dimensional lists of effects, which are much easier to handle for the office application users.

### F.1 Slide Animation

Each `<draw:page>` 10.2.4 element may have an `<anim:par>` 15.4.2 element that defines the animation of that page during a running slideshow. This `<anim:par>` element should contain one `<anim:seq>` 15.4.3 element which is the main sequence for shape effects and zero or more `<anim:seq>` elements that define interactive sequences for shapes that contain animation interactions. The animation elements are executed after the slide has executed its initial transition.

The `<anim:par>` element may further contain an `<anim:par>` element that defines the slide transition. Its `smil:begin` 19.4.41 attribute shall have the value `<id>.begin`, where `<id>` is the id of the slide's `<draw:page>` element itself. This `<anim:par>` element can contain basic animation elements as defined in chapter 15. These elements shall specify the `<draw:page>` element itself as target. If such an `<anim:par>` element for slide transitions is present, it overrides the following presentation page formatting properties:

- `<presentation:sound>`. See 10.8.2.
- `presentation:transition-speed`. See 20.238.
- `presentation:transition-style`. See 20.239.
- `presentation:transition-type`. See 20.240.

For backward compatibility reasons, producers that use an `<anim:par>` element to define the slide transition should also add the above presentation page formatting properties. This also enables consumers without SMIL support to display the slide transitions.

**Example:** The following example defines a slide wipe transition.

```
<draw:page draw:id="id1">
  <anim:par presentation:node-type="timing-root">
    <anim:par smil:begin="id1.begin">
      <anim:transitionFilter smil:dur="2s"
        smil:targetElement="id1"
        smil:type="slideWipe"
        smil:subtype="fromBottom"
        smil:direction="reverse"/>
    </anim:par>
  </anim:par>
</draw:page>
```

### F.2 Main Sequence

The main sequence is a `<anim:seq>` 15.4.3 element which contains the effects that should start after the slide has executed its initial transition. Since this is a sequential container, its child nodes are executed one after each other. If a child node's `smil:begin` 19.4.41 attribute

has the value `indefinite` execution is stalled until the user advances the slideshow by a mouse or key interaction.

The first level of child nodes in the main sequence should be `<anim:par>` 15.4.2 elements that group animation elements that are started with the same user interaction. The second level of child nodes should be `<anim:par>` elements that group animations elements that start at the same time. The third level of child nodes should be `<anim:par>` elements that group the animation elements for a single effect.

**Example:** The following example shows a main sequence with the effects A, B, C and D. Effect A is started on user interaction, effect B is started simultaneously with A. Effect C is started 4 seconds after the effects A and B. Effect D is started on the next user interaction:

```
<amin:par> <!-- timing root-->
  <anim:seq> <!-- main sequence-->
    <anim:par smil:begin="indefinite">
      <!-- first user interaction -->
      <anim:par smil:begin="0s" smil:dur="4s">
        <!-- first group of effects to execute -->
        <anim:par> <!-- effect a -->
          <!-- nodes for effect a-->
        </anim:par>
        <anim:par> <!-- effect b -->
          <!-- nodes for effect b-->
        </anim:par>
      </anim:par>
      <anim:par smil:begin="4s">
        <!-- second group of effects to execute -->
        <anim:par> <!-- effect c -->
          <!-- nodes for effect c-->
        </anim:par>
      </anim:par>
    </anim:par>
    <anim:par>
      <!-- second user interaction-->
      <anim:par smil:begin="indefinite">
        <!-- first group of effects to execute -->
        <anim:par> <!-- effect d -->
          <!-- nodes for effect d-->
        </anim:par>
      </anim:par>
    </anim:par>
  </anim:seq>
</anim:par>
```

### F.3 Interactive Sequence

An interactive sequence is a `<anim:seq>` 15.4.3 element that should have the same structure as a main sequence. The only difference is that the `<anim:par>` 15.4.2 element in the first level has a `smil:begin` 15.4.3 attribute with a value like `[shape-id].click`, where `[shape-id]` identifies a drawing shapes by its `draw:id` 19.187 attribute. These animation elements are triggered when the user interacts with the element defined by `[shape-id]`.

---

## Appendix G Changes From ODF 1.3 (Non Normative)

This appendix describes changes that are related to Part 3 of this specification.

