

Standard ECMA-376

1st Edition / December 2006

Office Open XML File Formats

Standard

Standard
ECMA-376

1st Edition / December 2006

**Office Open XML
File Formats**

Office Open XML

Part 1: Fundamentals

December 2006

Table of Contents

1		
2	Foreword.....	xi
3	Introduction	xii
4	1. Scope	1
5	2. Conformance	2
6	2.1 Goal	2
7	2.2 Issues	2
8	2.3 What this Standard Specifies.....	3
9	2.4 Document Conformance	3
10	2.5 Application Conformance	3
11	2.6 Interoperability Guidelines.....	3
12	3. Normative References.....	5
13	4. Definitions	6
14	5. Notational Conventions	8
15	6. Acronyms and Abbreviations.....	9
16	7. General Description	10
17	8. Overview	11
18	8.1 Packages and Parts.....	11
19	8.2 Consumers and Producers.....	11
20	8.3 WordprocessingML	11
21	8.4 SpreadsheetML.....	12
22	8.5 PresentationML	13
23	8.6 Supporting MLs.....	14
24	8.6.1 DrawingML	14
25	8.6.2 VML	15
26	8.6.3 Custom XML Data Properties	15
27	8.6.4 File Properties	15
28	8.6.5 Math	15
29	8.6.6 Bibliography	15
30	9. Packages.....	16
31	9.1 Constraints on Office Open XML's Use of OPC.....	16
32	9.1.1 Part Names.....	16
33	9.1.2 Part Addressing	16
34	9.1.3 Fragments.....	16
35	9.1.4 Physical Packages	16
36	9.1.5 Interleaving	16
37	9.1.6 Unknown Parts	17
38	9.1.7 Trash Items.....	17
39	9.1.8 Invalid Parts.....	17
40	9.1.9 Unknown Relationships.....	17

1	9.2 Relationships in Office Open XML	17
2	10. Markup Compatibility and Extensibility	23
3	10.1 Constraints on Office Open XML's Use of Markup Compatibility and Extensibility	23
4	10.1.1 PreserveElements and PreserveAttributes	23
5	10.1.2 Office Open XML Native Extensibility Constructs	23
6	11. WordprocessingML	24
7	11.1 Glossary of WordprocessingML-Specific Terms	24
8	11.2 Package Structure	25
9	11.3 Part Summary	27
10	11.3.1 Alternative Format Import Part	28
11	11.3.2 Comments Part	29
12	11.3.3 Document Settings Part	31
13	11.3.4 Endnotes Part	33
14	11.3.5 Font Table Part	35
15	11.3.6 Footer Part	36
16	11.3.7 Footnotes Part	38
17	11.3.8 Glossary Document Part	41
18	11.3.9 Header Part	43
19	11.3.10 Main Document Part	46
20	11.3.11 Numbering Definitions Part	48
21	11.3.12 Style Definitions Part	51
22	11.3.13 Web Settings Part	52
23	11.4 Document Template	53
24	11.5 Framesets	54
25	11.6 Master Documents and Subdocuments	55
26	11.7 Mail Merge Data Source	56
27	11.8 Mail Merge Header Data Source	57
28	11.9 XSL Transformation	58
29	12. SpreadsheetML	59
30	12.1 Glossary of SpreadsheetML-Specific Terms	59
31	12.2 Package Structure	60
32	12.3 Part Summary	62
33	12.3.1 Calculation Chain Part	63
34	12.3.2 Chartsheet Part	64
35	12.3.3 Comments Part	65
36	12.3.4 Connections Part	67
37	12.3.5 Custom Property Part	68
38	12.3.6 Custom XML Mappings Part	69
39	12.3.7 Dialogsheet Part	70
40	12.3.8 Drawings Part	72
41	12.3.9 External Workbook References Part	73
42	12.3.10 Metadata Part	75
43	12.3.11 Pivot Table Part	78
44	12.3.12 Pivot Table Cache Definition Part	79
45	12.3.13 Pivot Table Cache Records Part	81
46	12.3.14 Query Table Part	82

1	12.3.15 Shared String Table Part	83
2	12.3.16 Shared Workbook Revision Headers Part	84
3	12.3.17 Shared Workbook Revision Log Part	85
4	12.3.18 Shared Workbook User Data Part	87
5	12.3.19 Single Cell Table Definitions Part	87
6	12.3.20 Styles Part	89
7	12.3.21 Table Definition Part	90
8	12.3.22 Volatile Dependencies Part	91
9	12.3.23 Workbook Part	92
10	12.3.24 Worksheet Part	94
11	12.4 External Workbooks	96
12	13. PresentationML	98
13	13.1 Glossary of PresentationML-Specific Terms	98
14	13.2 Package Structure	98
15	13.3 Part Summary	101
16	13.3.1 Comment Authors Part	102
17	13.3.2 Comments Part	103
18	13.3.3 Handout Master Part	104
19	13.3.4 Notes Master Part	106
20	13.3.5 Notes Slide Part	107
21	13.3.6 Presentation Part	109
22	13.3.7 Presentation Properties Part	111
23	13.3.8 Slide Part	111
24	13.3.9 Slide Layout Part	113
25	13.3.10 Slide Master Part	115
26	13.3.11 Slide Synchronization Data Part	116
27	13.3.12 User Defined Tags Part	117
28	13.3.13 View Properties Part	118
29	13.4 HTML Publish Location	119
30	13.5 Slide Synchronization Server Location	120
31	14. DrawingML	122
32	14.1 Glossary of DrawingML-Specific Terms	122
33	14.2 Part Summary	122
34	14.2.1 Chart Part	123
35	14.2.2 Chart Drawing Part	125
36	14.2.3 Diagram Colors Part	126
37	14.2.4 Diagram Data Part	127
38	14.2.5 Diagram Layout Definition Part	128
39	14.2.6 Diagram Style Part	130
40	14.2.7 Theme Part	131
41	14.2.8 Theme Override Part	133
42	14.2.9 Table Styles Part	134
43	15. Shared	136
44	15.1 Glossary of Shared Terms	136
45	15.2 Part Summary	136
46	15.2.1 Additional Characteristics Part	137

1	15.2.2	Audio Part.....	138
2	15.2.3	Bibliography Part.....	139
3	15.2.4	Custom XML Data Storage Part.....	140
4	15.2.5	Custom XML Data Storage Properties Part	141
5	15.2.6	Digital Signature Origin Part.....	141
6	15.2.7	Digital Signature XML Signature Part	142
7	15.2.8	Embedded Control Persistence Part	143
8	15.2.9	Embedded Object Part	146
9	15.2.10	Embedded Package Part.....	148
10	15.2.11	File Properties	149
11	15.2.12	Font Part.....	154
12	15.2.13	Image Part	154
13	15.2.14	Printer Settings Part	155
14	15.2.15	Thumbnail Part.....	156
15	15.2.16	Video Part.....	157
16	15.2.17	VML Drawing Part	158
17	15.3	Hyperlinks.....	159
18	Annex A. Bibliography		161
19	Annex B. Index.....		163
20			

1 Foreword

2 This multi-part Standard deals with Office Open XML Format-related technology, and consists of the following
3 parts:

- 4 • **Part 1: "Fundamentals" (this document)**
- 5 • Part 2: "Open Packaging Conventions"
- 6 • Part 3: "Primer"
- 7 • Part 4: "Markup Language Reference"
- 8 • Part 5: "Markup Compatibility and Extensibility"

9 Parts 2 and 4 include a number of annexes that refer to data files provided in electronic form only.

1 Introduction

2 This Part is one piece of a Standard that describes a family of XML schemas, collectively called *Office Open XML*,
3 which define the XML vocabularies for word-processing, spreadsheet, and presentation documents, as well as
4 the packaging of documents that conform to these schemas.

5 The goal is to enable the implementation of the Office Open XML formats by the widest set of tools and
6 platforms, fostering interoperability across office productivity applications and line-of-business systems, as well
7 as to support and strengthen document archival and preservation, all in a way that is fully compatible with the
8 large existing investments in Microsoft Office documents.

9 The following organizations have participated in the creation of this Standard and their contributions are
10 gratefully acknowledged:

11 Apple, Barclays Capital, BP, The British Library, Essilor, Intel, Microsoft, NextPage, Novell, Statoil, Toshiba, and
12 the United States Library of Congress

1. Scope

- 2 This Standard defines Office Open XML's vocabularies and document representation and packaging. It also
3 specifies requirements for consumers and producers of Office Open XML.

2. Conformance

The text in this Standard is divided into *normative* and *informative* categories. Unless documented otherwise, any feature shall be implemented as specified by the normative text describing that feature in this Standard. Text marked informative (using the mechanisms described in §7) is for information purposes only. Unless stated otherwise, all text is normative.

Use of the word “shall” indicates required behavior.

Any behavior that is not explicitly specified by this Standard is implicitly unspecified (§4).

2.1 Goal

The goal of this clause is to define conformance, and to provide interoperability guidelines in a way that fosters broad and innovative use of the Office Open XML file format, while maximizing interoperability and preserving investment in existing files and applications (§4). By meeting this goal, this Standard benefits the following audiences:

- Developers that design, implement, or maintain Office Open XML applications.
- Developers that interact programmatically with Office Open XML applications.
- Governmental or commercial entities that procure Office Open XML applications.
- Testing organizations that verify conformance of specific Office Open XML applications to this Standard. (Note that this Standard does not include a test suite.)
- Educators and authors who teach about Office Open XML applications.

2.2 Issues

To achieve the above goal, the following issues need to be considered:

1. The application domain encompasses a range of possible consumers (§4) and producers (§4) so broad that defining specific application behaviors would restrict innovation. For example, stipulating visual layout would be inappropriate for a consumer that extracts data for machine consumption, or that renders text in sound. Another example is that restricting capacity or precision runs the risk of diluting the value of future advances in hardware.
2. Commonsense user expectations regarding the interpretation of an Office Open XML package (§4) play such an important role in that package's value that a purely syntactic definition of conformance would fail to effect a useful level of interoperability. For example, such a definition would admit an application that reads a package, and then writes it in a manner that, though syntactically valid, differs arbitrarily from the original.
3. Legitimate operations on a package include deliberate transformations, making blanket change prohibitions inappropriate in the conformance definition. For example, collapsing spreadsheet formulas to their calculated values, or converting complex presentation graphics to static bitmaps,

could be correct for an application whose published purpose is to perform those operations. Again, commonsense user expectation makes the difference.

4. Existing files and applications exercise a broad range of formats and functionality that, if required by the conformance definition, would add an impractical amount of bulk to the This Standard and could inadvertently obligate new applications to implement a prohibitive amount of functionality. This issue is caused by the breadth of currently available functionality and is compounded by the existence of legacy formats.

2.3 What this Standard Specifies

To address the issues listed above, this Standard constrains both syntax and semantics, but it is not intended to predefine application behavior. Therefore, it includes, among others, the following three types of information:

1. Schemas and an associated validation procedure for validating document syntax against those schemas. (The validation procedure includes un-zipping, locating files, processing the extensibility elements and attributes, and XML Schema validation.)
2. Additional syntax constraints in written form, wherever these constraints cannot feasibly be expressed in the schema language.
3. Descriptions of element semantics. The semantics of an element refers to its intended interpretation by a human being.

2.4 Document Conformance

Document conformance is purely syntactic; it involves only Items 1 and 2 in §2.3 above.

- A conforming document shall conform to the schema (Item 1) and any additional syntax constraints (Item 2).
- The document character set shall conform to the Unicode Standard and ISO/IEC 10646-1, with either the UTF-8 or UTF-16 encoding form, as required by the XML 1.0 standard.
- Any XML element or attribute not explicitly included in this Standard shall use the extensibility mechanisms described by Parts 4 and 5 of this Standard.

2.5 Application Conformance

Application conformance is purely syntactic; it also involves only Items 1 and 2 in §2.3 above.

- A conforming consumer shall not reject any conforming documents of the document type (§4) expected by that application.
- A conforming producer shall be able to produce conforming documents.

2.6 Interoperability Guidelines

[*Guidance:* The following interoperability guidelines incorporate semantics (Item 3 in §2.3 above).

For the guidelines to be meaningful, a software application should be accompanied by publicly available documentation that describes what subset of this Standard it supports. The documentation should highlight

1 any behaviors that would, without that documentation, appear to violate the semantics of document
2 elements. Together, the application and documentation should satisfy the following conditions.

- 3 1. The application need not implement operations on all elements defined in this Standard. However, if it
4 does implement an operation on a given element, then that operation should use semantics for that
5 element that are consistent with this Standard.
- 6 2. If the application moves, adds, modifies, or removes element instances with the effect of altering
7 document semantics, it should declare the behavior in its documentation.

8 The following scenarios illustrate these guidelines.

- 9 • A presentation editor that interprets the preset shape geometry “rect” as an ellipse does not observe
10 the first guideline because it implements “rect” but with incorrect semantics.
- 11 • A batch spreadsheet processor that saves only computed values even if the originally consumed cells
12 contain formulas, may satisfy the first condition, but does not observe the second because the
13 editability of the formulas is part of the cells’ semantics. To observe the second guideline, its
14 documentation should describe the behavior.
- 15 • A batch tool that reads a word-processing document and reverses the order of text characters in every
16 paragraph with “Title” style before saving it can be conforming even though this Standard does not
17 anticipate this behavior. This tool’s behavior would be to transform the title “Office Open XML” into
18 “LMX nepO eciffo”. Its documentation should declare its effect on such paragraphs. *end guidance*

3. Normative References

The following normative documents contain provisions, which, through reference in this text, constitute provisions of this Standard. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. However, parties to agreements based on this Standard are encouraged to investigate the possibility of applying the most recent editions of the normative documents indicated below. For undated references, the latest edition of the normative document referred to applies. Members of ISO and IEC maintain registers of currently valid International Standards.

8

9 ISO/IEC 2382.1:1993, *Information technology — Vocabulary — Part 1: Fundamental terms*.

10 ISO/IEC 10646:2003 (all parts), *Information technology — Universal Multiple-Octet Coded Character Set (UCS)*.

4. Definitions

For the purposes of this Standard, the following definitions apply. Other terms are defined where they appear in *italic* type or on the left side of a syntax rule. Terms explicitly defined in this Standard are not to be presumed to refer implicitly to similar terms defined elsewhere. [*Note*: This part uses OPC-related terms, which are defined in Part 2: "Open Packaging Conventions". *end note*]

application — A consumer or producer.

behavior — External appearance or action.

behavior, implementation-defined — Unspecified behavior where each implementation documents that behavior, thereby promoting predictability and reproducibility within any given implementation. (This term is sometimes called "application-specific behavior".)

behavior, locale-specific — Behavior that depends on local conventions of nationality, culture, and language.

behavior, unspecified — Behavior where this Standard imposes no requirements. [*Note*: To add an extension, an implementer must use the extensibility mechanisms described by this Standard rather than trying to do so by giving meaning to otherwise unspecified behavior. *end note*]

document type — One of the three types of Office Open XML documents: Wordprocessing, Spreadsheet, and Presentation, defined as follows:

- A document whose package-relationship item contains a relationship to a Main Document part (§11.3.10) is a document of type Wordprocessing.
- A document whose package-relationship item contains a relationship to a Workbook part (§12.3.23) is a document of type Spreadsheet.
- A document whose package-relationship item contains a relationship to a Presentation part (§13.3.6) is a document of type Presentation.

An Office Open XML document can contain one or more embedded Office Open XML packages (§15.2.10) with each embedded package having any of the three document types. However, the presence of these embedded packages does not change the type of the document.

DrawingML — A set of conventions for specifying the location and appearance of drawing elements in an Office Open XML document.

extension — Any XML element or attribute not explicitly included in this Standard, but that uses the extensibility mechanisms described by this Standard.

- 1 **Office Open XML document** — A package containing ZIP items as required by, and satisfying Parts 1 and 4 of,
2 this Standard. A rendition of a data stream formatted using the wordprocessing, spreadsheet, or presentation
3 ML and its related MLs as described in this Standard. Such a document is represented as a package.
- 4 **package**— A ZIP archive that conforms to the Open Packaging Conventions specification defined in Part 2 of
5 this Standard.
- 6 **package, embedded**— A package that has been stored as the target of a valid Embedded Package relationship
7 (§15.2.10) in an Office Open XML document
- 8 **PresentationML** — A set of conventions for representing an Office Open XML document of type Presentation.
- 9 **relationship** —The kind of connection between a source part and a target part in a package. Relationships
10 make the connections between parts directly discoverable without looking at the content in the parts, and
11 without altering the parts themselves. (See also Package Relationships.)
- 12 **relationships part** — A part containing an XML representation of relationships.
- 13 **relationship, explicit** — A relationship in which a resource is referenced from a source part's XML using the
14 Id attribute of a Relationship tag.
- 15 **relationship, implicit** — A relationship that is not explicit.
- 16 **SpreadsheetML** — A set of conventions for representing an Office Open XML document of type Spreadsheet.
- 17 **WordprocessingML** — A set of conventions for representing an Office Open XML document of type
18 Wordprocessing.

5. Notational Conventions

The following typographical conventions are used in this Standard:

1. The first occurrence of a new term is written in italics. [*Example: ... is considered normative. end example*]
2. A term defined as a basic definition is written in bold. [*Example: **behavior** — External ... end example*]
3. The name of an XML element is written using an Element style. [*Example: The root element is document. end example*]
4. The name of an XML element attribute is written using an Attribute style. [*Example: ... an id attribute. end example*]
5. An XML element attribute value is written using a constant-width style. [*Example: ... value of CommentReference. end example*]
6. An XML element type name is written using a Type style. [*Example: ... as values of the xsd:anyURI data type. end example*]

1 6. Acronyms and Abbreviations

2 **This clause is informative**

3 The following acronyms and abbreviations are used throughout this Standard:

4 IEC — the International Electrotechnical Commission

5 ISO — the International Organization for Standardization

6 W3C — World Wide Web Consortium

7 **End of informative text**

7. General Description

This Standard is intended for use by implementers, academics, and application programmers. As such, it contains a considerable amount of explanatory material that, strictly speaking, is not necessary in a formal specification.

This Part is divided into the following subdivisions:

1. Front matter (clauses 1–7);
2. Overview (clause 8);
3. Main body (clauses 9–14);
4. Annexes

Examples are provided to illustrate possible forms of the constructions described. References are used to refer to related clauses. Notes are provided to give advice or guidance to implementers or programmers. Rationale provides explanatory material as to why something is or is not in this Standard. Annexes provide additional information or summarize the information contained in this Standard.

Clauses 1–5, 7, and 9–14 form a normative part of this Part; and the Introduction, clauses 6 and 8, as well as the annexes, notes, examples, rationale, guidance, and the index, are informative.

Except for whole clauses or annexes that are identified as being informative, informative text that is contained within normative text is indicated in the following ways:

1. [*Example*: code fragment, possibly with some narrative ... *end example*]
2. [*Note*: narrative ... *end note*]
3. [*Rationale*: narrative ... *end rationale*]
4. [*Guidance*: narrative ... *end guidance*]

1 8. Overview

2 **This clause is informative.**

3 This clause contains an overview of Office Open XML.

4 8.1 Packages and Parts

5 An Office Open XML document is represented as a series of related *parts* that are stored in a container called a
6 *package*. Information about the *relationships* between a package and its parts is stored in the package's
7 *package-relationship ZIP item*. Information about the *relationships* between two parts is stored in the *part-*
8 *relationship ZIP item* for the source part. A package is an ordinary ZIP archive, which contains that package's
9 content-type item, relationship items, and parts. (Packages are discussed further in Part 2.)

10 A WordprocessingML document contains a part for the body of the text; it might also contain a part for an
11 image referenced by that text, and parts defining document characteristics, styles, and fonts. A SpreadsheetML
12 document contains a separate part for each worksheet; it might also contain parts for images. A
13 PresentationML document contains a separate part for each slide.

14 8.2 Consumers and Producers

15 A tool that can read and understand a package is called a *consumer*, while one that can create a package is
16 called a *producer*. An application can be a consumer, a producer, or both. For example, when a word processor
17 creates a new document, it acts as a producer. When it is used to open an existing document for reading or
18 search purposes, it acts as a consumer. When it is used to open an existing document, edit it, and save the
19 result, it acts as both consumer and producer. Similar scenarios exist for spreadsheet and presentation
20 applications.

21 8.3 WordprocessingML

22 This subclause introduces the overall form of a WordprocessingML package, and identifies some of its main
23 element types. (See Part 3 for a more detailed introduction.)

24 A WordprocessingML package has a relationship of type *officeDocument*, which specifies the location of the
25 main part in the package. For a WordprocessingML document, that part contains the main text of the
26 document.

27 A WordprocessingML package's main part starts with a word processing root element. That element contains a
28 *body*, which, in turn, contains one or more *paragraphs* (as well as tables, pictures, and the like). A paragraph
29 contains one or more runs, where a *run* is a container for one or more pieces of *text* having the same set of
30 properties. Like many collection element types, each run and paragraph can have associated with it a set of
31 *properties*. For example, a run might have the property *bold*, which indicates that run's text is to be displayed
32 in a bold typeface.

1 A WordprocessingML document is organized into *sections*, and the layout of a page on which the text appears
 2 within a section is controlled by that section's properties. For example, each section can have its own *headers*
 3 and *footers*.

4 One relationship from the document part specifies the document's styles. A *style* defines a text display format.
 5 A style can have properties, which can be applied to individual paragraphs or runs. Styles make runs more
 6 compact by reducing the number of repeated definitions and properties, and the amount of work required to
 7 make changes to the document's appearance. With styles, the appearance of all the pieces of text that share a
 8 common style can be changed in one place, in that style's definition.

9 A series of paragraphs can have *numbering* applied to them via a numbering definition instance or a
 10 numbering style.

11 Data in a WordprocessingML document can be organized in a *table*, a two-dimensional grid of *cells* organized
 12 into *rows* and *columns*. Cells and whole tables can have associated properties. A cell can contain text and
 13 paragraphs, for example.

14 Text within a WordprocessingML document can be determined dynamically via the use of *fields*. Fields consist
 15 of *field instructions* (the text that dictates the field's dynamic behavior) and the *field result* (the text resulting
 16 from the dynamic calculation of the field instructions. For example, page numbers are represented as fields. A
 17 *hyperlink* consists of two pieces of information: the hyperlink itself—the text the user will click—and the target
 18 for the link. Potential targets include external files, e-mail addresses, web sites, and bookmarks within the
 19 document itself.

20 A WordprocessingML document can also contain *custom markup*, user-defined semantics applied to arbitrary
 21 document content.

22 A WordprocessingML document is not stored as one large body in a single part; instead, the elements that
 23 implement certain groupings of functionality are stored in separate parts. For example, all footnotes in a
 24 document are stored in one footnote part, while each section can have up to three different header parts and
 25 three different footer parts, to support headers and footers on odd-numbered pages, even-numbered pages,
 26 and the first page.

27 **8.4 SpreadsheetML**

28 This subclause introduces the overall form of a SpreadsheetML package, and identifies some of its main
 29 element types. (See Part 3 for a more detailed introduction.)

30 A SpreadsheetML package has a relationship of type *officeDocument*, which specifies the location of the main
 31 part in the package. For a SpreadsheetML document, that part contains the workbook definition.

32 A SpreadsheetML package's main part starts with a spreadsheet root element. That element is a *workbook*,
 33 which refers to one or more *worksheets*, which, in turn, contain the data. A worksheet is a two-dimensional
 34 grid of *cells* that are organized into *rows* and *columns*.

1 The cell is the primary place in which data is stored and operated on. A cell can have a number of
 2 characteristics, such as numeric, text, date, or time *formatting*; *alignment*; *font*; *color*; and a *border*. Each cell is
 3 identified by a *cell reference*, a combination of its column and row headings.

4 Each horizontal set of cells in a worksheet is called a *row*, and each row has a heading numbered sequentially,
 5 starting at 1. Each vertical set of cells in a worksheet is called a *column*, and each column has an alphabetic
 6 heading named sequentially from A–Z, then AA–AZ, BA–BZ, and so on.

7 Instead of data, a cell can contain a *formula*, which is a recipe for calculating a value. Some formulas—called
 8 *functions*—are predefined, while others are user-defined. Examples of predefined formulas are AVERAGE, MAX,
 9 MIN, and SUM. A function takes one or more arguments on which it operates, producing a result. For example,
 10 in the formula SUM(B1:B4), there is one argument, B1:B4, which is the range of cells B1–B4, inclusive.

11 Other features that a SpreadsheetML document can contain include the following: *comments*, *hyperlinks*,
 12 *images*, and sorted and filtered *tables*.

13 A SpreadsheetML document is not stored as one large body in a single part; instead, the elements that
 14 implement certain groupings of functionality are stored in separate parts. For example, all the data for a
 15 worksheet is stored in that worksheet's part, all string literals from all worksheets are stored in a single shared
 16 string part, and each worksheet having comments has its own comments part.

17 8.5 PresentationML

18 This subclause introduces the overall form of a PresentationML package, and identifies some of its main
 19 element types. (See Part 3 for a more detailed introduction.)

20 A PresentationML package has a relationship of type officeDocument, which specifies the location of the main
 21 part in the package. For a PresentationML document, that part contains the presentation definition.

22 A PresentationML package's main part starts with a presentation root element. That element contains a
 23 *presentation*, which, in turn, refers to a *slide* list, a *slide master* list, a *notes master* list, and a *handout master*
 24 list. The slide list refers to all of the slides in the presentation; the slide master list refers to all of the slide
 25 masters used in the presentation; the notes master contains information about the formatting of notes pages;
 26 and the handout master describes how a handout looks.

27 A *handout* is a printed set of slides that can be handed out to an *audience* for future reference.

28 As well as text and graphics, each slide can contain *comments* and *notes*, can have a *layout*, and can be part of
 29 one or more *custom presentations*. (A comment is an annotation intended for the person maintaining the
 30 presentation slide deck. A note is a reminder or piece of text intended for the presenter or the audience.)

31 Other features that a PresentationML document can contain include the following: *animation*, *audio*, *video*,
 32 and *transitions* between slides.

A PresentationML document is not stored as one large body in a single part; instead, the elements that implement certain groupings of functionality are stored in separate parts. For example, all comments in a document are stored in one comment part while each slide has its own part.

8.6 Supporting MLs

This subclause introduces the set of markup languages used across package types. (See Part 3 for a more detailed introduction.)

The three markup languages described above define the structure of a package that is either a document (WordprocessingML), a spreadsheet (SpreadsheetML), or a presentation (PresentationML). However, there is also a set of shared markup languages used for common elements such as charts, diagrams, and drawing objects. These MLs are discussed below.

8.6.1 DrawingML

DrawingML specifies the location and appearance of drawing elements in a package. For example, these elements could be, but are not limited to, shapes, pictures, and tables. The root element of a DrawingML XML fragment specifies the presence of a drawing at this location in the document.

A *shape* is a geometric object such as a circle, square, or rectangle; a *picture* is an image presented inside the document; and a *table* is a two-dimensional grid of *cells* organized into *rows* and *columns*. Cells and whole tables can have associated properties. A cell can contain text, for example.

DrawingML also specifies the location and appearance of charts in a package. The root element of a chart part is *chart*, and specifies the appearance of the chart at this location in the document.

In addition, DrawingML specifies package-wide appearance characteristics, such as the package's theme. The *theme* of a document specifies the *color scheme*, *fonts*, and *effects*, which can be referenced by parts of the document—such as text, drawings, charts, and diagrams—in order to create a consistent visual presentation.

A *chart* is a presentation of data in a graphical fashion, such as a pie chart, bar chart, line chart, in order to make trends and exceptions in the data more visually apparent.

DrawingML also specifies the location and appearance of diagrams in a document. Together, the following four parts define a diagram:

- The *data* part (§14.2.4) specifies individual items of information presented in the diagram. Typically, each piece is a simple line of text, but depending on the diagram, an item of data might also be an image.
- The *layout* part (§14.2.5) specifies how the data and shapes are laid out to create the resulting diagram.
- The *colors* part (§14.2.3) specifies the color which is applied to each individual shape in the diagram.
- The *styles* part (§14.2.6) defines how each individual shape in the diagram maps to the document's theme.

8.6.2 VML

VML specifies the appearance and content of certain shapes in a document. This is used for shapes such as text boxes, as well as shapes which must be stored to maintain compatibility with earlier versions of consumer/producer applications.

[*Note*: The VML format is a legacy format originally introduced with Office 2000 and is included and fully defined in this Standard for backwards compatibility reasons. The DrawingML format is a newer and richer format created with the goal of eventually replacing any uses of VML in the Office Open XML formats. VML should be considered a deprecated format included in Office Open XML for legacy reasons only and new applications that need a file format for drawings are strongly encouraged to use preferentially DrawingML. *end note*]

A shape definition is typically specified using two elements: `shapeData`, which stores information about the shape, and `shape`, which stores the shape definition and appearance directly.

8.6.3 Custom XML Data Properties

Custom XML Data properties allow the ability to store arbitrary XML in a package, along with schema information used by that XML.

8.6.4 File Properties

The *core file properties* of a package enable users to discover, get, and set common sets of properties from within that package, regardless of whether it's a WordprocessingML, SpreadsheetML, or PresentationML package, or another use of OPC. Such properties include creator name, creation date, title, and description.

Extended file properties are specific to Office Open XML packages. For example, for a WordprocessingML package, these properties include the number of characters, words, lines, paragraphs, and pages in the document. For a SpreadsheetML package, these properties include worksheet titles. For a PresentationML package, these properties include presentation format, the number of slides, the number of notes, and whether or not any slides are hidden.

Custom file properties are defined by the user. Examples include the name of the client for whom the document was prepared, a date/time on which some event happened, a document number, or some Boolean status flag. Each custom file property has a value, and that value has a type.

8.6.5 Math

Math specifies the structure and appearance of equations in a document; it is specified with a root element of `math`.

8.6.6 Bibliography

Bibliography specifies the structure for all references stored within a document, for use in citations or a bibliography.

End of informative text.

9. Packages

An Office Open XML document is stored as a package, whose format is defined by Part 2: "Open Packaging Conventions". This subclause contains information regarding Office Open XML's use of OPC.

Throughout this Standard, the Open Packaging Conventions are referred to by their abbreviated name, OPC.

9.1 Constraints on Office Open XML's Use of OPC

While the OPC specification is designed for the representation of Office Open XML documents, it could also support a much broader range of applications. As a result, the use of some OPC features is restricted within Office Open XML documents. These additional requirements are discussed in the following subordinate subclauses. Any requirement not mentioned here is inherited from the OPC specification.

9.1.1 Part Names

An Office Open XML part name shall contain only ASCII characters, in non-escaped or escaped form. The following ASCII characters are permitted in non-escaped form: "!", "\$", "%", "&", "'", "(", ")", "*", "+", ",", "-", ".", the decimal digits "0"–"9", ":", ";", "=", "@", the Latin alphabetic characters "A"–"Z" and "a"–"z", "_", and "~". All other ASCII characters are permitted only when escaped as an encoded triplet of the form "%HH", where H is a hexadecimal digit.]

9.1.2 Part Addressing

Parts in an Office Open XML package targeted by relationships are addressed in relationship markup through part names. External document resources targeted by a relationship can be addressed using both relative and absolute references.

9.1.3 Fragments

Fragment identifiers are supported as part of all Office Open XML external relationship targets and some Office Open XML internal relationship targets.

9.1.4 Physical Packages

Each Office Open XML document is implemented as a ZIP archive.

9.1.5 Interleaving

Office Open XML document parts shall be arranged contiguously in a package as defined by simple ordering.

Parts within an Office Open XML package shall not be interleaved. All parts shall be stored as complete ZIP items, and the interleaving functionality defined in Part 2 of this Standard shall not be used. [Note: Part 2 of this Standard specifies a method for interleaving parts, which is a very useful capability for stream processing. In order to simplify initial implementations of the Standard, interleaving is not used in this current version of

1 the Office Open XML formats but it may be used in further versions of the standard or by other formats that
 2 leverage OPC. *end note*]

3 **9.1.6 Unknown Parts**

4 With the exception of relationship parts, all other parts in an Office Open XML document that are not the
 5 target of a valid relationship are considered *unknown parts*. Unknown parts shall be ignored on document
 6 consumption and can, but need not, be discarded on production.

7 **9.1.7 Trash Items**

8 *Trash items* represent parts that have been discarded or are no longer in use. Trash items shall not conform to
 9 OPC part naming guidelines as defined in Part 2 and shall not be associated with a content type. All trash
 10 items shall follow the naming scheme: [trash]/hhhh.dat where h represents a hexadecimal value.

11 [Example: A package has two parts that must be updated in-place but both parts have grown beyond their
 12 growth hints. The newer updated parts are added as new ZIP items while the original parts are renamed to:

13 [trash]/0000.dat

14 [trash]/0001.dat

15 *end example*]

16 **9.1.8 Invalid Parts**

17 ZIP archive items that do not conform to OPC part naming guidelines or are not associated with a content type
 18 shall not be allowed in an Office Open XML document, with the exception of items specifically defined by
 19 Part 2: “Open Packaging Conventions” and trash items.

20 **9.1.9 Unknown Relationships**

21 All relationships not defined within this Standard are considered *unknown relationships*. Unknown
 22 relationships are valid within an Office Open XML document provided that they conform to relationship
 23 markup guidelines as defined by the OPC specification. Specifically:

- 24 • Conforming consumers shall not fail to load a document containing unknown relationships.
- 25 • Conforming producers are encouraged to roundtrip and preserve unknown relationships and their
- 26 target parts.

27 **9.2 Relationships in Office Open XML**

28 In OPC, relationships describe references from parts to other internal resources in the package or to external
 29 resources. They represent the type of connection between a source part and a target resource, and make the
 30 connection directly discoverable without looking at the part contents, so they are quick to resolve.

