

15. SpreadsheetML Reference Material

15.1 General

[Note: For further information on the mapping of elements and attributes to OPC parts, see the Bibliography entry, “Information on elements, attributes, and OPC parts in ISO/IEC 29500 (OOXML)” . *end note*]

15.2 Table of Contents

This subclause is informative.

15.3	Workbook	208
15.3.1	Additional attribute for fileSharing element (Part 1, §18.2.12)	208
15.3.2	Additional attribute for webPublishing element (Part 1, §18.2.24)	208
15.3.3	Additional attributes for workbookProtection element (Part 1, §18.2.29)	209
15.3.4	Modified content for Date Conversion for Serial Date-Times (Part 1, §18.17.4.1)	215
15.4	Worksheets	216
15.4.1	Worksheets	216
15.4.1.1	legacyDrawing (Legacy Drawing Reference)	216
15.4.1.2	legacyDrawingHF (Legacy Drawing Reference in Header Footer)	216
15.4.1.3	Additional attribute for dataConsolidate element (Part 1, §18.3.1.29)	217
15.4.1.4	Additional attributes for protectedRange element (Part 1, §18.3.1.71)	217
15.4.1.5	Additional attribute for sheetProtection element (Part 1, §18.3.1.84)	218
15.4.1.6	Additional attribute for sheetProtection element (Part 1, §18.3.1.85)	218
15.4.2	AutoFilter Settings	218
15.4.2.1	Attributes with modified descriptions for dynamicFilter element (Part 1, §18.3.2.5)	218
15.5	Styles	219
15.5.1	left (Leading Edge Border)	219
15.5.2	right (Trailing Edge Border)	220
15.6	Pivot Tables	220
15.6.1	Pivot Tables	220
15.6.1.1	Additional attribute for pivotCacheDefinition element (Part 1, §18.10.1.67)	220
15.7	External Data Connections	220
15.7.1	Additional attribute for textPr element (Part 1, §18.13.12)	220
15.8	Simple Types	221
15.8.1	Additional enumeration values for ST_PivotAreaType (Part 1, §18.18.58)	221
15.8.2	ST_UnsignedShortHex (Unsigned Short Hex)	221
15.8.3	Removed enumeration values for ST_CellType (Part 1, §18.18.11)	221
15.9	Formulas	221
15.9.1	Attribute synonym for c element (Part 1, §18.6.1)	221
15.9.2	Additional representation for dates and times (Part 1, §18.17.4)	222

15.10 Changed attributes..... 222

15.10.1 Changed attribute for externalReference element (Part 1, §18.2.8)..... 222

15.10.2 Changed attribute for pivotCache element (Part 1, §18.2.17)..... 222

15.10.3 Changed attribute for sheet element (Part 1, §18.2.19)..... 222

15.10.4 Changed attribute for control element (Part 1, §18.3.1.19)..... 223

15.10.5 Changed attribute for controlPr element (Part 1, §18.3.1.20)..... 223

15.10.6 Changed attribute for customPr element (Part 1, §18.3.1.22)..... 223

15.10.7 Changed attribute for dataRef element (Part 1, §18.3.1.30)..... 224

15.10.8 Changed attribute for drawing element (Part 1, §18.3.1.36)..... 224

15.10.9 Changed attribute for drawingHF element (Part 1, §18.3.1.37)..... 224

15.10.10 Changed attribute for hyperlink element (Part 1, §18.3.1.47)..... 224

15.10.11 Changed attribute for objectPr element (Part 1, §18.3.1.56)..... 225

15.10.12 Changed attribute for oleObject element (Part 1, §18.3.1.59)..... 225

15.10.13 Changed attribute for pageSetup element (Part 1, §18.3.1.63)..... 225

15.10.14 Changed attribute for pageSetup element (Part 1, §18.3.1.64)..... 225

15.10.15 Changed attribute for picture element (Part 1, §18.3.1.67)..... 226

15.10.16 Changed attribute for pivotSelection element (Part 1, §18.3.1.69)..... 226

15.10.17 Changed attribute for tablePart element (Part 1, §18.3.1.94)..... 226

15.10.18 Changed attribute for pivotCacheDefinition element (Part 1, §18.10.1.67)..... 226

15.10.19 Changed attribute for rangeSet element (Part 1, §18.10.1.79)..... 226

15.10.20 Changed attribute for worksheetSource element (Part 1, §18.10.1.95)..... 227

15.10.21 Changed attribute for header element (Part 1, §18.11.1.1)..... 227

15.10.22 Changed attribute for externalBook element (Part 1, §18.14.7)..... 227

15.10.23 Changed attribute for oleLink element (Part 1, §18.14.11)..... 227

End of informative text.