The following datatypes are new:

error 18.3.43 [Office-3759](#)

nonNegativePercent 18.3.44 [Office-4122](#)

The following normative references have changed:

[ISO8601] [Office-4112](#)

[MathML] [Office-4148](#)

[UNICODE] [Office-4030](#)

The following normative references are new:

[CSS4WRitingModes] [Office-4030](#)

The following descriptions have changed:

SMIL Presentation Animations – General 10.9.1 [Office-4023](#)

Appendix D.1 D.1 [Office-4086](#)

Appendix D.1.1 D.1.1 [Office-4086](#)

Appendix D.2 D.2 [Office-4086](#)

OpenDocument Document 2.2.1 [Office-4093](#)

Appendix E.1 E.1 [Office-4030](#), [Office-4115](#)

The following elements are new:

<number:num-list-format> 16.29.30 [Office-4108](#)

<number:num-list-label> 16.29.31 [Office-4108](#)

<style:background-fill> 17.26 [Office-4125](#)

The following elements have changed:

<chart:legend> 11.4 [Office-4033](#)

<dr3d:cube:> 10.5.4 [Office-4072](#)

<dr3d:extrude> 10.5.6 [Office-4072](#)

<dr3d:rotate> 10.5.7 [Office-4072](#)

<dr3d:scene> 10.5.2 [Office-4072](#), [Office-4142](#)

<dr3d:sphere> 10.5.5 [Office-4072](#)

```
<draw:handle> 10.6.3 Office-4110
<draw:marker> 16.42.8 Office-3685
<draw:measure> 10.3.12 Office-2949
<draw:page> 10.2.4 Office-3022
<draw:caption> 10.3.11 Office-1909, Office-3761
<draw:circle> 10.3.8 Office-3761, Office-4142
<draw:connector> 10.3.10 Office-3761, Office-4142
<draw:custom-shape> 10.6.1 Office-3761, Office-4142
<draw:ellipse> 10.3.9 Office-3761, Office-4142
<draw:frame> 10.4.2 Office-4142
<draw:g> 10.3.15 Office-4142
<draw:image> 10.4.4 Office-3761
<draw:line> 10.3.3 Office-3761, Office-4142
<draw:measure> 10.3.12 Office-3761, Office-4142
<draw:page-thumbnail> 10.3.14 Office-4142
<draw:path> 10.3.7 Office-3761, Office-4142
<draw:polygon> 10.3.5 Office-3761, Office-4142
<draw:polyline> 10.3.4 Office-3761, Office-4142
<draw:rect> 10.3.2 Office-3761, Office-4142
<draw:regular-polygon> 10.3.6 Office-3761, Office-4142
<draw:text-box> 10.4.3 Office-1909
<office:annotation> 14.1 Office-1909
<style:chart-properties> 17.22 Office-3936
<style:graphic-properties> 17.21 Office-4142
<style:paragraph-properties> 17.6 Office-3846
<style:style> 16.2 Office-4004
<text:ruby-base> 6.4.2 Office-4120
```

The following element has moved:

<presentation:sound> from 10.8.2 to 10.8.2

The following attributes are new:

```
chart:major-origin 20.436 Office-3936
chart:minor-logarithmic 20.437 Office-3936
draw:allow-overlap 20.438 Office-4047
draw:decorative 20.441 Office-4132
draw:extrusion-metal-type 19.923 Office-4123
draw:handle-position-x 19.924 Office-4110
draw:handle-position-y 19.925 Office-4110
draw:handle-polar-pole-x 19.926 Office-4110
```

draw:handle-polar-pole-y 19.927 [Office-4110](#)  
draw:handle-polar-radius 19.928 [Office-4110](#)  
draw:handle-polar-angle 19.929 [Office-4110](#)  
draw:name\_<dr3d:cube> 19.197.31 [Office-4072](#)  
draw:name\_<dr3d:extrude> 19.197.32 [Office-4072](#)  
draw:name\_<dr3d:rotate> 19.197.33 [Office-4072](#)  
draw:name\_<dr3d:scene> 19.197.34 [Office-4072](#)  
draw:name\_<dr3d:sphere> 19.197.35 [Office-4072](#)  
form:input-required 19.921 [Office-4121](#)  
style:margin-gutter 20.439 [Office-4105](#)  
style:num-list-format-name 19.922 [Office-4108](#)  
text:relative-tab-stop-position 20.440 [Office-3846](#)

The following attributes changed

chart:interval-major 20.28 [Office-3936](#)  
chart:interval-minor-divisor 20.29 [Office-3936](#)  
db:table-definition 19.85.4 [Office-4029](#)  
draw:enhanced-path 19.145 [Office-3710](#)  
draw:enhanced-path 19.145 [Office-4026](#)  
draw:enhanced-path 19.145 [Office-4027](#)  
draw:extrusion-specularity 19.168 [Office-4122](#)  
draw:fill 20.118 [Office-4125](#)  
draw:fill-color 20.119 [Office-4125](#)  
draw:fill-gradient-name 20.120 [Office-4125](#)  
draw:fill-hatch-name 20.121 [Office-4125](#)  
draw:formula 19.171 [Office-3816](#)  
draw:marker-end-center 20.144 [Office-3685](#)  
draw:marker-start-center 20.147 [Office-3685](#)  
draw:opacity 20.152 [Office-4125](#)  
draw:opacity-name 20.153 [Office-4125](#)  
draw:transform 19.228 [Office-3750](#)  
draw:wrap-influence-on-position 20.181 [Office-3005](#)  
fo:break-after 20.184 [Office-3751](#)  
fo:break-before 20.185 [Office-3751](#)  
fo:line-height 20.204 [Office-3997](#)  
fo:text-indent 20.225 [Office-4092](#)  
form:default-button 19.265 [Office-4101](#)  
number:calendar 19.341 [Office-4112](#)  
office:value-type 19.389 [Office-3759](#)  
office:version 19.390 [Office-4140](#)