31 The same ZIP item can be the target of multiple relationships. [Note: Having multiple paths to a target can
 32 make access to that target more convenient. *end note*]

1 Office Open XML imposes constraints on relationships, described in subsequent clauses of this part.
 2 Relationships in Office Open XML are either explicit or implicit.

3 For an explicit relationship, a resource is referenced from a source part's XML using the Id attribute of a
 4 Relationship tag. [Example: A document part can have a relationship to a hyperlink only if that hyperlink's
 5 Relationship element's Id attribute value is referenced explicitly by the document part's XML. *end example*]
 6 [Note: Because this mechanism is used generically across multiple tag types, explicit relationships can be
 7 extracted from an Office Open XML document without prior knowledge of tag semantics. *end note*]. Certain
 8 relationships shall be *explicit*. All other relationships are *implicit* [Note: The syntax for specifying an implicit
 9 relationship varies among tag types. *end note*]. Relationships that are required or permitted, and restrictions
 10 on those relationships are described in §10–15 of this Part.

11 [Example: Consider a WordprocessingML document that contains the following footnote sentence fragment,
 12 "... produced by Ecma¹ (<http://www.ecma-international.org/>).", which contains a footnote and a hyperlink to a
 13 web site. The relationship from a source to a footnote is implicit while that to a hyperlink is explicit.

14 The Main Document part's relationship file contains the following:

```
15 <Relationships ...>
16   <Relationship Id="rId5" Type=".../footnotes"
17     Target="footnotes.xml"/>
18   <Relationship Id="rId7" Type=".../hyperlink"
19     Target="http://www.ecma-international.org/" TargetMode="External"/>
20 </Relationships>
```

21 All footnotes for a WordprocessingML document are contained in the same Footnotes part. Let's look at how
 22 the Main Document refers to the footnote. At the point at which the footnote reference is inserted, the
 23 following XML is present:

```
24 <w:r>
25   <w:footnoteReference w:id="2"/>
26 </w:r>
```

27 The w:id="2" refers to the footnote with id=2 in the Footnotes part, the relevant piece of which is:

```
28 <w:footnote w:id="2">
29   ...
30   Ecma is an international standards development organization (SDO).
31   ...
32 </w:footnote>
```

33 In the case of the hyperlink, the main document part makes an explicit reference to this relationship when it
 34 refers to the hyperlink, by using the following:

```

1      <w:hyperlink r:id="rId7" w:history="1">
2      ...
3      </w:hyperlink>

```

4 The important distinction here is that there is no explicit reference to a relationship ID designating the
5 Footnotes part. The reference to the footnote with `id=2` is “understood” to be in the Footnotes part that must
6 always exist if there are any footnotes in the document. *end example*]

7 [Example: The following figure shows how the source, relationship item, and the target relate to each other for
8 implicit and explicit relationships, respectively. The target does not have to be a file, however.

9 The dots correspond to attributes of relevant elements. Where one attribute refers to a piece in another part,
10 this is indicated by arrows. Solid arrows indicate that the value of the source directly corresponds to the value
11 at the target (for instance, `id=rId4` in the source part corresponds to `id=rId4` in the relationship item).

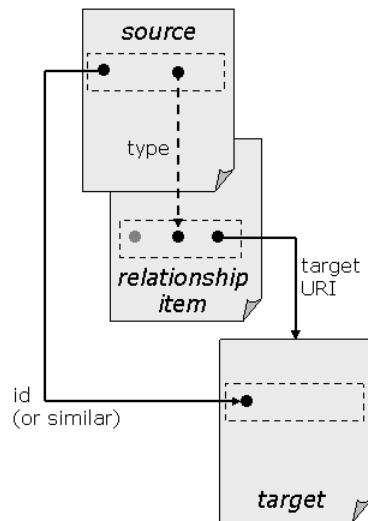
12 Dotted arrows indicate that the value of the source only implicit corresponds to the value of the target (for
13 instance, “footnoteReference” in the source indicates the type “footnotes” in the relationship item).

14 The main difference between the two types of relationship is that for implicit relationships, the id of the
15 reference refers to an element with the same id in the target part, whereas for explicit relationships, the id
16 refers to a relationship with the same id in the relationship item.

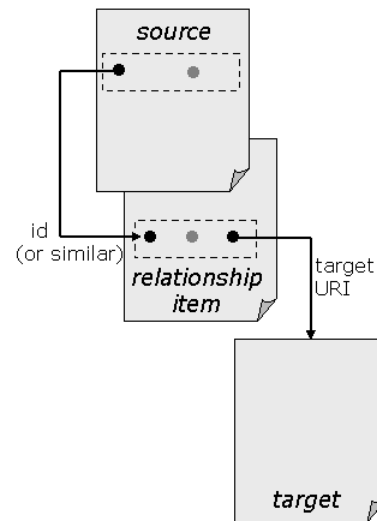
17 Both relationship types use the target URI of the relationship in the relationship item to locate the target.

18 For explicit relationships, the id in the source XML links to a relationship item with a direct explicit reference to
19 the target. For implicit relationships, the relationship item is implied by the containing tag (e.g., footnote) and
20 the id in the source XML is used to locate the correct element within the implied target.

Implicit relationship

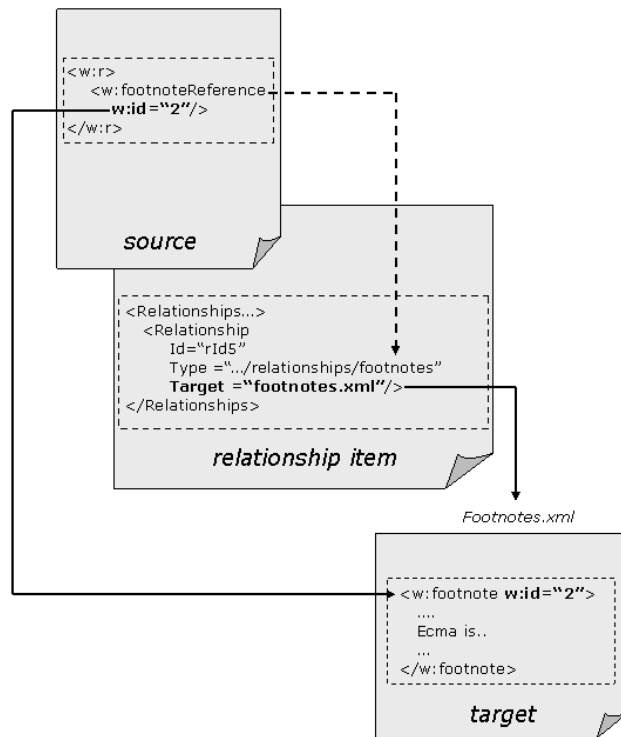


Explicit relationship



- 1
- 2 *end example]*
- 3 *[Example:* The following figure shows the implicit relationship for the footnote example described earlier.

Example Implicit relationship

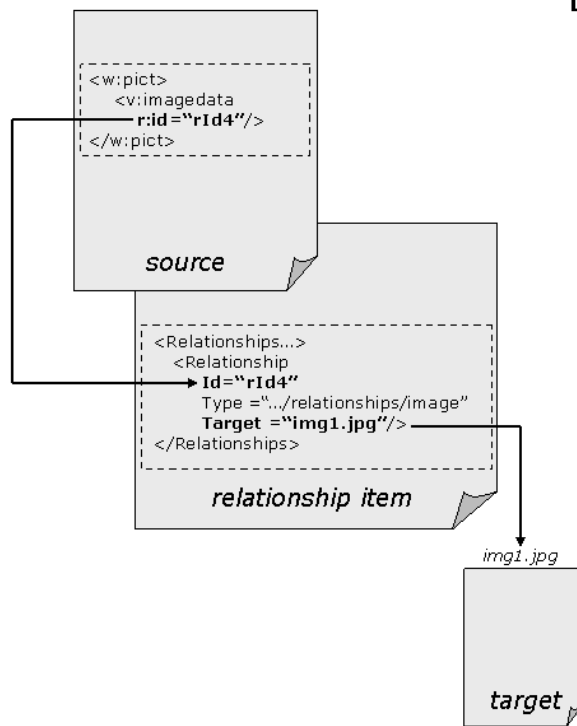


1

2 *end example]*

3 *[Example:* The following figure shows an explicit relationship.

Example Explicit relationship



1

2 *end example]*

10. Markup Compatibility and Extensibility

Office Open XML documents are designed to allow for innovation by extending their capabilities via a scheme defined by Part 5: "Markup Compatibility and Extensibility". This subclause contains information regarding Office Open XML's use of the Markup Compatibility constructs.

10.1 Constraints on Office Open XML's Use of Markup Compatibility and Extensibility

While the Markup Compatibility and Extensibility specification is designed for and used by Office Open XML documents, it could also be used to support a much broader range of applications. As a result, the use of some Markup Compatibility and Extensibility features is restricted within Office Open XML documents. These additional requirements are discussed in the following subordinate subclauses. Any requirement not mentioned here is inherited from the Markup Compatibility and Extensibility specification.

10.1.1 PreserveElements and PreserveAttributes

The PreserveElements and PreserveAttributes elements, as defined in Part 5, allow a markup language to specify the conditions under which extensions should be round-tripped, even when their contents are edited. Within the context of the markup languages explicitly defined by this Standard, no such conditions are specified, and therefore applications are not obliged to support these hints at any point in an Office Open XML document. Instead, the well-defined extensibility constructs defined below should be used.

All other constructs defined in Part 5 shall be supported.

10.1.2 Office Open XML Native Extensibility Constructs

Clause 12 of Part 5 specifies the ability for a markup language to define additional constructs for extensibility of a specific markup language. Within the context of Office Open XML documents, the extLst element(s) defined in individual markup languages shall allow the round-tripping of all unknown content regardless of the state of the PreserveElements and PreserveAttributes elements. See Part 4 for additional information on the valid use of the extLst construct.

11. WordprocessingML

This clause contains specifications for relationship items and parts that are specific to WordprocessingML. Parts that can occur in a WordprocessingML document, but are not WordprocessingML-specific, are specified in §15.2. Unless stated explicitly, all references to relationship items, content-type items, and parts in this clause refer to WordprocessingML ZIP items.

11.1 Glossary of WordprocessingML-Specific Terms

The following terms are used in the context of a WordprocessingML document:

comment — A note that an author or reviewer attaches to a piece of text in a document. Although a consumer may chose to display comments, they are not considered part of the body of the document. A comment includes the text of the note, the comment author's name and initials, and date of creation, among other things.

document setting — A reusable element in a template. [*Note:* Such elements include boilerplate text, cover pages, equations, footers, headers, tables, text boxes, and watermarks. *end note*]

glossary document — An additional WordprocessingML document story used to store reusable fragments of rich WordprocessingML content. It is called the glossary document as this story contains one or more fragments that can be indexed and extracted by name, like items in a glossary.

master document — A document that is the parent of one or more subdocuments. [*Note:* A master document can be used to manage a multipart document, such as a book having several chapters. In such as case, the master document might contain the cover page, front matter, table of contents, and cross-reference index, while each chapter and appendix resides in its own subdocument. *end note*]

section — A portion of a document in which certain page formatting options can be set. [*Note:* A new section is created to change such properties as line numbering, number of columns, or headers and footers. *end note*]

subdocument — A piece of a master document. [*Note:* A chapter or appendix might be a subdocument in a book. *end note*]

supplementary document storage location — A part within a WordprocessingML document in which fragments of WordprocessingML content can be stored separate from the printed page. See also **glossary document**

template — A document that is a pattern for creating other documents. A template can contain text, formatting, and graphics, among other things, such that documents based on it automatically have access to these elements.

1 11.2 Package Structure

2 A WordprocessingML package shall contain a package-relationship item and a content-type item. The package-
3 relationship item shall contain implicit relationships with targets of the following type:

- 4 • One Main Document part (§11.3.10)

5 The package-relationship item is permitted to contain implicit relationships with targets of the following type:

- 6 • Digital Signature Origin (§15.2.6)
- 7 • File Property parts (§15.2.11) (Application-Defined File Properties, Core File Properties, and Custom
8 File Properties), as appropriate.
- 9 • Thumbnail (§15.2.14).

10 The required and optional relationships between parts are defined in §11.3 and its subordinate clauses.

11 *[Example:* The following package represents the minimal conformant WordprocessingML package as defined
12 *by this Standard:*

13 First, the content type for relationship parts and the Main Document part (the only required part) must be
14 defined (physically located at `/[Content_Types].xml` in the package):

```
15 <Types xmlns="...">
16   <Default Extension="rels"
17     ContentType="application/vnd.openxmlformats-
18       package.relationships+xml"/>
19   <Override PartName="/document.xml"
20     ContentType="application/vnd.openxmlformats-
21       officedocument.wordprocessingml.document.main+xml"/>
22 </Types>
```

23 Next, the single required relationship (the package-level relationship to the Main Document part) must be
24 defined (physically located at `/_rels/.rels` in the package):

```
25 <Relationships xmlns="...">
26   <Relationship Id="rId1"
27     Type="http://schemas.openxmlformats.org/officeDocument/2006/
28       relationships/officeDocument"
29     Target="document.xml"/>
30 </Relationships>
```

31 Finally, the minimum content for the Main Document part must be defined (physically located at
32 `/document.xml` in the package):

```

1      <w:document xmlns:w="...">
2          <w:body>
3              <w:p/>
4          </w:body>
5      </w:document>

```

6 *end example]*

7 [Example: Consider a WordprocessingML document that is an early draft of this Standard. Here's an example
8 of the hierarchical folder structure that might be used for the ZIP items in the package for that document. As
9 shown, one part, Main Document (stored in the ZIP item /word/document.xml), has its own relationship item:

10	/[Content_Types].xml	<i>Content-type item</i>
11	/_rels/.rels	<i>Package-relationship item</i>
12	/docProps/app.xml	<i>Application-Defined File Properties part</i>
13	/docProps/core.xml	<i>Core File Properties part</i>
14	/word/document.xml	<i>Main Document part</i>
15	/word/_rels/document.xml.rels	<i>Part-relationship item</i>
16	/word/comments.xml	<i>Comment part</i>
17	/word/endnotes.xml	<i>Endnotes part</i>
18	/word/fontTable.xml	<i>Font Table part</i>
19	/word/footer1.xml	<i>Footer parts</i>
20	/word/footer2.xml	
21	/word/footer3.xml	
22	/word/footer4.xml	
23	/word/footnotes.xml	<i>Footnotes part</i>
24	/word/header1.xml	<i>Header parts</i>
25	/word/header2.xml	
26	/word/header3.xml	
27	/word/header4.xml	
28	/word/header5.xml	
29	/word/header6.xml	
30	/word/numbering.xml	<i>Numbering Definitions part</i>
31	/word/settings.xml	<i>Document Settings part</i>
32	/word/styles.xml	<i>Style Definitions part</i>
33	/word/theme/theme1.xml	<i>Theme part</i>

34 The package-relationship item contains the following:

```

35      <Relationships xmlns="...">
36          <Relationship Id="rId3"
37              Type="http://.../extended-properties" Target="docProps/app.xml"/>
38          <Relationship Id="rId2"
39              Type="http://.../core-properties" Target="docProps/core.xml"/>

```

```

1      <Relationship Id="rId1"
2          Type="http://.../officeDocument" Target="word/document.xml"/>
3      </Relationships>

```

4 *end example]*

5 11.3 Part Summary

6 The subclauses subordinate to this one describe in detail each of the part types specific to WordprocessingML.

7 [Note: For convenience, information from those subclauses is summarized in the following table:

Part	Relationship Target of	Root Element	Ref.
Alternative Format Import	Comments, Endnotes, Footer, Footnotes, Header, or Main Document	Not applicable	§11.3.1
Comments	Glossary Document or Main Document	comments	§11.3.2
Document Settings	Glossary Document or Main Document	settings	§11.3.3
Endnotes	Glossary Document or Main Document	endnotes	§11.3.4
Font Table	Glossary Document or Main Document	fonts	§11.3.5
Footer	Glossary Document or Main Document	fttr	§11.3.6
Footnotes	Glossary Document or Main Document	footnotes	§11.3.7
Glossary Document	Main Document	glossaryDocument	§11.3.8
Header	Glossary Document or Main Document	hdr	§11.3.9
Main Document	WordprocessingML package	document	§11.3.10
Numbering Definitions	Glossary Document or Main Document	numbering	§11.3.11
Style Definitions	Glossary Document or Main Document	styles	§11.3.12
Web Settings	Glossary Document or Main Document	webSettings	§11.3.13

8 *end note]*

1 11.3.1 Alternative Format Import Part

Content Type:	Any content, support for which is application-defined. [<i>Note</i> : Some examples of formats which might be supported include: <ul style="list-style-type: none"> • Text = application/txt • RTF = application/rtf • HTML = application/html • XML = application/xml <i>end note</i>]
Root Namespace:	not applicable
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/aFChunk

2

3 An alternative format import part allows content specified in an alternate format (HTML, MHTML, RTF, earlier
4 versions of WordprocessingML, or plain text) to be embedded directly in a WordprocessingML document in
5 order to allow that content to be migrated to the WordprocessingML format.

6 Any document part that permits a `p` element can also contain an `altChunk` element, whose `id` attribute refers
7 to a relationship. That relationship shall target a part within the package, which contains the content to be
8 imported into this WordprocessingML document.

9 A package is permitted to contain zero or more Alternative Format Import parts, each of which shall have a
10 corresponding alternate format file that is the target of an explicit relationship from a Comments (§11.3.2),
11 Endnotes (§11.3.4), Footer (§11.3.6), Footnotes (§11.3.7), Header (§11.3.9), or Main Document (§11.3.10) part.
12 This relationship shall be explicitly referenced using its relationship ID in the source part using the appropriate
13 XML syntax (i.e.; in the `id` attribute on the `altChunk` element), and the presence of this relationship without
14 such a reference shall be considered invalid.

15 A WordprocessingML consumer shall treat the contents of such legacy text files as if they were formatted
16 using equivalent WordprocessingML, and if that consumer is also a WordprocessingML producer, it shall emit
17 the legacy text in WordprocessingML format.

18 This Standard does not specify how one might create a WordprocessingML package that contains Alternative
19 Format Import relationships and `altChunk` elements.

20 However, a conforming producer shall not create a WordprocessingML package that contains Alternative
21 Format Import relationships and elements. [*Note*: The Alternative Format Import machinery provides a one-
22 time conversion facility. A producer could have an extension that allows it to generate a package containing
23 these relationships and elements, yet when run in conforming mode, does not do so. *end note*]

24 [*Example*: The following Main Document part-relationship item contains a relationship to an Alternative
25 Format Import part:


```

1    <Relationships xmlns="...">
2      <Relationship Id="rId5"
3        Type="http://.../aFChunk" Target="Demo.html"
4        TargetMode="Internal"/>
5    </Relationships>

```

6 The Main Document part contains the following XML fragment:

```

7    <w:body>
8      ...
9      <w:p/>
10     <w:altChunk r:id="rId5"/>
11     <w:p/>
12     ...
13 </w:body>

```

14 which results in the entire contents of Demo.html being converted and brought into the document at that
15 point (assuming that the content type of Demo.html is supported by the application consuming this
16 WordprocessingML file). *end example*]

17 An Alternative Format Import part shall be located within the package containing the source relationship
18 (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

19 An Alternative Format Import part shall not have any explicit or implicit relationships to parts defined by this
20 Standard.

21 11.3.2 Comments Part

Content Type:	application/vnd.openxmlformats-officedocument.wordprocessingml.comments+xml
Root Namespace:	http://schemas.openxmlformats.org/wordprocessingml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/comments

22

23 An instance of this part type contains the information about each comment in the document.

24 A package shall contain no more than two Comments parts. If it exists, one instance of that part shall be the
25 target of an implicit relationship from the Main Document (§11.3.10) part, and the other shall be the target of
26 an implicit relationship from the Glossary Document (§11.3.8) part.

27 *[Example:* The following Main Document part-relationship item contains a relationship to the Contents part,
28 which is stored as the ZIP item comment.xml:

```

1      <Relationships xmlns="...">
2          <Relationship Id="rId93"
3              Type="http://.../comments" Target="comments.xml"/>
4      </Relationships>

```

5 *end example]*

6 The root element for a Comment part shall be comments.

7 *[Example:*

```

8      <w:comments ... >
9          <w:comment>
10             ...
11          </w:comment>
12          ...
13      </w:comments>

```

14 *end example]*

15 The XML markup for a comment in a Main Document part uses the commentReference element.

16 *[Example:* Consider the case in which the Main Document part contains the text "... in the Standard.", and
 17 there is an comment inserted immediately after the period:

```

18      <w:p ...>
19          ...
20          <w:r>
21              <w:t>... in the Standard.</w:t>
22          </w:r>
23          <w:r>
24              <w:commentReference w:id="1"/>
25          </w:r>
26      </w:p>

```

27 *end example]*

28 Each comment has a corresponding comment element in the Comments part, which contains the text of the
 29 comment.

30 *[Example:* The text of the comment is "This is my comment.":

```

1      <w:comments xmlns:w="..."
2          <w:comment w:id="1">
3              <w:p>
4                  <w:r>
5                      <w:t>This is my comment.</w:t>
6                  </w:p>
7              </w:comment>
8          </w:comments>

```

9 *end example]*

10 A Comments part shall be located within the package containing the source relationship (expressed
11 syntactically, the TargetMode attribute of the Relationship element shall be Internal).

12 A Comments part is permitted to contain explicit relationships to the following parts defined by this Standard:

- 13 • Alternative Format Import (§11.3.1)
- 14 • Chart (§14.2.1)
- 15 • Diagrams: Diagram Colors (§14.2.3), Diagram Data (§14.2.4), Diagram Layout Definition (§14.2.5) and
16 Diagram Styles (§14.2.6)
- 17 • Embedded Control Persistence (§15.2.8)
- 18 • Embedded Object (§15.2.9)
- 19 • Embedded Package (§15.2.10)
- 20 • Hyperlinks (§15.3)
- 21 • Images (§15.2.13)
- 22 • Video (§15.2.16)

23 A Comments part shall not have any implicit or explicit relationships to any other part defined by this Standard.

24 11.3.3 Document Settings Part

Content Type:	application/vnd.openxmlformats-officedocument.wordprocessingml.settings+xml
Root Namespace:	http://schemas.openxmlformats.org/wordprocessingml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/settings

26 An instance of this part type contains all the document's properties.

27 A package shall contain no more than two Document Settings parts. If it exists, one instance of that part shall
28 be the target of an implicit relationship from the Main Document (§11.3.10) part, and the other shall be the
29 target of an implicit relationship from the Glossary Document (§11.3.8) part.

1 *[Example:* The following Main Document part-relationship item contains a relationship to a Document Settings
 2 part, which is stored in the ZIP item documentProperties1.xml:

```
3     <Relationships xmlns="...">
4         <Relationship Id="rId4"
5             Type="http://.../settings" Target="settings.xml"/>
6     </Relationships>
```

7 *end example]*

8 The root element for a part of this content type shall be settings.

9 *[Example:*

```
10     <w:settings ... >
11         ...
12         <w:defaultTabStop w:val="360"/>
13         <w:footnotePr>
14             ...
15         </w:footnotePr>
16         <w:endnotePr>
17             ...
18         </w:endnotePr>
19         <w:rsids>
20             ...
21         </w:rsids>
22         ...
23     </w:settings>
```

24 *end example]*

25 A Document Settings part shall be located within the package containing the source relationship (expressed
 26 syntactically, the TargetMode attribute of the Relationship element shall be Internal).

27 A Document Settings part is permitted to contain explicit relationships to the following parts defined by this
 28 Standard:

- 29 • Document Template (§11.4)
- 30 • Mail Merge Data Source (§11.7)
- 31 • Mail Merge Header Data Source (§11.8)
- 32 • XSL Transformation (§11.9)

33 A Document Settings part shall not have any implicit or explicit relationships to any other part defined by this
 34 Standard.

1 11.3.4 Endnotes Part

Content Type:	application/vnd.openxmlformats-officedocument.wordprocessingml.endnotes+xml
Root Namespace:	http://schemas.openxmlformats.org/wordprocessingml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/endnotes

2

3 An instance of this part type contains all the endnotes for the document.

4 A package shall contain no more than two Endnotes parts. If it exists, one instance of that part shall be the
5 target of an implicit relationship from the Main Document (§11.3.10) part, and the other shall be the target of
6 an implicit relationship from the Glossary Document (§11.3.8) part.

7 *[Example:* The following Main Document part-relationship item contains a relationship to the Endnotes part,
8 *which is stored as the ZIP item endnotes.xml:*

```

9     <Relationships xmlns="...">
10       <Relationship Id="rId6"
11         Type="http://.../endnotes" Target="endnotes.xml"/>
12     </Relationships>

```

13 *end example]*

14 The root element for an Endnotes part shall be endnotes.

15 *[Example:*

```

16     <w:endnotes xmlns:w="..." ...>
17       <w:endnote ...>
18         ...
19       </w:endnote>
20       <w:endnote ...>
21         ...
22       </w:endnote>
23     </w:endnotes>

```

24 *end example]*

25 The XML markup for an endnote in a Main Document part uses the endnoteReference element.

26 *[Example:* Consider the case in which the Main Document part contains the text "... in the Standard.", and
27 *there is an endnote inserted immediately after the period:*

```

1      <w:p ...>
2          ...
3          <w:r>
4              <w:t>... in the Standard.</w:t>
5          </w:r>
6          <w:r>
7              <w:rPr>
8                  <w:rStyle w:val="EndnoteReference"/>
9              </w:rPr>
10             <w:endnoteReference w:id="5"/>
11          </w:r>
12      </w:p>

```

13 *end example]*

14 Each endnote has a corresponding endnote element in the Endnotes part, which contains the text of the
 15 endnote, and the endnoteRef element.

16 [Example: The text of the endnote is "This can be downloaded from <http://www.aabbcc.com/index.html>."
 17 where "<http://www.aabbcc.com/index.html>" is marked as a hyperlink:

```

18      <w:endnotes xmlns:w="...">
19          <w:endnote w:id="5">
20              <w:p>
21                  <w:r>
22                      <w:rPr>
23                          <w:rStyle w:val="EndnoteReference"/>
24                      </w:rPr>
25                      <w:endnoteRef/>
26                  </w:r>
27                  <w:r>
28                      <w:t xml:space="preserve"> This can be downloaded from </w:t>
29                  </w:r>
30                  <w:hyperlink r:id="rId2">
31                      <w:r>
32                          <w:rPr>
33                              <w:rStyle w:val="Hyperlink"/>
34                          </w:rPr>
35                          <w:t>http://www.aabbcc.com/index.html</w:t>
36                      </w:r>
37                  </w:hyperlink>

```

```

1      <w:r>
2      <w:t>.</w:t>
3    </w:p>
4  </w:endnote>
5 </w:endnotes>

```

end example]

An Endnotes part shall be located within the package containing the source relationship (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

An Endnotes part is permitted to contain explicit relationships to the following parts defined by this Standard:

- Alternative Format Import (§11.3.1)
- Chart (§14.2.1)
- Diagrams: Diagram Colors (§14.2.3), Diagram Data (§14.2.4), Diagram Layout Definition (§14.2.5) and Diagram Styles (§14.2.6)
- Embedded Control Persistence (§15.2.8)
- Embedded Object (§15.2.9)
- Embedded Package (§15.2.10)
- Hyperlinks (§15.3)
- Images (§15.2.13)
- Video (§15.2.16)

An Endnotes part shall not have any implicit or explicit relationships to any other part defined by this Standard.

11.3.5 Font Table Part

Content Type:	application/vnd.openxmlformats-officedocument.wordprocessingml.fontTable+xml
Root Namespace:	http://schemas.openxmlformats.org/wordprocessingml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/fontTable

An instance of this part type contains information about each of the fonts used by content in the document. When a consumer reads a WordprocessingML document, it shall use this information to determine which fonts to use to display the document when the specified fonts are not available on the consumer's system.

A package shall contain no more than two Font Table parts. If it exists, one instance of that part shall be the target of an implicit relationship in the part-relationship item for the Main Document (§11.3.10) part, and the other instance shall be the target of an implicit relationship from the Glossary Document (§11.3.8) part.

[Example: The following Main Document part-relationship item contains a relationship to the Font Table part, which is stored as the ZIP item fontTable.xml:

```

1      <Relationships xmlns="...">
2        <Relationship Id="rId1"
3          Type="http://.../fontTable" Target="fontTable.xml"/>
4      </Relationships>

```

5 *end example]*

6 The root element for a part of this content type shall be fonts.

7 *[Example:*

```

8      <w:fonts ... >
9        <w:font w:name="Calibri">
10         <w:panose1 w:val="020F0502020204030204"/>
11         <w:charset w:val="00"/>
12         <w:family w:val="swiss"/>
13         <w:pitch w:val="variable"/>
14         <w:sig w:usb0="A00002EF" w:usb1="4000207B" w:usb2="00000000"
15           w:usb3="00000000" w:csb0="0000009F" w:csb1="00000000"/>
16         </w:font>
17      </w:fonts>

```

18 *end example]*

19 A Font Table part shall be located within the package containing the source relationship (expressed
20 syntactically, the TargetMode attribute of the Relationship element shall be Internal).

21 A Font Table part is permitted to contain explicit relationships to the following parts defined by this Standard:

- 22 • Fonts (§15.2.12)

23 A Font Table part shall not have any implicit or explicit relationships to any other part defined by this Standard.

24 11.3.6 Footer Part

Content Type:	application/vnd.openxmlformats-officedocument.wordprocessingml.footer+xml
Root Namespace:	http://schemas.openxmlformats.org/wordprocessingml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/footer

25

26 An instance of this part type contains the information about a footer displayed for one or more sections.

27 A package is permitted to contain zero or one Footer part for each kind of footer (first page, odd page, or even
28 page) in each section of the document. Each Footer part shall be the target of an explicit relationship in the
29 part-relationship item for the Main Document (§11.3.10) part, or the Glossary Document (§11.3.8) part.

1 [Example: The Main Document part-relationship item contains one relationship, for the odd footer part, which
2 is stored as the ZIP item footer3.xml:

```
3 <Relationships xmlns="...">
4 <Relationship Id="rId91"
5 Type="http://.../footer" Target="footer3.xml"/>
6 </Relationships>
```

7 *end example]*

8 The root element for a Footer part type shall be ftr.

9 [Example:

```
10 <w:ftr xmlns:w="..." ...>
11 ...
12 </w:ftr>
```

13 *end example]*

14 The XML markup for a footer in a section of a Main Document part involves the footerReference element in
15 that section's sectPr element which explicitly references the relationship for the header.

16 [Example: Consider the case in which a section in the Main Document part contains odd and even headers, and
17 an odd footer:

```
18 <w:document xmlns:w="...">
19 ...
20 <w:sectPr>
21 <w:footerReference w:val="rId89" w:type="default"/>
22 <w:footerReference w:val="rId90" w:type="even"/>
23 <w:footerReference w:val="rId91" w:type="first"/>
24 <w:type w:val="oddPage"/>
25 <w:pgSz w:w="11909" w:h="16834" w:code="9"/>
26 <w:pgMar w:top="1440" w:right="1152" w:bottom="1440"
27 w:left="1152" w:header="720" w:footer="720" w:gutter="0"/>
28 <w:lnNumType w:countBy="1"/>
29 <w:pgNumType w:numFmt="lowerRoman"/>
30 <w:cols w:space="720"/>
31 </w:sectPr>
32 </w:document>
```

33 *end example]*

34 Each footer has a corresponding ftr element in a Footer part, which contains the text of the footer.

[Example: Here is the odd footer corresponding to the example above. It has the page number centered and displayed using lowercase Roman numerals (as set by the pgNumType element above):

```

<w:ftr xmlns:w="...">
  <w:p>
    <w:pPr>
      <w:pStyle w:val="Centered"/>
    </w:pPr>
    <w:fldSimple w:instr="PAGE">
      <w:r>
        <w:rPr>
          <w:noProof/>
        </w:rPr>
        <w:t>i</w:t>
      </w:r>
    </w:fldSimple>
  </w:p>
</w:ftr>

```

end example]

A Footer part shall be located within the package containing the source relationship (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

A Footer part is permitted to have explicit relationships to the following parts defined by this Standard:

- Alternative Format Import (§11.3.1)
- Chart (§14.2.1)
- Diagrams: Diagram Colors (§14.2.3), Diagram Data (§14.2.4), Diagram Layout Definition (§14.2.5) and Diagram Styles (§14.2.6)
- Embedded Control Persistence (§15.2.8)
- Embedded Object (§15.2.9)
- Embedded Package (§15.2.10)
- Hyperlinks (§15.3)
- Images (§15.2.13)
- Video (§15.2.16)

A Footer part shall not have any implicit or explicit relationships to any other part defined by this Standard.

11.3.7 Footnotes Part

Content Type:	application/vnd.openxmlformats-officedocument.wordprocessingml.footnotes+xml
Root Namespace:	http://schemas.openxmlformats.org/wordprocessingml/2006/main

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/footnotes
----------------------	---

1

2 An instance of this part type contains all the footnotes for the document.

3 A package shall contain no more than two Footnotes parts. If it exists, one instance of that part shall be the
4 target of an implicit relationship from the Main Document (§11.3.10) part, and the other shall be the target of
5 an implicit relationship from the Glossary Document (§11.3.8) part.

