<<ISO and IEC logos go here>>

**Information technology — Document description and**

**processing languages — Office Open XML File Formats —**

Part 1:

**Fundamentals and Markup Language Reference**

TECHNICAL CORRIGENDUM 2

*Technologies de l'information — Description des documents et langages de traitement — Formats de fichier
"Office Open XML" —
Partie 1: …*

*RECTIFICATIF TECHNIQUE 2*

Technical Corrigendum 2 to ISO/IEC 29500-1:2012 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 34, *Document description and processing languages*.

It contains corrections that resolve various Defect Reports submitted against ISO/IEC 29500-1:2012.

A correction can involve changes to one or more clauses or subclauses; it can even apply to multiple Parts of ISO/IEC 29500. For changes to ISO/IEC 29500-1:2012, each such change has its own entry below.

Changes are presented in ascending clause, subclause, and page number order.

This is the second Technical Corrigendum for ISO/IEC 29500-1:2012. No amendments to ISO/IEC 29500-1:2012 have been published.**ISO/IEC 29500-1:2012/Cor.2:2016(E)**

**Notational conventions**

The title of each change is the complete reference to the clause or subclause being corrected. In all cases, the title begins with the clause or subclause number, the clause or subclause name, and the page number. In those cases containing changes to a particular row of a table, the value in that row’s first column is appended to the title. As the lines in each XML schema subclause are numbered, corrections to schemas also contain the numbers of the lines being corrected.

A change can contain any one or more of the following kinds of edits:

1. Addition of text: New text is displayed in blue and is underlined, as demonstrated here.
2. Deletion of text: ~~Deleted text is displayed in red and is struck-through, as demonstrated here.~~
3. Change of format of text: Text whose format (but not its content) has changed is displayed in green and is double-underlined, as demonstrated here.

Many changes involve edits to large paragraphs, tables, and/or XML fragments. In such cases, the changes contain only as much unchanged content as is necessary to establish the correct context of each change. Omitted content is identified via the use of ellipses (…).

Within a change, intent that cannot be represented visually as an edit is written as an instruction in italic and delimited by curly brackets; for example: *{In paragraph 2, item 4, and in paragraph 4, make the numbers in the text “17–23” hyperlinked forward references to Clauses 17 and 23.}*

Contents

[Introduction 1](#_Toc437595895)

[Changes 1](#_Toc437595896)

[1. §17.7.2, “Style Hierarchy”, p. 609 1](#_Toc437595897)

[2. §17.18.41, “ST\_Hint (Font Type Hint)”, pp. 1,393–1,394 2](#_Toc437595898)

[3. §18.8.30, “numFmt (Number Format)”, p. 1,769 3](#_Toc437595899)

[4. §19.2.1.13, “font (Embedded Font Name)”, pp. 2,523–2,524 3](#_Toc437595900)

[5. §20.1.10.xx, “ST\_PitchFamily (Pitch Family)”, new subclause, p. xx 6](#_Toc437595901)

[6. §21.1.2.3.1, “cs (Complex Script Font)”, attribute pitchFamily, p. 3,220 7](#_Toc437595902)

[7. §21.1.2.3.3, “ea (East Asian Font)”, attribute pitchFamily, pp. 3,227–3,228 8](#_Toc437595903)

[8. §21.1.2.3.7, “latin (Latin Font)”, attribute pitchFamily, pp. 3,234–3,235 9](#_Toc437595904)

[9. §21.1.2.3.10, “sym (Symbol Font)”, attribute pitchFamily, pp. 3,242–3,243 10](#_Toc437595905)

[10. §21.1.2.4.6, “buFont (Specified)”, attribute pitchFamily, pp. 3,254–3,255 11](#_Toc437595906)

Introduction

This Technical Corrigendum contains corrections that resolve various Defect Reports submitted against ISO/IEC 29500-1:2012.