15.3 Workbook

15.3.1 Additional attribute for fileSharing element (Part 1, §18.2.12)

The following additional attributes can be specified for a document of a transitional conformance class:

Attributes	Description
reservationPassword (Write Reservation Password)	<p>Specifies the legacy hash of the password required for editing this workbook.</p> <p>The hash is generated using the logic defined in the revisionsPassword attribute of the workbookProtection element (Part 1, §18.2.29).</p> <p>The possible values for this attribute are defined by the ST_UnsignedShortHex simple type (§15.8.2).</p>

15.3.2 Additional attribute for webPublishing element (Part 1, §18.2.24)

The following additional attributes can be specified for a document of a transitional conformance class:

Attributes	Description
codePage (Code Page)	<p>This attribute is used only for compatibility with the existing corpus of binary documents, and is ignored if the characterSet attribute is present. Specifies the encoding the application uses when a Web page is saved. A code page is a table that relates the binary character codes used by a program to keys on the keyboard or to the appearance of characters on the display. Code pages are a means of providing support for the languages used in different countries.</p> <p>[Note: There are a number of code page technologies. One example of potential values can be found at: http://www.unicode.org/Public/MAPPINGS/ end note]</p> <p>The default value for this attribute is the workbook's encoding.</p> <p>The possible values for this attribute are defined by the W3C XML Schema unsignedInt datatype.</p>

15.3.3 Additional attributes for workbookProtection element (Part 1, §18.2.29)

The following additional attributes can be specified for a document of a transitional conformance class:

Attributes	Description
revisionsPassword (Legacy Revisions Password)	<p>Specifies the legacy hash of the password required for unlocking revisions in this workbook. The hash is generated from an 8-bit wide character. The input string shall be in UTF-16LE format (if there is a leading BOM character (U+FEFF) in the encoded password it is removed before hash calculation), and these 16-bit Unicode characters shall be converted down to 8 bits before the hash is computed, using the following logic:</p> <p>[Note: This legacy conversion attempts to fit UTF-16 encoded characters into a single-byte character set. As such, if the input string uses characters from multiple character sets, many characters are unmapped in the destination character set and take on the default value, 0x3F. For this reason, it is recommended that applications choose a character set which maps the maximum number of characters from the input string and explicitly declare the character set used in the revisionsCharacterSet attribute. Not doing so will inhibit interoperability. end note]</p> <p>For SpreadsheetML password hash purposes, Unicode UTF-16 input code points are converted to a single or double byte character set.</p> <p>Code points with no representation in the target character set are replaced with Unicode character 0x3f (?).</p> <p>The values permitted by this attribute are names and aliases listed in the IANA character set listing found at http://www.iana.org/assignments/character-sets. For single byte character sets, each Unicode code point is replaced by a single byte or 0x3f if an appropriate character doesn't exist in the character set.</p> <p>For double byte character sets, each Unicode code point is replaced by either a single</p>

Attributes	Description
	<p>byte, or a two byte sequence, depending on the input character, or 0x3f if an appropriate character doesn't exist in the character set. In our tables the target is a single byte sequence if the most significant byte is 0x00, otherwise it is a double byte sequence, with the lead byte being the most significant byte.</p> <p>To convert, first check if conversion is being done to a single or double byte code page and load the appropriate WCTABLE code page table.</p> <p>For each input character, look up the code point in the WCTABLE. There are 3 possibilities: Not found, single byte, or double byte.</p> <ul style="list-style-type: none"> • If the input character is not found, append 0x3f and continue to the next character. • If the result is a single byte, check to make sure the entry in the MBTABLE matches the input. If it matches, append the single byte to the output. If it does not match, append 0x3f to the output. • If the result is a double byte, check to make sure the entry in the DBCSEENTRY table for the appropriate lead byte matches the input character. If it matches, append the lead byte and trail byte to the output. If it does not match, append 0x3f to the output. <p>The following pseudocode describes how this conversion should be done:</p> <pre> int WideCharToMultiByte(wchar_t* wszInput, byte* szOutput) { // Remember output start so we can return length byte* szOutputStart = szOutput; // Load Character Set Tables and determine // double/single byte nature. // This will depend on how the character sets are // represented on // the target machine. TABLECLASS represents some abstract // representation of this structure here. TABLECLASS pTables = LoadCharacterSetTables(); Bool bDoubleByte = IsCharacterSetDoubleByte(); while (*wszInput != 0) { if (bDoubleByte) szOutput = AppendDoubleByte(pTables, *wszInput, szOutput); else szOutput = AppendSingleByte(pTables, *wszInput, szOutput); } </pre>