6 *[Example:* The Main Document part-relationship item contains a relationship to the Footnotes part, which is
7 stored as the ZIP item footnotes.xml:

```
8     <Relationships xmlns="...">
9         <Relationship Id="rId5"
10             Type="http://.../footnotes" Target="footnotes.xml"/>
11     </Relationships>
```

12 *end example]*

13 The root element for a Footnotes part shall be footnotes.

14 *[Example:*

```
15     <w:footnotes xmlns:w="..." ...>
16         <w:footnote ...>
17             ...
18         </w:footnote>
19         <w:footnote ...>
20             ...
21         </w:footnote>
22     </w:footnotes>
```

23 *end example]*

24 The XML markup for a footnote in a Main Document part involves the footnoteReference element.

25 *[Example:* Consider the case in which the Main Document part contains the text "... in the Standard.", and
26 there is a footnote inserted immediately after the period:

```

1      <w:p ...>
2          ...
3      <w:r>
4          <w:t>... in the Standard.</w:t>
5      </w:r>
6      <w:r>
7          <w:rPr>
8              <w:rStyle w:val="FootnoteReference"/>
9          </w:rPr>
10         <w:footnoteReference w:id="5"/>
11     </w:r>
12 </w:p>

```

13 *end example]*

14 Each footnote has a corresponding footnote element in the Footnotes part, which contains the text of the
 15 footnote and the footnoteRef element.

16 [Example: The text of the footnote is "This can be downloaded from <http://www.aabbcc.com/index.html>."
 17 where "<http://www.aabbcc.com/index.html>" is marked as a hyperlink:

```

18 <w:footnotes xmlns:w="..."
19 <w:footnote w:id="5">
20     <w:p>
21         <w:r>
22             <w:rPr>
23                 <w:rStyle w:val="FootnoteReference"/>
24             </w:rPr>
25             <w:footnoteRef/>
26         </w:r>
27         <w:r>
28             <w:t xml:space="preserve">This can be downloaded from </w:t>
29         </w:r>
30         <w:hyperlink r:id="rId2" w:history="1">
31             <w:r>
32                 <w:rPr>
33                     <w:rStyle w:val="Hyperlink"/>
34                 </w:rPr>
35                 <w:t>http://www.aabbcc.com/index.html</w:t>
36             </w:r>
37         </w:hyperlink>

```

```

1      <w:r>
2          <w:t>.</w:t>
3      </w:r>
4  </w:p>
5  </w:footnote>
6  </w:footnotes>

```

7 *end example]*

8 A Footnotes part shall be located within the package containing the source relationship (expressed
9 syntactically, the TargetMode attribute of the Relationship element shall be Internal).

10 A Footnotes part is permitted to have explicit relationships to the following parts defined by this Standard:

- 11 • Alternative Format Import (§11.3.1)
- 12 • Chart (§14.2.1)
- 13 • Diagrams: Diagram Colors (§14.2.3), Diagram Data (§14.2.4), Diagram Layout Definition (§14.2.5) and
14 Diagram Styles (§14.2.6)
- 15 • Embedded Control Persistence (§15.2.8)
- 16 • Embedded Object (§15.2.9)
- 17 • Embedded Package (§15.2.10)
- 18 • Hyperlinks (§15.3)
- 19 • Images (§15.2.13)
- 20 • Video (§15.2.16)

21 A Footer part shall not have any implicit or explicit relationships to any other part defined by this Standard.

22 11.3.8 Glossary Document Part

Content Type:	application/vnd.openxmlformats-officedocument.wordprocessingml.document.glossary+xml
Root Namespace:	http://schemas.openxmlformats.org/wordprocessingml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/glossaryDocument

24 An instance of this part type is a supplementary document storage location which stores the definition and
25 content for content that shall be carried with the document for future insertion and/or use, but which shall not
26 be visible within the contents of the main document story. [*Example:* A legal contract template might include
27 one or more optional clauses that shall not appear in the document until those clauses are inserted explicitly
28 via a user action. To store these optional clauses until they are inserted, their contents are placed in the
29 glossary document part. *end example]*

30 [*Note:* This part is intended for storage of optional "document fragments" which are often used to perform
31 document assembly. The use of the word *glossary* is a reference to the fact that each of these entries was

1 historically referenced by its first word in legacy word processing applications, like definitions of terms in a
 2 traditional glossary. *end note*]

3 The root element for a part of this content type shall be `glossaryDocument`.

4 *[Example:* The following part contains two building blocks. The first block is named "rainbow colors", belongs
 5 to a category called "Misc", belongs to a gallery called "docParts", and contains the text "The colors ... and
 6 violet." The details of the second block have been omitted:

```

7   <w:glossaryDocument xmlns:w="..." >
8     <w:docParts>
9       <w:docPart>
10        <w:docPartPr>
11          <w:name w:val="rainbow colors"/>
12          <w:style w:val="Normal"/>
13          <w:category>
14            <w:name w:val="Misc"/>
15            <w:gallery w:val="docParts"/>
16          </w:category>
17        </w:docPartPr>
18        <w:docPartBody>
19          <w:p>
20            <w:r>
21              <w:t>The colors of the rainbow are red, orange, yellow,
22                green, blue, indigo, and violet.</w:t>
23            </w:r>
24          </w:p>
25        </w:docPartBody>
26      </w:docPart>
27    <w:docPart>
28      ...
29    </w:docPart>
30  </w:docParts>
31 </w:glossaryDocument>

```

32 *end example]*

33 A package shall contain at most one Glossary Document part, and that part shall be the target of an implicit
 34 relationship from the Main Document (§11.3.10) part.

35 *[Example:* The following Main Document part-relationship item contains a relationship to a Glossary Document
 36 part, which is stored in the ZIP item `glossary/document.xml`:

```

1      <Relationships xmlns="...">
2          <Relationship Id="rId4"
3              Type="http://.../glossaryDocument" Target="glossary/document.xml"/>
4      </Relationships>

```

5 *end example]*

6 A Glossary Document part shall be located within the package containing the source relationship (expressed
7 syntactically, the TargetMode attribute of the Relationship element shall be Internal).

8 A Glossary Document part is permitted to have implicit relationships to the following parts defined by this
9 Standard:

- 10 • Comments (§11.3.2)
- 11 • Document Settings (§11.3.3)
- 12 • Endnotes (§11.3.4)
- 13 • Font Table (§11.3.5)
- 14 • Footnotes (§11.3.7)
- 15 • Numbering Definitions (§11.3.11)
- 16 • Style Definitions (§11.3.11)

17 A Glossary Document part is permitted to have explicit relationships to the following parts defined by this
18 Standard:

- 19 • Alternative Format Import (§11.3.1)
- 20 • Chart (§14.2.1)
- 21 • Diagrams: Diagram Colors (§14.2.3), Diagram Data (§14.2.4), Diagram Layout Definition (§14.2.5) and
22 Diagram Styles (§14.2.6)
- 23 • Embedded Control Persistence (§15.2.8)
- 24 • Embedded Object (§15.2.9)
- 25 • Embedded Package (§15.2.10)
- 26 • Footer (§11.3.6)
- 27 • Header (§11.3.9)
- 28 • Hyperlinks (§15.3)
- 29 • Images (§15.2.13)
- 30 • Printer Settings (§15.2.14)
- 31 • Video (§15.2.16)

32 A Glossary Document part shall not have implicit or explicit relationships to any other part defined by this
33 Standard.

34 **11.3.9 Header Part**

Content Type:	application/vnd.openxmlformats-officedocument.wordprocessingml.header+xml
---------------	---

Root Namespace:	http://schemas.openxmlformats.org/wordprocessingml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/header

1

2 An instance of this part type contains the information about a header displayed for one or more sections.

3 A package shall contain zero or one Header part for each kind of header (first page, odd page, or even page) in
 4 each section of the document. Each Header part shall be the target of an explicit relationship from the Main
 5 Document (§11.3.10) part or the Glossary Document (§11.3.8) part.

6 *[Example:* The Main Document part-relationship item contains two relationships: one for the even header part
 7 (which is stored as the ZIP item header2.xml) and one for the odd header part (which is stored as the ZIP item
 8 header3.xml):

```

9     <Relationships xmlns="...">
10         <Relationship Id="rId89" Type="http://.../header" Target="header2.xml"/>
11         <Relationship Id="rId90" Type="http://.../header" Target="header3.xml"/>
12     </Relationships>

```

13 *end example]*

14 The root element for a Header part shall be `hdr`.

15 *[Example:*

```

16     <w:hdr xmlns:w="..." ...>
17         ...
18     </w:hdr>

```

19 *end example]*

20 The XML markup for a header in a section of a Main Document part involves the `headerReference` element in
 21 that section's `sectPr` element.

22 *[Example:* Consider the case in which a section in the Main Document part contains odd and even headers, and
 23 an odd footer:


```

1    <w:body>
2    ...
3    <w:sectPr w:rsidR="00363F31" w:rsidSect="008D4B40">
4        <w:headerReference w:val="rId89" w:type="default"/>
5        <w:headerReference w:val="rId90" w:type="even"/>
6        <w:headerReference w:val="rId91" w:type="first"/>
7        <w:type w:val="oddPage"/>
8        <w:pgSz w:w="11909" w:h="16834" w:code="9"/>
9        <w:pgMar w:top="1440" w:right="1152" w:bottom="1440"
10           w:left="1152" w:header="720" w:footer="720" w:gutter="0"/>
11        <w:lnNumType w:countBy="1"/>
12        <w:pgNumType w:fmt="lowerRoman"/>
13        <w:cols w:space="720"/>
14    </w:sectPr>
15 </w:body>

```

end example]

Each header has a corresponding `hdr` element in a Header part, which contains the text of the header.

[Example: Here is the even header corresponding to the examples above:

```

19    <w:hdr xmlns:w="...">
20        <w:p>
21            <w:pPr>
22                <w:pStyle w:val="Header"/>
23            </w:pPr>
24            <w:r>
25                <w:t>My Test Document</w:t>
26            </w:r>
27        </w:p>
28    </w:hdr>

```

Here is the odd header corresponding to the examples above:

```

1      <w:hdr xmlns:w="...">
2          <w:p>
3              <w:pPr>
4                  <w:pStyle w:val="Header"/>
5              </w:pPr>
6              <w:r>
7                  <w:tab/>
8                  <w:t>Table of Contents</w:t>
9              </w:r>
10         </w:p>
11     </w:hdr>

```

12 *end example]*

13 A Header part shall be located within the package containing the source relationship (expressed syntactically,
14 the TargetMode attribute of the Relationship element shall be Internal).

15 A Header part is permitted to have explicit relationships to the following parts defined by this Standard:

- 16 • Alternative Format Import (§11.3.1)
- 17 • Chart (§14.2.1)
- 18 • Diagrams: Diagram Colors (§14.2.3), Diagram Data (§14.2.4), Diagram Layout Definition (§14.2.5) and
19 Diagram Styles (§14.2.6)
- 20 • Embedded Control Persistence (§15.2.8)
- 21 • Embedded Object (§15.2.9)
- 22 • Embedded Package (§15.2.10)
- 23 • Hyperlinks (§15.3).
- 24 • Images (§15.2.13)
- 25 • Video (§15.2.16)

26 A Header part shall not have any implicit or explicit relationships to other parts defined by this Standard.

27 **11.3.10 Main Document Part**

Content Type(s):	application/vnd.openxmlformats-officedocument.wordprocessingml.main+xml application/vnd.openxmlformats-officedocument.wordprocessingml.template.main+xml
Root Namespace:	http://schemas.openxmlformats.org/wordprocessingml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/officeDocument

28

29 An instance of this part type contains the body of the document.

1 A package shall contain a Main Document part (§11.3.10) part. The Main Document part shall be the target of
 2 a relationship in the package-relationship item.

3 The root element for a part of this content type shall be document.

4 *[Example: Given the following package-relationship item excerpt:*

```
5     <Relationships xmlns="...">
6       <Relationship Id="rId1"
7         Type="http://.../officeDocument" Target="word/document.xml"/>
8     </Relationships>
```

9 /word/document.xml" contains the following:

```
10    <w:document ...>
11      <w:body>
12        ...
13      </w:body>
14    </w:document>
```

15 *end example]*

16 A Main Document part shall be located within the package containing the source relationship (expressed
 17 syntactically, the TargetMode attribute of the Relationship element shall be Internal).

18 A Main Document part is permitted to have implicit relationships to the following parts defined by this
 19 Standard:

- 20 • Additional Characteristics (§15.2.1)
- 21 • Bibliography (§15.2.3)
- 22 • Comments (§11.3.2)
- 23 • Custom XML Data Storage (§15.2.4)
- 24 • Document Settings (§11.3.3)
- 25 • Endnotes (§11.3.4)
- 26 • Font Table (§11.3.5)
- 27 • Footnotes (§11.3.7)
- 28 • Glossary Document (§11.3.8)
- 29 • Numbering Definitions (§11.3.11)
- 30 • Style Definitions (§11.3.12)
- 31 • Theme (§14.2.7)
- 32 • Thumbnail (§15.2.14)

33 A Main Document part is permitted to contain explicit relationships to the following parts defined by this
 34 Standard:

- Alternative Format Import (§11.3.1)
- Chart (§14.2.1)
- Diagrams: Diagram Colors (§14.2.3), Diagram Data (§14.2.4), Diagram Layout Definition (§14.2.5) and Diagram Styles (§14.2.6)
- Embedded Control Persistence (§15.2.8)
- Embedded Object (§15.2.9)
- Embedded Package (§15.2.10)
- Footer (§11.3.6)
- Header (§11.3.9)
- Hyperlinks (§15.3)
- Images (§15.2.13)
- Printer Settings (§15.2.14)
- Subdocument (§11.6)
- Video (§15.2.16)

A Main Document shall not have implicit or explicit relationships to any other part defined by this Standard.

11.3.11 Numbering Definitions Part

Content Type:	application/vnd.openxmlformats-officedocument.wordprocessingml.numbering+xml
Root Namespace:	http://schemas.openxmlformats.org/wordprocessingml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/numbering

An instance of this part type contains a definition for the structure of each unique numbering definition in this document.

[Example: If a set of paragraphs are added to a document which have a circle bullet at the first level, a square bullet at the second level, and a checkmark bullet at the third level, such as the following:

- First level
 - Second level
 - ✓ Third level

The numbering definition part will contain the definition for each of these levels (their bullet style, indent, etc.) even if the second and third levels are not actually used in the document *end example*]

A package shall contain no more than two Numbering Definitions parts. If they exist, one instance of that part shall be the target of an implicit relationship from the Main Document (§11.3.10) part, and the other shall be the target of an implicit relationship from the Glossary Document (§11.3.8) part.

1 *[Example:*

```
2     <Relationships xmlns="...">
3       <Relationship Id="rId2"
4         Type="http://.../numbering" Target="numbering.xml"/>
5     </Relationships>
```

6 *end example]*

7 The XML markup for a list usage involves a reference to a numbering definition via the child elements of the
8 numPr element.

9 *[Example:* Here we have a paragraph set using the style Text, followed by a list of things which have the
10 paragraph style ListBullet, followed by another paragraph set using the style Text:

```
11     <w:p>
12       <w:pPr>
13         <w:pStyle w:val="Text"/>
14       </w:pPr>
15       <w:r>
16         <w:t>The kinds of fruit needed are:</w:t>
17       </w:r>
18     </w:p>
19     <w:p>
20       <w:pPr>
21         <w:pStyle w:val="ListBullet"/>
22         <w:numPr>
23           <w:ilvl w:val="0" />
24           <w:numId w:val="5" />
25         </w:numPr>
26       </w:pPr>
27       <w:r>
28         <w:t>Apples</w:t>
29       </w:r>
30     </w:p>
```

```

1      <w:p>
2          <w:pPr>
3              <w:pStyle w:val="ListBullet"/>
4              <w:numPr>
5                  <w:ilvl w:val="0" />
6                  <w:numId w:val="5" />
7              </w:numPr>
8          </w:pPr>
9          <w:r>
10             <w:t>Oranges</w:t>
11          </w:r>
12      </w:p>
13      <w:p>
14          <w:pPr>
15              <w:pStyle w:val="Text"/>
16          </w:pPr>
17          <w:r>
18             <w:t>Other items may be needed too.</w:t>
19          </w:r>
20      </w:p>

```

21 *end example]*

22 The root element for a Numbering Definition part shall be numbering, with each numbering definition being
 23 defined by an abstractNum element.

24 *[Example:*

```

25      <w:numbering xmlns:w="...">
26          <w:abstractNum w:numId="11">
27              <w:nsid w:val="394E2425"/>
28              <w:multiLevelType w:val="hybridMultilevel"/>
29              <w:tmpl w:val="F628E89A"/>
30              <w:lvl w:ilvl="0" w:tplc="151C4798">
31                  <w:start w:val="1"/>
32                  <w:numFmt w:val="bullet"/>
33                  <w:pStyle w:val="ListBullet"/>
34                  <w:lvlText w:val="...">
35                  <w:lvlJc w:val="left"/>
36              <w:pPr>
37                  <w:tabs>
38                      <w:tab w:val="list" w:pos="720"/>
39                  </w:tabs>
40                  <w:ind w:left="720" w:hanging="360"/>
41              </w:pPr>

```

```

1      <w:rPr>
2          <w:rFonts w:ascii="Symbol" w:hAnsi="Symbol" w:hint="default"/>
3      </w:rPr>
4  </w:lvl>
5  ...
6  </w:abstractNum>
7  </w:numbering>

```

end example]

A Numbering Definitions part shall be located within the package containing the source relationship (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

A Numbering Definitions part is permitted to contain explicit relationships to the following parts defined by this Standard:

- Images (§15.2.13)

A Numbering Definitions part shall not have any implicit or explicit relationships to any other part defined by this Standard.

11.3.12 Style Definitions Part

Content Type:	application/vnd.openxmlformats-officedocument.wordprocessingml.styles+xml
Root Namespace:	http://schemas.openxmlformats.org/wordprocessingml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/styles

An instance of this part type contains the definition for a set of styles used by this document.

A package shall contain at most two Style Definitions parts. One instance of that part shall be the target of an implicit relationship from the Main Document (§11.3.10) part, and the other shall be the target of an implicit relationship in from the Glossary Document (§11.3.8) part.

[Example:

```

23  <Relationships xmlns="...">
24      <Relationship Id="rId3"
25          Type="http://.../styles" Target="styles.xml"/>
26  </Relationships>

```

end example]

The root element for a Styles Definition part shall be styles, which is a container for one or more style elements.

[Example: Here is the style ListBullet (which is used in a Main Document Part in §11.3.10):

```
<w:styles xmlns:wx="..." xmlns:w="..." ... xml:space="preserve">
  <w:style w:type="paragraph" w:styleId="ListBullet">
    <w:name w:val="List Bullet"/>
    <w:basedOn w:val="Text"/>
    <w:autoRedefine/>
    <w:rsid w:val="00081289"/>
    <w:pPr>
      <w:pStyle w:val="ListBullet"/>
      <w:numPr>
        <w:numId w:val="1"/>
      </w:numPr>
      <w:tabs>
        <w:tab w:val="clear" w:pos="360"/>
      </w:tabs>
      <w:ind w:left="648"/>
    </w:pPr>
  </w:style>
</w:styles>
```

end example]

A Style Definitions part shall be located within the package containing the source relationship (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

A Style Definitions part shall not have implicit or explicit relationships to any part defined by this Standard.

11.3.13 Web Settings Part

Content Type:	application/vnd.openxmlformats-officedocument.wordprocessingml.webSettings+xml
Root Namespace:	http://schemas.openxmlformats.org/wordprocessingml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/webSettings

An instance of this part type contains the definition for web-specific settings used by this document.

A package shall contain at most two Web Settings parts. One instance of that part shall be the target of an implicit relationship from the Main Document (§11.3.10) part, and the other shall be the target of an implicit relationship from the Glossary Document (§11.3.8) part.

[Example:


```

1      <Relationships xmlns="...">
2          <Relationship Id="rId3"
3              Type="http://.../webSettings" Target="webSettings.xml"/>
4      </Relationships>

```

5 *end example]*

6 The root element for a Web Settings part shall be webSettings.

7 *[Example:*

```

8      <w:webSettings ...>
9          <w:frameset>
10             ...
11             <w:frame>
12                 <w:sz w:val="216" />
13                 <w:name w:val="Frame2" />
14                 <w:sourceFileName r:id="rId1" />
15             </w:frame>
16             <w:frame>
17                 <w:name w:val="Frame1" />
18                 <w:sourceFileName r:id="rId2" />
19             </w:frame>
20          </w:frameset>
21      </w:webSettings>

```

22 *end example]*

23 A Web Settings part shall be located within the package containing the source relationship (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

25 A Web Settings part is permitted to contain explicit relationships to the following parts defined by this Standard:

- 27 • Frameset (§11.5)

28 A Web Settings part shall not contain implicit or explicit relationships to any other part defined by this Standard.

30 11.4 Document Template

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/attachedTemplate
----------------------	---

31

32 A *document template* can be represented by an instance of a WordprocessingML package, and contains styles, numbering definitions, and so on that are made available when documents based on that template are edited.

1 A WordprocessingML document can refer to another document as its document template, by having a
 2 Document Settings part (§11.3.3) that contains an explicit relationship to the file location of the necessary
 3 document template using the id attribute on the attachedTemplate element.

4 [Example: Consider a document specifying a document template located at c:\template.docx:

```
5     <Relationships xmlns="...">
6       <Relationship Id="rId1"
7         Type="http://.../attachedTemplate" Target="file:///c:\template.docx"
8         TargetMode="External"/>
9     </Relationships>
```

10 The document's Document Settings part contains an attachedTemplate element that explicitly references this
 11 relationship:

```
12     <w:settings ... >
13       <w:attachedTemplate r:id="rId1"/>
14     </w:settings>
```

15 *end example]*

16 11.5 Framesets

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/frame
----------------------	---

17

18 A *frameset* is a WordprocessingML document which specifies the location and placement of other
 19 WordprocessingML documents (which, when used in this context, are referred to as *frames*). A frameset shall
 20 be represented by an instance of a WordprocessingML document with a Web Settings part (§11.3.13) whose
 21 relationship item targets each of that frameset's frames.

22 [Example: Consider a frameset document having two frames. The frameset's Web Settings part-relationships
 23 item contains the following, in which frame1.docx and frame2.docx are packages containing the corresponding
 24 frames:

```
25     <Relationships xmlns="...">
26       <Relationship Id="rId1"
27         Type="http://.../frame" Target="frame1.docx" TargetMode="External"/>
28       <Relationship Id="rId2"
29         Type="http://.../frame" Target="frame2.docx" TargetMode="External"/>
30     </Relationships>
```

31 The frameset document's Web Settings part contains a frameset element that references its frames:

```

1    <w:webSettings ...>
2    <w:frameset>
3    ...
4    <w:frame>
5        <w:sz w:val="216" />
6        <w:name w:val="Frame2" />
7        <w:sourceFileName r:id="rId1" />
8    </w:frame>
9    <w:frame>
10        <w:name w:val="Frame1" />
11        <w:sourceFileName r:id="rId2" />
12    </w:frame>
13 </w:frameset>
14 </w:webSettings>

```

end example]

A frame shall be represented by an instance of a WordprocessingML package.

A frame shall be located external to the package containing the source relationship (expressed syntactically, the TargetMode attribute of the Relationship element shall be External).

11.6 Master Documents and Subdocuments

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/subDocument
----------------------	---

A master document shall be represented by an instance of a WordprocessingML document whose Main Document (§11.3.10) part targets each of that master document's subdocuments.

[Rationale: Sometimes, it is convenient to deal with a document as a collection of pieces, especially when those pieces might be edited by different authors in a collaborative group. Perhaps it simply makes sense to think about a book as a collection of chapters rather than as one big document. The breaking-up of a document into such pieces can be achieved by having a master document with one or more subdocuments.
end rationale]

[Example: Consider a master document, whose three subdocuments are called Start, Middle, and End, respectively. Master's Main Document part has a corresponding relationships part that contains the following, in which Start.docx, Middle.docx, and End.docx are packages containing the corresponding subdocuments:

```

31    <Relationships xmlns="...">
32        <Relationship Id="rId5"
33            Type="http://.../subDocument"
34            Target="Start.docx" TargetMode="External"/>

```

```

1      <Relationship Id="rId6"
2        Type="http://.../SubDocument"
3        Target="Middle.docx" TargetMode="External"/>
4      <Relationship Id="rId7"
5        Type="http://.../SubDocument"
6        Target="End.docx" TargetMode="External"/>
7    </Relationships>

```

The master document's Main Document part contains subDoc elements that reference its subdocuments:

```

9    <w:document xmlns:r="..." xmlns:wx="..." ...>
10      <w:body>
11        <w:p ...>
12          <w:pPr>
13            ...
14          </w:pPr>
15        </w:p>
16        <w:subDoc r:id="rId5"/>
17        ...
18        <w:subDoc r:id="rId6"/>
19        ...
20        <w:subDoc r:id="rId7"/>
21        ...
22      </w:body>
23    </w:document>

```

end example]

A subdocument shall be represented by an instance of a WordprocessingML package.

A subdocument shall be located external to the package containing the source relationship (expressed syntactically, the TargetMode attribute of the Relationship element shall be External).

11.7 Mail Merge Data Source

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/mailMergeSource
----------------------	---

A document that stores information about a mail merge operation is permitted to contain a Document Settings part (§11.3.3) whose relationship item targets the file location of the necessary data source using this relationship.

[Example: Consider a document specifying a mail merge whose data source is located at <http://www.openxmlformats.org/data.txt>:

```

1    <Relationships xmlns="...">
2      <Relationship Id="rId1"
3        Type="http://.../mailMergeSource"
4        Target="http://www.openxmlformats.org/data.txt"
5        TargetMode="External"/>
6    </Relationships>

```

7 The document's Document Settings part contains a dataSource element that explicitly references this
8 relationship:

```

9    <w:settings ...>
10      <w:mailMerge>
11        ...
12        <w:dataSource r:id="rId1" />
13        ...
14      </w:mailMerge>
15    </w:settings>

```

16 *end example]*

17 A mail merge data source shall be located external to the package containing the source relationship
18 (expressed syntactically, the TargetMode attribute of the Relationship element shall be External).

19 11.8 Mail Merge Header Data Source

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/mailMergeHeaderSource
----------------------	---

20

21 A document that stores information about a mail merge operation is permitted to contain a Document Settings
22 part (§11.3.3) whose relationship item targets the file location of the necessary header data source using this
23 relationship.

24 *[Example:* Consider a document specifying a mail merge whose header data source is located at
25 `http://www.openxmlformats.org/header.txt`:

```

26    <Relationships xmlns="...">
27      <Relationship Id="rId2"
28        Type="http://.../mailMergeHeaderSource"
29        Target=http://www.openxmlformats.org/header.txt
30        TargetMode="External"/>
31    </Relationships>

```

32 The document's Document Settings part contains a headerSource element that explicitly references this
33 relationship:

```

1      <w:settings ...>
2          <w:mailMerge>
3              ...
4              <w:headerSource r:id="rId2" />
5              ...
6          </w:mailMerge>
7      </w:settings>

```

8 *end example]*

9 A mail merge header data source shall be located external to the package containing the source relationship
10 (expressed syntactically, the TargetMode attribute of the Relationship element shall be External).

11 11.9 XSL Transformation

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/transform
----------------------	---

12
13 A document can store information about an XSL Transformation which might be applied on save, by containing
14 a Document Settings part (§11.3.3) whose part relationship item contains an explicit relationship to the file
15 location of the XSL Transformation using this relationship.

16 [Example: Consider a document specifying an XSL Transformation located at
17 <http://www.openxmlformats.org/test.xsl>:

```

18      <Relationships xmlns="...">
19          <Relationship Id="rId8" Type="http://.../transform"
20              Target="http://www.openxmlformats.org/test.xsl"
21              TargetMode="External"/>
22      </Relationships>

```

23 The document's Document Settings part contains a saveThroughXslt element that explicitly references this
24 relationship:

```

25      <w:settings ...>
26          ...
27          <w:saveThroughXslt r:id="rId8" />
28          ...
29      </w:settings>

```

30 *end example]*

31 An XSL transformation shall be located external to the package containing the source relationship (expressed
32 syntactically, the TargetMode attribute of the Relationship element shall be External).

12. SpreadsheetML

This clause contains specifications for relationship items and parts that are specific to SpreadsheetML. Parts than can occur in a SpreadsheetML document, but are not SpreadsheetML-specific, are specified in §15.2. Unless stated explicitly, all references to relationship items, content-type items, and parts in this clause refer to SpreadsheetML ZIP items.

12.1 Glossary of SpreadsheetML-Specific Terms

The following terms are used in the context of a SpreadsheetML document:

cell — The location at the intersection of a row and column, in which numeric or textual data or a formula is stored. A cell can have a number of characteristics, such as numeric or text formatting, alignment, font, color, and border.

cell reference — An individual cell's designation using a combination of its column and row headings, as in A13, H19, and BX1200. A **relative cell reference** in a formula automatically changes when the formula is copied down a column or across a row. An **absolute cell reference** is fixed. Absolute references don't change when a formula is copied from one cell to another. A **mixed cell reference** has either an absolute column and a relative row, or an absolute row and a relative column.

chart — A graphical representation of data, as in a bar, column, line, pie chart, for example.

column — Any vertical set of cells in a worksheet. Each column has an alphabetic heading. Columns are named sequentially, going from A–Z, then AA–AZ, BA–BZ, and so on.

comment — A reminder or annotation that can be attached to a cell.

connection — The means by which external data—that is, data stored outside of a workbook (in a database or on a Web server, for example)—can be imported into a worksheet.

formula — A recipe for calculating a value. Some formulas are predefined; others are user-defined.

function — A predefined formula, such as AVERAGE, MAX, MIN, and SUM. A function takes one or more arguments on which it operates, producing a result. [*Note: In the formula =SUM(B1:B4), there is one argument, B1:B4, which is the range of cells B1–B4, inclusive. end note*]

pivot table — A kind of table that is used to manage and analyze related data that is stored elsewhere.

row — Any horizontal set of cells in a worksheet. Each row has a numeric heading. Rows are numbered sequentially, starting at 1.

1 **table** — A rectangular-shaped set of related rows and columns that can be sorted, filtered, and totaled as a
 2 group. Rows in a table can be hidden by applying **autofilters** to one or more columns.

3 **workbook** — A collection of worksheets.

4 **worksheet** — A two-dimensional grid of cells that are organized into rows and columns.

5 12.2 Package Structure

6 A SpreadsheetML package shall contain a package-relationship item and a content-type item. The package-
 7 relationship item shall contain implicit relationships with targets of the following type:

- 8 • One Workbook part (12.3.23).

9 The package-relationship item is permitted to contain implicit relationships with targets of the following type:

- 10 • Digital Signature Origin (§15.2.6)
- 11 • File Property parts (§15.2.11) (Application-Defined File Properties, Core File Properties, and Custom
 12 File Properties), as appropriate.
- 13 • Thumbnail (§15.2.14).

14 The required and optional relationships between parts are defined in §12.3 and its subordinate clauses.

15 *[Example:* The following package represents the minimal conformant SpreadsheetML package as defined by
 16 this Standard:

17 First, the content types for relationship parts, the Workbook part, and at least one Sheet part must be defined
 18 (physically located at `/[Content_Types].xml` in the package):

```

19 <Types xmlns="...">
20   <Default Extension="rels"
21     ContentType="application/vnd.openxmlformats-package.relationships+xml"
22     />
23   <Override PartName="/workbook.xml"
24     ContentType="application/vnd.openxmlformats-officedocument.
25     spreadsheetml.sheet.main+xml" />
26   <Override PartName="/sheet1.xml"
27     ContentType="application/vnd.openxmlformats-
28     officedocument.spreadsheetml.worksheet+xml" />
29 </Types>
```

30 Next, the required package-level relationship to the Workbook part must be defined (physically located at
 31 `/_rels/.rels` in the package):

```

32 <Relationships xmlns="...">
```



```

1      <Relationship Id="rId1"
2      Type=http://schemas.openxmlformats.org/officeDocument/2006/
3      relationships/officeDocument"
4      Target="workbook.xml" />
5  </Relationships>

```

6 Next, the minimum content for the Workbook part must be defined (physically located at /workbook.xml in
7 the package):

```

8      <workbook xmlns="..." xmlns:r="...">
9      <sheets>
10      <sheet name="1" sheetId="1" r:id="rId1" />
11      </sheets>
12  </workbook>

```

13 Next, the required workbook-level relationship to the single Sheet part must be defined, (physically located at
14 /_rels/workbook.xml.rels in the package):

```

15  <Relationships xmlns="...">
16  <Relationship Id="rId1"
17  Type="http://schemas.openxmlformats.org/officeDocument/2006/
18  relationships/worksheet" Target="sheet1.xml" />
19  </Relationships>

```

20 Finally, the minimum content for a single Sheet part must be defined (physically located at /sheet1.xml in the
21 package):

```

22  <worksheet xmlns="..." xmlns:r="...">
23  <sheetData />
24  </worksheet>

```

25 *end example]*

26 [Example: Consider a SpreadsheetML document that contains a workbook having three worksheets. Here's an
27 example of the hierarchical folder structure that might be used for the ZIP items in the package for that
28 document. As shown, one part, Workbook (stored in the ZIP item /xl/workbook.xml), has its own relationship
29 item:

30	/_rels/.rels	<i>Package-relationship item</i>
31	/[Content_Types].xml	<i>Content-type item</i>
32	/docProps/app.xml	<i>Application-Defined File Properties part</i>
33	/docProps/core.xml	<i>Core File Properties part</i>
34	/xl/workbook.xml	<i>Workbook part</i>
35	/xl/_rels/workbook.xml.rels	<i>Part-relationship item</i>

```

1    /xl/calcChain.xml           Calculation Chain part
2    /xl/sharedStrings.xml      Shared String Table part
3    /xl/styles.xml             Styles part
4    /xl/volatileDependencies.xml Volatile Dependencies part
5    /xl/theme/theme1.xml       Theme part
6    /xl/worksheets/sheet1.xml   Worksheet parts
7    /xl/worksheets/sheet2.xml
8    /xl/worksheets/sheet3.xml

```

9 The package-relationship item contains the following:

```

10    <Relationships xmlns="...">
11      <Relationship Id="rId3"
12        Type="http://.../extended-properties" Target="docProps/app.xml"/>
13      <Relationship Id="rId2"
14        Type="http://.../core-properties" Target="docProps/core.xml"/>
15      <Relationship Id="rId1"
16        Type="http://.../officeDocument" Target="xl/workbook.xml"/>
17    </Relationships>

```

18 *end example]*

19 12.3 Part Summary

20 The subclauses subordinate to this one describe in detail each of the part types specific to SpreadsheetML.

21 [Note: For convenience, information from those subclauses is summarized in the following table:

Part	Relationship Target of	Root Element	Ref.
Calculation Chain	Workbook	calcChain	§12.3.1
Chartsheet	Workbook	chartsheet	§12.3.2
Comments	Chartsheet, Dialogsheet, Worksheet	comments	§12.3.3
Connections	Workbook	connections	§12.3.4
Custom Property	Workbook	Not applicable	§12.3.5
Custom XML Mappings	Workbook	mapInfo	§12.3.6
Dialogsheet	Workbook	dialogSheet	§12.3.7
Drawings	Chartsheet, Worksheet	wsDr	§12.3.8
External Workbook References	Workbook	externalReference	§12.3.9
Metadata	Workbook	metadata	§12.3.10
Pivot Table	Worksheet	pivotTableDefinition	§12.3.11

Part	Relationship Target of	Root Element	Ref.
Pivot Table Cache Definition	Pivot Table, Workbook	pivotCacheDefinition	§12.3.12
Pivot Table Cache Records	Pivot Table Cache Definition	pivotCacheRecords	§12.3.13
Query Table	Worksheet	queryTable	§12.3.14
Shared String Table	Workbook	sst	§12.3.15
Shared Workbook Revision Headers	Workbook	headers	§12.3.16
Shared Workbook Revision Log	Shared Workbook Revision Headers	revisions	§12.3.17
Shared Workbook User Data	Workbook	users	§12.3.18
Single Cell Table Definitions	Dialogsheet, Worksheet	singleCells	§12.3.19
Styles	Workbook	styleSheet	§12.3.20
Table Definition	Dialogsheet, Worksheet	table	§12.3.21
Volatile Dependencies	Workbook	volTypes	§12.3.22
Workbook	SpreadsheetML package	workbook	§12.3.23
Worksheet	Workbook	worksheet	§12.3.24

1 *end note]*

2 **12.3.1 Calculation Chain Part**

Content Type:	application/vnd.openxmlformats-officedocument.spreadsheetml.calcChain+xml
Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/calcChain

3

4 An instance of this part type contains an ordered set of references to all cells in all worksheets in the workbook
5 whose value is calculated from any formula. The ordering allows inter-related cell formulas to be calculated in
6 the correct order when a worksheet is loaded for use.