presentation:show-logo 19.419 [Office-4103](#)  
style:first-page-number 20.266 [Office-3953](#)  
style:num-letter-sync 19.505, 20.323 [Office-4104](#)  
style:page-usage 19.509 [Office-4105](#)  
style:repeat 20.341 [Office-4106](#)  
style:vertical-pos 20.397 [Office-4073](#)  
style:vertical-rel 20.398 [Office-4073](#), [Office-4129](#)  
svg:viewBox 19.574 [Office-3824](#)  
style:writing-mode General 20.404.1 [Office-4030](#)  
style:writing-mode\_<style:graphic-properties> 20.404.2 [Office-4030](#)  
style:writing-mode\_<style:page-layout-properties> 20.404.3 [Office-4030](#)  
style:writing-mode\_<style:paragraph-properties> 20.404.4 [Office-4030](#)  
style:writing-mode\_<style:section-properties> 20.404.5 [Office-4030](#)  
style:writing-mode\_<style:table-cell-properties> 20.404.6 [Office-4030](#)  
style:writing-mode\_<style:table-properties> 20.404.7 [Office-4030](#)  
table:data-type 19.615.2 [Office-4107](#)  
table:name <table:operation> 19.677.11 [Office-4102](#)  
table:operator 19.688 [Office-4107](#)  
table:search-criteria-must-apply-to-whole-cell 19.711 [Office-4107](#)  
text:condition general 19.782.1 [Office-4022](#)  
text:main-entry-style-name 19.838 [Office-4119](#)  
text:reference-format 19.860 [Office-4035](#)  
text:start-value\_<text:list-level-style-number> 19.874.4 [Office-4118](#)  
text:start-value\_<text:outline-level-style> 19.874.7 [Office-4118](#)  
text:style-name\_<text:index-entry-bibliography> 19.880.10 [Office-4119](#)  
text:style-name\_<text:index-entry-chapter> 19.880.11 [Office-4119](#)  
text:style-name\_<text:index-entry-link-start> 19.880.13 [Office-4119](#)  
text:style-name\_<text:index-entry-link-end> 19.880.12 [Office-4119](#)  
text:style-name\_<text:index-entry-page-number> 19.880.14 [Office-4119](#)  
text:style-name\_<text:index-entry-span> 19.880.15 [Office-4119](#)  
text:style-name\_<text:index-entry-tab-stop> 19.880.16 [Office-4119](#)  
text:style-name\_<text:index-entry-text> 19.880.17 [Office-4119](#)  
text:style-name\_<text:index-title-template> 19.880.20 [Office-4119](#)  
text:style-name\_<text:line-numbering-configuration> 19.880.21 [Office-4119](#)  
text:style-name\_<text:list-level-style-number> 19.880.23 [Office-4119](#)  
text:style-name\_<text:list-level-style-bullet> 19.880.24 [Office-4119](#)  
text:style-name\_<text:outline-level-style> 19.880.28 [Office-4119](#)  
text:style-name\_<text:ruby-text> 19.880.31 [Office-4119](#)

The following attribute has moved:

table:tab-color from 19.731 to 20.442

Removed elements, attributes and values and consequential changes

table:cell-range-address 19.597.6 within <chart:plot-area> [Office-3741](#)

chart:data-source-has-labels 19.17 [Office-4134](#)

chart:series-source 20.54 [Office-4134](#)

Deprecated Elements

<meta:date-string> 14.3 [Office-2390](#)

<presentation:animation-group> 10.8.9 [Office-4023](#)

<presentation:animations> 10.8.1 [Office-4023](#)

<presentation:dim> 10.8.7 [Office-4023](#)

<presentation:hide-shape> 10.8.5 [Office-4023](#)

<presentation:hide-text> 10.8.6 [Office-4023](#)

<presentation:play> 10.8.8 [Office-4023](#)

<presentation:show-shape> 10.8.3 [Office-4023](#)

<presentation:show-text> 10.8.4 [Office-4023](#)

Deprecated Attributes

draw:handle-polar 19.178 [Office-4110](#)

draw:handle-position 19.179 [Office-4110](#)

presentation:delay 19.395 [Office-4023](#)

presentation:path-id 19.409 [Office-4023](#)