Attributes	Description
	<pre> // Read next input wchar_t wszInput++; } // Null terminate the output *szOutput = 0; // Return output length return szOutput - szOutputStart; } byte* AppendSingleByte(TABLECLASS pTables, wchar_t wcIn, byte* szOutput) { // Look up byte that we want to append. byte bOut = pTables->LookUpSingleByte(wcIn); // Make sure that bOut matches the input, otherwise use ? // (ie: no best fit behavior allowed) if (wcIn != pTables->LookUpWideChar(bOut)) bOut = 0x3f; *szOutput = bOut; szOutput++; return szOutput; } byte* AppendDoubleByte(TABLECLASS pTables, wchar_t wcIn, byte* szOutput) { // Look up bytes that we want to append. UINT16 bytesOut = pTables->LookUpDoubleByte(wcIn); // See if it is a single or double byte sequence if (bytesOut & 0xFF00) { // It is a double byte sequence // Make sure that bytesOut matches the input, otherwise use ? // (ie: no best fit behavior allowed) if (wcIn != pTables->LookUpWideChar(bytesOut)) { // Use ?, it will be added below bytesOut = 0x003f; } else { </pre>

Attributes	Description
	<pre> // It matched, use the lead byte we found // trail byte will be added below *szOutput = bytesOut >> 8; szOutput++; } else { // It is a single byte sequence // Make sure that bytesOut matches the input, otherwise use ? // (ie: no best fit behavior allowed) if (wcIn != pTables->LookUpWideChar(bytesOut & 0xFF)) bytesOut = 0x003f; } // Add the single or trail byte *szOutput = bytesOut & 0xFF; szOutput++; return szOutput; } class pTables { // Construction depends on how you choose to store & load the // table files byte LookUpSingleByte(wchar_t wcIn) { // How you access the table depends on your storage mechanism. // Look up the line in WCTABLE where the first column matches wcIn, // and then return the byte value from the second column. if (exists WCTABLE{wcIn}) return WCTABLE{wcIn}.SecondColumn; // If it doesn't exist, return ? return 0x3f; } UINT16 LookUpDoubleByte(wchar_t wcIn) { // How you access the table depends on your storage mechanism. // Look up the line in WCTABLE where the first column </pre>

Attributes	Description
	<pre> matches wcIn, // and then return the double byte value from the second column. if (exists WCTABLE{wcIn}) return WCTABLE{wcIn}.SecondColumn; // If it doesn't exist, return ? return 0x003f; } // Overload that looks up wide chars from single byte code points. wchar_t LookUpWideChar(byte bIn) { // How you access the table depends on your storage mechanism. // Look up the line in MBTABLE where the first column matches bIn, // and then return the wchar_t value from the second column. if (exists MBTABLE{bIn}) return MBTABLE{bIn}.SecondColumn; // If it doesn't exist, return ? return 0x003f; } // Overload that looks up wide chars from double byte code points wchar_t LookUpWideChar(UINT16 bytesIn) { // How you access the table depends on your storage mechanism. // First find the DBCSTABLE where the LeadByte matches // the lead (most significant) input byte. if (exists DBCSTABLE{bytesIn >> 8}) { DbcTable = DBCSTABLE{bytesIn >> 8}; // Look up the line in DbcTable where the first column // matches the input trail (least significant) byte, // and then return the wchar_t value from the second column. if (exists DbcTable{bytesIn & 0xFF}) return DbcTable{bytesIn & 0xFF}.SecondColumn; } </pre>