A correction can involve changes to one or more clause or subclauses; it can even apply to multiple Parts of ISO/IEC 29500. For changes to Part 1, each such change has its own entry below, and the number of the Defect Report that lead to any particular change is written immediately following that change’s title, in the form “[DR *99*-*9999*]”.

Changes are presented in ascending clause, subclause, and page number order.

Changes

# §17.7.2, “Style Hierarchy”, p. 609

[DR 12-0005, DR 12-00025]

*{Replace the existing diagram with the following one.}*

Application order

Document Defaults

Table

Paragraph

Numbering

Character

Direct Formatting

# §17.18.41, “ST\_Hint (Font Type Hint)”, pp. 1,393–1,394

[DR 09-0040]

This simple type s~~S~~pecifies information ~~the font type which shall be~~ used to decide how to format any characters in the current run for which the font type is otherwise ambiguous.

~~There are c~~Certain characters ~~which are not explicitly stored in the document, and~~ can be mapped into ~~multiple~~ more than one of the font slot categories ~~of the four mentioned above~~described in the parent element. This attribute shall be used to ~~arbitrate that conflict, and~~ determine how ambiguities in this run shall be handled. [*Note*: This ~~is primarily~~ can be used to handle the formatting on the paragraph mark glyph, and other characters that are not stored as text in the WordprocessingML document. Some printable characters can be mapped to more than one font slot, such as Unicode glyph U+2026 ‘HORIZONTAL ELLIPSIS’. *end note*]

[*Example*: Consider the run representing the paragraph mark glyph, which is not stored as a physical character. Since this could therefore be formatted with any of the fonts specified for the run, this ambiguity is resolved using the following WordprocessingML:

<w:pPr>
 <w:rPr>
 <w:rFonts w:hint="eastAsia" />
 </w:rPr>
</w:pPr>

The hint attribute specifies that some characters in the run ~~must~~ use an East Asian ~~the eastAsia~~ font ~~(theme or not, whichever is in use for East Asian text) as applied for this run~~. *end example*]

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

This simple type is restricted to the values listed in the following table:

|  |  |
| --- | --- |
| Enumeration Value | Description |
| cs (Complex Script Font) | Specifies that the font hint for this text run shall be to use the Complex Script font defined ~~on~~ for the run via the style hierarchy. |
| default (~~High ANSI~~no ~~Font~~font hint) | Specifies that ~~the font~~no hint shall apply ~~for~~ to this text run ~~shall be to use the High ANSI font defined on the run via the style hierarchy~~. |
| eastAsia (East Asian Font) | Specifies that the font hint for this text run shall be to use the East Asian font defined ~~on~~ for the run via the style hierarchy. |

[*Note*: The W3C XML Schema definition of this simple type’s content model ([ST\_Hint](#xsd_s_9fa48a8d-e0a9-4324-a2aa-19ac6a77e2)) is located in §TBD. *end note*]

# §18.8.30, “numFmt (Number Format)”, p. 1,769

[DR 14-0006]

*{The corresponding correction in ISO/IEC 29500-1:2012/Cor.1:2015 is incorrect, and is replaced by the following:}*

…

Following is a listing of number formats whose formatCode value is implied rather than explicitly saved in the file. In this case, a numFmtId value is written on the xf record, but no corresponding numFmt element is written. Some of these Ids can be interpreted differently, depending on the UI language of the implementing application.

Ids not specified in the listing, such as 5, 6, 7, and 8, shall follow the number format specified by the formatCode attribute.

[Note: To maximize interoperability, implementers should restrict the content of this attribute to enumerations present in the lists below. Additional values may be used, but interoperability will only be possible via mutual agreement between implementers. end note]

When values not present in the lists below are used, the behavior is implementation-defined.

