# Office Open XML Themes, Schemes, and Fonts

.Michael Bowman. 22 Apr 2013 8:18 PM | 0

We frequently get questions about how Office selects substitute fonts when a specified font is not found. This article describes how Office Open XML-compatible software can select fonts in text runs. It includes information about the font selection process that is used in Office 2010.

The Office Open XML (OOXML) specification provides software developers with a standard way to store information in documents produced by workplace productivity applications such as word processors, electronic spreadsheets, and presentation programs. But the methods that developers use to implement Office Open XML standard is up to the developer. One common challenge faced by implementers is how to select substitute fonts for OOXML text runs, themes, or schemes, for which the specified font is not found. This article presents one possible implementation by explaining how Office 2010 selects fonts when it performs font substitution. Please be aware that this is not the only possible implementation. Rather, it is only one example of how things might be done.

## Overview

To understand the font selection process, it is first necessary to have some background about how OOXML stores formatting information in documents.

### Text formatting in OOXML

Text in OOXML documents is contained in text runs. A text run is a group of characters that a document groups together in one parent field. An example of a parent field is a paragraph. Every paragraph contains at least one text run that contains the paragraph's text. It is possible for a paragraph to contain more than one text run.

Text runs can occur in a variety of parent fields. Examples of other fields that contain text runs are hyperlinks, custom XML elements, structured document tags, and smart tags.

Text runs can contain child fields that specify formatting and fonts. One of the most relevant to the current discussion is the `rPr` field, which specifies the run's properties. The `rPr` field, in turn, can contain an `rFont` tag, which selects the fonts to use in a text run. Here is an example.

```
<w:r>

    <w:rPr>

        <w:rFonts w:ascii="Courier
New" w:cs="Times New Roman" />

    </w:rPr>


<w:t>English العربية </w:t>

</w:r>
```

The XML code shown above selects the Courier New font for all ASCII characters and uses the Times New Roman font for all Complex Script characters, which are discussed shortly.

The characters in a text run are stored in memory using their Unicode values. The Unicode standard partitions the character values into specific ranges.

1. ASCII (character values 0-128)
2. High ANSI
3. Complex Script
4. East Asian

Because these character values have such distinct groupings, applications that process OOXML documents can glean information about what font to use by testing the character values in the text runs. Applications often use this as one step in the process of selecting a substitute font when the font specified in the OOXML document cannot be found in the document and is not installed on the computer.

### Text styling in OOXML

OOXML enables documents to contain styles that specify a particular set of fonts to use in a document. This is known as the *font scheme*. The font scheme can be grouped together with the other schemes, such as the color scheme, to form a theme. The theme can be used to give groups of documents a consistent appearance. A user can, for instance, define a corporate theme that gives all documents produced by a corporation a similar look.

OOXML schemes can have major fonts and minor fonts. Usually, major fonts are used for styles such as headings, whereas minor fonts are generally applied to body and paragraph text.

## Latin, East Asian, and Complex Script fonts

It is possible that themes and schemes may be used worldwide and therefore must accommodate many languages and their associated fonts. For example, a scheme can contain a minor font field that specifies fonts to be used when the document contains Japanese, Arabic, English, and Tibetan text. OOXML provides global support by using industry-standard classifications for fronts that group them into three basic types: Latin, East Asian, and Complex Script. It enables theme designers to specify default fonts for each of these groups. In addition, it offers specific XML tags that let you select default fonts for the individual languages that you want to support in your theme. Consider the following example.

```
<minorFont>

   <latin typeface="Calibri"/>

   <ea typeface="Arial"/>

   <cs typeface="Arial"/>

   <font script="Jpan"
typeface="ＭＳ Ｐゴシック"/>

   <font script="Hang"
typeface="HY중고딕"/>

   <font
script="Hans" typeface="隶书"/>

   <font
script="Hant" typeface="微軟正黑體"/>

   <font
script="Arab" typeface="Traditional Arabic"/>

   <font
script="Hebr" typeface="Arial"/>

   <font
script="Thai" typeface="Cordia New"/>

   <font
script="Ethi" typeface="Nyala"/>

   <font
script="Beng" typeface="Vrinda"/>

   <font
script="Gujr" typeface="Shruti"/>

   <font
script="Khmr" typeface="DaunPenh"/>

   <font
script="Knda" typeface="Tunga"/>

</minorFont>
```