Attributes	Description
	<pre> // Either the lead byte table or specific trail byte // doesn't exist in the table, return ? return 0x003f; } } </pre> <p>The resulting value is hashed using the low-order word algorithm defined in §14.8.1. This step assumes that all words are unsigned, the word size is two bytes, and that bit-level shift-left/shift-right operations shift in the direction of the highest-order and lowest-order bit, respectively. [Example: 0x61 SHR 1 is 0xC2, as 01100001 shifted one position in the direction of its highest-order bit is 11000010. end example]</p> <p>[Example: This algorithm can be represented by the following pseudocode:</p> <pre> // Function Input: // szPassword: NULL terminated C-Style string // cchPassword: The number of characters in szPassword (not // including the NULL terminator) unsigned_short GetPasswordHash(const char *szPassword, int cchPassword) { unsigned_short wPasswordHash; const char *pch; wPasswordHash = 0; if (cchPassword > 0) { pch = &szPassword[cchPassword]; while (pch-- != szPassword) { wPasswordHash = ((wPasswordHash >> 14) & 0x01) ((wPasswordHash << 1) & 0x7fff); wPasswordHash ^= *pch; } wPasswordHash = ((wPasswordHash >> 14) & 0x01) ((wPasswordHash << 1) & 0x7fff); wPasswordHash ^= cchPassword; wPasswordHash ^= (0x8000 ('N' << 8) 'K'); } return(wPasswordHash); } end example] </pre> <p>The possible values for this attribute are defined by the ST_UnsignedShortHex simple type (§15.8.2).</p>

Attributes	Description
revisionsPasswordCharacterSet (Revisions Password Character Set)	<p>Name of the character set associated with the legacy revisionsPassword hash. The values permitted by this attribute are names and aliases listed in the IANA CHARACTER SETS listing found at http://www.iana.org/assignments/character-sets.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
workbookPassword (Legacy Workbook Password)	<p>Specifies the legacy hash of the password required for unlocking revisions in this workbook.</p> <p>The hash is generated using the logic defined in the preceding revisionsPassword attribute.</p> <p>The possible values for this attribute are defined by the ST_UnsignedShortHex simple type (§15.8.2).</p>
workbookPasswordCharacterSet (Workbook Password Character Set)	<p>Name of the character set associated with the workbookPassword hash. The values permitted by this attribute are the names and aliases listed in the IANA CHARACTER SETS listing found at http://www.iana.org/assignments/character-sets.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

15.3.4 Modified content for Date Conversion for Serial Date-Times (Part 1, §18.17.4.1)

When interpreting a document of a transitional conformance class, Part 1, §18.17.4.1 is replaced by the following text:

A serial date-time is a number that represents a date and time. This signed value is in units of days relative to the base date for the selected date system. Serial date-times increase by 1 into each successive day, and decrease by 1 into each preceding day. Fractional portions of serial date-times represent fractions of a single day. [Example: When using the 1900 date system, which has a base date of 30th December 1899, a serial date-time of 1.5 represents midday on the 31st December 1899 (serial date-time day 1); that is, 1899-12-31T12:00. A serial date-time of -4.25 represents 6 pm on the 25th December 1899; that is, 1899-12-25T18:00. end example] The base dates and the related serial date-times represent local date and time.

Two different bases are used for converting dates to and from serial date-times:

- In the *1900 date system*, the lower limit is January 1, 1900, 00:00:00, which has a serial date-time of 1. The upper limit is December 31, 9999, 23:59:59, which has a serial date-time of 2,958,465.9999884. The base date for this date base system is December 31, 1899, which has a serial date-time of 0.
- In the *1904 date system*, the lower limit is January 1st, 0001, 00:00:00, which has a serial date-time of -695055. The upper limit is December 31st, 9999, 23:59:59.999, which has a serial date-time of

2,957,003.9999884. The base date for this system is midnight (00:00:00) on the morning of January 1st, 1904, which has a serial date-time of 0.

A serial date-time outside the temporal range for the selected date system is invalid.

The date system is specified by the value of the date1904 attribute of the workbookPr element. [Example:

1900 date system: <workbookPr showObjects="all"/>

1904 date system: <workbookPr date1904="1" showObjects="all"/>

end example]

15.4 Worksheets

15.4.1 Worksheets

15.4.1.1 legacyDrawing (Legacy Drawing Reference)

This element is present when the sheet contains drawing shapes defined by VML. In this case, the element contains an explicit relationship whose ID points to the part containing the VML definitions.