**All Languages**

…

# §19.2.1.13, “font (Embedded Font Name)”, pp. 2,523–2,524

[DR 09-0037, DR 09-0055]

Font Substitution Logic:

…

[Note: Not all characters for a typeface must be stored. It is up to the generating application to determine which characters are to be stored in the corresponding font data files. end note]

If ambiguities exist between properties in the markup of a body of text that refers to an embedded font (such as the latin element specified in §21.1.2.3.7) and properties in the markup of the corresponding instance of this embedded font element, the determination whether to use that embedded font is application-dependent behavior. If ambiguities exist between properties in the markup of an instance of this embedded font element and properties within the corresponding embedded Font part as specified in §15.2.13, the determination whether to use that embedded font is application-dependent behavior.

|  |  |
| --- | --- |
| Attributes | Description |
| charset (Similar Character Set) | Specifies the character set that is supported by the parent font. This information can be used in font substitution logic to locate an appropriate substitute font when this font is not available. This information is determined by querying the font when present and shall not be modified when the font is not available.The value of this attribute shall be interpreted as follows:

| **Value** | **Description** |
| --- | --- |
| 0x00 | Specifies the ANSI character set. (IANA name iso-8859-1) |
| 0x01 | Specifies the default character set. |
| 0x02 | Specifies the Symbol character set. This value specifies that the characters in the Unicode private use area (U+FF00 to U+FFFF) of the font should be used to display characters in the range U+0000 to U+00FF. |
| 0x4D | Specifies a Macintosh (Standard Roman) character set. (IANA name macintosh) |
| 0x80 | Specifies the JIS character set. (IANA name shift\_jis) |
| 0x81 | Specifies the Hangul character set. (IANA name ks\_c\_5601-1987) |
| 0x82 | Specifies a Johab character set. (IANA name KS C-5601-1992) |
| 0x86 | Specifies the GB-2312 character set. (IANA name GBK) |
| 0x88 | Specifies the Chinese Big Five character set. (IANA name Big5) |
| 0xA1 | Specifies a Greek character set. (IANA name windows-1253) |
| 0xA2 | Specifies a Turkish character set. (IANA name iso-8859-9) |
| 0xA3 | Specifies a Vietnamese character set. (IANA name windows-1258) |
| 0xB1 | Specifies a Hebrew character set. (IANA name windows-1255) |
| 0xB2 | Specifies an Arabic character set. (IANA name windows-1256) |
| 0xBA | Specifies a Baltic character set. (IANA name windows-1257) |
| 0xCC | Specifies a Russian character set. (IANA name windows-1251) |
| 0xDE | Specifies a Thai character set. (IANA name windows-874) |
| 0xEE | Specifies an Eastern European character set. (IANA name windows-1250) |
| 0xFF | Specifies an OEM character set not defined by ISO/IEC 29500. |
| Any other value | Application-defined, can be ignored. |

The possible values for this attribute are defined by the W3C XML Schema byte datatype. |
| panose (Panose Setting) | Specifies the Panose-1 classification number for the current font using the mechanism defined in §4.2.7.17 of ISO/IEC 14496-22:2007.The possible values for this attribute are defined by the ST\_Panose simple type (§22.9.2.8). |
| pitchFamily (Similar Font Family) | Specifies the font pitch as well as the font family for the corresponding font.This information is determined by querying the font when present and shall not be modified when the font is not available. This information can be used in font substitution logic to locate an appropriate substitute font when this font is not available.The possible values for this attribute are defined by the ST\_PitchFamily simple type (§20.1.10.xx). |
| typeface (Text Typeface)… | … |

…

# §20.1.10.xx, “ST\_PitchFamily (Pitch Family)”, new subclause, p. xx

[DR 09-0055]

**20.1.10.xx ST\_PitchFamily (Pitch Family)**

This simple type specifies a font pitch.