The preceding XML shows the definition of a group of minor fonts for a scheme. As you can see, it selects specific default fonts for the Latin, East Asian, and Complex Script groups. In addition, it defines default fonts for a variety of languages. In addition to the `rFont` tag, OOXML documents can also use the `rPr` tag and language code IDs (LCIDs) to select fonts for a text run, as shown in the following example.

```
<w:r>

    <w:rPr lang="zh-CN"
dirty="0" smtClean="0">

        <w:solidFill>

            <w:schemeClr
val="bg1"/>

        </w:solidFill>
    </w:rPr>

    <w:t>你好</w:t>

</w:r>
```

This XML snippet demonstrates how a document might select simplified Chinese to display the text. But it does not explicitly state exactly which font to use for the simplified Chinese characters. The application processing this document must look up the appropriate font in the major or minor font table. If the font is not found, it must use a font selection process to substitute the best available font for the one specified in the font table.

## The ISO 29500 font selection process

According to the ISO 29500 standard, font selection is an implementation-specific process for each application program. But to follow the standard, applications should give priority to the font or fonts selected by the `rFonts` tag. If the `rFonts` tag is not present, it selects the nearest font specified in the style hierarchy. If no styles are present or no styles specify a font, then it should use the default fonts specified by the application.

It is possible that the font specified in the `rFonts` tag or the style hierarchy is not embedded in the document and not installed on the user's computer. If the application finds that this is the case, it must use a font substitution algorithm to select the appropriate font. If the font specified in the `rFonts` tag is not available, then the application should search the style hierarchy for the most appropriate font.

For example, if the `rFonts` tag selects a font for simplified Chinese that is not embedded in the document and not present on the computer, the application can check to determine whether the style hierarchy contains a specification for a simplified Chinese font that is available. If the font is not present or not specified, the application checks the major or minor font table for simplified Chinese. If the font is still not specified or is not available, then the application should use the default East Asian font specified in the major or minor font table. If, yet again, there is no font specified in the major or minor font table, or the font cannot be accessed, the application must then either use an application-specific default font or perform font substitution.

When an application performs font substitution, the ISP 29500 standard recommends that it select the font that matches most closely with the font that it is trying to substitute for. Each glyph in the font should have the most similar appearance possible to the glyphs in the specified font. The font that the application selects should also try to match the physical characteristics of the selected font. The ISO 29500 standard recommends matching the following characteristics in the order of descending priority.

1. panose1
2. sig
3. charset
4. pitch
5. family
6. altName

Of course, developers are free to apply more sophisticated logic to the font selection process.

The following table presents a more specific list of elements that may be useful to your application during the font substitution process.

### Font substitution hints

| Element Name | Attribute | Description |
|---|---|---|
| subFontBySize | N/A | Require exact size during font substitution. |
| font | charset | Character set that is supported by the parent font |
| font | family | Font family |
| font | panose | Panose-1 classification number. |
| font | pitchFamily | Font pitch and the font family. |
| font | typeface | Typeface name of the font. |

## The Office 2010 font selection process

All applications that implement the ISO 29500 standard do so in accordance with their business needs. Therefore, when the standard allows then to do so, they use algorithms and implementation methods that serve their customers' best interests. Microsoft is no exception. Microsoft Office implements the ISO 29500 standard for a worldwide audience and therefore must meet the needs of a wide range of customer requirements. As a result, its implementation of the font selection process is often used as an example for other software. Also, many software vendors see a need for their software to be compatible with Office 2010. For these reasons, the font selection algorithm used by Office 2010 is presented here.

### Step 1: Follow the ISO 29500 standard

Office 2010 follows the standard ISO 29500 font selection process in that it gives preferences to the font or fonts selected in the `rFonts` tag, if it is present. It then uses the style hierarchy and the default fonts in the major and minor font tables exactly as the standard directs. If it finds a suitable font, then it uses that font.

If no suitable font is found, then the ISO 29500 standard directs that Office 2010, like any other software application, must perform font substitution using an implementation-dependent algorithm. Therefore, it moves on to the next step in its font selection process.