[Example:

<drawing r:id="rId1"/>

end example]

Attributes	Description
id (Relationship Id)	This value references a relationship Id for the sheet. The relationship shall point to the part containing the VML definition.
Namespace: .../officeDocument /2006/relationships	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

[Note: The W3C XML Schema definition of this element's content model ([CT_LegacyDrawing](#)) is located in §A.3.
end note]

15.4.1.2 legacyDrawingHF (Legacy Drawing Reference in Header Footer)

This element specifies the explicit relationship to the part containing the VML defining pictures rendered in the header / footer of the sheet.

Attributes	Description
id (Relationship Id)	This value references a relationship Id for the sheet. The relationship shall point to the part containing the VML definition.
Namespace:	

Attributes	Description
.../officeDocument/2006/relationships	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

[Note: The W3C XML Schema definition of this element's content model ([CT_LegacyDrawing](#)) is located in §A.3.
end note]

15.4.1.3 Additional attribute for dataConsolidate element (Part 1, §18.3.1.29)

The following additional attributes can be specified for a document of a transitional conformance class:

Attributes	Description
leftLabels (Starting Column Labels)	Semantically equivalent to startLabels. The possible values for this attribute are defined by the W3C XML Schema boolean datatype.

15.4.1.4 Additional attributes for protectedRange element (Part 1, §18.3.1.71)

The following additional attributes can be specified for a document of a transitional conformance class:

Attributes	Description
password (Legacy Password)	Specifies the legacy hash of the password required for editing this range. The hash is generated using the logic defined in the revisionsPassword attribute of the workbookProtection element (Part 1, §18.2.29). The possible values for this attribute are defined by the ST_UnsignedShortHex simple type (§15.8.2).
securityDescriptor (Security Descriptor)	Optional setting to specify the relative security descriptor. The security descriptor defines user accounts who can edit this range without providing a password to access the range. The format of a securityDescriptor is application defined; however, it is recommended that the following format be used for interoperability between implementations: <ul style="list-style-type: none"> username@domain If multiple user accounts are specified in the securityDescriptor attribute, each account shall be delimited by parentheses. [Example: This example demonstrates two user accounts in the security descriptor attribute: <pre><protectedRanges> <protectedRange sqref="A1:C5" name="Range1" securityDescriptor="(user1@iso.org)(user2@iso.org)"/></pre>

Attributes	Description
	<p><code></protectedRanges></code></p> <p><i>end example]</i></p> <p>If an application is unable to resolve the meaning of the securityDescriptor, it shall treat the attribute as if it had been removed.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

15.4.1.5 Additional attribute for sheetProtection element (Part 1, §18.3.1.84)

The following additional attributes can be specified for a document of a transitional conformance class:

Attributes	Description
password (Password)	<p>Specifies the hash of the password required for editing this chart sheet.</p> <p>The hash is generated using the logic defined in the revisionPassword attribute of the workbookProtection element (Part 1, §18.2.29).</p> <p>The possible values for this attribute are defined by the ST_UnsignedShortHex simple type (§15.8.2).</p>

15.4.1.6 Additional attribute for sheetProtection element (Part 1, §18.3.1.85)

The following additional attributes can be specified for a document of a transitional conformance class:

Attributes	Description
password (Legacy Password)	<p>Specifies the legacy hash of the password required for editing this worksheet.</p> <p>The hash is generated using the logic defined in the revisionsPassword attribute of the workbookProtection element (Part 1, §18.2.29).</p> <p>The possible values for this attribute are defined by the ST_UnsignedShortHex simple type (§15.8.2).</p>

15.4.2 AutoFilter Settings

15.4.2.1 Attributes with modified descriptions for dynamicFilter element (Part 1, §18.3.2.5)

The following attributes have modified descriptions when specified for a document of a transitional conformance class:

Attributes	Description
maxVal (Max Value)	A maximum value for dynamic filter. maxVal/maxValIso shall be required for today,

