

Complex text layout

Complex text layout (**CTL**) or **complex text rendering** is the typesetting of writing systems in which the shape or positioning of a *grapheme* depends on its relation to other graphemes. The term is used in the field of software *internationalization*, where each grapheme is a *character*.

Scripts which require CTL for proper display may be known as **complex scripts**. Examples include the Arabic alphabet and scripts of the Brahmic family, such as Devanagari, Khmer script or the Thai alphabet. Many scripts do not require CTL. For instance, the Latin alphabet or Chinese characters can be typeset by simply displaying each character one after another in straight rows or columns. However, even these scripts have alternate forms or optional features (such as *cursive* writing) which require CTL to produce on computers.

Characteristics requiring CTL

The main characteristics of CTL complexity are:

- Bi-directional text, where characters may be written from either right-to-left or left-to-right direction.
- Context-sensitive shaping and *ligatures*, where a character may change its shape, dependent on its location and/or the surrounding characters. For example, a character in Arabic script can have as many as four different shape-forms, depending on context.
- Ordering, where the displayed order of the characters is not the same as the logical order. For example, in Devanagari, which is written from left to right, the grapheme for "short i" appears to the left of ("before") the consonant that it follows: in कि *ki*, the ि *-i* should render on the left, its bow reaching until above the क *k-* to the right.

Not all occurrences of these characteristics require CTL. For example, the Greek alphabet has context-sensitive shaping of the letter *sigma*, which appears as ς at the end of a word and σ elsewhere. However, these two forms are normally stored as different characters; for instance, Unicode has both U+03C2 ς GREEK SMALL LETTER FINAL SIGMA and U+03C3 σ GREEK SMALL LETTER SIGMA, and does not treat them as equivalent. For collation and comparison purposes, software should consider the string "ῥῖος Ἀχιλλεύς" equivalent to "ῥῖοσ Ἀχιλλεύς",^[1] but for typesetting purposes they are distinct and CTL is not required to choose the correct form.

Implementations

Most text-rendering software that is capable of CTL will include information about specific scripts, and so will be able to render them correctly without *font files* needing to supply instructions on how to lay out characters. Such software is usually provided in a *library*; examples include:

- Core Text for macOS
- Uniscribe (with Universal Shaping Engine) and DirectWrite for Microsoft Windows
- HarfBuzz, a cross-platform library
- Pango, a cross-platform library which nowadays incorporates HarfBuzz

However, such software is unable to properly render any script for which it lacks instructions, which can include many minority scripts. The alternative approach is to include the rendering instructions in the font file itself. Rendering software still needs to be capable of reading and following the instructions, but this is relatively simple.

Examples of this latter approach include Apple Advanced Typography (AAT) and Graphite. Both of these names encompass both the instruction format and the software supporting it; AAT is included on Apple operating systems, while Graphite is available for Microsoft Windows and Linux-based systems.