[*Note:* Although the type name is ST\_PitchFamily, the integer value of this attribute specifies the font family with the higher 4 bits and the font pitch with the lower 4 bits. *end note*]

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

This simple type is restricted to the values listed in the following table:

| **Value** | **Description** |
| --- | --- |
| 0x00 | DEFAULT PITCH + UNKNOWN FONT FAMILY |
| 0x01 | FIXED PITCH + UNKNOWN FONT FAMILY |
| 0x02 | VARIABLE PITCH + UNKNOWN FONT FAMILY |
| 0x10 | DEFAULT PITCH + ROMAN FONT FAMILY |
| 0x11 | FIXED PITCH + ROMAN FONT FAMILY |
| 0x12 | VARIABLE PITCH + ROMAN FONT FAMILY |
| 0x20 | DEFAULT PITCH + SWISS FONT FAMILY |
| 0x21 | FIXED PITCH + SWISS FONT FAMILY |
| 0x22 | VARIABLE PITCH + SWISS FONT FAMILY |
| 0x30 | DEFAULT PITCH + MODERN FONT FAMILY |
| 0x31 | FIXED PITCH + MODERN FONT FAMILY |
| 0x32 | VARIABLE PITCH + MODERN FONT FAMILY |
| 0x40 | DEFAULT PITCH + SCRIPT FONT FAMILY |
| 0x41 | FIXED PITCH + SCRIPT FONT FAMILY |
| 0x42 | VARIABLE PITCH + SCRIPT FONT FAMILY |
| 0x50 | DEFAULT PITCH + DECORATIVE FONT FAMILY |
| 0x51 | FIXED PITCH + DECORATIVE FONT FAMILY |
| 0x52 | VARIABLE PITCH + DECORATIVE FONT FAMILY |

[*Note*: The W3C XML Schema definition of this simple type’s content model ([ST\_PitchFamily](#xsd_s_3dbeeb50-6fdd-490e-86c9-b7a5f36004)) is located in §A.3. *end note*]

# §21.1.2.3.1, “cs (Complex Script Font)”, attribute pitchFamily, p. 3,220

[DR 09-0055]

|  |  |
| --- | --- |
| Attributes | Description |
| pitchFamily (Similar Font Family) | Specifies the font pitch as well as the font family for the corresponding font. ~~Because the value of this attribute is determined by an octet value this value shall be interpreted as follows:~~

| **~~Value~~** | **~~Description~~** |
| --- | --- |
| ~~0x00~~ | ~~DEFAULT PITCH + UNKNOWN FONT FAMILY~~ |
| ~~0x01~~ | ~~FIXED PITCH + UNKNOWN FONT FAMILY~~ |
| ~~0x02~~ | ~~VARIABLE PITCH + UNKNOWN FONT FAMILY~~ |
| ~~0x10~~ | ~~DEFAULT PITCH + ROMAN FONT FAMILY~~ |
| ~~0x11~~ | ~~FIXED PITCH + ROMAN FONT FAMILY~~ |
| ~~0x12~~ | ~~VARIABLE PITCH + ROMAN FONT FAMILY~~ |
| ~~0x20~~ | ~~DEFAULT PITCH + SWISS FONT FAMILY~~ |
| ~~0x21~~ | ~~FIXED PITCH + SWISS FONT FAMILY~~ |
| ~~0x22~~ | ~~VARIABLE PITCH + SWISS FONT FAMILY~~ |
| ~~0x30~~ | ~~DEFAULT PITCH + MODERN FONT FAMILY~~ |
| ~~0x31~~ | ~~FIXED PITCH + MODERN FONT FAMILY~~ |
| ~~0x32~~ | ~~VARIABLE PITCH + MODERN FONT FAMILY~~ |
| ~~0x40~~ | ~~DEFAULT PITCH + SCRIPT FONT FAMILY~~ |
| ~~0x41~~ | ~~FIXED PITCH + SCRIPT FONT FAMILY~~ |
| ~~0x42~~ | ~~VARIABLE PITCH + SCRIPT FONT FAMILY~~ |
| ~~0x50~~ | ~~DEFAULT PITCH + DECORATIVE FONT FAMILY~~ |
| ~~0x51~~ | ~~FIXED PITCH + DECORATIVE FONT FAMILY~~ |
| ~~0x52~~ | ~~VARIABLE PITCH + DECORATIVE FONT FAMILY~~ |