Attributes	Description
	<p>yesterday, tomorrow, nextWeek, thisWeek, lastWeek, nextMonth, thisMonth, lastMonth, nextQuarter, thisQuarter, lastQuarter, nextYear, thisYear, lastYear, and yearToDate.</p> <p>The above criteria are based on a value range. [<i>Example</i>: If today's date is September 22nd, then the range for thisWeek is the values greater than or equal to September 17 and less than September 24. <i>end example</i>] In the thisWeek range, the lower value is expressed using val or valIso. The higher value is expressed using maxVal or maxValIso.</p> <p>These dynamic filter shall not require val/valIso or maxVal/maxValIso: Q1, Q2, Q3, Q4, M1, M2, M3, M4, M5, M6, M7, M8, M9, M10, M11 and M12.</p> <p>The above criteria shall not specify the range using val/valIso and maxVal/maxValIso because Q1 always starts from M1 to M3, and M1 is always January.</p> <p>These types of dynamic filters shall use val and shall not use maxVal/maxValIso: aboveAverage and belowAverage.</p> <p>If maxValIso and maxVal are both present, maxValIso shall take precedence.</p> <p>The possible values for this attribute are defined by the W3C XML Schema double datatype.</p>
val (Value)	<p>A minimum numeric or serial date value for dynamic filter. (See description of ValIso to understand when val is required.)</p> <p>If valIso and val are both present, valIso shall take precedence.</p> <p>The possible values for this attribute are defined by the W3C XML Schema double datatype.</p>
valIso (ISO Value)	<p>A minimum date value for dynamic filter. (See description of maxVal/maxValIso to understand when val/valIso is required.)</p> <p>The possible values for this attribute are defined by the W3C XML Schema dateTime datatype.</p>

15.5 Styles

15.5.1 left (Leading Edge Border)

Semantically equivalent to start (Part 1, §18.8.37).

Attributes	Description
style (Line Style)	<p>The line style for this border.</p> <p>The possible values for this attribute are defined by the ST_BorderStyle simple type (Part</p>

Attributes	Description
	1, §18.18.3).

[Note: The W3C XML Schema definition of this element's content model ([CT_BorderPr](#)) is located in §A.3. *end note*]

15.5.2 right (Trailing Edge Border)

Semantically equivalent to end (Part 1, §18.8.16).

Attributes	Description
style (Line Style)	The line style for this border. The possible values for this attribute are defined by the ST_BorderStyle simple type (Part 1, §18.18.3).

[Note: The W3C XML Schema definition of this element's content model ([CT_BorderPr](#)) is located in §A.3. *end note*]

15.6 Pivot Tables

15.6.1 Pivot Tables

15.6.1.1 Additional attribute for pivotCacheDefinition element (Part 1, §18.10.1.67)

The following additional attributes can be specified for a document of a transitional conformance class:

Attributes	Description
refreshedDate (PivotCache Last Refreshed Date)	Specifies the date when the cache was last refreshed. This attribute depends on whether the application exposes mechanisms via the user interface whereby the end-user can refresh the cache. If refreshedDateIso and refreshedDate are both present, refreshedDateIso shall take precedence. The possible values for this attribute are defined by the W3C XML Schema double datatype.

15.7 External Data Connections

15.7.1 Additional attribute for textPr element (Part 1, §18.13.12)

The following additional attributes can be specified for a document of a transitional conformance class:

Attributes	Description
codePage (Code Page)	<p>Code page associated with the text file. This attribute is used only for backwards compatibility, and is ignored if the characterSet attribute is present.</p> <p>[Note: There are a number of code page technologies. One example of potential values can be found at: http://www.unicode.org/Public/MAPPINGS end note]</p> <p>The possible values for this attribute are defined by the W3C XML Schema unsignedInt datatype.</p>

15.8 Simple Types

The following additional simple type information in the <http://schemas.openxmlformats.org/spreadsheetml/2006/main> namespace is used for documents of a transitional conformance class.

15.8.1 Additional enumeration values for ST_PivotAreaType (Part 1, §18.18.58)

The following additional enumeration values can be specified for a document of a transitional conformance class.

Enumeration Value	Description
topRight (Top Corner, Trailing Edge)	Semantically equivalent to topEnd.

15.8.2 ST_UnsignedShortHex (Unsigned Short Hex)

This simple type defines the Hex representation of an unsigned short.

This simple type's contents are a restriction of the W3C XML Schema hexBinary datatype.

This simple type also specifies the following restrictions:

- This simple type's contents have a length of exactly 4 hexadecimal digit(s).

[Note: The W3C XML Schema definition of this simple type's content model ([ST_UnsignedShortHex](#)) is located in §A.3. end note]

15.8.3 Removed enumeration values for ST_CellType (Part 1, §18.18.11)

For transitional documents, the restriction on the simple type ST_CellType having the value "d" (ISO 8601 format) is removed.