### Step 2: Convert the LCID to a script tag

If an LCID is specified for the text run, Office 2010 converts the LCID to a script tag. It uses the following exhaustive table to perform the conversion.

## LCID to script tag conversion

| LCID | Script Tag | Value |
|---|---|---|
| lidAmerican | 0x409 | Latn |
| lidAustralian | 0xc09 | Latn |
| lidBritish | 0x809 | Latn |
| lidEnglishCanadian | 0x1009 | Latn |
| lidCatalan | 0x403 | Latn |
| lidDanish | 0x406 | Latn |
| lidDutch | 0x413 | Latn |
| lidDutchBelgian | 0x813 | Latn |
| lidPapiamentu | 0x479 | Latn |
| lidFinnish | 0x40b | Latn |
| lidFrench | 0x40c | Latn |
| lidFrenchCanadian | 0xc0c | Latn |
| lidGerman | 0x407 | Latn |
| lidSwissGerman | 0x807 | Latn |
| lidAustrianGerman | 0xc07 | Latn |
| lidGermanLuxembourg | 0x1007 | Latn |
| lidGermanLiechtenstein | 0x1407 | Latn |
| lidItalian | 0x410 | Latn |
| lidNorskBokmal | 0x414 | Latn |
| lidNorskNynorsk | 0x814 | Latn |
| lidPortBrazil | 0x416 | Latn |
| lidPortIberian | 0x816 | Latn |

| | | |
|---|---|---|
| lidSpanish | 0x40a | Latn |
| lidSwedish | 0x41d | Latn |
| lidRussian | 0x419 | Cyrl |
| lidCzech | 0x405 | Latn |
| lidHungarian | 0x40e | Latn |
| lidPolish | 0x415 | Latn |
| lidTurkish | 0x41f | Latn |
| lidFarsi | 0x429 | Arab |
| lidBasque | 0x42d | Latn |
| lidSlovenian | 0x424 | Latn |
| lidLatvian | 0x426 | Latn |
| lidLithuanian | 0x427 | Latn |
| lidRomanian | 0x418 | Latn |
| lidRomanianMoldavia | 0x818 | Latn |
| lidBulgarian | 0x402 | Cyrl |
| lidSerbianLatin | 0x241a | Latn |
| lidSerbianCyrillic | 0x281a | Cyrl |
| lidCroatian | 0x41a | Latn |
| lidGaelicScots | 0x491 | Latn |
| lidGaelicIrish | 0x83c | Latn |
| lidSutu | 0x430 | Latn |
| lidTsonga | 0x431 | Latn |
| lidTswana | 0x432 | Latn |
| lidVenda | 0x433 | Latn |
| lidXhosa | 0x434 | Latn |
| lidZulu | 0x435 | Latn |
| lidAfrikaans | 0x436 | Latn |
| lidKoreanExtWansung | 0x412 | Hang |
| lidUkranian | 0x422 | Cyrl |
| lidGreek | 0x408 | Grek |
| lidEstonian | 0x425 | Latn |
| lidGalician | 0x456 | Latn |
| lidArabic | 0x401 | Arab |
| lidHebrew | 0x40d | Hebr |
| lidJapanese | 0x411 | Jpan |
| lidSlovak | 0x41b | Latn |
| lidIraq | 0x801 | Arab |
| lidEgyptian | 0xc01 | Arab |
| lidLibya | 0x1001 | Arab |