7 A package shall contain no more than one Calculation Chain part. If it exists, that part shall be the target of an
8 implicit relationship from the Workbook part (§12.3.23).

9 *[Example:* The following Workbook part-relationship item contains a relationship to the Calculation Chain part,
10 *which is stored in the ZIP item calcChain.xml:*

```

1    <Relationships xmlns="...">
2      <Relationship Id="rId7"
3        Type="http://.../calcChain" Target="calcChain.xml"/>
4    </Relationships>

```

5 *end example]*

6 The root element for a part of this content type shall be calcChain.

7 [Example: Cells D8, E8, and F8 each contain a value that is the result of calculations that shall be performed in
8 the order E8, D8, F8:

```

9    <calcChain xmlns="...">
10      <c r="E8" i="1"/>
11      <c r="D8"/>
12      <c r="F8" s="1"/>
13    </calcChain>

```

14 *end example]*

15 A Calculation Chain part shall be located within the package containing the source relationship (expressed
16 syntactically, the TargetMode attribute of the Relationship element shall be Internal).

17 A Calculation Chain part shall not have implicit or explicit relationships to any part defined by this Standard.

18 12.3.2 Chartsheet Part

Content Type:	application/vnd.openxmlformats-officedocument.spreadsheetml.chartsheet+xml
Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/chartsheet

19

20 An instance of this part type represents a chart that is stored in its own sheet.

21 A package is permitted to contain zero or more Chartsheet parts. Each such part shall be the target of an
22 explicit relationship from the Workbook part (§12.3.23).

23 [Example: The following Workbook part-relationship item contains three relationships to Chartsheet parts,
24 which are stored in the ZIP items chartsheets/sheetN.xml:

```

1    <Relationships xmlns="...">
2      <Relationship Id="rId2"
3        Type="http://.../chartsheet" Target="chartsheets/sheet1.xml"/>
4      <Relationship Id="rId5"
5        Type="http://.../chartsheet" Target="chartsheets/sheet2.xml"/>
6      <Relationship Id="rId6"
7        Type="http://.../chartsheet" Target="chartsheets/sheet3.xml"/>
8    </Relationships>

```

end example]

The root element for a part of this content type shall be chartsheet.

[*Example:* sheet1.xml refers to a drawing that is the target of a relationship in the Chartsheet part's relationship item:

```

13    <chartsheet xmlns:r="..." ...>
14      <sheetViews>
15        <sheetView scale="64"/>
16      </sheetViews>
17      <drawing r:id="rId1"/>
18    </chartsheet>

```

end example]

A Chartsheet part shall be located within the package containing the source relationship (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

A Chartsheet part is permitted to have implicit relationships to the following parts defined by this Standard:

- Printer Settings (§15.2.14)

A Chartsheet part is permitted to have explicit relationships to the following parts defined by this Standard:

- Drawings (§12.3.8)

A Chartsheet part shall not have implicit or explicit relationships to any other part defined by this Standard.

12.3.3 Comments Part

Content Type:	application/vnd.openxmlformats-officedocument.spreadsheetml.comments+xml
Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/comments

1 An instance of this part type contains all the comments for a given worksheet, as well as the names of the
2 authors of those comments.

3 A package shall contain exactly one Comments part for each worksheet that contains one or more comments.
4 If a Comments part exists, it shall be the target of an implicit relationship from the Workbook part (§12.3.23).

5 *[Example: The following Worksheet part-relationship item contains a relationship to the Comments part, which*
6 *is stored in the ZIP item comments2.xml:*

```
7     <Relationships xmlns="...">
8         <Relationship Id="rId2"
9             Type="http://.../comments" Target="../comments2.xml"/>
10    </Relationships>
```

11 *end example]*

12 The root element for a part of this content type shall be comments.

13 *[Example: This Comments part results from a workbook that has one or more comments from each of two*
14 *people: James Jones and Mary Smith:*

```
15    <comments xmlns:st="..." >
16        <authors>
17            <author>James Jones</author>
18            <author>Mary Smith</author>
19        </authors>
20        <commentList>
21            <comment r="C7" authorId="0">
22                <text>
23                    <st:r>
24                        <st:rPr>
25                            ...
26                        </st:rPr>
27                        <st:t>James Jones:</st:t>
28                    </st:r>
29                    <st:r>
30                        <st:rPr>
31                            ...
32                        </st:rPr>
33                        <st:t>Check that this date is correct.</st:t>
34                    </st:r>
35                </text>
36            </comment>
```

```

1      <comment r="E7" authorId="1">
2          ...
3      </comment>
4  </commentList>
5  </comments>

```

6 *end example]*

7 A Comments part shall be located within the package containing the source relationship (expressed
8 syntactically, the TargetMode attribute of the Relationship element shall be Internal).

9 A Comments part shall not implicit or explicit relationships to any part defined by this Standard.

10 12.3.4 Connections Part

Content Type:	application/vnd.openxmlformats-officedocument.spreadsheetml.connections+xml
Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/connections

11
12 An instance of this part type describes all of the connections currently established for a workbook.

13 A package shall contain no more than one Connections part, and that part shall be the target of an implicit
14 relationship from the Workbook part (§12.3.23).

15 *[Example:* The following Workbook part-relationship item contains a relationship to the Connections part,
16 which is stored in the ZIP item connections.xml:

```

17 <Relationships xmlns="...">
18     <Relationship Id="rId5"
19         Type="http://.../connections" Target="connections.xml"/>
20 </Relationships>

```

21 *end example]*

22 The root element for a part of this content type shall be connections.

23 *[Example:* A workbook has three connections, two from one worksheet, and one from another.
24 connections.xml defines these three connections:

```

1    <connections ...>
2      <connection id="1" odcFile="..." keepAlive="1" name="..." type="5"
3        refreshedVersion="2" background="1" saveData="1">
4        <dbPr connection="Provider=MSDASQL.1;Persist Security Info=True;Data
5        Source=dBASE Files;Extended Properties=&quot;DSN=dBASE Files;DBQ=E:\MY
6        DOCUMENTS;DefaultDir=E:\MY
7        DOCUMENTS;DriverId=533;MaxBufferSize=2048;PageTimeout=5;&quot;;Initial
8        Catalog=E:\MY DOCUMENTS" command="`E:\MY DOCUMENTS`\`ADDRESS`"
9        commandType="3"/>
10     </connection>
11     <connection id="2" ...>
12       <dbPr ... />
13     </connection>
14     <connection id="3" ...>
15       <dbPr ... />
16     </connection>
17   </connections>

```

end example]

A Connections part shall be located within the package containing the source relationship (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

A Connections part shall not have implicit or explicit relationships to any part defined by this Standard..

12.3.5 Custom Property Part

Content Type:	application/vnd.openxmlformats-officedocument.spreadsheetml.customProperty
Root Namespace:	Not applicable
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/customProperty

This binary part supports the storage of arbitrary user-defined data.

A package shall contain at most one Custom Property part, and that part shall be the target of an implicit relationship from the Workbook (§12.3.23) part.

[Example: The following Workbook part-relationship item contains a relationship to the Custom Property part, which is stored in the ZIP item CustomProperty.bin:

```

29    <Relationships xmlns="...">
30      <Relationship Id="rId7"
31        Type="http://.../customProperty" Target="CustomProperty.bin"/>
32    </Relationships>

```


1 *end example]*

2 A Custom Property part shall be located within the package containing the source relationship (expressed
3 syntactically, the TargetMode attribute of the Relationship element shall be Internal).

4 A Custom Property part shall not have implicit or explicit relationships to any part defined by this Standard.

5 **12.3.6 Custom XML Mappings Part**

Content Type:	application/xml
Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/xmlMaps

6

7 An instance of this part type contains a schema for an XML file, and information on the behavior that is used
8 when allowing this custom XML schema to be mapped into the spreadsheet.

9 A package shall contain no more than one Custom XML Mappings part, and that part shall be the target of an
10 implicit relationship from the Workbook part (§12.3.23). The Worksheet part into which this data is imported
11 shall also have a relationship file that targets one or more Table Definition (§12.3.21) parts and/or one or more
12 Single Cell Table Definitions (§12.3.19) parts.

13 [Example: The following Workbook part-relationship item contains a relationship to the Custom XML Mappings
14 part, which is stored in the ZIP item xmlMaps.xml:

```
15 <Relationships xmlns="...">
16   <Relationship Id="rId9"
17     Type="http://.../xmlMaps" Target="xmlMaps.xml"/>
18 </Relationships>
```

19 *end example]*

20 The root element for a part of this content type shall be mapInfo.

21 [Example: xmlMaps.xml contains the following:

```
22 <mapInfo SelectionNamespaces="">
23   <Schema ID="Schema1">
24     <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
25       <xsd:element nillable="true" name="names">
26         <xsd:complexType>
27           <xsd:sequence minOccurs="0">
28             <xsd:element minOccurs="0" maxOccurs="unbounded"
29               nillable="true" name="name" form="unqualified">
```

```

1      <xsd:complexType>
2      <xsd:sequence minOccurs="0">
3      <xsd:element minOccurs="0" nillable="true"
4      type="xsd:string" name="firstname"
5      form="unqualified"/>
6      <xsd:element minOccurs="0" nillable="true"
7      type="xsd:string" name="initial"
8      form="unqualified"/>
9      <xsd:element minOccurs="0" nillable="true"
10     type="xsd:string" name="lastName"
11     form="unqualified"/>
12     </xsd:sequence>
13   </xsd:complexType>
14 </xsd:element>
15 </xsd:sequence>
16 </xsd:complexType>
17 </xsd:element>
18 </xsd:schema>
19 </Schema>
20 <Map ID="1" Name="names_Map" RootElement="names" SchemaID="Schema1"
21   ShowImportExportValidationErrors="false" AutoFit="true"
22   Append="false"
23   PreserveSortAFLayout="true" PreserveFormat="true">
24   <DataBinding FileBinding="Test.xml" DataBindingLoadMode="1"/>
25 </Map>
26 </mapInfo>

```

end example]

A Custom XML Mappings part shall be located within the package containing the source relationship (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

A Custom XML Mappings part shall not have implicit or explicit relationships to any other part defined by this Standard.

12.3.7 Dialogsheet Part

Content Type:	application/vnd.openxmlformats-officedocument.spreadsheetml.dialogsheet+xml
Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/dialogsheet

An instance of this part type contains information about a legacy custom dialog box for a user form.

1 A package is permitted to contain one or more Dialogsheet parts, and each such part shall be the target of an
 2 explicit relationship from the Workbook part (§12.3.23).

3 [Example: The following Workbook part-relationship item contains relationships to a Dialogsheet part, which is
 4 stored in the ZIP item dialogsheets/sheet1.xml:

```
5 <Relationships xmlns="...">
6   <Relationship Id="rId2"
7     Type="http://.../dialogsheets" Target="dialogsheets/sheet1.xml"/>
8 </Relationships>
```

9 The Workbook part contains the following:

```
10 <workbook xmlns:r="..." ...>
11   ...
12   <sheets>
13     ...
14     <sheet name="Dialog1" tabId="4" type="dialog" r:id="rId2"/>
15   </sheets>
16   ...
17 </workbook>
```

18 *end example]*

19 The root element for a part of this content type shall be dialogsheet.

20 [Example: sheet1.xml contains the following:

```
21 <dialogsheets xmlns:r="..." ...>
22   <sheetPr>
23     <pageSetUpPr/>
24   </sheetPr>
25   <sheetViews>
26     ...
27   </sheetViews>
28   ...
29   <legacyDrawing r:id="rId1"/>
30 </dialogsheets>
```

31 *end example]*

32 A Dialogsheet part shall be located within the package containing the source relationship (expressed
 33 syntactically, the TargetMode attribute of the Relationship element shall be Internal).

34 A Dialogsheet part is permitted to have implicit relationships to the following parts defined by this Standard:

- 35 • Printer Settings (§15.2.14)

1 A Dialogsheet part is permitted to have explicit relationships to the following parts defined by this Standard:

- 2 • Embedded Control Persistence (§15.2.8)
- 3 • Drawings (§12.3.8)
- 4 • Embedded Object (§15.2.9)

5 A Dialogsheet part shall not have implicit or explicit relationships to any other part defined by this Standard.

6 **12.3.8 Drawings Part**

Content Type:	application/vnd.openxmlformats-officedocument.drawing+xml
Root Namespace:	http://schemas.openxmlformats.org/drawingml/2006/spreadsheetDrawing
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/drawing

7

8 An instance of this part type contains the presentation and layout information for one or more drawing
9 elements that are present on this worksheet.

10 A package is permitted to contain one or more Drawings parts, and each such part shall be the target of an
11 explicit relationship from a Worksheet part (§12.3.24), or a Chartsheet part (§12.3.2). There shall be only one
12 Drawings part per worksheet or chartsheet.

13 *[Example:* The following Chartsheet part-relationship item contains a relationship to a Drawings part, which is
14 stored in the ZIP item ../drawings/drawing1.xml:

```

15 <Relationships xmlns="...">
16   <Relationship Id="rId1"
17     Type="http:// .../drawing" Target="../drawings/drawing1.xml"/>
18 </Relationships>

```

19 *end example]*

20 The root element for a part of this content type shall be wsDr.

21 *[Example:* drawing1.xml refers to a chart that is the target of a relationship in the Drawing part's relationship
22 item:

```

1  <xdr:wsDr xmlns:xdr="..." xmlns:a="...">
2    <xdr:absoluteAnchor>
3      <xdr:pos x="1518046" y="-1443632"/>
4      <xdr:extents cx="8587382" cy="5848945"/>
5      <xdr:graphicFrame macro="">
6        <xdr:nvGraphicFramePr>
7          <xdr:cNvPr id="24" name="Chart 24" descr=""/>
8          <xdr:cNvGraphicFramePr/>
9        </xdr:nvGraphicFramePr>
10       <xdr:xfrm>
11         <a:off x="0" y="0"/>
12         <a:ext cx="0" cy="0"/>
13       </xdr:xfrm>
14       <a:graphic>
15         <a:graphicData uri="http://.../chart">
16           <a:chart relId="rId1"/>
17         </a:graphicData>
18       </a:graphic>
19     </xdr:graphicFrame>
20     <xdr:clientData/>
21   </xdr:absoluteAnchor>
22 </xdr:wsDr>

```

end example]

A Drawings part shall be located within the package containing the source relationship (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

A Drawings part is permitted to have explicit relationships to the following parts defined by this Standard:

- Chart (§14.2.1)
- Diagrams: Diagram Colors (§14.2.3), Diagram Data (§14.2.4), Diagram Layout Definition (§14.2.5) and Diagram Styles (§14.2.6)
- Hyperlinks (§15.3)
- Images (§15.2.13)

A Drawings part shall not have any implicit or explicit relationships to any other part defined by this Standard.

12.3.9 External Workbook References Part

Content Type:	application/vnd.openxmlformats-officedocument.spreadsheetml.externalLink+xml
Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/externalLink

1

2 An instance of this part specifies information about data referenced in other SpreadsheetML packages.

3 *[Example: Consider two workbooks, wb1 and wb2, stored in packages called wb1.xlsx and wb2.xlsx,*
 4 *respectively. The value of a cell on a worksheet in wb1 can be computed using the value of one or more cells in*
 5 *wb2. This is done by having wb1 contain an external reference to wb2. end example]*

6 A package is permitted to contain one or more External Workbook References parts, and those parts shall be
 7 the target of an explicit relationship in the Workbook part (§12.3.23).

8 *[Example: A Workbook part for wb1 contains the following, which indicates that somewhere in its three*
 9 *worksheets, an external reference is made to a target specified in relationship id rId4 of the part's relationship*
 10 *item:*

```
11     <workbook xmlns:r="..." />
12     ...
13     <sheets>
14         <sheet name="Sheet1" tabId="1" r:id="rId1"/>
15         <sheet name="Sheet2" tabId="2" r:id="rId2"/>
16         <sheet name="Sheet3" tabId="3" r:id="rId3"/>
17     </sheets>
18     ...
19     <externalReferences>
20         <externalReference r:id="rId4"/>
21     </externalReferences>
22     ...
23 </workbook>
```

24 That part's relationship item contains the following:

```
25     <Relationships xmlns="...">
26         <Relationship Id="rId4"
27             Type="http://.../externalLink"
28             Target="externalReferences/externalReference1.xml"/>
29     </Relationships>
```

30 *end example]*

31 The root element for a part of this content type shall be externalLink.

32 *[Example: externalReference1.xml contains:*

```

1      <externalLink xmlns:r="..." r:id="rId1">
2          <externalBook>
3              <sheetNames>
4                  <sheetName val="Sheet1"/>
5                  <sheetName val="Sheet2"/>
6                  <sheetName val="Sheet3"/>
7              </sheetNames>
8              <sheetDataSet>
9                  <sheetData sheetId="0">
10                     <row r="7">
11                         <cell r="C8">
12                             <v>0</v>
13                         </cell>
14                     </row>
15                 </sheetData>
16                 <sheetData sheetId="1"/>
17                 <sheetData sheetId="2"/>
18             </sheetDataSet>
19         </externalBook>
20     </externalLink>

```

This part's relationship item contains the following:

```

22     <Relationships ...>
23         <Relationship Id="rId1"
24             Type=".../externalReference"
25             Target="wb2.xlsx" TargetMode="External"/>
26     </Relationships>

```

where wb2.xlsx is the workbook in which one or more cells' values are used in calculating the values of a cell in workbook wb1. *end example*]

An External Workbook References part shall be located within the package containing the source relationship (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

An External Workbook References part shall specify an explicit relationship to one or more this Standard External Workbooks (§12.4).

An External Workbook References part shall not have any implicit or explicit relationships to other parts defined by this Standard.

12.3.10 Metadata Part

Content Type:	application/vnd.openxmlformats-officedocument.spreadsheetml.sheetMetadata+xml
---------------	---

Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/sheetMetadata

1

2 An instance of this part type contains information relating to a cell whose value is related to one or more other
3 cells via OnLine Analytical Processing (OLAP) technology.

4 A package shall contain no more than one Cell Metadata part, and that part shall be the target of an implicit
5 relationship from the Workbook part (§12.3.23).

6 *[Example:* The following Workbook part-relationship item contains a relationship to the Metadata part, which
7 is stored in the ZIP item metadata.xml. Cell B3 contains the formula CUBEMEMBER ("externalData",
8 "[Account].[All Account]"):

```
9     <Relationships xmlns="...">
10       <Relationship Id="rId10"
11         Type="http://.../sheetMetadata" Target="metadata.xml"/>
12     </Relationships>
```

13 *end example]*

14 The root element for a part of this content type shall be metadata.

15 *[Example:* metadata.xml contains the following:

```
16     <metadata ...>
17       <metadataTypes count="1">
18         <metadataType name="XLMDX" minSupportedVersion="120000" copy="1"
19           pasteAll="1" pasteValues="1" merge="1" splitFirst="1" rowColShift="1"
20           clearFormats="1" clearComments="1" assign="1" coerce="1"/>
21       </metadataTypes>
22       <metadataStrings count="2">
23         <s v="externalData"/>
24         <s v="[Account].[All Account]"/>
25       </metadataStrings>
26       <mdxMetadata count="1">
27         <m n="0" f="m">
28           <t c="1">
29             <n v="1"/>
30           </t>
31         </m>
32       </mdxMetadata>
```



```

1      <valueMetadata count="1">
2      <b>
3      <r t="1" v="0"/>
4      </b>
5      </valueMetadata>
6  </metadata>

```

7 The corresponding Connections part contains the following:

```

8  <connections ...>
9      <connection id="1" odcFile="..." keepAlive="1" name="externalData"
10      description="..." type="5" refreshedVersion="3" background="1">
11      <dbPr connection="Provider=MSOLAP.2;..." command="Budget" commandType="1"/>
12      <olapPr sendLocale="1" rowDrillCount="1000" serverFill="1"
13      serverNumberFormat="1" serverFont="1" serverFontColor="1"/>
14      </connection>
15  </connections>

```

16 The corresponding Volatile Dependencies part contains the following:

```

17  <volTypes ...">
18      <volType type="cubeFunctions">
19      <main first="externalData">
20      <tp t="e">
21      <v>#N/A</v>
22      <stp>1</stp>
23      <r r="B3" s="1"/>
24      </tp>
25      </main>
26      </volType>
27  </volTypes>

```

28 The corresponding Pivot Table Cache Definition part contains the following:

```

29  <pivotCacheDefinition ... saveData="0" refreshedBy="..."
30  refreshedDate="2005-11-28T16:55:44" backgroundQuery="1" createdVersion="3"
31  refreshedVersion="3" recordCount="0">
32  <cacheSource type="external" connectionID="1"/>
33  <cacheFields count="0"/>
34  <cacheHierarchies count="6">
35      ...
36  </cacheHierarchies>

```

```

1      <kpis count="0"/>
2      <tupleCache>
3          <queryCache count="3">
4              <query mdx="[product].[category]"/>
5              <query mdx=""/>
6              <query mdx="[Account].[All Account]">
7                  <tpls c="1">
8                      <tpl hier="0" item="4294967295"/>
9                  </tpls>
10             </query>
11         </queryCache>
12     </tupleCache>
13 </pivotCacheDefinition>

```

end example]

A Metadata part shall be located within the package containing the source relationship (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

A Metadata part shall not have implicit or explicit relationships to any part defined by this Standard.

12.3.11 Pivot Table Part

Content Type:	application/vnd.openxmlformats-officedocument.spreadsheetml.pivotTable+xml
Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/pivotTable

An instance of this part type contains a pivot table definition.

A package shall contain exactly one Pivot Table part per pivot table, and each such part shall be the target of an implicit relationship in the relationship part for the Worksheet part (§12.3.24) that corresponds to the worksheet containing the pivot table.

[Example: The following Worksheet part-relationship item contains a relationship to two Pivot Table parts, which are stored in the ZIP items ../pivotTables/pivotTableN.xml:

```

26  <Relationships xmlns="...">
27      <Relationship Id="rId1"
28          Type="http://.../pivotTable" Target="../pivotTables/pivotTable1.xml"/>
29      <Relationship Id="rId2"
30          Type="http://.../pivotTable" Target="../pivotTables/pivotTable2.xml"/>
31  </Relationships>

```

1 *end example]*

2 The root element for a part of this content type shall be pivotTableDefinition.

3 [Example: pivotTable1.xml contains the following:

```

4 <pivotTableDefinition ... cache="4" applyNumberFormats="0" applyBorderFormats="0"
5   applyFontFormats="0" applyPatternFormats="0" applyAlignmentFormats="0"
6   applyWidthHeightFormats="1" dataCaption="Data" updatedVersion="3"
7   minRefreshableVersion="3" useAutoFormatting="1" itemPrintTitles="1"
8   createdVersion="3" indent="0" outline="1" outlineData="1">
9   <location ref="H4:H5" firstHeaderRow="1" firstDataRow="1" firstDataCol="0"/>
10  <pivotFields count="1">
11    <pivotField dataField="1" numFmtId="0" outline="1"
12      subtotalTop="1" showAll="0" measureFilter="0" sortType="manual"/>
13  </pivotFields>
14  <rowItems count="1">
15    <i t="data"/>
16  </rowItems>
17  <colItems count="1">
18    <i t="data"/>
19  </colItems>
20  <dataFields count="1">
21    <dataField name="Sum of 1000" fld="0" subtotal="average"
22      baseField="0" baseItem="0" numFmtId="0"/>
23  </dataFields>
24  <tableStyle name="TableStyle2" showRowHeaders="1" showColHeaders="1"
25    showRowStripes="1" showColStripes="1"/>
26 </pivotTableDefinition>

```

27 *end example]*

28 A Pivot Table part shall be located within the package containing the source relationship (expressed
 29 syntactically, the TargetMode attribute of the Relationship element shall be Internal).

30 A Pivot Table part is permitted to have implicit relationships to the following parts defined by this Standard:

- 31 • Pivot Table Cache Definition (§12.3.12).

32 A Pivot Table part shall not have any implicit or explicit relationships to other parts defined by this Standard.

33 12.3.12 Pivot Table Cache Definition Part

Content Type:	application/vnd.openxmlformats-officedocument.spreadsheetml.pivotCacheDefinition+xml
Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/pivotCacheDefinition
----------------------	---

1

2 An instance of this part type contains a cache definition for a pivot table.

3 A package shall contain exactly one Pivot Table Cache Definition part per pivot table, and each such part shall
4 be the target of an implicit relationship from a Pivot Table (§12.3.11) part as well as an explicit relationship
5 from a Workbook (§12.3.23) part.

6 *[Example:* The following Pivot Table part-relationship item contains a relationship to the Pivot Table Cache
7 Definition part, which is stored in the ZIP item ../pivotCache/pivotCacheDefinition2.xml:

```
8     <Relationships xmlns="...">
9         <Relationship Id="rId1"
10             Type=http://.../pivotCacheDefinition
11             Target="../pivotCache/pivotCacheDefinition2.xml"/>
12     </Relationships>
```

13 *end example]*

14 The root element for a part of this content type shall be pivotCacheDefinition.

15 *[Example:* pivotCacheDefinition2.xml contains the following:

```
16     <pivotCacheDefinition ... r:id="rId1" refreshedBy="John Jones"
17         refreshedDate="2005-11-18T16:47:49" createdVersion="3"
18         refreshedVersion="3" recordCount="11">
19         <cacheSource type="worksheet">
20             <worksheet range="C4:C15" sheet="Sheet1"/>
21         </cacheSource>
22         <cacheFields count="1">
23             <cacheField name="1000">
24                 <sharedItems containsSemiMixedTypes="0" containsString="0"
25                     containsNumber="1" containsInteger="1" minValue="234
26                     maxValue="2543"/>
27             </cacheField>
28         </cacheFields>
29     </pivotCacheDefinition>
```

30 *end example]*

31 A Pivot Table Cache Definition part shall be located within the package containing the source relationship
32 (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

A Pivot Table Cache Definition part is permitted to have an explicit relationship to the following part defined by this Standard:

- Pivot Table Cache Records (§12.3.13).

A Pivot Table Cache Definition part shall not have any implicit or explicit relationships to other parts defined by this Standard.

12.3.13 Pivot Table Cache Records Part

Content Type:	application/vnd.openxmlformats-officedocument.spreadsheetml.pivotCacheRecords+xml
Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/spreadsheetml/pivotCacheRecords

An instance of this part type contains the cache records for a pivot table.

A package shall contain exactly one Pivot Table Cache Records part per pivot table, and each such part shall be the target of an explicit relationship in the Pivot Table Cache Definition (§12.3.12) part for the corresponding pivot table.

[Example: The following Pivot Table Cache Definition part-relationship item contains a relationship to the Pivot Table Cache Records part, which is stored in the ZIP item pivotCacheRecords2.xml:

```
<Relationships xmlns="...">
  <Relationship Id="rId1"
    Type="http://.../pivotCacheRecords" Target="pivotCacheRecords2.xml"/>
</Relationships>
```

end example]

The root element for a part of this content type shall be pivotCacheRecords.

[Example: pivotCacheRecords2.xml contains the following:

```
<pivotCacheRecords ... count="11">
  <r>
    <n v="1234"/>
  </r>
  ...
  <r>
    <n v="876"/>
  </r>
</pivotCacheRecords>
```

1 *end example]*

2 A Pivot Table Cache Records part shall be located within the package containing the source relationship
3 (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

4 A Pivot Table Cache Records part shall not have implicit or explicit relationships to any other part defined by
5 this Standard.

6 **12.3.14 Query Table Part**

Content Type:	application/vnd.openxmlformats-officedocument.spreadsheetml.queryTable+xml
Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/queryTable

7

8 An instance of this part type contains information that describes how the source table is connected to an
9 external data source, and defines the properties that is used when this table is refreshed from that source.

10 A package is permitted to contain one Query Table part per table, and each of those parts shall be the target of
11 an implicit relationship from the corresponding Table Definitions (§12.3.21) part.

12 *[Example:* The following Table part-relationship item contains a relationship to the Query Table part
13 corresponding to the connections details for that table. These parts are stored in the ZIP items
14 *../queryTables/queryTable1.xml:*

```

15     <Relationships xmlns="...">
16         <Relationship Id="rId1"
17             Type="http://.../queryTable"
18             Target="../queryTables/queryTable1.xml"/>
19     </Relationships>

```

20 *end example]*

21 The root element for a part of this content type shall be queryTable.

22 *[Example:* queryTable2.xml deals with a connection to a database file having the seven fields shown:

```

23     <queryTable ... name="+Connect to New Data Source_1"
24         growShrinkType="insertDelete" connectionId="2" autoFormatId="16"
25         applyNumberFormats="0" applyBorderFormats="0" applyFontFormats="1"
26         applyPatternFormats="1" applyAlignmentFormats="0"
27         applyWidthHeightFormats="0">

```

```

1      <queryTableRefresh nextId="8">
2          <queryTableFields count="7">
3              <queryTableField id="1" name="ACCOUNT"/>
4              <queryTableField id="2" name="CHECKNUM"/>
5              <queryTableField id="3" name="DATE"/>
6              <queryTableField id="4" name="AMOUNT"/>
7              <queryTableField id="5" name="PAYEE"/>
8              <queryTableField id="6" name="CHARGECODE"/>
9              <queryTableField id="7" name="DESCRIPT"/>
10         </queryTableFields>
11     </queryTableRefresh>
12 </queryTable>

```

end example]

A Query Table part shall be located within the package containing the source relationship (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

A Query Table part shall not have implicit or explicit relationships to any other part defined by this Standard.

12.3.15 Shared String Table Part

Content Type:	application/vnd.openxmlformats-officedocument.spreadsheetml.sharedStrings+xml
Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/sharedStrings

An instance of this part type contains one occurrence of each unique string that occurs on all worksheets in a workbook.

A package shall contain exactly one Shared String Table part, and that part shall be the target of an implicit relationship from the Workbook part (§12.3.23).

[Example: The following Workbook part-relationship item contains a relationship to the Shared String Table part, which is stored in the ZIP item sharedStrings.xml:

```

25     <Relationships xmlns="...">
26         <Relationship Id="rId6"
27             Type="http://.../sharedStrings" Target="sharedStrings.xml"/>
28     </Relationships>

```

end example]

The root element for a part of this content type shall be sst.

[Example: Here are three of the six strings used in the worksheets:

```
<sst xmlns:st="..." ... totalCount="6" uniqueCount="6">
  <sstItem>
    <t>Expenses Log</t>
  </sstItem>
  <sstItem>
    <t>Period Start</t>
  </sstItem>
  <sstItem>
    <t>Period End</t>
  </sstItem>
  ...
</sst>
```

end example]

A Shared String Table part shall be located within the package containing the source relationship (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

A Shared String Table part shall not have implicit or explicit relationships to any other part defined by this Standard.

12.3.16 Shared Workbook Revision Headers Part

Content Type:	application/vnd.openxmlformats-officedocument.spreadsheetml.revisionHeaders+xml
Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/revisionHeaders

An instance of this part type contains information about each of the editing sessions performed on the parent workbook at the worksheet level (worksheets added and rearranged in each session).

A package shall contain at most one Shared Workbook Revision Headers part. If it exists, that part shall be the target of an implicit relationship from the Workbook (§12.3.23) part.