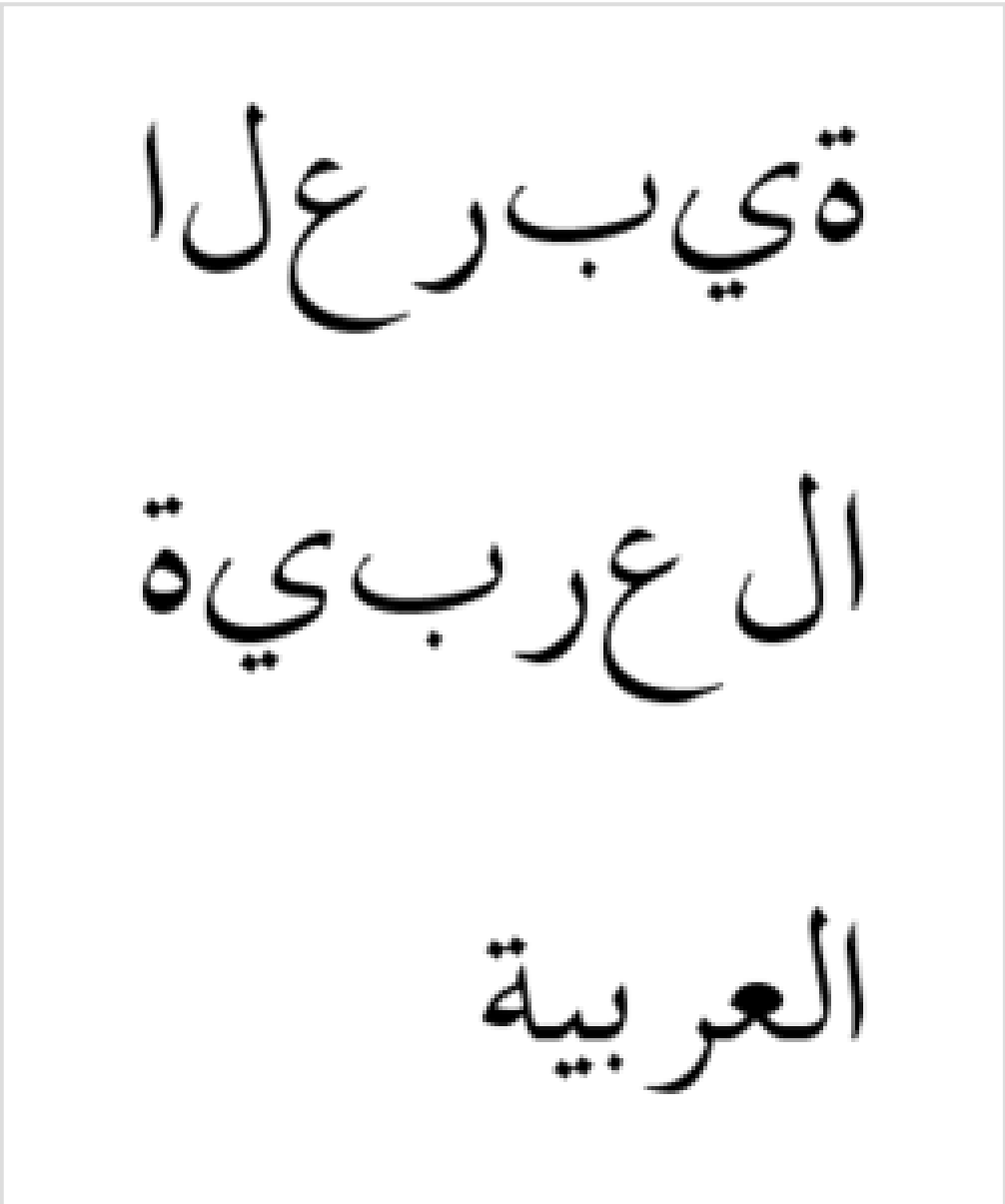
The OpenType format is primarily intended for systems using the first approach (layout knowledge in the renderer, not the font), but it has a few features that assist with CTL, such as contextual ligatures. AAT and Graphite instructions can be embedded in OpenType font files.

See also

- Typography
- Unicode
- Writing systems which require complex text layout:
 - Arabic alphabet
 - Most of the Brahmic family of scripts
 - N'Ko script



The Devanagari ddhrya-ligature, as displayed in the JanaSanskritSans (https://web.archive.org/web/20110716160603/http://tdil.mit.gov.in/download/janasanskrit.htm) font, which should be invoked by the layout engine to render the sequence द + ष् + ध + ष् + र + ष् + य = दध्य.


 The word العربية *al-arabiyyah*, "the Arabic [language]" in Arabic, in successive stages of rendering. The first line shows the letters in left-to-right order and unjoined, as they might appear in an application without complex text layout. In the second line, bidirectional display has been applied, and in the third the glyph-shaping mechanism has rendered the letters according to context.

- [Tengwar](#) (diacritics and numbers)

References

- ↑ "FAQ - Greek Language & Script" (<https://www.unicode.org/faq/greek.html#5>). Unicode Consortium. 2012-12-03. Retrieved 2013-09-13. "It is easier to simply equate the two sigma codes for operations which are concerned with word content, for example."

External links

- Examples of complex rendering (http://scripts.sil.org/cms/scripts/page.php?site_id=nrsi&item_id=CmplxRndExamples) — [SIL international](#)'s examples of complex writing systems around the world
- <http://www.opengroup.org/desktop/ctl/> — [The Open Group's Desktop Technologies](#)
- <https://web.archive.org/web/20050206102353/http://www.mozilla.org/projects/ctl/> — also other CTL scripts
- <https://web.archive.org/web/20121020041740/http://sila.mozdev.org/> — [Graphite and Mozilla integration project](#)
- http://developers.sun.com/techtopics/global/products_platforms/solaris/reference/whitepapers/#ctl — [Solaris Globalization Whitepapers](#)
- <http://www.microsoft.com/globaldev/DrIntl/faqs/Complex.aspx> — [Microsoft Global Development and Computing Portal](#)
- <http://linux.thai.net/~thep/> — information about Thai language processing
- <http://www.freedesktop.org/wiki/Software/HarfBuzz> at [Freedesktop.org](#)
- http://www.d-type.com/unicode_text/ — [D-Type Unicode Text Module](#) — Portable software library for complex text
- <https://github.com/salshaaban/BidiRenderer> — An application that illustrates the shaping and layout of complex text in bidirectional paragraphs using FriBidi, FreeType, and HarfBuzz
- <https://github.com/Tehreer/Tehreer-Android> — A library that gives full control over text related technologies such as bidirectional algorithm, open type shaping, text typesetting and text rendering
- <https://github.com/Tehreer/Tehreer-Cocoa> — [Standalone font/text engine for iOS](#)

Retrieved from "https://en.wikipedia.org/w/index.php?title=Complex_text_layout&oldid=1168211785"

-