| | | |
|---|---|---|
| lidAlgerian | 0x1401 | Arab |
| lidMorocco | 0x1801 | Arab |
| lidTunisia | 0x1c01 | Arab |
| lidOman | 0x2001 | Arab |
| lidYemen | 0x2401 | Arab |
| lidSyria | 0x2801 | Arab |
| lidJordan | 0x2c01 | Arab |
| lidLebanon | 0x3001 | Arab |
| lidKuwait | 0x3401 | Arab |
| lidUAE | 0x3801 | Arab |
| lidBahrain | 0x3c01 | Arab |
| lidQatar | 0x4001 | Arab |
| lidChineseTrad | 0x404 | Hant |
| lidChineseSimp | 0x804 | Hans |
| lidHongkong | 0xc04 | Hant |
| lidSingapore | 0x1004 | Hans |
| lidMacau | 0x1404 | Hant |
| lidEnglishNewZealand | 0x1409 | Latn |
| lidEnglishIreland | 0x1809 | Latn |
| lidEnglishSouthAfrica | 0x1c09 | Latn |
| lidEnglishJamaica | 0x2009 | Latn |
| lidEnglishCaribbean | 0x2409 | Latn |
| lidEnglishBelize | 0x2809 | Latn |
| lidEnglishTrinidad | 0x2c09 | Latn |
| lidEnglishZimbabwe | 0x3009 | Latn |
| lidEnglishPhilippines | 0x3409 | Latn |
| lidEnglishIndonesia | 0x3809 | Latn |
| lidEnglishHongKong | 0x3c09 | Latn |
| lidEnglishIndia | 0x4009 | Latn |
| lidEnglishMalaysia | 0x4409 | Latn |
| lidEnglishSingapore | 0x4809 | Latn |
| lidSpanish | 0x40a | Latn |
| lidSpanishMexican | 0x80a | Latn |
| lidMexican | 0x80a | Latn |
| lidSpanishModern | 0xc0a | Latn |
| lidGuatemala | 0x100a | Latn |
| lidCostaRica | 0x140a | Latn |

| | | |
|---|---|---|
| lidPanama | 0x180a | Latn |
| lidDominicanRepublic | 0x1c0a | Latn |
| lidSpanishSA | 0x200a | Latn |
| lidVenezuela | 0x200a | Latn |
| lidColombia | 0x240a | Latn |
| lidPeru | 0x280a | Latn |
| lidArgentina | 0x2c0a | Latn |
| lidEcuador | 0x300a | Latn |
| lidChile | 0x340a | Latn |
| lidUruguay | 0x380a | Latn |
| lidParguay | 0x3c0a | Latn |
| lidBolivia | 0x400a | Latn |
| lidElSalvador | 0x440a | Latn |
| lidHonduras | 0x480a | Latn |
| lidNicaragua | 0x4c0a | Latn |
| lidPuertoRico | 0x500a | Latn |
| lidSpanishUS | 0x540a | Latn |
| lidFrenchBelgian | 0x80c | Latn |
| lidFrenchSwiss | 0x100c | Latn |
| lidFrenchLuxembourg | 0x140c | Latn |
| lidFrenchMonaco | 0x180c | Latn |
| lidFrenchWestIndies | 0x1c0c | Latn |
| lidFrenchReunion | 0x200c | Latn |
| lidFrenchCongoDRC | 0x240c | Latn |
| lidFrenchZaire | 0x240c | Latn |
| lidFrenchSenegal | 0x280c | Latn |
| lidFrenchCameroon | 0x2c0c | Latn |
| lidFrenchCotedIvoire | 0x300c | Latn |
| lidFrenchMali | 0x340c | Latn |
| lidFrenchHaiti | 0x3c0c | Latn |
| lidFrenchMorocco | 0x380c | Latn |
| lidIcelandic | 0x40f | Latn |
| lidItalianSwiss | 0x810 | Latn |
| lidRhaetoRomanic | 0x417 | Latn |
| lidRomanic | 0x417 | Latn |
| lidRussianMoldavia | 0x819 | Cyrl |
| lidCroat | 0x41a | Latn |
| lidSerbianLatinSerbMont | 0x81a | Latn |
| lidSerbianCyrillicSerbMont | 0xc1a | Cyrl |