[Example: The following Workbook part-relationship item contains a relationship to the Shared Workbook Revision Headers part, which is stored in the ZIP item handout revisions/revisionHeaders.xml:

```
<Relationships xmlns="...">
  <Relationship Id="rId9"
    Type="http://.../revisionHeaders"
    Target="revisions/revisionHeaders.xml"/>
</Relationships>
```


1 *end example]*

2 The root element for a part of this content type shall be headers.

3 [Example: revisionHeaders.xml contains the following:

```

4   <headers xmlns:r="..." guid="{233BEE23-EB5C-4542-905D-0230EFFED88B}"
5     diskRevisions="1" revisionId="4" version="3">
6     <header guid="..." dateTime="..." maxSheetId="4" userName="..." r:id="rId1">
7       <sheetIdMap count="3">
8         ...
9       </sheetIdMap>
10    </header>
11    ...
12    <header guid="..." dateTime="..." maxSheetId="4" userName="..." r:id="rId3">
13      <sheetIdMap count="3">
14        ...
15      </sheetIdMap>
16    </header>
17  </headers>

```

18 *end example]*

19 A Shared Workbook Revision Headers part shall be located within the package containing the source
20 relationship (expressed syntactically, the TargetMode attribute of the Relationship element shall be
21 Internal).

22 A Shared Workbook Revision Headers part is permitted to have explicit relationships to the following parts
23 defined by this Standard:

- 24 • Shared Workbook Revision Log (§12.3.17)

25 A Shared Workbook Revision Headers part shall not have any implicit or explicit relationships to other parts
26 defined by this Standard.

27 12.3.17 Shared Workbook Revision Log Part

Content Type:	application/vnd.openxmlformats-officedocument.spreadsheetml.revisionLog+xml
Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/revisionLog

28

29 An instance of this part type contains information about edits performed on individual cells in the parent
30 workbook's worksheets in each editing session.

1 A package shall contain one Shared Workbook Revision Log part for each session's set of changes, and those
 2 parts shall be the target of an explicit relationship from the Shared Workbook Revision Headers (§12.3.16)
 3 part.

4 [Example: The following Shared Workbook Revision Headers part-relationship item contains a number of
 5 relationships to Shared Workbook Revision Log parts, which are stored in the ZIP item revisionLogN.xml:

```

6     <Relationships xmlns="...">
7       <Relationship Id="rId1"
8         Type="http://.../revisionLog" Target="revisionLog1.xml"/>
9       ...
10      <Relationship Id="rId6"
11        Type="http://.../revisionLog" Target="revisionLog6.xml"/>
12    </Relationships>

```

13 *end example]*

14 The root element for a part of this content type shall be revisions.

15 [Example: revisionLog2.xml contains the following:

```

16    <revisions xmlns:xs="..." ...>
17      <rfmt sheetId="1" sqref="B4:B15">
18        <dxfs>
19          <xs:fill>
20            <xs:pattern patternType="solid">
21              <xs:fgColor type="icv" val="64"/>
22              <xs:bgColor type="rgb" val="4278252287"/>
23            </xs:pattern>
24          </xs:fill>
25        </dxfs>
26      </rfmt>
27      <rcv guid="{CBCE5672-5A4D-48C9-A120-F72804F8CF64}" action="delete"/>
28      <rcv guid="{CBCE5672-5A4D-48C9-A120-F72804F8CF64}" action="add"/>
29    </revisions>

```

30 *end example]*

31 A Shared Workbook Revision Log part shall be located within the package containing the source relationship
 32 (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

33 A Shared Workbook Revision Log part shall not have implicit or explicit relationships to any other part defined
 34 by this Standard.

12.3.18 Shared Workbook User Data Part

Content Type:	application/vnd.openxmlformats-officedocument.spreadsheetml.userNames+xml
Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/usernames

2

3 An instance of this part type contains a list of all the users that are sharing the parent workbook.

4 A package shall contain at most one Shared Workbook User Data part, and that part shall be the target of an
5 implicit relationship in the Workbook (§12.3.23) part.

6 *[Example: The following Workbook part-relationship item contains a relationship to the Shared Workbook User*
7 *Data part, which is stored in the ZIP item revisions/userNames.xml:*

```
8     <Relationships xmlns="...">
9         Relationship Id="rId8"
10             Type="http://.../usernames" Target="revisions/userNames.xml"/>
11     </Relationships>
```

12 *end example]*

13 The root element for a part of this content type shall be users.

14 *[Example: usernames.xml shows that there are two users sharing this workbook:*

```
15     <users ... count="2">
16         <usrinfo guid="{B5A024F7-40BE-4A48-9B6D-B1655241C84D}"
17             name="Mary Jones" id="-264292310" dateTime="2005-11-18T18:53:16"/>
18         <usrinfo .../>
19     </users>
```

20 *end example]*

21 A Shared Workbook User Data part shall be located within the package containing the source relationship
22 (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

23 A Shared Workbook User Data part shall not have implicit or explicit relationships to any other part defined by
24 this Standard.

25 12.3.19 Single Cell Table Definitions Part

Content Type:	application/vnd.openxmlformats-officedocument.spreadsheetml.tableSingleCells+xml
Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/tableSingleCells
----------------------	---

1

2 An instance of this part type contains information on how to map non-repeating elements from a custom XML
3 file into cells in a worksheet. [*Note*: Repeating custom XML elements are mapped using a Table (§12.3.21). *end*
4 *note*]

5 A package shall contain at most one Single Cell Table Definitions part per worksheet, and that part shall be the
6 target of an implicit relationship from a Worksheet (§12.3.24) part. A Single Cell Table Definitions part can
7 describe one or more single cell table definitions for any given worksheet.

8 [*Example*: The following Worksheet part-relationship item contains a relationship to the Single Cell Table
9 Definitions part, which is stored in the ZIP item ../tables/tableSingleCells1.xml:

```
10 <Relationships xmlns="...">
11   <Relationship Id="rId1"
12     Type="http://.../tableSingleCells"
13     Target="../tables/tableSingleCells1.xml"/>
14 </Relationships>
```

15 *end example*

16 The root element for a part of this content type shall be singleCells.

17 [*Example*: A worksheet contains two single cell table definitions; e.g., ../tables/tableSingleCells1.xml contains
18 the following, where the elements id and count are nested inside element names:

```
19 <singleCells ...>
20   <singleCell id="1" name="Table1" displayName="Table1" ref="B4">
21     <cellPr id="1" uniqueName="id">
22       <xmlPr mapId="1" xpath="/names/id" xmlDataType="string"/>
23     </cellPr>
24   </singleCell>
25   <singleCell id="2" name="Table2" displayName="Table2" ref="B7">
26     <cellPr id="1" uniqueName="count">
27       <xmlPr mapId="1" xpath="/names/count" xmlDataType="integer"/>
28     </cellPr>
29   </singleCell>
30 </singleCells>
```

31 *end example*

32 A Single Cell Table Definitions part shall be located within the package containing the source relationship
33 (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

1 A Single Cell Table Definitions part shall not have implicit or explicit relationships to any other part defined by
2 this Standard.

3 12.3.20 Styles Part

Content Type:	application/vnd.openxmlformats-officedocument.spreadsheetml.styles+xml
Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/mains
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/styles

4

5 An instance of this part type contains all the characteristics for all the cells in the workbook. Such information
6 includes numeric and text formatting, alignment, font, color, and border.

7 A package shall contain no more than one Styles part, and that part shall be the target of an implicit
8 relationship from the Workbook (§12.3.23) part.

9 *[Example:* The following Workbook part-relationship item contains a relationship to the Styles part, which is
10 stored in the ZIP item styles.xml:

```
11 <Relationships xmlns="...">
12   <Relationship Id="rId5"
13     Type="http://.../styles" Target="styles.xml"/>
14 </Relationships>
```

15 *end example]*

16 The root element for a part of this content type shall be styleSheet.

17 *[Example:*

```
18 <styleSheet xmlns="...">
19   <numFmts count="5">
20     <numFmt numFmtId="164" formatCode=""$&quot;;#,##0.00"/>
21     <numFmt numFmtId="165"
22       formatCode=""Yes&quot;;&quot;Yes&quot;;&quot;No&quot;"/>
23     <numFmt numFmtId="166"
24       formatCode=""True&quot;;&quot;True&quot;;&quot;False&quot;"/>
25     <numFmt numFmtId="167"
26       formatCode=""On&quot;;&quot;On&quot;;&quot;Off&quot;"/>
27     <numFmt numFmtId="168"
28       formatCode="[$€-2]\ #,##0.00_);[Red]\([$€-2]\ #,##0.00\)" />
29   </numFmts>
```

```

1      <fonts count="5">
2          ...
3      </fonts>
4      <fills count="4">
5          ...
6      </fills>
7      <borders count="1">
8          ...
9      </borders>
10     ...
11     <colors>
12         ...
13     </colors>
14 </styleSheet>

```

15 *end example]*

16 A Styles part shall be located within the package containing the source relationship (expressed syntactically,
17 the TargetMode attribute of the Relationship element shall be Internal).

18 A Styles part shall not have implicit or explicit relationships to any other part defined by this Standard.

19 12.3.21 Table Definition Part

Content Type:	application/vnd.openxmlformats-officedocument.spreadsheetml.table+xml
Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/table

20

21 An instance of this part type contains a description of a single table and its autofilter information. (The data for
22 the table is stored in the corresponding Worksheet part.)

23 A package shall contain one Table Definition part per table, and each such part shall be the target of an implicit
24 relationship from the Worksheet (§12.3.24) part that corresponds to the worksheet containing that table.

25 [Example: The following Worksheet part-relationship item contains relationships to two Table Definition parts,
26 which are stored in the ZIP items ../tables/tableN.xml:

```

27     <Relationships xmlns="...">
28         <Relationship Id="rId2"
29             Type="http://.../table" Target="../tables/table1.xml"/>
30         <Relationship Id="rId3"
31             Type="http://.../table" Target="../tables/table2.xml"/>
32     </Relationships>

```

1 *end example]*

2 The root element for a part of this content type shall be table.

3 [Example: table2.xml describes a table that spans a 2-column range of cells, F2:G19:

```

4     <table xmlns:af="..." ... id="2" name="Table2" displayName="Table2" ref="F2:G19"
5         totalsRowShown="0" headerRowDxfId="7">
6         <autoFilter ref="F2:G19"/>
7         <tableColumns count="2">
8             <tableColumn id="1" name="Salesman" dataDxfId="9" totalsRowDxfId="6"/>
9             <tableColumn id="2" name="Units" dataDxfId="8" totalsRowDxfId="5"/>
10        </tableColumns>
11        <tableStyle name="TableStyle2" showFirstColumn="0" showLastColumn="0"
12            showRowStripes="1" showColumnStripes="1"/>
13    </table>

```

14 When the filter "Salesman equal to Smith" is applied, the autoFilter element in table2.xml is extended, as
 15 follows:

```

16    <autoFilter ref="F2:G19">
17        <af:filterColumn colId="0">
18            <af:filters>
19                <af:filter val="Smith"/>
20            </af:filters>
21        </af:filterColumn>
22    </autoFilter>

```

23 *end example]*

24 A Table Definition part shall be located within the package containing the source relationship (expressed
 25 syntactically, the TargetMode attribute of the Relationship element shall be Internal).

26 A Table Definition part is permitted to explicit relationships to the following parts defined by this Standard:

- 27 • Query Table (§12.3.14)

28 A Table Definition part shall not have any implicit or explicit relationships to any other part defined by this
 29 Standard.

30 12.3.22 Volatile Dependencies Part

Content Type:	application/vnd.openxmlformats-officedocument.spreadsheetml.volatileDependencies+xml
Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/volatileDependencies
----------------------	---

1

2 An instance of this part type contains information involving real-time data formulas in a workbook.

3 A package shall contain exactly one Volatile Dependencies part, and that part shall be the target of an implicit
4 relationship from the Workbook (§12.3.23) part.

5 *[Example:* The following Workbook part-relationship item contains a relationship to the Volatile Dependencies
6 part, which is stored in the ZIP item volatileDependencies.xml:

```
7     <Relationships xmlns="...">
8         <Relationship Id="rId8"
9             Type="http://.../volatileDependencies"
10            Target="volatileDependencies.xml"/>
11     </Relationships>
```

12 *end example]*

13 The root element for a part of this content type shall be volTypes.

14 *[Example:*

```
15     <volTypes xmlns="..." />
```

16 *end example]*

17 A Volatile Dependencies part shall be located within the package containing the source relationship (expressed
18 syntactically, the TargetMode attribute of the Relationship element shall be Internal).

19 A Volatile Dependencies part shall not have implicit or explicit relationships to any other part defined by this
20 Standard.

21 12.3.23 Workbook Part

Content Type(s):	application/vnd.openxmlformats-officedocument.spreadsheetml.main+xml application/vnd.openxmlformats-officedocument.spreadsheetml.template.main+xml
Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/officeDocument

22

23 An instance of this part type contains workbook data and references to all of its worksheets.

1 A package shall contain exactly one Workbook part, and that part shall be the target of a relationship in the
2 package-relationship item.

3 *[Example:* The following SpreadsheetML package-relationship item contains a relationship to the Workbook
4 part, which is stored in the ZIP item xl/workbook.xml:

```
5     <Relationships xmlns="...">
6         <Relationship Id="rId1"
7             Type="http://.../officeDocument" Target="xl/workbook.xml"/>
8     </Relationships>
```

9 *end example]*

10 The root element for a part of this content type shall be workbook.

11 *[Example:* This workbook has three worksheets, named January, February, and March:

```
12     <workbook xmlns="..." xmlns:r="...">
13         <fileVersion lastEdited="4" lowestEdited="4" rupBuild="3417"/>
14         <bookViews>
15             <workbookView xWindow="240" yWindow="15" windowWidth="8505"
16                 windowHeight="6240"/>
17         </bookViews>
18         <sheets>
19             <sheet name="January" tabId="1" r:id="rId1"/>
20             <sheet name="February" tabId="2" r:id="rId2"/>
21             <sheet name="March" tabId="3" r:id="rId3"/>
22         </sheets>
23         <workbookPr showObjects="all"/>
24         <webPublishing codePage="1252"/>
25         <calcPr calcId="122211" fullCalcOnLoad="1"/>
26     </workbook>
```

27 *end example]*

28 A Workbook part shall be located within the package containing the source relationship (expressed
29 syntactically, the TargetMode attribute of the Relationship element shall be Internal).

30 A Workbook part is permitted to have implicit relationships to the following parts defined by this Standard:

- 31 • Additional Characteristics (§15.2.1)
- 32 • Bibliography (§15.2.3)
- 33 • Calculation Chain (§12.3.1)
- 34 • Cell Metadata (§12.3.10)
- 35 • Connections (§12.3.4)
- 36 • Custom XML Mappings (§12.3.6)

- Custom XML Data Storage (§15.2.4)
- Shared String Table (§12.3.15)
- Shared Workbook Revision Headers (§12.3.16)
- Shared Workbook User Data (§12.3.18)
- Styles (§12.3.20)
- Theme (§14.2.7)
- Thumbnail (§15.2.14)
- Volatile Dependencies (§12.3.22)

A Workbook part is permitted to have explicit relationships to the following parts defined by this Standard:

- Chartsheet (§12.3.2)
- Dialogsheet (§12.3.7)
- External Workbook References (§12.3.8)
- Pivot Table Cache Definition (§12.3.12)
- Worksheet (§12.3.24)

A Workbook part shall not have implicit or explicit relationships to any other part defined by this Standard.

12.3.24 Worksheet Part

Content Type:	application/vnd.openxmlformats-officedocument.spreadsheetml.worksheet+xml
Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/worksheet

An instance of this part type contains all the data, formulas, and characteristics associated with a given worksheet.

A package shall contain exactly one Worksheet part per worksheet, and those parts shall be the target of an explicit relationship from the Workbook (§12.3.23) part. Specifically, the id attribute on the sheet element shall reference the desired worksheet part.

[Example: The following Workbook part-relationship item contains three relationships to Worksheet parts, which are stored in the ZIP items worksheets/sheetN.xml:

```

1    <Relationships xmlns="...">
2      <Relationship Id="rId1"
3        Type="http://.../worksheet" Target="worksheets/sheet1.xml"/>
4      <Relationship Id="rId2"
5        Type="http://.../worksheet" Target="worksheets/sheet2.xml"/>
6      <Relationship Id="rId3"
7        Type="http://.../worksheet" Target="worksheets/sheet3.xml"/>
8    </Relationships>

```

9 *end example]*

10 The root element for a part of this content type shall be worksheet.

11 *[Example:* This worksheet, has cells in the range B1 to F8. Row 8 contains three cells whose values are
 12 calculated using the following formulas: D8=SUM(D5:D7), E8=SUM(E5:E7), and F8= D8-E8:

```

13    <worksheet xmlns="..." ...>
14      <sheetPr/>
15      <dimension range="B1:F8"/>
16      ...
17      <sheetData>
18        <row r="1" spans="2:6" ht="360">
19          <c r="B1" s="1" t="s">
20            <v>0</v>
21          </c>
22        </row>
23        ...
24        <row r="8" spans="2:6" ht="360">
25          <c r="D8" s="5">
26            <f>SUM(D5:D7)</f>
27            <v>2280.5299999999997</v>
28          </c>
29          <c r="E8" s="5">
30            <f>SUM(E5:E7)</f>
31            <v>1251.31</v>
32          </c>
33          <c r="F8" s="6">
34            <f>D8-E8</f>
35            <v>1029.2199999999998</v>
36          </c>
37        </row>
38      </sheetData>
39      ...
40    </worksheet>

```

1 *end example]*

2 A Worksheet part shall be located within the package containing the source relationship (expressed
3 syntactically, the TargetMode attribute of the Relationship element shall be Internal).

4 A Worksheet part is permitted to contain implicit relationships to the following parts defined by this Standard:

- 5 • Comments (§12.3.3)
- 6 • Pivot Table Definitions (§12.3.11)
- 7 • Printer Settings (§15.2.14)
- 8 • Single Cell Table Definitions (§12.3.19)
- 9 • Table Definition (§12.3.21)

10 A Worksheet part is permitted to contain explicit relationships to the following parts defined by this Standard:

- 11 • Drawings (§12.3.8)
- 12 • Embedded Control Persistence (§15.2.8)
- 13 • Embedded Object (§15.2.9)
- 14 • Embedded Package (§15.2.10)
- 15 • Hyperlinks (§15.3)
- 16 • Images (§15.2.13)
- 17 • VML Drawing (§15.2.17)

18 A Worksheet part shall not have implicit or explicit relationships to any other part defined by this Standard.

19 12.4 External Workbooks

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/externalLinkPath
----------------------	---

20

21 An *external workbook* is a SpreadsheetML package whose contents are being referenced by the current
22 SpreadsheetML package. When a package refers to external workbooks, it shall store the location of those
23 workbooks using this relationship.

24 A package is permitted to contain one or more External Workbook relationships, and those relationships shall
25 be an explicit relationship from the External Workbook References (§12.3.9) part.

26 [Example: An External Workbook References part, which references the package c:\sourceData.xlsx would
27 have an External Workbook relationship, which points at that file:

```

28 <Relationships xmlns="...">
29   <Relationship Id="rId1"
30     Type="http://.../externalLinkPath"
31     Target="c:\sourceData.xlsx" TargetMode="External"/>
32 </Relationships>
```

- 1 *end example]*
- 2 A external workbook shall be located external to the package containing the source relationship (expressed
- 3 syntactically, the TargetMode attribute of the Relationship element shall be External1).

13. PresentationML

This clause contains specifications for relationship items and parts that are specific to PresentationML. Parts than can occur in a PresentationML document, but are not PresentationML-specific, are specified in §15.2. Unless stated explicitly, all references to relationship items, content-type items, and parts in this clause refer to PresentationML ZIP items.

13.1 Glossary of PresentationML-Specific Terms

The following terms are used in the context of a PresentationML document:

comment — A note that an author or reviewer attaches to a piece of text in a document. Although a consumer may chose to display comments, they are not considered part of the body of the document. A comment includes the text of the note, the comment author's name and initials, and date of creation, among other things.

handout — A printed set of slides that can be handed out to an audience for future reference.

note — A slide annotation, reminder, or piece of text intended for the presenter or the audience.

presentation — A collection of slides intended to be viewed by an audience.

slide — A frame containing one or more pieces of text and/or images.

slide layout — The organization of elements on a slide.

13.2 Package Structure

A PresentationML package shall contain a package-relationship item and a content-type item. The package-relationship item shall contain implicit relationships with targets of the following type:

- One Presentation part (§13.3.6).

The package-relationship item is permitted to contain implicit relationships with targets of the following type:

- Digital Signature Origin (§15.2.6)
- File Property parts (§15.2.11) (Application-Defined File Properties, Core File Properties, and Custom File Properties), as appropriate.
- Thumbnail (§15.2.14).

The required and optional relationships between parts are defined in §13.3 and its subordinate clauses.

[Example: The following package represents the minimal conformant PresentationML package as defined by this Standard:

1 First, the content type for relationship parts and the Main Presentation part (the only required part) must be
2 defined (physically located at `/[Content_Types].xml` in the package):

```
3 <Types xmlns="...">
4   <Default Extension="rels"
5     ContentType="application/vnd.openxmlformats-
6       package.relationships+xml"/>
7   <Override PartName="/presentation.xml"
8     ContentType="application/vnd.openxmlformats-
9       officedocument.presentationml.presentation.main+xml"/>
10 </Types>
```

11 Next, the single required relationship (the package-level relationship to the Main Presentation part) must be
12 defined (physically located at `/_rels/.rels` in the package):

```
13 <Relationships xmlns="...">
14   <Relationship Id="rId1"
15     Type="http://schemas.openxmlformats.org/officeDocument/2006/
16       relationships/officeDocument"
17     Target="presentation.xml"/>
18 </Relationships>
```

19 Finally, the minimum content for the Main Presentation part must be defined (physically located at
20 `/presentation.xml` in the package):

```
21 <p:presentation xmlns:p="...">
22   <p:notesSz cx="913607" cy="913607"/>
23 </p:presentation>
```

24 *end example]*

25 [Example: Consider a simple PresentationML document containing two slides, which both use an image as a
26 template. Here's an example of the hierarchical folder structure that might be used for the ZIP items in the
27 package for that document. As shown, a number of parts have their own relationship items:

28	<code>/_rels/.rels</code>	<i>Package-relationship item</i>
29	<code>/[Content_Types].xml</code>	<i>Content-type item</i>
30	<code>/docProps/app.xml</code>	<i>Application-Defined File</i>
31	<i>Properties part</i>	
32	<code>/docProps/core.xml</code>	<i>Core File Properties part</i>
33	<code>/docProps/custom.xml</code>	<i>Custom File Properties part</i>
34	<code>/docProps/thumbnail.wmf</code>	<i>Package thumbnail image</i>
35	<code>/ppt/presentation.xml</code>	<i>Presentation part</i>
36	<code>/ppt/_rels/presentation.xml.rels</code>	<i>Part-relationship item</i>

1	/ppt/presProps.xml	<i>Presentation Properties part</i>
2	/ppt/tableStyles.xml	<i>Table Styles part</i>
3	/ppt/viewProps.xml	<i>View Properties part</i>
4	/ppt/handoutMasters/handoutMaster1.xml	<i>Handout Master part</i>
5	/ppt/handoutMasters/_rels/handoutMaster1.xml.rels	
6		<i>Part-relationship item</i>
7	/ppt/media/image1.jpeg	<i>Slide template image</i>
8	/ppt/notesMasters/notesMaster1.xml	<i>Notes Master part</i>
9	/ppt/notesMasters/_rels/notesMaster1.xml.rels	
10		<i>Part-relationship item</i>
11	/ppt/notesSlides/notesSlide1.xml	<i>Notes Slide parts</i>
12	/ppt/notesSlides/notesSlide2.xml	
13	/ppt/notesSlides/_rels/notesSlide1.xml.rels	
14		<i>Part-relationship items</i>
15	/ppt/notesSlides/_rels/notesSlide2.xml.rels	
16	/ppt/slideLayouts/slideLayout1.xml	<i>Slide Layout parts 1-6</i>
17	/ppt/slideLayouts/slideLayout2.xml	
18	/ppt/slideLayouts/slideLayout3.xml	
19	/ppt/slideLayouts/slideLayout4.xml	
20	/ppt/slideLayouts/slideLayout5.xml	
21	/ppt/slideLayouts/slideLayout6.xml	
22	/ppt/slideLayouts/_rels/slideLayout1.xml.rels	
23		<i>Part-relationship items</i>
24	/ppt/slideLayouts/_rels/slideLayout2.xml.rels	
25	/ppt/slideLayouts/_rels/slideLayout3.xml.rels	
26	/ppt/slideLayouts/_rels/slideLayout4.xml.rels	
27	/ppt/slideLayouts/_rels/slideLayout5.xml.rels	
28	/ppt/slideLayouts/_rels/slideLayout6.xml.rels	
29	/ppt/slideMasters/slideMaster1.xml	<i>Slide Master part</i>
30	/ppt/slideMasters/_rels/slideMaster1.xml.rels	
31		<i>Part-relationship item</i>
32	/ppt/slides/slide1.xml	<i>Slide parts</i>
33	/ppt/slides/slide2.xml	
34	/ppt/slides/_rels/slide1.xml.rels	<i>Part-relationship items</i>
35	/ppt/slides/_rels/slide2.xml.rels	
36	/ppt/theme/theme1.xml	<i>Theme parts</i>
37	/ppt/theme/theme2.xml	
38	/ppt/theme/theme3.xml	

Theme Override parts

```

1  /ppt/theme/themeOverride1.xml
2  /ppt/theme/themeOverride2.xml
3  /ppt/theme/themeOverride3.xml
4  /ppt/theme/themeOverride4.xml
5  /ppt/theme/themeOverride5.xml
6  /ppt/theme/themeOverride6.xml
7  /ppt/theme/themeOverride7.xml
8  /ppt/theme/themeOverride8.xml
9  /ppt/theme/themeOverride9.xml
10 /ppt/theme/themeOverride10.xml

```

11 The package-relationship item contains the following:

```

12 <Relationships xmlns="...">
13   <Relationship Id="rId1"
14     Type="http://.../officeDocument" Target="ppt/presentation.xml"/>
15   <Relationship Id="rId3"
16     Type="http://.../core-properties" Target="docProps/core.xml"/>
17   <Relationship Id="rId2"
18     Type="http://.../thumbnail" Target="docProps/thumbnail.wmf"/>
19   <Relationship Id="rId4"
20     Type="http://.../extended-properties" Target="docProps/app.xml"/>
21 </Relationships>

```

22 *end example]*

23 13.3 Part Summary

24 The subclauses subordinate to this one describe in detail each of the part types specific to PresentationML.

25 [Note: For convenience, information from those subclauses is summarized in the following table:

Part	Relationship Target of	Root Element	Ref.
Comment Authors	Presentation	cmAuthorLst	§13.3.1
Comments	Slide	cmLst	§13.3.2
Handout Master	Presentation	handoutMaster	§13.3.3
Notes Master	Notes Slide, Presentation	notesMaster	§13.3.4
Notes Slide	Slide	notes	§13.3.5
Presentation	PresentationML package	presentation	§13.3.6
Presentation Properties	Presentation	presentationPr	§13.3.7
Slide	Presentation	sld	§13.3.8

Part	Relationship Target of	Root Element	Ref.
Slide Layout	Slide Master, Notes Slide, Presentation, Slide, Slide Master	sldLayout	§13.3.9
Slide Master	Presentation, Slide Layout	sldMaster	§13.3.10
Slide Synchronization Data	Slide	sldSyncPr	§13.3.11
User-Defined Tags	Presentation, Slide	tagLst	§13.3.12
View Properties	Presentation	viewPr	§13.3.13

1 *end note]*

2 **13.3.1 Comment Authors Part**

Content Type:	application/vnd.openxmlformats-officedocument.presentationml.commentAuthors+xml
Root Namespace:	http://schemas.openxmlformats.org/presentationml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/commentAuthors

3

4 An instance of this part type contains information about each author who has added a comment to the
5 document. That information includes the author's name, initials, a unique author-ID, a last-comment-index-
6 used count, and a display color. (The color can be used when displaying comments to distinguish comments
7 from different authors.)

8 A package shall contain at most one Comment Authors part. If it exists, that part shall be the target of an
9 implicit relationship from the Presentation (§13.3.6) part.

10 *[Example:* The following Presentation part relationship item contains a relationship to the Comment Authors
11 part, which is stored in the ZIP item commentAuthors.xml:

```

12     <Relationships xmlns="...">
13         <Relationship Id="rId8"
14             Type="http://.../commentAuthors" Target="commentAuthors.xml"/>
15     </Relationships>

```

16 *end example]*

17 The root element for a part of this content type shall be cmAuthorLst.

18 *[Example:* Two people have authored comments in this document: Mary Smith and Peter Jones. Her initials are
19 "mas", her author-ID is 0, and her comments' display color index is 0. Since Mary's last-comment-index-used
20 value is 3, the next comment-index to be used for her will be 4. His initials are "pjj", his author-ID is 1, and his

1 comments' display color index is 1. Since Peter's last-comment-index-used value is 1, the next comment-index
2 to be used for him will be 2:

```
3   <p:cmAuthorLst xmlns:p="..." ...>
4     <p:cmAuthor id="0" name="Mary Smith" initials="mas" lastIdx="3"
5       clrIdx="0"/>
6     <p:cmAuthor id="1" name="Peter Jones" initials="pjj" lastIdx="1"
7       clrIdx="1"/>
8   </p:cmAuthorLst>
```

9 *end example]*

10 A Comment Authors part shall be located within the package containing the source relationship (expressed
11 syntactically, the TargetMode attribute of the Relationship element shall be Internal).

12 A Comment Authors part shall not have implicit or explicit relationships to any other part defined by this
13 Standard.

14 13.3.2 Comments Part

Content Type:	application/vnd.openxmlformats-officedocument.presentationml.comments+xml
Root Namespace:	http://schemas.openxmlformats.org/presentationml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/comments

15

16 An instance of this part type contains the comments for a single slide. Each comment is tied to its author via an
17 author-ID. Each comment's index number and author-ID combination are unique.

18 A package shall contain one Comments part for each slide containing one or more comments, and each of
19 those parts shall be the target of an implicit relationship from its corresponding Slide (§13.3.8) part.

20 *[Example:* The following Slide part-relationship item contains a relationship to a Comments part, which is
21 stored in the ZIP item ../comments/comment2.xml:

```
22   <Relationships xmlns="...">
23     <Relationship Id="rId4"
24       Type="http://../comments"
25       Target="../comments/comment2.xml"/>
26   </Relationships>
```

27 *end example]*

28 The root element for a part of this content type shall be cmLst .

[Example: The Comments part contains three comments, two created by one author, and one created by another, all at the dates and times shown. The index numbers are assigned on a per-author basis, starting at 1 for an author's first comment:

```
<p:cmLst xmlns:p="..." ...>
  <p:cm authorId="0" dt="2005-11-13T17:00:22.071" idx="1">
    <p:pos x="4486" y="1342"/>
    <p:text>Comment text goes here.</p:text>
  </p:cm>
  <p:cm authorId="0" dt="2005-11-13T17:00:34.849" idx="2">
    <p:pos x="3607" y="1867"/>
    <p:text>Another comment's text goes here.</p:text>
  </p:cm>
  <p:cm authorId="1" dt="2005-11-15T00:06:46.919" idx="1">
    <p:pos x="1493" y="2927"/>
    <p:text>comment ...</p:text>
  </p:cm>
</p:cmLst>
```

end example]

A Comments part shall be located within the package containing the source relationship (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

A Comments part shall not have implicit or explicit relationships to any other part defined by this Standard.

13.3.3 Handout Master Part

Content Type:	application/vnd.openxmlformats-officedocument.presentationml.handoutMaster+xml
Root Namespace:	http://schemas.openxmlformats.org/presentationml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/handoutMaster

An instance of this part type contains the look, position, and size of the slides, notes, header and footer text, date, or page number on the presentation's handout.

A package shall contain at most one Handout Master part, and it shall be the target of an explicit relationship from the Presentation (§13.3.6) part.

[Example: The following Presentation part-relationship item contains a relationship to the Handout Master part, which is stored in the ZIP item handoutMasters/handoutMaster1.xml:

```

1      <Relationships xmlns="...">
2      <Relationship Id="rId5"
3      Type="http://.../handoutMaster"
4      Target="handoutMasters/handoutMaster1.xml"/>
5      </Relationships>

```

6 *end example]*

7 The root element for a part of this content type shall be handoutMaster.

8 *[Example:*

```

9      <p:handoutMaster xmlns:p="...">
10      <p:cSld name="">
11      ...
12      </p:cSld>
13      <p:clrMap ... />
14      </p:handoutMaster>

```

15 *end example]*

16 A Handout Master part shall be located within the package containing the source relationship (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

18 A Handout Master part is permitted to have implicit relationships to the following parts defined by this Standard:

- 20 • Additional Characteristics (§15.2.1)
- 21 • Bibliography (§15.2.3)
- 22 • Custom XML Data Storage (§15.2.4)
- 23 • Theme (§14.2.7)
- 24 • Thumbnail (§15.2.14)

25 A Handout Master part is permitted to have explicit relationships to the following parts defined by this Standard:

- 27 • Audio (§15.2.2)
- 28 • Chart (§14.2.1)
- 29 • Diagrams: Diagram Colors (§14.2.3), Diagram Data (§14.2.4), Diagram Layout Definition (§14.2.5) and Diagram Styles (§14.2.6)
- 31 • Embedded Control Persistence (§15.2.8)
- 32 • Embedded Object (§15.2.9)
- 33 • Embedded Package (§15.2.10)
- 34 • Hyperlink (§15.3)
- 35 • Image (§15.2.13)

- Video (§15.2.14)
- VML Drawing (§15.2.17)

A Handout Master part shall not have implicit or explicit relationships to any other part defined by this Standard.