This information is determined by querying the font when present and shall not be modified when the font is not available. This information can be used in font substitution logic to locate an appropriate substitute font when this font is not available.~~[~~*~~Note~~*~~: Although the attribute name is pitchFamily, the integer value of this attribute specifies the font family with higher 4 bits and the font pitch with lower 4 bits.~~ *~~end note~~*~~]~~The possible values for this attribute are defined by the ST\_PitchFamily simple type (§20.1.10.xx)~~W3C XML Schema byte datatype~~. |

# §21.1.2.3.3, “ea (East Asian Font)”, attribute pitchFamily, pp. 3,227–3,228

[DR 09-0055]

|  |  |
| --- | --- |
| Attributes | Description |
| pitchFamily (Similar Font Family) | Specifies the font pitch as well as the font family for the corresponding font. ~~Because the value of this attribute is determined by a byte variable this value shall be interpreted as follows:~~

| **~~Value~~** | **~~Description~~** |
| --- | --- |
| ~~0x00~~ | ~~DEFAULT PITCH + UNKNOWN FONT FAMILY~~ |
| ~~0x01~~ | ~~FIXED PITCH + UNKNOWN FONT FAMILY~~ |
| ~~0x02~~ | ~~VARIABLE PITCH + UNKNOWN FONT FAMILY~~ |
| ~~0x10~~ | ~~DEFAULT PITCH + ROMAN FONT FAMILY~~ |
| ~~0x11~~ | ~~FIXED PITCH + ROMAN FONT FAMILY~~ |
| ~~0x12~~ | ~~VARIABLE PITCH + ROMAN FONT FAMILY~~ |
| ~~0x20~~ | ~~DEFAULT PITCH + SWISS FONT FAMILY~~ |
| ~~0x21~~ | ~~FIXED PITCH + SWISS FONT FAMILY~~ |
| ~~0x22~~ | ~~VARIABLE PITCH + SWISS FONT FAMILY~~ |
| ~~0x30~~ | ~~DEFAULT PITCH + MODERN FONT FAMILY~~ |
| ~~0x31~~ | ~~FIXED PITCH + MODERN FONT FAMILY~~ |
| ~~0x32~~ | ~~VARIABLE PITCH + MODERN FONT FAMILY~~ |
| ~~0x40~~ | ~~DEFAULT PITCH + SCRIPT FONT FAMILY~~ |
| ~~0x41~~ | ~~FIXED PITCH + SCRIPT FONT FAMILY~~ |
| ~~0x42~~ | ~~VARIABLE PITCH + SCRIPT FONT FAMILY~~ |
| ~~0x50~~ | ~~DEFAULT PITCH + DECORATIVE FONT FAMILY~~ |
| ~~0x51~~ | ~~FIXED PITCH + DECORATIVE FONT FAMILY~~ |
| ~~0x52~~ | ~~VARIABLE PITCH + DECORATIVE FONT FAMILY~~ |

This information is determined by querying the font when present and shall not be modified when the font is not available. This information can be used in font substitution logic to locate an appropriate substitute font when this font is not available.The possible values for this attribute are defined by the ST\_PitchFamily simple type (§20.1.10.xx)~~W3C XML Schema byte datatype~~. |

# §21.1.2.3.7, “latin (Latin Font)”, attribute pitchFamily, pp. 3,234–3,235

[DR 09-0055]

|  |  |
| --- | --- |
| Attributes | Description |
| pitchFamily (Similar Font Family) | Specifies the font pitch as well as the font family for the corresponding font. ~~Because the value of this attribute is determined by an octet value this value shall be interpreted as follows:~~