| | | |
|---|---|---|
| lidCroatSerbo | 0x81a | Latn |
| lidBosniaHerzegovina | 0x101a | Latn |
| lidBosnianBosniaHerzegovinaLatin | 0x141a | Latn |
| lidSerbianBosniaHerzegovinaLatin | 0x181a | Latn |
| lidSerbianBosniaHerzegovinaCyrillic | 0x1c1a | Cyrl |
| lidBosnianBosniaHerzegovinaCyrillic | 0x201a | Cyrl |
| lidSerbianMontenegroLatin | 0x2c1a | Latn |
| lidSerbianMontenegroCyrillic | 0x301a | Cyrl |
| lidAlbanian | 0x41c | Latn |
| lidSwedishFinland | 0x81d | Latn |
| lidThai | 0x41e | Thai |
| lidUrdu | 0x420 | Arab |
| lidBahasa | 0x421 | Latn |
| lidIndonesian | 0x421 | Latn |
| lidByelorussian | 0x423 | Cyrl |
| lidTajik | 0x428 | Cyrl |
| lidVietnamese | 0x42a | Viet |
| lidArmenian | 0x42b | Armn |
| lidAzeriLatin | 0x42c | Latn |
| lidAzeriCyrillic | 0x82c | Cyrl |
| lidSorbian | 0x42e | Latn |
| lidLowerSorbian | 0x82e | Latn |
| lidMacedonian | 0x42f | Cyrl |
| lidGeorgian | 0x437 | Geor |
| lidFaeroese | 0x438 | Latn |
| lidHindi | 0x439 | Deva |
| lidMaltese | 0x43a | Latn |
| lidSamiLappish | 0x43b | Latn |
| lidNorthSamiSwe | 0x83b | Latn |
| lidNorthernSamiFi | 0xc3b | Latn |
| lidLuleSamiNor | 0x103b | Latn |
| lidLuleSamiSwe | 0x143b | Latn |
| lidSouthSamiNor | 0x183b | Latn |
| lidSouthSamiSwe | 0x1c3b | Latn |
| lidSkoltSami | 0x203b | Latn |
| lidInariSami | 0x243b | Latn |
| lidYiddish | 0x43d | Hebr |
| lidMalaysian | 0x43e | Latn |

| | | |
|---|---|---|
| lidMalayBrunei | 0x83e | Latn |
| lidKazakh | 0x43f | Cyrl |
| lidKyrgyz | 0x440 | Cyrl |
| lidSwahili | 0x441 | Latn |
| lidTurkmen | 0x442 | Latn |
| lidUzbekLatin | 0x443 | Latn |
| lidUzbekCyrillic | 0x843 | Cyrl |
| lidTatar | 0x444 | Cyrl |
| lidBengali | 0x445 | Beng |
| lidBengaliBangladesh | 0x845 | Beng |
| lidPunjabi | 0x446 | Guru |
| lidPunjabiPakistan | 0x846 | Arab |
| lidGujarati | 0x447 | Gujr |
| lidOriya | 0x448 | Orya |
| lidTamil | 0x449 | Taml |
| lidTelugu | 0x44a | Telu |
| lidKannada | 0x44b | Knda |
| lidMalayalam | 0x44c | Mlym |
| lidAssamese | 0x44d | Beng |
| lidMarathi | 0x44e | Deva |
| lidSanskrit | 0x44f | Deva |
| lidMongolian | 0x450 | Cyrl |
| lidMongolianMongo | 0x850 | Mong |
| lidTibetan | 0x451 | Tibt |
| lidBhutanese | 0x851 | Tibt |
| lidWelsh | 0x452 | Latn |
| lidKhmer | 0x453 | Khmr |
| lidLao | 0x454 | Laoo |
| lidBurmese | 0x455 | Mymr |
| lidKonkani | 0x457 | Deva |
| lidManipuri | 0x458 | Beng |
| lidSindhi | 0x459 | Deva |
| lidSindhiPakistan | 0x859 | Arab |
| lidSyriac | 0x45a | Syrc |
| lidSinhalese | 0x45b | Sinh |
| lidCherokee | 0x45c | Cher |
| lidInuktitut | 0x45d | Cans |
| lidInuktitutLatin | 0x85d | Latn |
| lidAmharic | 0x45e | Ethi |