13.3.4 Notes Master Part

Content Type:	application/vnd.openxmlformats-officedocument.presentationml.notesMaster+xml
Root Namespace:	http://schemas.openxmlformats.org/presentationml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/notesMaster

An instance of this part type contains information about the content and formatting of all notes pages.

A package shall contain at most one Notes Master part, and that part shall be the target of an implicit relationship from the Notes Slide (§13.3.5) part, as well as an explicit relationship from the Presentation (§13.3.6) part.

[*Example:* The following Presentation part-relationship item contains a relationship to the Notes Master part, which is stored in the ZIP item notesMasters/notesMaster1.xml:

```
<Relationships xmlns="...">
  <Relationship Id="rId4"
    Type="http://.../notesMaster" Target="notesMasters/notesMaster1.xml"/>
</Relationships>
```

end example]

The root element for a part of this content type shall be notesMaster.

[*Example:*

```
<p:notesMaster xmlns:p="...">
  <p:cSld name="">
    ...
  </p:cSld>
  <p:clrMap ... />
</p:notesMaster>
```

end example]

A Notes Master part shall be located within the package containing the source relationship (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

1 A Notes Master part is permitted to have implicit relationships to the following parts defined by this Standard:

- 2 • Additional Characteristics (§15.2.1)
- 3 • Bibliography (§15.2.3)
- 4 • Custom XML Data Storage (§15.2.4)
- 5 • Theme (§14.2.7)
- 6 • Thumbnail (§15.2.14)

7 A Notes Master part is permitted to have explicit relationships to the following parts defined by this Standard:

- 8 • Audio (§15.2.2)
- 9 • Chart (§14.2.1)
- 10 • Diagrams: Diagram Colors (§14.2.3), Diagram Data (§14.2.4), Diagram Layout Definition (§14.2.5) and
- 11 Diagram Styles (§14.2.6)
- 12 • Embedded Control Persistence (§15.2.8)
- 13 • Embedded Object (§15.2.9)
- 14 • Embedded Package (§15.2.10)
- 15 • Hyperlink (§15.3).
- 16 • Image (§15.2.13)
- 17 • Video (§15.2.14)
- 18 • VML Drawing (§15.2.17)

19 The Notes Master part shall not have implicit or explicit relationships to any other part defined by this
20 Standard.

21 13.3.5 Notes Slide Part

Content Type:	application/vnd.openxmlformats-officedocument.presentationml.notesSlide+xml
Root Namespace:	http://schemas.openxmlformats.org/presentationml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/notesSlide

22

23 An instance of this part type contains the notes for a single slide.

24 A package shall contain one Notes Slide part for each slide that contains notes. If they exist, those parts shall
25 each be the target of an implicit relationship from the Slide (§13.3.8) part.

26 *[Example:* The following Slide part-relationship item contains a relationship to a Notes Slide part, which is
27 stored in the ZIP item ../notesSlides/notesSlide1.xml:

```

1      <Relationships xmlns="...">
2          <Relationship Id="rId3"
3              Type="http://.../notesSlide" Target="../notesSlides/notesSlide1.xml"/>
4      </Relationships>

```

5 *end example]*

6 The root element for a part of this content type shall be notes.

7 *[Example:*

```

8      <p:notes xmlns:p="...">
9          <p:cSld name="">
10              ...
11          </p:cSld>
12          <p:clrMapOvr>
13              <a:masterClrMapping/>
14          </p:clrMapOvr>
15      </p:notes>

```

16 *end example]*

17 A Notes Slide part shall be located within the package containing the source relationship (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

19 A Notes Slide part is permitted to have implicit relationships to the following parts defined by this Standard:

- 20 • Additional Characteristics (§15.2.1)
- 21 • Bibliography (§15.2.3)
- 22 • Custom XML Data Storage (§15.2.4)
- 23 • Notes Master (§13.3.4)
- 24 • Theme Override (§14.2.8)
- 25 • Thumbnail (§15.2.14)

26 A Notes Slide part is permitted to have explicit relationships to the following parts defined by this Standard:

- 27 • Audio (§15.2.2)
- 28 • Chart (§14.2.1)
- 29 • Diagrams: Diagram Colors (§14.2.3), Diagram Data (§14.2.4), Diagram Layout Definition (§14.2.5) and Diagram Styles (§14.2.6)
- 31 • Embedded Control Persistence (§15.2.8)
- 32 • Embedded Object (§15.2.9)
- 33 • Embedded Package (§15.2.10)
- 34 • Hyperlink (§15.3).
- 35 • Image (§15.2.13)

- Video (§15.2.14)
- VML Drawing (§15.2.17)

The Notes Slide part shall not have implicit or explicit relationships to any other part defined by this Standard.

13.3.6 Presentation Part

Content Type:	application/vnd.openxmlformats-officedocument.presentationml.presentation.main+xml application/vnd.openxmlformats-officedocument.presentationml.slideshow.main+xml application/vnd.openxmlformats-officedocument.presentationml.template.main+xml
Root Namespace:	http://schemas.openxmlformats.org/presentationml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/officeDocument

An instance of this part type contains the definition for a slide presentation.

A package shall contain exactly one Presentation part, and that part shall be the target of a relationship in the package-relationship item.

[*Example:* The following PresentationML's package-relationship item contains a relationship to the Presentation part, which is stored in the ZIP item ppt/presentation.xml:

```
<Relationships xmlns="...">
  <Relationship Id="rId1"
    Type="http://.../officeDocument" Target="ppt/presentation.xml"/>
</Relationships>
```

end example]

The root element for a part of this content type shall be presentation.

[*Example:* This presentation contains two slides:

```
<p:presentation xmlns:p="..." ... >
  <p:sldMasterIdLst>
    <p:sldMasterId
      xmlns:rel="http://.../relationships" rel:id="rId1"/>
  </p:sldMasterIdLst>
  <p:notesMasterIdLst>
    <p:notesMasterId
      xmlns:rel="http://.../relationships" rel:id="rId4"/>
  </p:notesMasterIdLst>
```

```

1      <p:handoutMasterIdLst>
2          <p:handoutMasterId
3              xmlns:rel="http://.../relationships" rel:id="rId5"/>
4      </p:handoutMasterIdLst>
5      <p:sldIdLst>
6          <p:sldId id="267"
7              xmlns:rel="http://.../relationships" rel:id="rId2"/>
8          <p:sldId id="256"
9              xmlns:rel="http://.../relationships" rel:id="rId3"/>
10     </p:sldIdLst>
11     <p:sldSz cx="9144000" cy="6858000"/>
12     <p:notesSz cx="6858000" cy="9144000"/>
13 </p:presentation>

```

14 *end example]*

15 A Presentation part shall be located within the package containing the source relationship (expressed
16 syntactically, the TargetMode attribute of the Relationship element shall be Internal).

17 A Presentation part is permitted to have implicit relationships to the following parts defined by this Standard:

- 18 • Additional Characteristics (§15.2.1)
- 19 • Comments (§13.3.2)
- 20 • Bibliography (§15.2.3)
- 21 • Custom XML Data Storage (§15.2.4)
- 22 • Font (§15.2.12)
- 23 • Presentation Properties (§13.3.7)
- 24 • Table Styles (§14.2.9)
- 25 • Theme (§14.2.7)
- 26 • Thumbnail (§15.2.14)
- 27 • View Properties (§13.3.13).

28 A Presentation part is permitted to have explicit relationships to the following parts defined by this Standard:

- 29 • Notes Master (§13.3.4)
- 30 • Handout Master (§13.3.3)
- 31 • Slide (§13.3.8)
- 32 • Slide Layout (§13.3.9)
- 33 • Slide Master (§13.3.10)
- 34 • User Defined Tags (§13.3.12)

13.3.7 Presentation Properties Part

Content Type:	application/vnd.openxmlformats-officedocument.presentationml.presentationProperties+xml
Root Namespace:	http://schemas.openxmlformats.org/presentationml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/presProps

2

3 An instance of this part type contains all the presentation's properties.

4 A package shall contain exactly one Presentation Properties part, and that part shall be the target of an implicit
5 relationship from the Presentation (§13.3.6) part.

6 *[Example:* The following Presentation part-relationship item contains a relationship to the Presentation
7 Properties part, which is stored in the ZIP item presProps.xml:

```
8     <Relationships xmlns="...">
9         <Relationship Id="rId6"
10             Type="http://.../presProps" Target="presProps.xml"/>
11     </Relationships>
```

12 *end example]*

13 The root element for a part of this content type shall be presentationPr.

14 *[Example:*

```
15     <p:presentationPr xmlns:p="..." ...>
16         <p:clrMru>
17             ...
18         </p:clrMru>
19         ...
20     </p:presentationPr>
```

21 *end example]*

22 A Presentation Properties part shall be located within the package containing the source relationship
23 (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

24 A Presentation Properties part shall not have implicit or explicit relationships to any other part defined by this
25 Standard.

26 13.3.8 Slide Part

Content Type:	application/vnd.openxmlformats-officedocument.presentationml.slide+xml
---------------	--

Root Namespace:	http://schemas.openxmlformats.org/presentationml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/slide

1

2 A Slide part contains the contents of a single slide.

3 A package shall contain one Slide part per slide, and each of those parts shall be the target of an explicit
4 relationship from the Presentation (§13.3.6) part.

5 *[Example:* Consider a PresentationML document having two slides. The corresponding Presentation part-
6 relationship item contains two relationships to Slide parts, which are stored in the ZIP items slides/slide1.xml
7 and slides/slide2.xml:

```
8     <Relationships xmlns="...">
9         <Relationship Id="rId2"
10             Type="http://.../slide" Target="slides/slide1.xml"/>
11         <Relationship Id="rId3"
12             Type="http://.../slide" Target="slides/slide2.xml"/>
13     </Relationships>
```

14 *end example]*

15 The root element for a part of this content type shall be sld.

16 *[Example:* slides/slide1.xml contains:

```
17     <p:sld xmlns:p="...">
18         <p:cSld name="">
19             ...
20         </p:cSld>
21         <p:clrMapOvr>
22             ...
23         </p:clrMapOvr>
24         <p:timing>
25             <p:tnLst>
26                 <p:par>
27                     <p:cTn id="1" dur="indefinite" restart="never" nodeType="tmRoot"/>
28                 </p:par>
29             </p:tnLst>
30         </p:timing>
31     </p:sld>
```

32 *end example]*

1 A Slide part shall be located within the package containing the source relationship (expressed syntactically, the
2 TargetMode attribute of the Relationship element shall be Internal).

3 A Slide part is permitted to have implicit relationships to the following parts defined by this Standard:

- 4 • Additional Characteristics (§15.2.1)
- 5 • Bibliography (§15.2.3)
- 6 • Comments (§12.3.3)
- 7 • Custom XML Data Storage (§15.2.4)
- 8 • Notes Slide (§13.3.5)
- 9 • Theme Override (§14.2.8)
- 10 • Thumbnail (§15.2.14)
- 11 • Slide Layout (§13.3.9)
- 12 • Slide Synchronization Data (§13.3.11)

13 A Slide part is permitted to have explicit relationships to the following parts defined by this Standard:

- 14 • Audio (§15.2.2)
- 15 • Chart (§14.2.1)
- 16 • Diagrams: Diagram Colors (§14.2.3), Diagram Data (§14.2.4), Diagram Layout Definition (§14.2.5) and
17 Diagram Styles (§14.2.6)
- 18 • Embedded Control Persistence (§15.2.8)
- 19 • Embedded Object (§15.2.9)
- 20 • Embedded Package (§15.2.10)
- 21 • Hyperlink (§15.3).
- 22 • Image (§15.2.13)
- 23 • User Defined Tags (§13.3.12)
- 24 • Video (§15.2.14)
- 25 • VML Drawing (§15.2.17)

26 A Slide part shall not have implicit or explicit relationships to any other part defined by this Standard.

27 13.3.9 Slide Layout Part

Content Type:	application/vnd.openxmlformats-officedocument.presentationml.slideLayout+xml
Root Namespace:	http://schemas.openxmlformats.org/presentationml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/slideLayout

28

29 An instance of this part type contains the definition for a slide layout template for this presentation. This
30 template defines the default appearance and positioning of drawing objects on this slide type when it is
31 created.

1 A package shall contain one or more Slide Layout parts, and those parts shall be the target of an explicit
 2 relationship in the Presentation (§13.3.6) part, as well as an implicit relationship from the Slide Master
 3 (§13.3.10) part associated with this slide layout.

4 *[Example:* The following Slide Master part-relationship item contains relationships to several Slide Layout
 5 parts, which are stored in the ZIP items ../slideLayouts/slideLayoutN.xml:

```

6     <Relationships xmlns="...">
7         <Relationship Id="rId1"
8             Type="http://.../slideLayout"
9             Target="../slideLayouts/slideLayout1.xml"/>
10        <Relationship Id="rId2"
11            Type="http://.../slideLayout"
12            Target="../slideLayouts/slideLayout2.xml"/>
13        <Relationship Id="rId3"
14            Type="http://.../slideLayout"
15            Target="../slideLayouts/slideLayout3.xml"/>
16    </Relationships>

```

17 *end example]*

18 The root element for a part of this content type shall be sldLayout.

19 *[Example:*

```

20    <p:sldLayout xmlns:p="..." matchingName="" type="title" preserve="1">
21        <p:cSld name="Title Slide">
22            ...
23        </p:cSld>
24        <p:clrMapOvr>
25            <a:masterClrMapping/>
26        </p:clrMapOvr>
27        <p:timing/>
28    </p:sldLayout>
29 </p:sldMaster>

```

30 *end example]*

31 A Slide Layout part is permitted to have implicit relationships to the following parts defined by this Standard:

- 32 • Additional Characteristics (§15.2.1)
- 33 • Bibliography (§15.2.3)
- 34 • Custom XML Data Storage (§15.2.4)
- 35 • Slide Master (§13.3.10)
- 36 • Theme Override (§14.2.8)

- Thumbnail (§15.2.14)

A Slide Layout part is permitted to have explicit relationships to the following parts defined by this Standard:

- Audio (§15.2.2)
- Chart (§14.2.1)
- Diagrams: Diagram Colors (§14.2.3), Diagram Data (§14.2.4), Diagram Layout Definition (§14.2.5) and Diagram Styles (§14.2.6)
- Embedded Control Persistence (§15.2.8)
- Embedded Object (§15.2.9)
- Embedded Package (§15.2.10)
- Hyperlink (§15.3).
- Image (§15.2.13)
- Video (§15.2.14)
- VML Drawing (§15.2.17)

A Slide Layout part shall not have implicit or explicit relationships to any other part defined by this Standard.

13.3.10 Slide Master Part

Content Type:	application/vnd.openxmlformats-officedocument.presentationml.slideMaster+xml
Root Namespace:	http://schemas.openxmlformats.org/presentationml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/slideMaster

An instance of this part type contains the master definition of formatting, text, and objects that appear on each slide in the presentation that is derived from this slide master.

A package shall contain one or more Slide Master parts, each of which shall be the target of an explicit relationship from the Presentation (§13.3.6) part, as well as an implicit relationship from any Slide Layout (§13.3.9) part where that slide layout is defined based on this slide master. Each can optionally be the target of a relationship in a Slide Layout (§13.3.9) part as well.

[Example: The following Presentation part-relationship item contains a relationship to the Slide Master part, which is stored in the ZIP item slideMasters/slideMaster1.xml:

```
<Relationships xmlns="...">
  <Relationship Id="rId1"
    Type="http://.../slideMaster" Target="slideMasters/slideMaster1.xml"/>
</Relationships>
```

end example]

1 The root element for a part of this content type shall be sldMaster.

2 *[Example:*

```

3     <p:sldMaster xmlns:p="">
4         <p:cSld name="">
5             ...
6         </p:cSld>
7         <p:clrMap ... />
8     </p:sldMaster>

```

9 *end example]*

10 A Slide Master part shall be located within the package containing the source relationship (expressed
11 syntactically, the TargetMode attribute of the Relationship element shall be Internal).

12 A Slide Master part is permitted to have implicit relationships to the following parts defined by this Standard:

- 13 • Additional Characteristics (§15.2.1)
- 14 • Bibliography (§15.2.3)
- 15 • Custom XML Data Storage (§15.2.4)
- 16 • Slide Layout (§13.3.9)
- 17 • Theme (§14.2.7)
- 18 • Thumbnail (§15.2.14)

19 A Slide Master part is permitted to have explicit relationships to the following parts defined by this Standard:

- 20 • Audio (§15.2.2)
- 21 • Chart (§14.2.1)
- 22 • Diagrams: Diagram Colors (§14.2.3), Diagram Data (§14.2.4), Diagram Layout Definition (§14.2.5) and
23 Diagram Styles (§14.2.6)
- 24 • Embedded Control Persistence (§15.2.8)
- 25 • Embedded Object (§15.2.9)
- 26 • Embedded Package (§15.2.10)
- 27 • Hyperlink (§15.3).
- 28 • Image (§15.2.13)
- 29 • Video (§15.2.14)
- 30 • VML Drawing (§15.2.17)

31 A Slide Master part shall not have implicit or explicit relationships to any other part defined by this Standard.

32 **13.3.11 Slide Synchronization Data Part**

Content Type:	application/vnd.openxmlformats-officedocument.presentationml.slideUpdateInfo+xml
---------------	--

Root Namespace:	http://schemas.openxmlformats.org/presentationml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/slideUpdateInfo

1

2 An instance of this part type contains properties specifying the current state of a slide that is being
3 synchronized with a version of that slide stored on a central server.

4 A package shall contain zero or one Slide Synchronization Data part for each slide stored in the presentation,
5 and that part shall be the target of an implicit relationship from the corresponding Slide (§13.3.8) part.

6 [Example: The following Slide part-relationship item contains a relationship to the Slide Synchronization Data
7 part, which is stored in the ZIP item slideUpdateInfo/slideUpdateInfo1.xml:

```
8 <Relationships xmlns="...">
9   <Relationship Id="rId1" Type="http://.../slideUpdateInfo"
10     Target="slideUpdateInfo/slideUpdateInfo1.xml"/>
11 </Relationships>
```

12 *end example]*

13 The root element for a part of this content type shall be sldSyncPr.

14 [Example:

```
15 <p:sldSyncPr xmlns:p="..." serverSldId="1"
16   serverSldModifiedTime="2006-08-12T01:31:08"
17   clientInsertedTime="2006-08-12T01:34:11.227" />
```

18 *end example]*

19 A Slide Synchronization Data part shall be located within the package containing the source relationship
20 (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

21 A Slide Synchronization Data part is permitted to have implicit relationships to the following parts defined by
22 this Standard:

- 23 • Slide Synchronization Server Location (§13.4)

24 A Slide Synchronization Data part shall not have implicit or explicit relationships to any other part defined by
25 this Standard.

26 13.3.12 User Defined Tags Part

Content Type:	application/vnd.openxmlformats-officedocument.presentationml.tags+xml
---------------	---

Root Namespace:	http://schemas.openxmlformats.org/presentationml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/tags

1

2 An instance of this part type contains a set of user-defined properties for an object in a presentation (each
3 property consisting of a name/value pair).

4 A package shall contain zero or more User Defined Tags parts, zero or one as the target of an explicit
5 relationship from the corresponding Presentation (§13.3.6) or Slide (§13.3.8) part.

6 *[Example:* The following Slide part-relationship item contains a relationship to the User Defined Tags part,
7 which is stored in the ZIP item tags/tag1.xml:

```
8     <Relationships xmlns="...">
9         <Relationship Id="rId1" Type="http://.../tag"
10             Target="tags/tag1.xml"/>
11     </Relationships>
```

12 *end example]*

13 The root element for a part of this content type shall be tagLst.

14 *[Example:*

```
15     <p:tagLst xmlns:p="..." >
16         <p:tag name="testTagName" val="testTagValue" />
17     </p:tagLst>
```

18 *end example]*

19 A User Defined Tags part shall be located within the package containing the source relationship (expressed
20 syntactically, the TargetMode attribute of the Relationship element shall be Internal).

21 A User Defined Tags part shall not have implicit or explicit relationships to any other part defined by this
22 Standard.

23 13.3.13 View Properties Part

Content Type:	application/vnd.openxmlformats-officedocument.presentationml.viewProps+xml
Root Namespace:	http://schemas.openxmlformats.org/presentationml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/viewProps

24

1 An instance of this part type contains display properties for this presentation.

2 A package shall contain zero or one View Properties part, and if it exists, that part shall be the target of an
3 implicit relationship from the Presentation (§13.3.6) part.

4 *[Example:* The following Presentation part-relationship item contains a relationship to the View Properties
5 part, which is stored in the ZIP item viewProps.xml:

```
6     <Relationships xmlns="...">
7       <Relationship Id="rId7"
8         Type="http://.../viewProps" Target="viewProps.xml"/>
9     </Relationships>
```

10 *end example]*

11 The root element for a part of this content type shall be viewPr.

12 *[Example:*

```
13     <p:viewPr xmlns:p="..." ...>
14       <p:normalViewPr showOutlineIcons="0">
15         ...
16       </p:normalViewPr>
17       <p:slideViewPr>
18         ...
19       </p:slideViewPr>
20       <p:outlineViewPr>
21         ...
22       </p:outlineViewPr>
23       <p:notesTextViewPr>
24         ...
25       </p:notesTextViewPr>
26       <p:gridSpacing cx="78028800" cy="78028800"/>
27     </p:viewPr>
```

28 *end example]*

29 A View Properties part shall be located within the package containing the source relationship (expressed
30 syntactically, the TargetMode attribute of the Relationship element shall be Internal).

31 A View Properties part shall not have implicit or explicit relationships to any other part defined by this
32 Standard.

33 13.4 HTML Publish Location

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/htmlPubSaveAs
----------------------	---

When a presentation specifies an external location to which an optional copy might be pushed in the HTML format, this relationship shall be used to target the location where the HTML copy of the presentation is published.

A package shall contain one HTML Publish Location relationship for each slide linked with an HTML publish location, and that relationships shall be an explicit relationship from the corresponding Presentation Properties (§13.3.7) part.

[Example: A Presentation Properties part, which stores an HTML Publish Location of <http://www.openxmlformats.org/test.htm> will contain the following relationship in that part's relationship part:

```
<Relationships xmlns="...">
  <Relationship Id="rId1"
    Type="http://.../htmlPubSaveAs"
    Target="http://www.openxmlformats.org/test.htm" type="External"/>
</Relationships>
```

end example]

An HTML publish location shall be located external to the package containing the source relationship (expressed syntactically, the TargetMode attribute of the Relationship element shall be External).

13.5 Slide Synchronization Server Location

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/slideUpdateUrl
----------------------	---

When a slide is being synchronized with a copy stored on a remote server, this relationship shall be used to target the location where the server copy of the slide is stored.

A package shall contain one Slide Synchronization Server Location relationship for each slide linked with server data, and that relationships shall be an implicit relationship from the corresponding Slide Synchronization Data (§13.3.11) part.

[Example: A Slide Synchronization Data part that stores information about a slide that is synchronized with a server located at <http://www.openxmlformats.org/slides/> will contain the following relationship in that part's relationship part item:

```
1      <Relationships xmlns="...">
2        <Relationship Id="rId1"
3          Type="http://.../slideupdateUrl"
4          Target="http://www.openxmlformats.org/slides/" type="External"/>
5      </Relationships>
```

6 *end example]*

7 A slide synchronization server location shall be located external to the package containing the source
8 relationship (expressed syntactically, the TargetMode attribute of the Relationship element shall be
9 External).

14. DrawingML

The relationship items and parts defined in this clause are used by one or more of WordprocessingML (§10), SpreadsheetML (§12), and PresentationML (§13) environments.

14.1 Glossary of DrawingML-Specific Terms

diagram — A picture or graphical representation that is displayed using a related set of color, data, layout, and style parts. Examples of diagram types are cycle, organization chart, pyramid, target, and Venn.

14.2 Part Summary

The subclauses subordinate to this one describe in detail each of the part types specific to DrawingML.

[Note: For convenience, information from those subclauses is summarized in the following table:

Part	Relationship Target of	Root Element	Ref.
Chart	WordprocessingML: Main Document SpreadsheetML: Drawings PresentationML: Handout Master, Notes Master, Notes Slide, Slide Layout, Slide Master, Slide All: Chart Drawing	chartSpace	§14.2.1
Chart Drawing	All: Chart	userShapes	§14.2.2
Diagram Colors	WordprocessingML: Main Document SpreadsheetML: Drawings PresentationML: Handout Master, Notes Master, Notes Slide, Slide Layout, Slide Master, Slide	colorsDef	§14.2.3
Diagram Data	WordprocessingML: Main Document SpreadsheetML: Drawings PresentationML: Handout Master, Notes Master, Notes Slide, Slide Layout, Slide Master, Slide	dataModel	§14.2.4
Diagram Layout Definition	WordprocessingML: Main Document SpreadsheetML: Drawings PresentationML: Handout Master, Notes Master, Notes Slide, Slide Layout, Slide Master, Slide	layoutDef	§14.2.5

Part	Relationship Target of	Root Element	Ref.
Diagram Style	WordprocessingML: Main Document SpreadsheetML: Drawings PresentationML: Handout Master, Notes Master, Notes Slide, Slide Layout, Slide Master, Slide	styleDef	§14.2.6
Theme	WordprocessingML: Main Document SpreadsheetML: Workbook PresentationML: Handout Master, Notes Master, Presentation, Slide Master	officeStyleSheet	§14.2.7
Theme Override	PresentationML: Notes Slide, Slide, Slide Layout	themeOverride	§14.2.8
Table Styles	PresentationML: Presentation	tblStyleLst	§14.2.9

1 *end note]*

2 **14.2.1 Chart Part**

Content Type:	application/vnd.openxmlformats-officedocument.drawingml.chart+xml
Root Namespace:	http://schemas.openxmlformats.org/drawingml/2006/chart
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/chart

3

4 An instance of this part type describes a chart.

5 A package shall contain a Chart part for each chart in the document. In a WordprocessingML document, each
6 such part shall be the target of an explicit relationship in a Main Document (§11.3.10) part. In a
7 SpreadsheetML document, each such part shall be the target of an explicit relationship in a Drawings (§12.3.8)
8 part. In a PresentationML document, each such part shall be the target of an explicit relationship in a Handout
9 Master (§13.3.3), Notes Master (§13.3.4), Notes Slide (§13.3.5), Slide (§13.3.8), Slide Layout (§13.3.9), or Slide
10 Master (§13.3.10) part. This part is permitted to also be the target of an explicit relationship in a Chart Drawing
11 (§14.2.2) part, if the chart that points at this Chart Drawing part is the target of a relationship from a
12 Chartsheet part. In other words, the only time a chart can embed another chart is if the parent chart is part of
13 a chartsheet.

14 *[Example:* The following Main Document part-relationship item contains relationships to two Chart parts,
15 *which are stored in the ZIP items ../charts/chartN.xml:*

```

1    <Relationships xmlns="...">
2      <Relationship Id="rId4"
3        Type="http://.../chart" Target="charts/chart1.xml"/>
4      <Relationship Id="rId5"
5        Type="http://.../chart" Target="charts/chart2.xml"/>
6    </Relationships>

```

7 The following Drawings part-relationship item contains a relationship to a Chart part, which is stored in the ZIP
8 item ../charts/chart1.xml:

```

9    <Relationships xmlns="...">
10     <Relationship Id="rId1"
11       Type="http://.../relationships/chart" Target="../charts/chart1.xml"/>
12   </Relationships>

```

13 The following Slide part-relationship item contains relationships to two Chart parts, which are stored in the ZIP
14 items ../charts/chartN.xml:

```

15    <Relationships xmlns="...">
16     <Relationship Id="rId4"
17       Type="http://.../chart" Target="../charts/chart1.xml"/>
18     <Relationship Id="rId5"
19       Type="http://.../chart" Target="../charts/chart2.xml"/>
20   </Relationships>

```

21 *end example]*

22 The root element for a part of this content type shall be chartSpace.

23 [Example: chart1.xml contains the following clustered bar chart:

```

24    <c:chartSpace ...>
25      <c:chart>
26        <c:title>
27          ...
28        </c:title>
29        <c:plotArea>
30          <c:layout>
31            ...
32          </c:layout>
33          <c:barChart>
34            ...
35          </c:barChart>
36        </c:plotArea>

```



```

1      <c:legend>
2      ...
3      </c:legend>
4  </c:chart>
5      ...
6  </c:chartSpace>

```

7 *end example]*

8 For WordprocessingML and PresentationML documents, the data for a chart is not stored in the Chart part
9 directly. Instead, it shall be stored in an embedded SpreadsheetML package (§12.2) targeted by an Embedded
10 Package (§15.2.10) part specified by that Chart part. For SpreadsheetML documents, the data for a chart is
11 stored directly in the Drawing's parent worksheet; no embedded SpreadsheetML package shall be used.

12 A Chart part shall be located within the package containing the source relationship (expressed syntactically,
13 the TargetMode attribute of the Relationship element shall be Internal).

14 A Chart part is permitted to have explicit relationships to the following parts defined by this Standard:

- 15 • Chart Drawing (§14.2.2)
- 16 • Embedded Package (§15.2.10)

17 A Chart part shall not have any implicit or explicit relationships to any other part defined by this Standard.

18 14.2.2 Chart Drawing Part

Content Type:	application/vnd.openxmlformats-officedocument.drawingml.chartshapes+xml
Root Namespace:	http://schemas.openxmlformats.org/drawingml/2006/chart
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/chartUserShapes

19
20 An instance of this part type contains all basic drawing elements (shapes) which are explicitly associated with
21 this chart. These drawing elements are automatically moved with the chart when it is moved and resized when
22 the chart is resized.

23 A package is permitted to contain one Chart Drawing part per chart part, and each such part shall be the target
24 of an explicit relationship from a Chart (§14.2.1) part.

25 *[Example:* The following Chart part-relationship item contains a relationship to a Chart Drawing part, which is
26 stored in the ZIP item ../drawings/drawing1.xml:

```

1    <Relationships xmlns="...">
2      <Relationship Id="rId2"
3        Type="http://.../chartUserShapes" Target="../drawings/drawing1.xml"/>
4    </Relationships>

```

5 *end example]*

6 The root element for a part of this content type shall be userShapes.

7 *[Example:*

```

8      <c:userShapes xmlns:cdr="..." xmlns:c="...">
9        <cdr:relSizeAnchor>
10         ...
11        </cdr:relSizeAnchor>
12      </c:userShapes>

```

13 *end example]*

14 A Chart Drawing part shall be located within the package containing the source relationship (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

16 A Chart Drawing part is permitted to have explicit relationships to the following parts defined by this Standard:

- 17 • Chart (§14.2.1)

18 A Chart Drawing part shall not have any implicit or explicit relationships to any other part defined by this Standard.

20 14.2.3 Diagram Colors Part

Content Type:	application/vnd.openxmlformats-officedocument.drawingml.diagramColors+xml
Root Namespace:	http://schemas.openxmlformats.org/drawingml/2006/diagram
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/diagramColors

21

22 An instance of this part type contains color information for a diagram.

23 A package shall contain exactly one Diagram Colors part per diagram. Each Diagram Colors part shall be the target of an explicit relationship in a WordprocessingML Main Document (§11.3.10), SpreadsheetML Drawings (§12.3.8), or PresentationML Slide (§13.3.8) part.

26 *[Example:* The following SpreadsheetML Drawings part-relationship item contains a relationship to two Diagram Colors parts, which are stored in the ZIP items ../graphics/colorsN.xml.

27

```

1    <Relationships xmlns="...">
2      <Relationship Id="rId4"
3        Type="http://.../diagramColors" Target="../graphics/colors1.xml"/>
4      <Relationship Id="rId8"
5        Type="http://.../diagramColors" Target="../graphics/colors2.xml"/>
6    </Relationships>

```

7 *end example]*

8 The root element for a part of this content type shall be colorsDef.

9 *[Example: colors1.xml contains the following:*

```

10   <cs:colorsDef xmlns:cs="..." uniqueId="..." minVer="12.0">
11     <cs:title lang="" val="Primary Accent 2"/>
12     <cs:desc lang="" val="Primary Accent 2"/>
13     <cs:catLst>
14       <cs:cat type="accent1" pri="11200"/>
15     </cs:catLst>
16     <cs:styleLbl ...>
17       ...
18     </cs:styleLbl>
19     ...
20     <cs:styleLbl ...>
21       ...
22     </cs:styleLbl>
23   </cs:colorsDef>

```

24 *end example]*

25 A Diagram Colors part shall be located within the package containing the source relationship (expressed
26 syntactically, the TargetMode attribute of the Relationship element shall be Internal).

27 A Diagram Colors part shall not have implicit or explicit relationships to any other part defined by this
28 Standard.

29 **14.2.4 Diagram Data Part**

Content Type:	application/vnd.openxmlformats-officedocument.drawingml.diagramData+xml
Root Namespace:	http://schemas.openxmlformats.org/drawingml/2006/diagram
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/diagramData

30

31 An instance of this part type contains the semantic data for a diagram.

A package shall contain exactly one Diagram Data part per diagram. Each Diagram Data part shall be the target of an explicit relationship in a WordprocessingML Main Document (§11.3.10); a SpreadsheetML Drawings part (§12.3.8); or a PresentationML Handout Master (§13.3.3), Notes Master (§13.3.4), Notes Slide (§13.3.5), Slide (§13.3.8), Slide Layout (§13.3.9), or Slide Master (§13.3.10) part.

[*Example:* The following SpreadsheetML Drawings part-relationship item contains a relationship to two Diagram Data parts, which are stored in the ZIP items ../graphics/dataN.xml.

```

7      <Relationships xmlns="...">
8        <Relationship Id="rId1"
9          Type="http://.../diagramData" Target="../graphics/data1.xml"/>
10       <Relationship Id="rId5"
11         Type="http://.../diagramData" Target="../graphics/data2.xml"/>
12     </Relationships>

```

end example]

The root element for a part of this content type shall be dataModel.