15.9 Formulas

15.9.1 Attribute synonym for c element (Part 1, §18.6.1)

The following additional attribute can be specified for a document of a transitional conformance class:

Attributes	Description
------------	-------------

Attributes	Description
ref (Cell Reference)	An A-1 style reference to a cell. The possible values for this attribute are defined by the ST_CellRef simple type (Part 1, §18.18.7).

This attribute is semantically equivalent to r (Part 1, §18.6.1).

Only one or the other of r and ref can be defined in any given instance.

15.9.2 Additional representation for dates and times (Part 1, §18.17.4)

For a document of a transitional conformance class, each unique instant in SpreadsheetML time shall be stored as an ISO 8601-formatted string or as a serial value.

15.10 Changed attributes

The following attributes, which are defined in subclauses within Part 1, §18, “SpreadsheetML”, have different source relationships when used in documents of the Transitional conformance class:

15.10.1 Changed attribute for externalReference element (Part 1, §18.2.8)

Attributes	Description
id (Relationship Id) Namespace: .../officeDocument /2006/relationships	Specifies a unique identifier that is used to identify a relationship to another part in the file. Relationship identifiers link the element definition with the part where data for the element is stored. The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

15.10.2 Changed attribute for pivotCache element (Part 1, §18.2.17)

Attributes	Description
id (Relationship Id) Namespace: .../officeDocument /2006/relationships	Specifies the identifier to a pivot cache definition part where cached data is stored. This attribute is required. The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

15.10.3 Changed attribute for sheet element (Part 1, §18.2.19)

Attributes	Description
id (Relationship Id) Namespace: .../officeDocument /2006/relationships	Specifies the identifier of the sheet part where the definition for this sheet is stored. This attribute is required. The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

15.10.4 Changed attribute for control element (Part 1, §18.3.1.19)

Attributes	Description
id (Relationship Id) Namespace: .../officeDocument /2006/relationships	This relationship ID references an Embedded Control Data part that contains control-specific properties and state information about this particular embedded control. The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

15.10.5 Changed attribute for controlPr element (Part 1, §18.3.1.20)

Attributes	Description
id (Relationship ID for Embedded Control Properties) Namespace: .../officeDocument /2006/relationships	<p>Specifies the relationship ID for the relationship which contains the properties for this embedded control. This property bag is contained in a separate part within the package.</p> <p>The relationship explicitly targeted by this attribute shall be of relationship type http://schemas.openxmlformats.org/officeDocument/2006/relationships/control or the document shall be considered non-conformant.</p> <p>If this attribute is omitted, then the embedded control shall be given no property bag when instantiated.</p> <p>[<i>Example:</i> Consider the following WordprocessingML markup for an embedded control in a document:</p> <pre><w:control r:id="rId5" w:id="CheckBox1" w:name="CheckBox1" w:shapeid="_x0000_s1027" w:class="shape" w:w="145" w:h="28" w:align="left" /></pre> <p>The id attribute in the relationship reference namespace specifies that the relationship with relationship ID rId5 must contain the property data for this embedded control. <i>end example</i></p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

15.10.6 Changed attribute for customPr element (Part 1, §18.3.1.22)

Attributes	Description
id (Relationship Id) Namespace: .../officeDocument /2006/relationships	<p>This relationship references the binary part containing the specified custom properties.</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

15.10.7 Changed attribute for dataRef element (Part 1, §18.3.1.30)

Attributes	Description
id (relationship Id) Namespace: .../officeDocument /2006/relationships	Used only when the source range is external to this workbook. The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

15.10.8 Changed attribute for drawing element (Part 1, §18.3.1.36)

Attributes	Description
id (Relationship id) Namespace: .../officeDocument /2006/relationships	Relationship Id referencing a part containing DrawingML definitions for this worksheet. The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

15.10.9 Changed attribute for drawingHF element (Part 1, §18.3.1.37)

Attributes	Description
id (Relationship ID for Embedded Control Properties) Namespace: .../officeDocument /2006/relationships	Specifies the relationship ID for the relationship to the DrawingML part that contains the drawing objects used in the header and footer. This DrawingML part is a separate part within the package. [Example: <div style="text-align: center;"> <code><drawingHF r:id="rId2" lho="7" lhf="6"/></code> </div> The id attribute in the relationship reference namespace specifies that the relationship with relationship ID rId5 must contain the drawing objects used in the header and footer. <i>end example</i>] The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