| | | |
|---|---|---|
| lidTamazight | 0x45f | Arab |
| lidTamazightLatin | 0x85f | Latn |
| lidKashmiri | 0x460 | Arab |
| lidKashmiriIndia | 0x860 | Deva |
| lidNepali | 0x461 | Deva |
| lidNepaliIndia | 0x861 | Deva |
| lidFrisian | 0x462 | Latn |
| lidPashto | 0x463 | Arab |
| lidFilipino | 0x464 | Latn |
| lidMaldivian | 0x465 | Thaa |
| lidEdo | 0x466 | Latn |
| lidFulfulde | 0x467 | Latn |
| lidHausa | 0x468 | Latn |
| lidIbibio | 0x469 | Latn |
| lidYoruba | 0x46a | Latn |
| lidQuechuaBol | 0x46b | Latn |
| lidQuechuaEcu | 0x86b | Latn |
| lidQuechuaPe | 0xc6b | Latn |
| lidSesothoSaLeboa | 0x46c | Latn |
| lidBashkir | 0x46d | Cyrl |
| lidLuxembourgish | 0x46e | Latn |
| lidGreenlandic | 0x46f | Latn |
| lidIgbo | 0x470 | Latn |
| lidKanuri | 0x471 | Latn |
| lidOromo | 0x472 | Latn |
| lidTigrignaEthiopic | 0x473 | Ethi |
| lidTigrignaEritrea | 0x873 | Ethi |
| lidGuarani | 0x474 | Latn |
| lidHawaiian | 0x475 | Latn |
| lidLatin | 0x476 | Latn |
| lidSomali | 0x477 | Latn |
| lidYi | 0x478 | Yiii |
| lidMapudungun | 0x47a | Latn |
| lidMohawk | 0x47c | Latn |
| lidBreton | 0x47e | Latn |
| lidUighur | 0x480 | Uigh |
| lidMaori | 0x481 | Latn |
| lidOccitan | 0x482 | Latn |
| lidCorsican | 0x483 | Latn |

| | | |
|---|---|---|
| lidAlsatian | 0x484 | Latn |
| lidSakha | 0x485 | Cyrl |
| lidKiche | 0x486 | Latn |
| lidKinyarwanda | 0x487 | Latn |
| lidWolof | 0x488 | Latn |
| lidDari | 0x48c | Arab |

### Step 3: Search the font collections

After Office 2010 obtains a script tag, it searches the major and minor font tables for a match with the script tag it obtained. If it finds a matching tag in the font table, it uses the font specified for that script tag. But if it does not find a matching script tag in the font table, it moves on to the next step in the process.

### Step 4: Determine the default font

At this point, Office 2010 attempts to determine the default font for the text run. It does this by checking the Unicode values of the characters in the run. It uses the Unicode values to categorize the text into font groups. The categorization is performed by using the following table.

## Unicode sub-ranges and their corresponding font groups

| Unicode Character Range | Font Group |
|---|---|
| U+0000–U+007F | Latin |
| U+0080–U+00A6 | Latin |
| U+00A9–U+00AF | Latin |
| U+00B2–U+00B3 | Latin |
| U+00B5–U+00D6 | Latin |
| U+00D8–U+00F6 | Latin |
| U+00F8–U+058F | Latin |
| U+0590–U+074F | Complex Script |
| U+0780–U+07BF | Complex Script |
| U+0900–U+109F | Complex Script |
| U+10A0–U+10FF | Latin |
| U+1200–U+137F | Latin |
| U+13A0–U+177F | Latin |
| U+1D00–U+1D7F | Latin |
| U+1E00–U+1FFF | Latin |
| U+1780–U+18AF | Complex Script |
| U+2000–U+200B | Latin |
| U+200C–U+200F | Complex Script |
| U+2010–U+2029 | Latin* |
| U+202A–U+202F | Complex Script |
| U+2030–U+2046 | Latin |
| U+204A–U+245F | Latin |
| U+2670–U+2671 | Complex Script |

| | |
|---|---|
| U+27C0–U+2BFF | Latin |
| U+3099–U+309A | East Asian |
| U+D835 | Latin |
| U+F000–U+F0FF | Symbol. Use symbol font. |
| U+FB00–U+FB17 | Latin |
| U+FB1D–U+FB4F | Complex Script |
| U+FE50–U+FE6F | Latin |
| All others | East Asian |

* For the quote characters in the range U+2018 – U+201E, use the East Asian font if the text has one of the following language identifiers: ii-CN, ja-JP, ko-KR, zh-CN,zh-HK, zh-MO, zh-SG, zh-TW

After Office 2010 has determined which font group the characters belong to, it selects the default font for that group.

## Conclusion

Font selection in applications compatible with Office Open XML can be a complex process. Using the guidelines that are presented in the ISO 29500 standard for font selection is usually the best starting point for developers. When an appropriate font is not found, the application must perform an implementation-specific font substitution process. Developers are free to follow the algorithm presented here, which is used by Office 2010, or they can create their own.

### Comments