[*Example:* data1.xml contains the following:

```

16     <dm:dataModel xmlns:dm="...">
17       <dm:ptLst>
18         ...
19       </dm:ptLst>
20       <dm:cxnLst>
21         ...
22       </dm:cxnLst>
23       <dm:bg/>
24       <dm:whole/>
25     </dm:dataModel>

```

end example]

A Diagram Data part shall be located within the package containing the source relationship (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

A Diagram Data part is permitted to have explicit relationships to the following parts defined by this Standard:

- Image (§15.2.13)

A Diagram Data part shall not have any implicit or explicit relationships to other parts defined by this Standard.

14.2.5 Diagram Layout Definition Part

Content Type:	application/vnd.openxmlformats-officedocument.drawingml.diagramLayout+xml
---------------	---

Root Namespace:	http://schemas.openxmlformats.org/drawingml/2006/diagram
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/diagramLayout

1

2 An instance of this part type is a template that describes how diagram-related data is mapped to a shape.

3 A package shall contain exactly one Diagram Layout Definition part per diagram. Each Layout Definition part
 4 shall be the target of an explicit relationship from a WordprocessingML Main Document (§11.3.10); a
 5 SpreadsheetML Drawings part (§12.3.8); or a PresentationML Handout Master (§13.3.3), Notes Master
 6 (§13.3.4), Notes Slide (§13.3.5), Slide (§13.3.8), Slide Layout (§13.3.9), or Slide Master (§13.3.10) part. If a
 7 document contains multiple diagrams having the same graphic layout definition, each of those diagrams shall
 8 have its own copy of that Diagram Layout Definition part.

9 *[Example:* The following SpreadsheetML Drawings part-relationship item contains a relationship to two
 10 Diagram Layout Definition parts, which are stored in the ZIP items ../graphics/layoutN.xml.

```

11     <Relationships xmlns="...">
12         <Relationship Id="rId2"
13             Type="http://.../diagramLayout" Target="../graphics/layout1.xml"/>
14         <Relationship Id="rId6"
15             Type="http://.../diagramLayout" Target="../graphics/layout2.xml"/>
16     </Relationships>

```

17 *end example]*

18 The root element for a part of this content type shall be layoutDef.

19 *[Example:* layout1.xml contains the following:

```

20     <lo:layoutDef xmlns:lo="..." uniqueId="...2" minVer="12.0" defStyle="">
21         <lo:title lang="" val="Hierarchy 2"/>
22         <lo:desc lang="" val=""/>
23         <lo:catLst>
24             <lo:cat type="hierarchy" pri="2000"/>
25         </lo:catLst>
26         <lo:sampData>
27             ...
28         </lo:sampData>
29         <lo:styleData>
30             ...
31         </lo:styleData>

```

```

1      <lo:clrData>
2      ...
3      </lo:clrData>
4      <lo:layoutNode name="Name0" styleLbl="" moveWith="">
5      ...
6      </lo:layoutNode>
7  </lo:layoutDef>

```

8 *end example]*

9 A Diagram Layout Definition part shall be located within the package containing the source relationship
10 (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

11 A Diagram Layout Definition part is permitted to have explicit relationships to the following parts and items
12 defined by this Standard:

- 13 • Image (§15.2.13)

14 A Diagram Layout Definition part shall not have any implicit or explicit relationships to other parts defined by
15 this Standard.

16 14.2.6 Diagram Style Part

Content Type:	application/vnd.openxmlformats-officedocument.drawingml.diagramStyle+xml
Root Namespace:	http://schemas.openxmlformats.org/drawingml/2006/diagram
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/diagramQuickStyle

18 An instance of this part type maps diagram semantic information to a document's theme.

19 A package shall contain exactly one Diagram Style part per diagram. Each Style part shall be the target of an
20 explicit relationship from a WordprocessingML Main Document (§11.3.10); a SpreadsheetML Drawings part
21 (§12.3.8); or a PresentationML Handout Master (§13.3.3), Notes Master (§13.3.4), Notes Slide (§13.3.5), Slide
22 (§13.3.8), Slide Layout (§13.3.9), or Slide Master (§13.3.10) part.

23 *[Example:* The following SpreadsheetML Drawings part-relationship item contains a relationship to two
24 Diagram Style parts, which are stored in the ZIP items ../graphics/quickStyleN.xml.

```

1    <Relationships xmlns="...">
2      <Relationship Id="rId3"
3        Type="http://.../diagramQuickStyle"
4        Target="../graphics/quickStyle1.xml"/>
5      <Relationship Id="rId7"
6        Type="http://.../diagramQuickStyle"
7        Target="../graphics/quickStyle2.xml"/>
8    </Relationships>

```

9 *end example]*

10 The root element for a part of this content type shall be styleDef.

11 [Example: quickStyle1.xml contains the following:

```

12    <qs:styleDef xmlns:qs="..." uniqueId="..." minVer="12.0">
13      <qs:title lang="" val="Style 2"/>
14      <qs:desc lang="" val="Style 2"/>
15      <qs:catLst>
16        <qs:cat type="simple" pri="10200"/>
17      </qs:catLst>
18      <qs:scene3d>
19        ...
20      </qs:scene3d>
21      <qs:style>
22        ...
23      </qs:style>
24      <qs:styleLbl name="...">
25        ...
26      </qs:styleLbl>
27      ...
28      <qs:styleLbl name="...">
29        ...
30      </qs:styleLbl>
31    </qs:styleDef>

```

32 *end example]*

33 A Diagram Style part shall be located within the package containing the source relationship (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

35 A Diagram Style part shall not have implicit or explicit relationships to other parts defined by this Standard.

36 14.2.7 Theme Part

Content Type:	application/vnd.openxmlformats-officedocument.theme+xml
---------------	---

Root Namespace:	http://schemas.openxmlformats.org/drawingml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/theme

1

2 An instance of this part type contains information about a document's *theme*, which is a combination of *color*
3 *scheme*, *font scheme*, and *format scheme* (the latter also being referred to as *effects*). For a WordprocessingML
4 document, the choice of theme affects the color and style of headings, among other things. For a
5 SpreadsheetML document, the choice of theme affects the color and style of cell contents and charts, among
6 other things. For a PresentationML document, the choice of theme affects the formatting of slides, handouts,
7 and notes via the associated master, among other things.

8 A WordprocessingML or SpreadsheetML package shall contain zero or one Theme part, which shall be the
9 target of an implicit relationship in a Main Document (§11.3.10) or Workbook (§12.3.23) part. A
10 PresentationML package shall contain zero or one Theme part per Handout Master (§13.3.3), Notes Master
11 (§13.3.4), Slide Master (§13.3.10) or Presentation (§13.3.6) part via an implicit relationship.

12 [Example: The following WordprocessingML Main Document part-relationship item contains a relationship to
13 the Theme part, which is stored in the ZIP item theme/theme1.xml:

```
14 <Relationships xmlns="...">
15   <Relationship Id="rId4"
16     Type="http://.../theme" Target="theme/theme1.xml"/>
17 </Relationships>
```

18 *end example]*

19 The root element for a part of this content type shall be officeStyleSheet.

20 [Example: theme1.xml contains the following, where the name attributes of the clrScheme, fontScheme, and
21 fmtScheme elements correspond to the document's color scheme, font scheme, and format scheme,
22 respectively:

```
23 <a:officeStyleSheet xmlns:a="...">
24   <a:baseStyles>
25     <a:clrScheme name="...">
26       ...
27     </a:clrScheme>
28     <a:fontScheme name="...">
29       ...
30     </a:fontScheme>
```



```

1      <a:fmtScheme name="...">
2      ...
3      </a:fmtScheme>
4      </a:baseStyles>
5      <a:objectDefaults/>
6      </a:officeStyleSheet>

```

7 *end example]*

8 A Theme part shall be located within the package containing the source relationship (expressed syntactically,
9 the TargetMode attribute of the Relationship element shall be Internal).

10 A Theme part is permitted to contain explicit relationships to the following parts defined by this Standard:

- 11 • Image (§15.2.13)

12 A Theme part shall not have any implicit or explicit relationships to other parts defined by this Standard.

13 14.2.8 Theme Override Part

Content Type:	application/vnd.openxmlformats-officedocument.themeOverride+xml
Root Namespace:	http://schemas.openxmlformats.org/drawingml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/themeOverride

14
15 An instance of this part type contains information about an object's *theme override*, which are overrides to the
16 *color scheme*, *font scheme*, and *format scheme* (the latter also being referred to as *effects*) for a particular
17 slide, notes slide, or handout.

18 A PresentationML package shall contain zero or one Theme Override part per Notes Slide (§13.3.5), Slide
19 (§13.3.8), or Slide Layout (§13.3.9) part via an implicit relationship.

20 [Example: The following WordprocessingML Main Document part-relationship item contains a relationship to
21 the Theme part, which is stored in the ZIP item theme/theme1.xml:

```

22      <Relationships xmlns="...">
23      <Relationship Id="rId1"
24      Type="http://.../themeOverride" Target="theme/themeoverride1.xml"/>
25      </Relationships>

```

26 *end example]*

27 The root element for a part of this content type shall be *ossOverride*.

28 [Example:

```

1      <a:ossOverride xmlns:a="..." >
2          ...
3      </a:ossOverride>

```

4 *end example]*

5 A Theme Override part shall be located within the package containing the source relationship (expressed
6 syntactically, the TargetMode attribute of the Relationship element shall be Internal).

7 A Theme Override part shall not contain implicit or explicit relationships to other parts defined by this
8 Standard.

9 **14.2.9 Table Styles Part**

Content Type:	application/vnd.openxmlformats-officedocument.presentationml.tableStyles+xml
Root Namespace:	http://schemas.openxmlformats.org/drawingml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/tableStyles

10

11 An instance of this part type contains information about the table styles used by tables in this presentation. A
12 table style defines characteristics such as row and column colors, heading row colors, and text.

13 A PresentationML package shall contain no more than one Table Styles part per Presentation (§13.3.6) part via
14 an implicit relationship.

15 *[Example:* The following Presentation part-relationship item contains a relationship to a Table Styles part,
16 *which is stored in the ZIP item tableStyles.xml:*

```

17      <Relationships xmlns="...">
18          <Relationship Id="rId1"
19              Type="http://.../tableStyles" Target="tableStyles.xml"/>
20      </Relationships>

```

21 *end example]*

22 The root element for a part of this content type shall be tblStyleLst.

23 *[Example:* tablestyles.xml contains the following:

```

24      <a:tblStyleLst xmlns:a="...">
25          <a:tblStyle>
26              ...
27          </a:tblStyle>
28      </a:tblStyleLst>

```

- 1 *end example]*
- 2 A Table Styles part shall be located within the package containing the source relationship (expressed
- 3 syntactically, the TargetMode attribute of the Relationship element shall be Internal).
- 4 A Table Styles part shall not contain implicit or explicit relationships to other parts defined by this Standard.

15. Shared

The relationship items and parts defined in this clause are used by one or more of WordprocessingML (§10), SpreadsheetML (§12), and PresentationML (§13) environments.

15.1 Glossary of Shared Terms

control — A region of active content within an Office Open XML document.

15.2 Part Summary

The subclauses subordinate to this one describe in detail each of the shared part types.

[Note: For convenience, information from those subclauses is summarized in the following table:

Part	Relationship Target of	Root Element	Ref.
Additional Characteristics	Numerous PresentationML, SpreadsheetML, and WordprocessingML parts	Characteristics	§15.2.1
Audio	Numerous PresentationML, SpreadsheetML, and WordprocessingML parts	Not applicable	§15.2.2
Bibliography	Numerous PresentationML, SpreadsheetML, and WordprocessingML parts	Sources	§15.2.3
Custom XML Data Storage	Numerous PresentationML, SpreadsheetML, and WordprocessingML parts	Not applicable	§15.2.4
Custom XML Data Storage Properties	Custom XML Data Storage	datastoreItem	§15.2.5
Digital Signature Origin	WordprocessingML, SpreadsheetML, or PresentationML package	Not applicable	§15.2.6
Digital Signature XML Signature	Digital Signature Origin	Signature	§15.2.7
Embedded Control Persistence	Numerous PresentationML, SpreadsheetML, and WordprocessingML parts	Not applicable	§15.2.8
Embedded Object	Numerous PresentationML, SpreadsheetML, and WordprocessingML parts	Not applicable	§15.2.9

Part	Relationship Target of	Root Element	Ref.
Embedded Package	Numerous PresentationML, SpreadsheetML, and WordprocessingML parts	Not applicable	§15.2.10
File Properties, Extended	WordprocessingML, SpreadsheetML, or PresentationML package	Properties	§ 15.2.11.3
File Properties, Core	WordprocessingML, SpreadsheetML, or PresentationML package	coreProperties	§15.2.11.1
File Properties, Custom	WordprocessingML, SpreadsheetML, or PresentationML package	properties	§15.2.11.2
Font	WordprocessingML Font Table part, PresentationML Presentation part	Not applicable	§15.2.12
Image	Numerous PresentationML, SpreadsheetML, and WordprocessingML parts	Not applicable	§15.2.13
Printer Settings	SpreadsheetML Chartsheet, Dialogsheet, Worksheet parts, WordprocessingML Main Document or Glossary Document parts	Not applicable	§15.2.14
Thumbnail	WordprocessingML, SpreadsheetML, or PresentationML package	Not applicable	§15.2.14
Video part	Numerous PresentationML and WordprocessingML parts	Not applicable	§15.2.16

1 *end note]*

2 **15.2.1 Additional Characteristics Part**

Content Type:	application/xml
Root Namespace:	http://schemas.openxmlformats.org/officeDocument/2006/additionalCharacteristics
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/customXml

3

4 An instance of this part type contains information about additional characteristics of the producer that
5 generated the document, when those characteristics cannot be specified using elements defined by this
6 Standard.

A package is permitted to contain zero or one Additional Characteristics parts, and each such part shall be the target of an implicit relationship from a Main Document (§11.3.10) part in a WordprocessingML package; a Workbook (§12.3.23) part in a SpreadsheetML package; or a Handout Master (§13.3.3), Notes Master (§13.3.4), Notes Slide (§13.3.5), Presentation (§13.3.6), Slide (§13.3.8), Slide Layout (§13.3.9), or Slide Master (§13.3.10) part in a PresentationML package.

[*Example:* The following Main Document part-relationship item contains a relationship to an Additional Characteristics part, which is stored in the ZIP item ../customXML/item2.xml:

```
<Relationships xmlns="...">
  <Relationship Id="rId1"
    Type="http://.../customXmlData" Target="../customXML/item2.xml"/>
</Relationships>
```

end example]

An Additional Characteristics part shall be located within the package containing the source relationship (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

An Additional Characteristics part is permitted to have implicit relationships to the following parts defined by this Standard:

- Custom XML Data Storage Properties (§15.2.5)

An Additional Characteristics part shall not have implicit or explicit relationships to any other part defined by this Standard.

15.2.2 Audio Part

Content Type:	Any supported audio type.	
	[Note: Some example content types are:	
	audio/aiff	http://developer.apple.com/documentation/QuickTime/INMAC/SOUND/imsoundmgr.30.htm
	audio/midi	http://www.midi.org/about-midi/specinfo.shtml
	audio/x-ms-wax	http://msdn.microsoft.com/library/en-us/wmplay10/mmp_sdk/asx_elementsintro.asp
	<i>end note]</i>	
Root Namespace:	not applicable	
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/audio	

An instance of this part type contains an audio file.

A PresentationML package is permitted to contain zero or more Sound parts, each of which shall be the target of a relationship in a Handout Master (§13.3.3), Notes Slide (§13.3.5), Notes Master (§13.3.4), Slide (§13.3.8), Slide Layout (§13.3.9), or Slide Master (§13.3.10) part-relationship item. [Example: The following Slide part-relationship item contains a relationship to a Sound part, which is stored as the file E:\Beethoven's Symphony No. 9.wma:

```
<Relationships xmlns="">
  <Relationship Id="rId1"
    Type="http://.../audio/x-ms-wma"
    Target="file:///E:\Beethoven's%20Symphony%20No.%209.wma"
    TargetMode="External"/>
</Relationships>
```

end example]

An Audio part may be located within or external to the package containing the source relationship (expressed syntactically, the TargetMode attribute of the Relationship element may be Internal or External).

An Audio part is not stored as XML; instead, it involves a relationship target that is an audio clip.

An Audio part shall not have implicit or explicit relationships to other parts defined by this Standard.

15.2.3 Bibliography Part

Content Type:	application/xml
Root Namespace:	http://schemas.openxmlformats.org/officeDocument/2006/bibliography
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/customXml

An instance of this part type contains bibliographic data for the current package.

A package is permitted to contain zero or one Bibliography part, and each such part shall be the target of an implicit relationship in a Main Document (§11.3.10) part in a WordprocessingML package; a Workbook (§12.3.23) part in a SpreadsheetML package; or a Handout Master (§13.3.3), Notes Master (§13.3.4), Notes Slide (§13.3.5), Slide (§13.3.8), Slide Layout (§13.3.9), or Slide Master (§13.3.10) part in a PresentationML package.

[Example: The following Main Document part-relationship item contains a relationship to a Bibliography part, which is stored in the ZIP item ../customXML/bib1.xml:

```
<Relationships xmlns="">
  <Relationship Id="rId1"
    Type="http://.../customXml" Target="../customXML/bib1.xml"/>
</Relationships>
```

1 *end example]*

2 A Bibliography part shall be located within the package containing the source relationship (expressed
3 syntactically, the TargetMode attribute of the Relationship element shall be Internal).

4 A Bibliography part is permitted to have implicit relationships to the following parts defined by this Standard:

- 5 • Custom XML Data Storage Properties (§15.2.5)

6 A Bibliography part shall not have implicit or explicit relationships to any other part defined by this Standard.

7 **15.2.4 Custom XML Data Storage Part**

Content Type:	application/xml
Root Namespace:	any XML allowed
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/customXml

8

9 An instance of this part type can contain arbitrary XML. As such, an instance of this part can be used to
10 roundtrip arbitrary custom XML data with this package.

11 A package is permitted to contain one or more Custom XML Data Storage parts, and each such part shall be the
12 target of an implicit relationship in a Main Document (§11.3.10) part in a WordprocessingML package; a
13 Workbook (§12.3.23) part in a SpreadsheetML package; or a Handout Master (§13.3.3), Notes Master
14 (§13.3.4), Notes Slide (§13.3.5), Presentation (§13.3.6), Slide (§13.3.8), Slide Layout (§13.3.9), or Slide Master
15 (§13.3.10) part in a PresentationML package.

16 [Example: The following Main Document part-relationship item contains a relationship to a Custom XML Data
17 Storage part, which is stored in the ZIP item ../customXML/item1.xml:

```
18 <Relationships xmlns="...">
19   <Relationship Id="rId1"
20     Type="http://.../customXmlData" Target="../customXML/item1.xml"/>
21 </Relationships>
```

22 *end example]*

23 A Custom XML Data Storage part shall be located within the package containing the source relationship
24 (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

25 A Custom XML Data Storage part is permitted to have implicit relationships to the following parts defined by
26 this Standard:

- 27 • Custom XML Data Storage Properties (§15.2.5)

A Custom XML Data Storage part shall not have implicit or explicit relationships to any other part defined by this Standard.

15.2.5 Custom XML Data Storage Properties Part

Content Type:	application/vnd.openxmlformats-officedocument.customXmlProperties+xml
Root Namespace:	http://schemas.openxmlformats.org/officeDocument/2006/customXmlDataProps
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/customXmlProps

An instance of this part type contains the set of properties which are specified for this custom XML data. These properties consist of a unique ID for the storage, as well as information on the set of XML schemas used by this custom XML data storage.

A package is permitted to contain zero or more Custom XML Data Storage Properties parts, and each such part shall be the target of an implicit relationship from a Custom XML Data Storage (§15.2.4) part.

[Example: The following Custom XML Data Storage part-relationship item contains a relationship to a Custom XML Data Storage Properties part, which is stored in the ZIP item itemProps1.xml:

```
<Relationships xmlns="...">
  <Relationship Id="rId1"
    Type="http://.../customXmlProps" Target="itemProps1.xml"/>
</Relationships>
```

end example]

The root element for a part of this content type shall be dataStoreItem.

[Example:

```
<ds:dataStoreItem ds:itemID="{D85...53A}" xmlns:ds="..." /> \
```

end example]

A Custom XML Data Storage Properties part shall be located within the package containing the source relationship (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

A Custom XML Data Storage Properties part shall not have implicit or explicit relationships to other parts defined by this Standard.

15.2.6 Digital Signature Origin Part

Content Type:	application/vnd.openxmlformats-package.digital-signature-origin
---------------	---

Root Namespace:	not applicable
Source Relationship:	http://schemas.openxmlformats.org/package/2006/relationships/digital-signature/origin

1

2 The Digital Signature Origin part is the starting point for navigating through the signatures in a package.

3 This part shall have no contents.

4 A package is permitted to contain zero or one Digital Signature Origin part in a package and it shall the target
5 of a relationship from the package-relationship item for a WordprocessingML, SpreadsheetML, or
6 PresentationML package.

7 *[Example:* The following package-relationship item contains a relationship to a Digital Signature Origin part,
8 *which is stored in the ZIP item origin.sigs:*

```

9     <Relationships xmlns="...">
10       <Relationship Id="rId1"
11         Type="http://.../origin" Target="../origin.sigs"/>
12     </Relationships>

```

13 *end example]*

14 A Digital Signature Origin part shall be located within the package containing the source relationship
15 (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

16 A Digital Signature Origin part is permitted to have implicit relationships to the following parts defined by this
17 Standard:

- 18 • Digital Signature XML Signature (§15.2.7)

19 A Digital Signature Origin part shall not have implicit or explicit relationships to any other part defined by this
20 Standard.

21 15.2.7 Digital Signature XML Signature Part

Content Type:	application/vnd.openxmlformats-package.digital-signature-xmlsignature+xml
Root Namespace:	http://schemas.openxmlformats.org/package/2006/digital-signature
Source Relationship:	http://schemas.openxmlformats.org/package/2006/relationships/digital-signature/signature

22

23 The Digital Signature XML Signature part contains digital signature markup.

1 A package is permitted to contain zero or more Digital Signature XML Signature parts, one for each digital
 2 signature in a package, and each of these shall be the target of an implicit relationship from the Digital
 3 Signature Origin (§15.2.6) part.

4 *[Example:* The following Digital Signature Origin part-relationship item contains a relationship to a Digital
 5 Signature XML Signature part, which is stored in the ZIP item sig1.xml:

```
6 <Relationships xmlns="...">
7   <Relationship Id="rId1"
8     Type="http://.../signature" Target="../sig1.xml"/>
9 </Relationships>
```

10 *end example]*

11 The root element for this part shall be Signature.

12 *[Example:*

```
13 <Signature xmlns="..." Id="idPackageSignature" >
14   ...
15 </Signature>
```

16 *end example]*

17 A Digital Signature XML Signature part shall be located within the package containing the source relationship
 18 (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

19 A Digital Signature XML Signature part shall not have implicit or explicit relationships to any part defined by
 20 this Standard.

21 15.2.8 Embedded Control Persistence Part

Content Type:	Any supported control type. [Note: There are a number of possible control types. One example of a potential control type would be an Active X control, which would use the following content type: application/vnd.ms-office.activeX+xml. end note]
Root Namespace:	not applicable
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/control

22

23 An instance of this part contains information about an embedded control in the package. This information is
 24 provided by the specified control when asked to persist.

A package is permitted to contain one or more Embedded Control Persistence parts, and each such part shall be the target of an explicit relationship in an Endnotes (§11.3.4), Footer (§11.3.6), Footnotes (§11.3.7), Header (§11.3.9), or Main Document (§11.3.10) part-relationship item in a WordprocessingML package; a Worksheet part (§12.3.24) 15.2.17 in a SpreadsheetML package; or a Handout Master (§13.3.3), Notes Slide (§13.3.5), Notes Master (§13.3.4), Slide (§13.3.8), Slide Layout (§13.3.9), Slide Master (§13.3.10) part-relationship item in a PresentationML package.

The content type of this part shall determine the format and contents of the embedded control.

[*Example:* The following example shows the persistence that could be used for an embedded control which is a Java applet within a WordprocessingML document:

```
<w:p>
  <w:r w:rsidR="005810E1">
    <w:object w:dxaOrig="1440" w:dyaOrig="1440">
      <v:shapetype id="_x0000_t75" coordsize="21600,21600" o:spt="75"
o:preferrelative="t" path="m@4@5l@4@11@9@11@9@5xe" filled="f" stroked="f">
        <v:stroke joinstyle="miter" />
        <v:formulas>
          <v:f eqn="if lineDrawn pixelLineWidth 0" />
          <v:f eqn="sum @0 1 0" />
          <v:f eqn="sum 0 0 @1" />
          <v:f eqn="prod @2 1 2" />
          <v:f eqn="prod @3 21600 pixelWidth" />
          <v:f eqn="prod @3 21600 pixelHeight" />
          <v:f eqn="sum @0 0 1" />
          <v:f eqn="prod @6 1 2" />
          <v:f eqn="prod @7 21600 pixelWidth" />
          <v:f eqn="sum @8 21600 0" />
          <v:f eqn="prod @7 21600 pixelHeight" />
          <v:f eqn="sum @10 21600 0" />
        </v:formulas>
        <v:path o:extrusionok="f" gradientshapeok="t" o:connecttype="rect" />
        <o:lock v:ext="edit" aspectratio="t" />
      </v:shapetype>
      <v:shape id="_x0000_i1027" type="#_x0000_t75"
style="width:1in;height:24pt" o:ole="">
        <v:imagedata r:id="rId4" o:title="" />
      </v:shape>
      <w:control r:id="rId5" w:name="CommandButton1" w:shapeid="_x0000_i1027"
/>
    </w:object>
  </w:r>
</w:p>
```

1 The relationship type for rld5 is: <http://schemas.openxmlformats.org/officeDocument/2006/relationships/control>

3 The XML content of the part referenced by rld5 could be:

```
4 <applet xlink:href="../../../Program%20Files/Application"
5 xlink:type="simple" xlink:show="embed" xlink:actuate="onLoad"
6 code="CalculateApplet.class" may-script="false"/>
```

7 *end example]*

8 [Example: The following example shows the persistence that could be used for an embedded control which is
9 an ActiveX control within a WordprocessingML document:

```
10 <w:p>
11 <w:r w:rsidR="005810E1">
12 <w:object w:dxaOrig="1440" w:dyaOrig="1440">
13 <v:shapetype id="_x0000_t75" coordsize="21600,21600" o:spt="75"
14 o:preferrelative="t" path="m@4@5l@4@11@9@11@9@5xe" filled="f" stroked="f">
15 <v:stroke joinstyle="miter" />
16 <v:formulas>
17 <v:f eqn="if lineDrawn pixelLineWidth 0" />
18 <v:f eqn="sum @0 1 0" />
19 <v:f eqn="sum 0 0 @1" />
20 <v:f eqn="prod @2 1 2" />
21 <v:f eqn="prod @3 21600 pixelWidth" />
22 <v:f eqn="prod @3 21600 pixelHeight" />
23 <v:f eqn="sum @0 0 1" />
24 <v:f eqn="prod @6 1 2" />
25 <v:f eqn="prod @7 21600 pixelWidth" />
26 <v:f eqn="sum @8 21600 0" />
27 <v:f eqn="prod @7 21600 pixelHeight" />
28 <v:f eqn="sum @10 21600 0" />
29 </v:formulas>
30 <v:path o:extrusionok="f" gradientshapeok="t" o:connecttype="rect" />
31 <o:lock v:ext="edit" aspectratio="t" />
32 </v:shapetype>
33 <v:shape id="_x0000_i1027" type="#_x0000_t75"
34 style="width:1in;height:24pt" o:ole="">
35 <v:imagedata r:id="rId4" o:title="" />
36 </v:shape>
37 <w:control r:id="rId5" w:name="CommandButton1" w:shapeid="_x0000_i1027"
38 />
39 </w:object>
40 </w:r>
```

1 </w:p>

2 The relationship type for rld5 is: [http://schemas.openxmlformats.org/officeDocument/](http://schemas.openxmlformats.org/officeDocument/2006/relationships/control)
3 [2006/relationships/control](http://schemas.openxmlformats.org/officeDocument/2006/relationships/control)

4 The content type of the part referenced by rld5 could be: [application/vnd.ms-office.activeX+xml](http://schemas.openxmlformats.org/officeDocument/2006/relationships/control)

5 The XML content of the part referenced by rld5 could be:

```
6 <ax:ocx ax:classid="{D7053240-CE69-11CD-A777-00DD01143C57}"
7   ax:persistence="persistPropertyBag"
8   xmlns:ax="http://schemas.microsoft.com/office/2006/activeX">
9   <ax:ocxPr ax:name="Caption" ax:value="CommandButton1" />
10  <ax:ocxPr ax:name="Size" ax:value="2540;847" />
11  <ax:ocxPr ax:name="FontName" ax:value="Calibri" />
12  <ax:ocxPr ax:name="FontHeight" ax:value="225" />
13  <ax:ocxPr ax:name="FontCharSet" ax:value="0" />
14  <ax:ocxPr ax:name="FontPitchAndFamily" ax:value="2" />
15  <ax:ocxPr ax:name="ParagraphAlign" ax:value="3" />
16 </ax:ocx>
```

17 *end example]*

18 An Embedded Control Persistence part shall be located within the package containing the source relationship
19 (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

20 An Embedded Control Persistence part shall not have any implicit or explicit relationships to other parts
21 defined by this Standard.

22 15.2.9 Embedded Object Part

Content Type:	application/vnd.openxmlformats-officedocument.oleObject
Root Namespace:	not applicable
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/oleObject

24 An instance of this part type can contain an embedded object produced by any embedded object server.

25 A package is permitted to contain zero or more Embedded Object parts, and each such part shall be the target
26 of an explicit relationship from a Comments (§11.3.2), Endnotes (§11.3.4), Footer (§11.3.6), Footnotes
27 (§11.3.7), Header (§11.3.9), or Main Document (§11.3.10) part in a WordprocessingML package; a Worksheet
28 part (§12.3.24) in a SpreadsheetML package; or a Handout Master (§13.3.3), Notes Slide (§13.3.5), Notes
29 Master (§13.3.4), Slide (§13.3.8), Slide Layout (§13.3.9), Slide Master (§13.3.10) part in a PresentationML
30 package.

1 A WordprocessingML document package is permitted to contain zero or more Embedded Object parts, each of
 2 which shall be the target of a relationship in a Main Document part-relationship item. Each Embedded Object
 3 part shall have an associated image, which appears in the document as a placeholder for the corresponding
 4 embedded object.

5 *[Example:* Consider the case in which a WordprocessingML document has embedded in it one video object and
 6 one audio object. The following Main Document part-relationship item contains relationships to two
 7 Embedded parts (one each for the video and audio), which are stored in the ZIP items
 8 embeddings/embeddedObjectN.bin:

```

9     <Relationships xmlns="...">
10       <Relationship Id="rId5"
11         Type="http://.../oleObject" Target="embeddings/embeddedObject1.bin"/>
12       <Relationship Id="rId7"
13         Type="http://.../oleObject" Target="embeddings/embeddedObject2.bin"/>
14       <Relationship Id="rId4"
15         Type="http://.../image" Target="media/image1.png"/>
16       <Relationship Id="rId6"
17         Type="http://.../image" Target="media/image2.png"/>
18     </Relationships>

```

19 *example]*

20 A SpreadsheetML document package is permitted to contain zero or more Embedded Object parts, each of
 21 which shall be the target of a relationship in a Worksheet part-relationship item.

22 *[Example:* Consider the case in which a SpreadsheetML document has embedded in it one video object and
 23 one audio object on one worksheet, and another audio object embedded in another worksheet. The following
 24 Worksheet Document part-relationship item contains relationships to two Embedded Object parts (one each
 25 for the video and audio), which are stored in the ZIP items ../embeddings/embeddedObjectN.bin:

```

26     <Relationships xmlns="...">
27       <Relationship Id="rId2"
28         Type="http://.../oleObject" Target="../embeddings/embeddedObject1.bin"/>
29       <Relationship Id="rId3"
30         Type="http://.../oleObject" Target="../embeddings/embeddedObject2.bin"/>
31     </Relationships>

```

32 *end example]*

33 A PresentationML document package is permitted to contain zero or more Embedded Object parts, each of
 34 which shall be the target of a relationship in a Slide part-relationship item.

35 *[Example:* Consider the case in which a PresentationML document has embedded in it one video object and
 36 one audio object on one slide, and another audio object embedded on another slide. The following Slide part-

relationship item contains relationships to two Embedded Object parts (one each for the video and audio), which are stored in the ZIP items ../embeddings/embeddedObjectN.bin:

```
<Relationships xmlns="...">
  <Relationship Id="rId6"
    Type="http://.../oleObject"
    Target="../embeddings/embeddedObject1.bin"/>
  <Relationship Id="rId7"
    Type="http://.../oleObject"
    Target="../embeddings/embeddedObject2.bin"/>
</Relationships>
```

end example]

An Embedded Object part may be located within or external to the package containing the source relationship (expressed syntactically, the TargetMode attribute of the Relationship element may be Internal or External).

An Embedded Object part is permitted to have an explicit relationship to the following parts defined by this Standard:

- Hyperlink (§15.3)

An Embedded Object part shall not have any implicit or explicit relationships to other parts defined by this Standard.

15.2.10 Embedded Package Part

Content Type:	application/vnd.openxmlformats-officedocument.package
Root Namespace:	not applicable
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/package

An instance of this part type contains a complete package. For example, a WordprocessingML document might contain a SpreadsheetML or PresentationML document, in which case, the WordprocessingML document package would contain an embedded package part that defined that SpreadsheetML or PresentationML document.