15.10.10 Changed attribute for hyperlink element (Part 1, §18.3.1.47)

Attributes	Description
id (Relationship Id) Namespace: .../officeDocument /2006/relationships	Relationship Id in this sheet's relationships part, expressing the target location of the resource. The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

15.10.11 Changed attribute for objectPr element (Part 1, §18.3.1.56)

Attributes	Description
id (Relationship ID to Embedded Object Data) Namespace: .../officeDocument/2006/relationships	<p>Specifies the relationship ID for the relationship that targets the Embedded Object Part containing the embedded object data.</p> <p>The specified relationship shall be of type http://schemas.openxmlformats.org/officeDocument/2006/oleObject or the document shall be considered non-conformant.</p> <p>[<i>Example</i>: Consider an XML element which has the following id attribute:</p> <pre><... r:id="rId1" /></pre> <p>The markup specifies the associated relationship part with relationship ID rId1 targets the part containing the corresponding embedded object information. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

15.10.12 Changed attribute for oleObject element (Part 1, §18.3.1.59)

Attributes	Description
id (Relationship Id) Namespace: .../officeDocument/2006/relationships	<p>Relationship Id of the relationship pointing to the object persistence part.</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

15.10.13 Changed attribute for pageSetup element (Part 1, §18.3.1.63)

Attributes	Description
id (Id) Namespace: .../officeDocument/2006/relationships	<p>Relationship Id of the devMode printer settings part.</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

15.10.14 Changed attribute for pageSetup element (Part 1, §18.3.1.64)

Attributes	Description
id (Id) Namespace: .../officeDocument/2006/relationships	<p>Relationship Id of the devMode printer settings part.</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

Attributes	Description
ps	

15.10.15 Changed attribute for picture element (Part 1, §18.3.1.67)

Attributes	Description
id (Relationship Id)	Relationship Id pointing to the image part.
Namespace: .../officeDocument /2006/relationships ps	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

15.10.16 Changed attribute for pivotSelection element (Part 1, §18.3.1.69)

Attributes	Description
id (Relationship Id)	Relationship Id pointing to the particular PivotTable Part corresponding to this selection.
Namespace: .../officeDocument /2006/relationships ps	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

15.10.17 Changed attribute for tablePart element (Part 1, §18.3.1.94)

Attributes	Description
id (Relationship Id)	This relationship Id is used to locate a particular table definition part.
Namespace: .../officeDocument /2006/relationships ps	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

15.10.18 Changed attribute for pivotCacheDefinition element (Part 1, §18.10.1.67)

Attributes	Description
id (Relationship Identifier)	Specifies the unique identifier that corresponds to the related pivotCacheRecords part. See (Part 1, §18.10.1.68) for more information.
Namespace: .../officeDocument /2006/relationships ps	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

15.10.19 Changed attribute for rangeSet element (Part 1, §18.10.1.79)

Attributes	Description
id (Relationship Id)	Specifies the unique identifier of the Workbook part where the range set is stored. See

Attributes	Description
Namespace: .../officeDocument /2006/relationships	Workbook (Part 1, §18.2) for more information. The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

15.10.20 Changed attribute for worksheetSource element (Part 1, §18.10.1.95)

Attributes	Description
id (Relationship Id) Namespace: .../officeDocument /2006/relationships	Specifies the identifier to the Sheet part whose data is stored in the cache. See the Sheet section (Part 1, §18.2) for more information. The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

15.10.21 Changed attribute for header element (Part 1, §18.11.1.1)

Attributes	Description
id (Relationship ID) Namespace: .../officeDocument /2006/relationships	This is the ID that is used to find the corresponding log record of the changes made for this header. Use the corresponding relationship expressed in the revisionHeaders part to locate the log record that lists the specific changes. The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

15.10.22 Changed attribute for externalBook element (Part 1, §18.14.7)

Attributes	Description
id (Relationship to supporting book file path) Namespace: .../officeDocument /2006/relationships	Relationship ID that references a link in the relationships collection. The target attribute in the associated relationship will specify the worksheet XML file in the current SpreadsheetML document ZIP archive that makes use of this externalbook. The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

15.10.23 Changed attribute for oleLink element (Part 1, §18.14.11)

Attributes	Description
id (Object Link Relationship)	Relationship ID that references a link in the relationships collection. The target attribute in the associated relationship will specify the external file name used for this oleLink.

Attributes	Description
Namespace: .../officeDocument /2006/relationships	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).