| **~~Value~~** | **~~Description~~** |
| --- | --- |
| ~~0x00~~ | ~~DEFAULT PITCH + UNKNOWN FONT FAMILY~~ |
| ~~0x01~~ | ~~FIXED PITCH + UNKNOWN FONT FAMILY~~ |
| ~~0x02~~ | ~~VARIABLE PITCH + UNKNOWN FONT FAMILY~~ |
| ~~0x10~~ | ~~DEFAULT PITCH + ROMAN FONT FAMILY~~ |
| ~~0x11~~ | ~~FIXED PITCH + ROMAN FONT FAMILY~~ |
| ~~0x12~~ | ~~VARIABLE PITCH + ROMAN FONT FAMILY~~ |
| ~~0x20~~ | ~~DEFAULT PITCH + SWISS FONT FAMILY~~ |
| ~~0x21~~ | ~~FIXED PITCH + SWISS FONT FAMILY~~ |
| ~~0x22~~ | ~~VARIABLE PITCH + SWISS FONT FAMILY~~ |
| ~~0x30~~ | ~~DEFAULT PITCH + MODERN FONT FAMILY~~ |
| ~~0x31~~ | ~~FIXED PITCH + MODERN FONT FAMILY~~ |
| ~~0x32~~ | ~~VARIABLE PITCH + MODERN FONT FAMILY~~ |
| ~~0x40~~ | ~~DEFAULT PITCH + SCRIPT FONT FAMILY~~ |
| ~~0x41~~ | ~~FIXED PITCH + SCRIPT FONT FAMILY~~ |
| ~~0x42~~ | ~~VARIABLE PITCH + SCRIPT FONT FAMILY~~ |
| ~~0x50~~ | ~~DEFAULT PITCH + DECORATIVE FONT FAMILY~~ |
| ~~0x51~~ | ~~FIXED PITCH + DECORATIVE FONT FAMILY~~ |
| ~~0x52~~ | ~~VARIABLE PITCH + DECORATIVE FONT FAMILY~~ |

This information is determined by querying the font when present and shall not be modified when the font is not available. This information can be used in font substitution logic to locate an appropriate substitute font when this font is not available.~~[~~*~~Note~~*~~: Although the attribute name is pitchFamily, the integer value of this attribute specifies the font family with higher 4 bits and the font pitch with lower 4 bits.~~ *~~end note~~*~~]~~The possible values for this attribute are defined by the ST\_PitchFamily simple type (§20.1.10.xx)~~W3C XML Schema byte datatype~~. |

# §21.1.2.3.10, “sym (Symbol Font)”, attribute pitchFamily, pp. 3,242–3,243

[DR 09-0055]

|  |  |
| --- | --- |
| Attributes | Description |
| pitchFamily (Similar Font Family) | Specifies the font pitch as well as the font family for the corresponding font. ~~Because the value of this attribute is determined by an octet value this value shall be interpreted as follows:~~

| **~~Value~~** | **~~Description~~** |
| --- | --- |
| ~~0x00~~ | ~~DEFAULT PITCH + UNKNOWN FONT FAMILY~~ |
| ~~0x01~~ | ~~FIXED PITCH + UNKNOWN FONT FAMILY~~ |
| ~~0x02~~ | ~~VARIABLE PITCH + UNKNOWN FONT FAMILY~~ |
| ~~0x10~~ | ~~DEFAULT PITCH + ROMAN FONT FAMILY~~ |
| ~~0x11~~ | ~~FIXED PITCH + ROMAN FONT FAMILY~~ |
| ~~0x12~~ | ~~VARIABLE PITCH + ROMAN FONT FAMILY~~ |
| ~~0x20~~ | ~~DEFAULT PITCH + SWISS FONT FAMILY~~ |
| ~~0x21~~ | ~~FIXED PITCH + SWISS FONT FAMILY~~ |
| ~~0x22~~ | ~~VARIABLE PITCH + SWISS FONT FAMILY~~ |
| ~~0x30~~ | ~~DEFAULT PITCH + MODERN FONT FAMILY~~ |
| ~~0x31~~ | ~~FIXED PITCH + MODERN FONT FAMILY~~ |
| ~~0x32~~ | ~~VARIABLE PITCH + MODERN FONT FAMILY~~ |
| ~~0x40~~ | ~~DEFAULT PITCH + SCRIPT FONT FAMILY~~ |
| ~~0x41~~ | ~~FIXED PITCH + SCRIPT FONT FAMILY~~ |
| ~~0x42~~ | ~~VARIABLE PITCH + SCRIPT FONT FAMILY~~ |
| ~~0x50~~ | ~~DEFAULT PITCH + DECORATIVE FONT FAMILY~~ |
| ~~0x51~~ | ~~FIXED PITCH + DECORATIVE FONT FAMILY~~ |
| ~~0x52~~ | ~~VARIABLE PITCH + DECORATIVE FONT FAMILY~~ |