A package is permitted to contain zero or more Embedded Package parts, and each such part shall be the target of an explicit relationship from a Chart (§14.2.1), Comments (§11.3.2), Endnotes (§11.3.4), Footer (§11.3.6), Footnotes (§11.3.7), Header (§11.3.9), or Main Document (§11.3.10) part in a WordprocessingML package; a Chart (§14.2.1), or Worksheet part (§12.3.24) in a SpreadsheetML package; or a Chart (§14.2.1),

Handout Master (§13.3.3), Notes Slide (§13.3.5), Notes Master (§13.3.4), Slide (§13.3.8), Slide Layout (§13.3.9), Slide Master (§13.3.10) part in a PresentationML package.

[*Example:* The following Presentation part-relationship item contains relationships to two Embedded Package parts: one is a SpreadsheetML package, which is stored in the ZIP item embeddings/Worksheet1.xlsx, the other is a PresentationML package, which is stored in the ZIP item embeddings/Presentation2.pptx. The image files are used as document display placeholders if the consumer cannot handle the embedded package type:

```
<Relationships xmlns="...">
  <Relationship Id="rId4"
    Type="http://.../image" Target="media/image1.emf"/>
  <Relationship Id="rId5"
    Type="http:package" Target="embeddings/Worksheet1.xlsx"/>
  <Relationship Id="rId6"
    Type="http://.../image" Target="media/image2.emf"/>
  <Relationship Id="rId7"
    Type="http://.../package" Target="embeddings/Presentation2.pptx"/>
</Relationships>
```

end example]

An Embedded Package part may be located within or external to the package containing the source relationship (expressed syntactically, the TargetMode attribute of the Relationship element may be Internal or External).

An Embedded Package part is permitted to have an explicit relationship to the following parts defined by this Standard:

- Hyperlink (§15.3)

An Embedded Package part shall not have any implicit or explicit relationships to other parts defined by this Standard.

15.2.11 File Properties

There are three kinds of file properties: , *core*, *custom*, and *extended*. The *core file properties* of a package enable users to discover, get, and set common sets of properties from within that package, regardless of whether it's a WordprocessingML, SpreadsheetML, or PresentationML package. *Extended file properties* are specific to Office Open XML packages, while *custom file properties* are defined by the user, with each custom file property having a name, a value, and a type.

15.2.11.1 Core File Properties Part

Content Type:	application/vnd.openxmlformats-package.core-properties+xml
Root Namespace:	http://schemas.openxmlformats.org/package/2006/metadata/core-properties

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/metadata/core-properties
----------------------	---

1

2 Core file properties enable users to discover, get, and set common sets of properties within packages. (These
3 properties include creator name, creation date, title, and description.) These properties are stored using the
4 appropriate Dublin Core properties whenever possible.

5 A package shall contain at most one Core File Properties part, and that part shall be the target of a relationship
6 in the package-relationship item for the document.

7 *[Example:* The following PresentationML's package-relationship item contains one relationship, for the Core
8 File Properties part stored in the ZIP item core.xml):

```
9     <Relationships xmlns="...">
10       <Relationship Id="rId3"
11         Type="http://.../core-properties" Target="docProps/core.xml"/>
12     </Relationships>
```

13 *end example]*

14 The root element for a part of this content type shall be coreProperties.

15 *[Example:*

```
16     <cp:coreProperties xmlns:cp="..." xmlns:dc="..." >
17       <dc:title>Example File</dc:title>
18       <dc:creator>Tristan Davis</dc:creator>
19       <cp:lastModifiedBy>Tristan Davis</cp:lastModifiedBy>
20       <cp:revision>1</cp:revision>
21     </cp:coreProperties>
```

22 *end example]*

23 A Core File Properties part shall be located within the package containing the source relationship (expressed
24 syntactically, the TargetMode attribute of the Relationship element shall be Internal).

25 A Core File Properties part shall not have implicit or explicit relationships to other parts defined by this
26 Standard.

27 15.2.11.2 Custom File Properties Part

Content Type:	application/vnd.openxmlformats-officedocument.custom-properties+xml
Root Namespace:	http://schemas.openxmlformats.org/officeDocument/2006/custom-properties

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/custom-properties
----------------------	---

1

2 An instance of this part contains the names of custom file properties that apply to the package, their values,
3 and the types of those values. A custom file property might be the name of the client for whom the document
4 was prepared, a date/time on which some event happened, a document number, or some Boolean status flag.

5 A package shall contain at most one Custom File Properties part, and that part shall be the target of a
6 relationship in the package-relationship item for the document.

7 *[Example: The following PresentationML's package-relationship item contains a relationship to a Custom File*
8 *Properties part, stored in the ZIP item docProps/custom.xml:*

```
9     <Relationships xmlns="...">
10       <Relationship Id="rId4"
11         Type="http://.../custom-properties" Target="docProps/custom.xml"/>
12     </Relationships>
```

13 *end example]*

14 The root element for a part of this content type shall be Properties.

15 *[Example: Here's some content markup from a WordprocessingML document, which contains four custom*
16 *properties: Client, having a text value of "ACME Corp."; Document number, having a numeric value of*
17 *1543; Recorded date, having a date/time value of 2005-12-01; and Special processing needed,*
18 *having a Boolean value of false:*

```
19     <Properties ... xmlns:vt="...">
20       <property fmtid="{D5C...9AE}" pid="2" name="Client">
21         <vt:lpwstr>ACME Corp.</vt:lpwstr>
22       </property>
23       <property fmtid="{D5C...9AE}" pid="3" name="Document number">
24         <vt:i4>1543</vt:i4>
25       </property>
26       <property fmtid="{D5C...9AE}" pid="4" name="Recorded date">
27         <vt:filetime>2005-12-01T05:00:00Z</vt:filetime>
28       </property>
29       <property fmtid="{D5C...9AE}" pid="5" name="Special processing needed">
30         <vt:bool>>false</vt:bool>
31       </property>
32     </Properties>
```

33 *end example]*

1 A Custom File Properties part shall be located within the package containing the source relationship (expressed
2 syntactically, the TargetMode attribute of the Relationship element shall be Internal).

3 A Custom File Properties part shall not have implicit or explicit relationships to other parts defined by this
4 Standard.

5 15.2.11.3 Extended File Properties Part

Content Type:	application/vnd.openxmlformats-officedocument.extended-properties+xml
Root Namespace:	http://schemas.openxmlformats.org/officeDocument/2006/extended-properties
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/extended-properties

6
7 An instance of this part contains properties specific to an Office Open XML document. *[Example: A*
8 *PresentationML document specifies the number of slides in this presentation when last saved by a producer.*
9 *end example]*

10 A package shall contain at most one Extended File Properties part, and that part shall be the target of a
11 relationship in the package-relationship item for the document.

12 *[Example:*

13 <Relationships xmlns="...">
14 <Relationship Id="rId4"
15 Type="http://.../extended-properties" Target="docProps/app.xml"/>
16 </Relationships>

17 *end example]*

18 The root element for a part of this content type shall be Properties.

19 *[Example: Here's some content markup from a WordprocessingML document:*

20 <Properties ...>
21 <Template>Normal.dotm</Template>
22 <TotalTime>0</TotalTime>
23 <Pages>1</Pages>
24 <Words>3</Words>
25 <Characters>22</Characters>

```

1      <Application>Sample Producer</Application>
2      <DocSecurity>0</DocSecurity>
3      <Lines>1</Lines>
4      <Paragraphs>1</Paragraphs>
5      ...
6      <AppVersion>12.0000</AppVersion>
7  </Properties>

```

8 here's some content markup from a SpreadsheetML document:

```

9  <Properties ...>
10    <Application>Sample Producer</Application>
11    <HeadingPairs>
12    ...
13    </HeadingPairs>
14    <TitlesOfParts>
15    ...
16    </TitlesOfParts>
17    <Company>Consultant</Company>
18    ...
19  </Properties>

```

20 and here's some content markup from a PresentationML document:

```

21  <Properties ...>
22    <Template>ppt_template_sdwest05</Template>
23    <TotalTime>3166</TotalTime>
24    <Words>37</Words>
25    <Application>Sample Producer</Application>
26    <PresentationFormat>On-screen Show</PresentationFormat>
27    <Paragraphs>15</Paragraphs>
28    <Slides>2</Slides>
29    <Notes>2</Notes>
30    ...
31    <HeadingPairs>
32    ...
33    </HeadingPairs>
34    <TitlesOfParts>
35    ...
36    </TitlesOfParts>
37    ...
38  </Properties>

```

39 *end example]*

1 A Extended File Properties part shall be located within the package containing the source relationship
2 (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

3 An Extended File Properties part shall not have implicit or explicit relationships to any other part defined by
4 this Standard.

5 15.2.12 Font Part

Content Type:	application/x-fontdata application/x-font-ttf
Root Namespace:	not applicable
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/font

6

7 An instance of this part type contains a given font embedded directly into the document. (This is useful when
8 using custom fonts or fonts that are not widely distributed.)

9 Fonts stored in a Font part can be stored in one of two formats, identified by the associated content type:

- 10 • application/x-fontdata specifies that the font shall be stored as a bitmapped font (each glyph is
11 stored as a raster image)
- 12 • application/x-font-ttf specifies that the font shall be stored in the TrueType or OpenType format

13 A package shall contain zero or more Font parts, and for each that exists, that part shall be the target of an
14 explicit relationship in the Font Table (§11.3.5), or Presentation (§13.3.6) part.

15 A Font part shall be located within the package containing the source relationship (expressed syntactically, the
16 TargetMode attribute of the Relationship element shall be Internal).

17 A Font part shall not have implicit or explicit relationships to other parts defined by this Standard.

18 15.2.13 Image Part

Content Type:	Any supported image type.	
	[Note: Some example content types are:	
	image/gif	http://www.w3.org/Graphics/GIF/spec-gif89a.txt
	image/png	ISO/IEC 15948:2003 http://www.libpng.org/pub/png/spec/
	image/tiff	http://partners.adobe.com/public/developer/tiff/index.html#spec
	image/pict	http://developer.apple.com/documentation/mac/QuickDraw/QuickDraw-2.html
	image/jpeg	http://www.w3.org/Graphics/JPEG/
	end note.]	

Root Namespace:	Not applicable
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/image

1

2 An image can be stored in a package as a ZIP item. Image ZIP items shall be identified by an image part
3 relationship and the appropriate content type.

4 A package is permitted to contain zero or more Image parts, and each such part shall be the target of an
5 explicit relationship from a Comments (§11.3.2), Endnotes (§11.3.4), Footer (§11.3.6), Footnotes (§11.3.7),
6 Header (§11.3.9), or Main Document (§11.3.10) part in a WordprocessingML package; a VML Drawing
7 (§15.2.17) part in a SpreadsheetML package; or a Handout Master (§13.3.3), Notes Slide (§13.3.5), Notes
8 Master (§13.3.4), Slide (§13.3.8), Slide Layout (§13.3.9), Slide Master (§13.3.10), or a VML Drawing (§15.2.17)
9 part in a PresentationML package.

10 *[Example:* The following PresentationML's package-relationship item contains one relationship, for the slide
11 template jpeg image stored in the ZIP item ../media/image1.jpeg:

```
12     <Relationships xmlns="...">
13         <Relationship Id="rId8"
14             Type="http://.../image" Target="../media/image1.jpeg"/>
15     </Relationships>
```

16 *end example]*

17 An Image part may be located within or external to the package containing the source relationship (expressed
18 syntactically, the TargetMode attribute of the Relationship element may be Internal or External).

19 An Image part shall not have implicit or explicit relationships to other parts defined by this Standard.

20 15.2.14 Printer Settings Part

Content Type:	application/vnd.openxmlformats-officedocument.spreadsheetml.printerSettings (in SpreadsheetML documents) application/vnd.openxmlformats-officedocument.wordprocessingml.printerSettings (in WordprocessingML documents)
Root Namespace:	not applicable
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/printerSettings

21

22 An instance of this part type contains information about the initialization and environment of a printer or a
23 display device. The layout of this data structure is application-defined. *[Example:* An Office Open XML producer

on Windows might store the DEVMODE structure defined here:

http://msdn.microsoft.com/library/default.asp?url=/library/en-us/gdi/prntspol_8nle.asp, while an application on the Mac OS might choose to store the print record defined here:

http://developer.apple.com/documentation/Carbon/Reference/CarbonPrintingManager_Ref/Reference/reference.html. *end example]*

A SpreadsheetML package is permitted to contain at most one Printer Settings part per Chartsheet, Dialogsheet, or Worksheet part, and that part shall be the target of an implicit relationship from a Chartsheet (§12.3.2), Dialogsheet (§12.3.7), or Worksheet (§12.3.24) part. A WordprocessingML package is permitted to contain zero or more Printer Settings parts, one per sectPr element, each a target of an explicit relationship from a Main Document (§11.3.10) or Glossary Document (§11.3.8) part.

[*Example:* The following SpreadsheetML Worksheet part-relationship item contains a relationship to a Printer Settings part, which is stored in the ZIP item ../printerSettings/printerSettings1.bin:

```
<Relationships xmlns="...">
  <Relationship Id="rId4"
    Type="http://../printerSettings"
    Target="../printerSettings/printerSettings1.bin"/>
</Relationships>
```

end example]

A Printer Settings part shall be located within the package containing the source relationship (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

A Printer Settings part shall not have implicit or explicit relationships to any other part defined by this Standard.

15.2.15 Thumbnail Part

Content Type:	<p>Any supported image type.</p> <p>[<i>Note:</i> Some example content types are:</p> <table border="1"> <tr> <td>image/gif</td><td>http://www.w3.org/Graphics/GIF/spec-gif89a.txt</td></tr> <tr> <td>image/png</td><td>ISO/IEC 15948:2003 http://www.libpng.org/pub/png/spec/</td></tr> <tr> <td>image/tiff</td><td>http://partners.adobe.com/public/developer/tiff/index.html#spec</td></tr> <tr> <td>image/pict</td><td>http://developer.apple.com/documentation/mac/QuickDraw/QuickDraw-2.html</td></tr> <tr> <td>image/jpeg</td><td>http://www.w3.org/Graphics/JPEG/</td></tr> </table> <p><i>end note.]</i></p>	image/gif	http://www.w3.org/Graphics/GIF/spec-gif89a.txt	image/png	ISO/IEC 15948:2003 http://www.libpng.org/pub/png/spec/	image/tiff	http://partners.adobe.com/public/developer/tiff/index.html#spec	image/pict	http://developer.apple.com/documentation/mac/QuickDraw/QuickDraw-2.html	image/jpeg	http://www.w3.org/Graphics/JPEG/
image/gif	http://www.w3.org/Graphics/GIF/spec-gif89a.txt										
image/png	ISO/IEC 15948:2003 http://www.libpng.org/pub/png/spec/										
image/tiff	http://partners.adobe.com/public/developer/tiff/index.html#spec										
image/pict	http://developer.apple.com/documentation/mac/QuickDraw/QuickDraw-2.html										
image/jpeg	http://www.w3.org/Graphics/JPEG/										
Root Namespace:	Not applicable										

Source Relationship:	http://schemas.openxmlformats.org/package/2006/relationships/metadata/thumbnail
----------------------	---

1

2 To help end-users identify parts of a package or the package as a whole, images, called *thumbnails*, may be
3 stored in that package. Each thumbnail image is generated by the package producer and is stored in the
4 package as a ZIP item.

5 Thumbnail ZIP items shall be identified by either a package-relationship item or a part-relationship item.
6 Packages shall not contain more than one thumbnail relationship associated with the package as a whole, or
7 more than one thumbnail relationship per package part.

8 *[Example: The following PresentationML's package-relationship item contains one relationship, for the*
9 *metafile image stored in the ZIP item thumbnail.wmf:*

```
10 <Relationships xmlns="...">
11   <Relationship Id="rId2"
12     Type="http://.../thumbnail" Target="docProps/thumbnail.wmf"/>
13 </Relationships>
```

14 *end example]*

15 A Thumbnail part shall be located within the package containing the source relationship (expressed
16 syntactically, the TargetMode attribute of the Relationship element shall be Internal).

17 A Thumbnail part shall not have implicit or explicit relationships to other parts defined by this Standard.

18 15.2.16 Video Part

Content Type:	Any supported video type.	
	[Note: Some example content types are:	
	video/x-ms-asf	http://www.microsoft.com/windows/windowsmedia/forpros/format/asfspec.aspx
	video/avi	http://www.the-labs.com/Video/odmlff2-avidef.pdf
	video/mpg	ISO/IEC 13818
	video/mpeg	ISO/IEC 13818
	video/x-ms-wm	http://www.microsoft.com/windows/windowsmedia/forpros/format/asfspec.aspx
	video/quicktime	http://developer.apple.com/softwarelicensing/agreements/quicktime.html
	end note]	
Root Namespace:	not applicable	

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/video
----------------------	---

1

2 An instance of this part type contains a video file.

3 A PresentationML package is permitted to contain zero or more Video parts, each of which shall be the target
4 of an explicit relationship in a Handout Master (§13.3.3), Notes Slide (§13.3.5), Notes Master (§13.3.4), Slide
5 (§13.3.8), Slide Layout (§13.3.9), or Slide Master (§13.3.10) part. A WordprocessingML package is permitted to
6 contain zero or more Video parts, each of which shall be the target of an explicit relationship from a Comments
7 (§11.3.2), Endnotes (§11.3.4), Footer (§11.3.6), Footnotes (§11.3.7), Header (§11.3.9), or Main Document
8 (§11.3.10) part.

9 *[Example:* The following Slide part-relationship item contains a relationship to a Video part, which is stored as
10 the file E:\Video demo.avi:

```
11     <Relationships xmlns="...">
12         <Relationship Id="rId2"
13             Type="http://.../video"
14             Target="file:///E:\Video%20demo.avi" TargetMode="External"/>
15     </Relationships>
```

16 *end example]*

17 A Video part is not stored as XML; instead, it involves a relationship target that is a video clip.

18 A Video part may be located within or external to the package containing the source relationship (expressed
19 syntactically, the TargetMode attribute of the Relationship element may be Internal or External).

20 A Video part shall not have implicit or explicit relationships to other parts defined by this Standard.

21 15.2.17 VML Drawing Part

Content Type:	application/vnd.openxmlformats-officedocument.vmlDrawing
Root Namespace:	not applicable
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/vmlDrawing

22

23 An instance of this part type contains markup in the Vector Markup Language (VML) syntax, which is used to
24 provide an alternative image representation of objects stored in a SpreadsheetML or PresentationML
25 document.

26 *[Note:* The VML format is a legacy format originally introduced with Office 2000 and is included and fully
27 defined in this Standard for backwards compatibility reasons. The DrawingML format is a newer and richer

1 format created with the goal of eventually replacing any uses of VML in the Office Open XML formats. VML
 2 should be considered a deprecated format included in Office Open XML for legacy reasons only and new
 3 applications that need a file format for drawings are strongly encouraged to use preferentially DrawingML. *end*
 4 *note*]

5 A package is permitted to contain zero or more VML Drawing parts, each of which shall be the target of an
 6 explicit relationship in a Handout Master (§13.3.3), Notes Slide (§13.3.5), Notes Master (§13.3.4), Slide
 7 (§13.3.8), Slide Layout (§13.3.9), or Slide Master (§13.3.10) part in a PresentationML document; or a
 8 Worksheet part (§12.3.24) in a SpreadsheetML document.

9 *[Example:* The following SpreadsheetML's package-relationship item contains one relationship, for the VML
 10 Drawing part stored in the ZIP item ../drawings/drawing1.vml:

```
11 <Relationships xmlns="...">
12   <Relationship Id="rId8"
13     Type="http://.../vmlDrawing" Target="../drawings/drawing1.vml"/>
14 </Relationships>
```

15 *end example]*

16 The root element for a part of this content type shall be xml in the null namespace, encapsulating an arbitrary
 17 amount of VML markup as defined by this Standard.

18 *[Example:* Consider the following VML Drawing part:

```
19 <xml>
20   <v:shape ...>
21     ...
22   </v:shape>
23   ...
24 </xml>
```

25 *end example]*

26 A VML Drawing part shall be located within the package containing the source relationship (expressed
 27 syntactically, the TargetMode attribute of the Relationship element shall be Internal).

28 A VML Drawing part is permitted to have explicit relationships to the following parts defined by this Standard:

- 29 • Image (§15.2.13)

30 A VML Drawing part shall not have implicit or explicit relationships to any other part defined by this Standard.

31 15.3 Hyperlinks

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/hyperlink
----------------------	---

1

2 A hyperlink can be stored in a package as a relationship. Hyperlinks shall be identified by containing a target
3 which specifies the destination of the given hyperlink.

4 *[Example:* The following WordprocessingML Footnote part's relationship part contains one relationship, for the
5 hyperlink <http://schemas.openxmlformats.org/wordprocessingml/>:

```
6     <Relationships xmlns="...">
7         <Relationship Id="rId1"
8             Type="http://.../hyperlink"
9             Target="http://schemas.openxmlformats.org/wordprocessingml/"
10            TargetMode="External"/>
11     </Relationships>
```

12 *end example]*

13 A hyperlink target may be located within or external to the package containing the source relationship
14 (expressed syntactically, the TargetMode attribute of the Relationship element may be Internal or
15 External).

Annex A. Bibliography

This clause is informative.

Character Sets from IANA, as specified at <http://www.iana.org/assignments/character-sets>

PANOSE Classification Guide, Version 1.2, Hewlett Packard Co., 1992.

RFC 2119, Bradner, Scott, 1997: “Key words for use in RFCs to Indicate Requirement Levels.”
<http://www.ietf.org/rfc/rfc2119.txt>.

RFC 2045, Borenstein, N., and N. Freed. “Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies.” The Internet Society. 1996. <http://www.rfc-editor.org>

RFC 2616, Berners-Lee, T., R. Fielding, H. Frystyk, J. Gettys, P. Leach, L. Masinter, and J. Mogul. “Hypertext Transfer Protocol—HTTP/1.1.” The Internet Society. 1999. <http://www.rfc-editor.org>

RFC 3066, Alvestrand, H. “Tags for the Identification of Languages.” The Internet Society. 2001.
<http://www.rfc-editor.org>

RFC 3339, Klyne, G. and C. Newman “Date and Time on the Internet: Timestamps.” The Internet Society. 2002.
<http://www.rfc-editor.org>

RFC 3629, Yergeau, F. “UTF-8, a transformation format of ISO 10646.” The Internet Society. 2003.
<http://www.rfc-editor.org>

RFC 3986, Berners-Lee, T., R. Fielding, and L. Masinter. “Uniform Resource Identifier (URI): Generic Syntax.” The Internet Society. 2005. <http://www.rfc-editor.org>

The Unicode Consortium. The Unicode Standard, Version 4.0, defined by: The Unicode Standard, Version 4.0 (Reading, MA, Addison-Wesley, 2003. ISBN 0-321-18578-1).

XML, Bray, Tim, Eve Maler, Jean Paoli, C. M. Sperberg-McQueen, and François Yergeau (editors). “Extensible Markup Language (XML) 1.0,” Third Edition. World Wide Web Consortium. 2004.
<http://www.w3.org/TR/2004/REC-xml-20040204/xml11-20040204/>

XML Base, Marsh, Jonathan. “XML Base.” World Wide Web Consortium. 2001.
<http://www.w3.org/TR/2001/REC-xmlbase-20010627/>

XML Namespaces, Bray, Tim, Dave Hollander, Andrew Layman, and Richard Tobin (editors). “Namespaces in XML 1.1.” World Wide Web Consortium. 2004. <http://www.w3.org/TR/2004/REC-xml-names11-20040204/>

XML Path Language Specification, Version 1.0, W3C Recommendation 16 November 1999
<http://www.w3.org/TR/xpath>

- 1 XML Schema Part 0: Primer Second Edition, W3C Recommendation 28 October 2004
- 2 <http://www.w3.org/TR/xmlschema-0/>
- 3 XML Schema Part 1: Structures Second Edition, W3C Recommendation 28 October 2004
- 4 <http://www.w3.org/TR/xmlschema-1/>
- 5 XML Schema Part 2: Datatypes Second Edition, W3C Recommendation 28 October 2004
- 6 <http://www.w3.org/TR/xmlschema-2/>
- 7 .ZIP File Format Specification from PKWARE, Inc., as specified in *appnote*, the Application Note on the Zip file
- 8 format, at <http://www.pkware.com>.
- 9 **End of informative text.**

1 Annex B. Index

2 This annex is informative.

3	4	Additional Characteristics part.....	137	44	consumer	11
5	Alternative Format Import part	28	45	control.....	136	
6	animation	13	46	Core File Properties part.....	149	
7	annex.....	10	47	Custom File Properties part.....	150	
8	application.....	6	48	custom markup.....	12	
9	Application-Defined File Properties part	152	49	Custom Property part	68	
10	audience	13	50	Custom XML data property	15	
11	audio.....	13	51	Custom XML Data Storage part	140	
12	Audio part.....	138	52	Custom XML Data Storage Properties part.....	141	
13	behavior	6	53	Custom XML Mappings part	69	
14	implementation-defined	6	54	diagram	122	
15	locale-specific	6	55	diagram color.....	14	
16	unspecified	6	56	Diagram Colors part.....	126	
17	Bibliography part.....	139	57	diagram data.....	14	
18	body.....	11	58	Diagram Data part	127	
19	Calculation Chain part	63	59	diagram layout.....	14	
20	cell	12, 14, 59	60	Diagram Layout Definition part	128	
21	alignment	13	61	diagram style	14	
22	border.....	13	62	Diagram Style part	130	
23	color	13	63	Dialogsheet part	70	
24	font.....	13	64	Digital Signature Origin part	141	
25	formatting	13	65	Digital Signature XML Signature part	142	
26	cell reference.....	13, 59	66	document setting.....	24	
27	chart	14, 59	67	Document Settings part.....	31	
28	Chart Drawing part.....	125	68	document template	53	
29	Chart part	123	69	document type	6	
30	Chartsheet part	64	70	DrawingML.....	6, 14	
31	color scheme	14, 132, 133	71	effect... 14, See format scheme, See format scheme		
32	column.....	12, 14, 59	72	Embedded Control Persistence part.....	143	
33	comment	13, 24, 59, 98	73	Embedded Object part	146	
34	Comment Authors part	102	74	Embedded Package part.....	148	
35	Comments part		75	Endnotes part	33	
36	PresentationML.....	103	76	example	10	
37	SpreadsheetML	65	77	extension	6	
38	WordprocessingML	29	78	External Workbook.....	96	
39	conformance	2	79	External Workbook References part	73	
40	application.....	3	80	field	12	
41	document	3	81	field instruction.....	12	
42	connection.....	59	82	field result.....	12	
43	Connections part.....	67	83	file property	15, 149	

1	application-defined . See Application-Defined File	49	valid	7
2	Properties part	50	package	7, 11
3	core15, 149 , See Core File Properties part	51	embedded.....	7
4	custom..... 15, 149 , See Custom Properties part	52	package-relationship ZIP item	11
5	extended	53	paragraph	11
6	font.....14	54	part	11
7	Font part..... 154	55	Additional Characteristics.....See Additional	
8	font scheme.....132, 133	56	Characteristics part	
9	Font Table part..... 35	57	Alternative Format Import.....See Alternative	
10	footer.....12	58	Format Import part	
11	Footer part	59	Application-Defined File Properties..... See	
12	Footnotes part..... 38	60	Application-Defined File Properties part	
13	format scheme	61	Audio..... See Audio part	
14	formula.....13, 59	62	Bibliography.....See Bibliography part	
15	frame.....54	63	Calculation ChainSee Calculation Chain part	
16	frameset..... 54	64	Chart	See Chart part
17	function	65	Chart Drawing.....See Chart Drawing part	
18	glossary document	66	Chartsheet	See Chartsheet part
19	Glossary Document part	67	Comment Authors See Comment Authors part	
20	handout.....13, 98	68	Comments	
21	handout master.....13	69	PresentationML	See Comments part,
22	Handout Master part	70	PresentationML	
23	header	71	SpreadsheetML..... See Comments part,	
24	Header part	72	SpreadsheetML	
25	HTML Publish Location..... 119	73	WordprocessingML..... See Comments part,	
26	hyperlink.....12, 13, 159	74	WordprocessingML	
27	IEC.. See International Electrotechnical Commission	75	Connections	See Connections part
28	image.....13, 154	76	Core File Properties See Core File Properties part	
29	informative text..... 2	77	Custom File Properties	See Custom File
30	International Electrotechnical Commission	78	Properties part	
31	International Organization for Standardization.....9	79	Custom Property..... See Custom Property part	
32	ISOSee International Organization for	80	Custom XML Data Storage.See Custom XML Data	
33	Standardization	81	Storage part	
34	ISO/IEC 10646.....5	82	Custom XML Data Storage PropertiesSee Custom	
35	layout	83	XML Data Storage Properties part	
36	mail merge data source..... 56	84	Custom XML Mappings..... See Custom XML	
37	mail merge header data source	85	Mappings part	
38	Main Document part..... 46	86	Diagram Colors	See Diagram Colors part
39	master document..... 24, 55	87	Diagram Data	See Diagram Data part
40	Metadata part	88	Diagram Layout Definition.... See Diagram Layout	
41	normative text..... 2, 10	89	Definition part	
42	note	90	Diagram Style..... See Diagram Style part	
43	notes master	91	Dialogsheet	See Dialogsheet part
44	Notes Master part..... 106	92	Digital Signature Origin..... See Digital Signature	
45	Notes Slide part..... 107	93	Origin part	
46	numbering.....12	94	Digital Signature XML Signature	See Digital
47	Numbering Definitions part	95	Signature XML Signature part	
48	Office Open XML	96	Document Settings .. See Document Settings part	

1	Embedded Control Persistence.... See Embedded	49	ThemeSee Theme part
2	Control Persistence part	50	Theme OverrideSee Theme Override part
3	Embedded Object..... See Embedded Object part	51	User Defined Tags.....See User Defined Tags part
4	Embedded Package .See Embedded Package part	52	VideoSee Video part
5	Endnotes See Endnotes part	53	View Properties See View Properties part
6	External Workbook References.....See External	54	Volatile DependenciesSee Volatile Dependencies
7	Workbook References part	55	part
8	Font See Font part	56	Web Settings..... See Web Settings part
9	Font Table..... See Font Table part	57	Workbook See Workbook part
10	FooterSee Footer part	58	Worksheet See Worksheet part
11	Footnotes See Footnotes part	59	part-relationship item..... 11
12	Glossary Document .See Glossary Document part	60	picture..... 14
13	Handout MasterSee Handout Master part	61	pivot table..... 59
14	HeaderSee Header part	62	Pivot Table Cache Definition part..... 79
15	Main Document See Main Document part	63	Pivot Table Cache Records part 81
16	Metadata..... See Metadata part	64	Pivot Table part..... 78
17	Notes Master..... See Notes Master part	65	presentation 13, 98
18	Notes Slide See Notes Slide part	66	custom 13
19	Numbering DefinitionsSee Numbering	67	Presentation part..... 109
20	Definitions part	68	Presentation Properties part 111
21	Pivot Table..... See Pivot Table part	69	PresentationML 7 , 13
22	Pivot Table Cache Definition See Pivot Table	70	Printer Settings part..... 155
23	Cache Definition part	71	producer 11
24	Pivot Table Cache Records See Pivot Table Cache	72	property..... 11
25	Records part	73	Query Table part..... 82
26	PresentationSee Presentation part	74	rationale..... 10
27	Presentation Properties See Presentation	75	reference 10
28	Properties part	76	relationship..... 7 , 11
29	Printer Settings..... See Printer Settings part	77	explicit..... 7 , 18
30	Query TableSee Query Table part	78	implicit 7 , 18
31	Shared String Table .See Shared String Table part	79	relationships part..... 7
32	Shared Workbook Revision Headers...See Shared	80	row 12, 14, 59
33	Workbook Revision Headers part	81	run..... 11
34	Shared Workbook Revision Log.....See Shared	82	section 12, 24
35	Workbook Revision Log part	83	shape 14
36	Shared Workbook User Data.....See Shared	84	Shared String Table part 83
37	Workbook User Data part	85	Shared Workbook Revision Headers part..... 84
38	Single Cell Table Definitions .See Single Cell Table	86	Shared Workbook Revision Log part 85
39	Definitions part	87	Shared Workbook User Data part 87
40	SlideSee Slide part	88	Single Cell Table Definitions part..... 87
41	Slide Layout See Slide Layout part	89	slide..... 13, 98
42	Slide Master See Slide Master part	90	slide layout..... 98
43	Slide Synchronization Data..... See Slide	91	Slide Layout part 113
44	Synchronization Data part	92	slide master 13
45	Style Definitions See Style Definitions part	93	Slide Master part 115
46	Styles See Styles part	94	Slide part..... 111
47	Table DefinitionSee Table Definition part	95	Slide Synchronization Data part 116
48	Table Styles See Table Styles part	96	Slide Synchronization Server Location..... 120

1	SpreadsheetML	7, 12	19	unknown parts.....	17
2	style	12	20	unknown relationships	17
3	Style Definitions part.....	51	21	User Defined Tags part	117
4	Styles part.....	89	22	video	13
5	subdocument	24, 55	23	Video part	157
6	supplementary document storage location.....	24	24	View Properties part.....	118
7	table	12, 13, 14, 60	25	VML.....	15
8	Table Definition part	90	26	Volatile Dependencies part	91
9	Table Styles part.....	134	27	W3C	See World Wide Web Consortium
10	template	24	28	Web Settings part	52
11	text	11	29	WordprocessingML.....	7, 11
12	theme	14, 132, 133	30	workbook.....	12, 60
13	Theme Override part.....	133	31	external.....	96
14	Theme part.....	131	32	Workbook part.....	92
15	thumbnail.....	156	33	worksheet	12, 60
16	transition	13	34	Worksheet part.....	94
17	trash items	17	35	World Wide Web Consortium	9
18	Unicode standard	161	36	XSL transformation	58