This information is determined by querying the font when present and shall not be modified when the font is not available. This information can be used in font substitution logic to locate an appropriate substitute font when this font is not available.~~[~~*~~Note~~*~~: Although the attribute name is pitchFamily, the integer value of this attribute specifies the font family with higher 4 bits and the font pitch with lower 4 bits.~~ *~~end note~~*~~]~~The possible values for this attribute are defined by the ST\_PitchFamily simple type (§20.1.10.xx)~~W3C XML Schema byte datatype~~. |

# §21.1.2.4.6, “buFont (Specified)”, attribute pitchFamily, pp. 3,254–3,255

[DR 09-0055]

|  |  |
| --- | --- |
| Attributes | Description |
| pitchFamily (Similar Font Family) | Specifies the font pitch as well as the font family for the corresponding font. ~~Because the value of this attribute is determined by an octet value this value shall be interpreted as follows:~~

| **~~Value~~** | **~~Description~~** |
| --- | --- |
| ~~0x00~~ | ~~DEFAULT PITCH + UNKNOWN FONT FAMILY~~ |
| ~~0x01~~ | ~~FIXED PITCH + UNKNOWN FONT FAMILY~~ |
| ~~0x02~~ | ~~VARIABLE PITCH + UNKNOWN FONT FAMILY~~ |
| ~~0x10~~ | ~~DEFAULT PITCH + ROMAN FONT FAMILY~~ |
| ~~0x11~~ | ~~FIXED PITCH + ROMAN FONT FAMILY~~ |
| ~~0x12~~ | ~~VARIABLE PITCH + ROMAN FONT FAMILY~~ |
| ~~0x20~~ | ~~DEFAULT PITCH + SWISS FONT FAMILY~~ |
| ~~0x21~~ | ~~FIXED PITCH + SWISS FONT FAMILY~~ |
| ~~0x22~~ | ~~VARIABLE PITCH + SWISS FONT FAMILY~~ |
| ~~0x30~~ | ~~DEFAULT PITCH + MODERN FONT FAMILY~~ |
| ~~0x31~~ | ~~FIXED PITCH + MODERN FONT FAMILY~~ |
| ~~0x32~~ | ~~VARIABLE PITCH + MODERN FONT FAMILY~~ |
| ~~0x40~~ | ~~DEFAULT PITCH + SCRIPT FONT FAMILY~~ |
| ~~0x41~~ | ~~FIXED PITCH + SCRIPT FONT FAMILY~~ |
| ~~0x42~~ | ~~VARIABLE PITCH + SCRIPT FONT FAMILY~~ |
| ~~0x50~~ | ~~DEFAULT PITCH + DECORATIVE FONT FAMILY~~ |
| ~~0x51~~ | ~~FIXED PITCH + DECORATIVE FONT FAMILY~~ |
| ~~0x52~~ | ~~VARIABLE PITCH + DECORATIVE FONT FAMILY~~ |

This information is determined by querying the font when present and shall not be modified when the font is not available. This information can be used in font substitution logic to locate an appropriate substitute font when this font is not available.~~[~~*~~Note~~*~~: Although the attribute name is pitchFamily, the integer value of this attribute specifies the font family with higher 4 bits and the font pitch with lower 4 bits.~~ *~~end note~~*~~]~~The possible values for this attribute are defined by the ST\_PitchFamily simple type (§20.1.10.xx)~~W3C XML Schema byte datatype~~. |