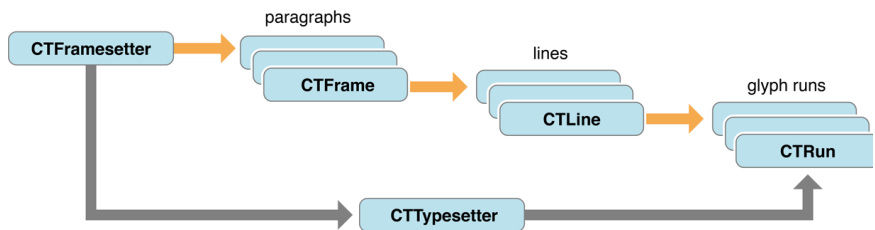


# About Core Text

Core Text is an advanced, low-level technology for laying out text and handling fonts. The Core Text API, introduced in Mac OS X v10.5 and iOS 3.2, is accessible from all OS X and iOS environments.

**Important:** Core Text is intended for developers who must do text layout and font handling at a low level, such as developers of layout engines. You should develop your app using a higher-level framework if possible—that is, use Text Kit in iOS (see *Text Programming Guide for iOS*) or the Cocoa text system in OS X (see *Cocoa Text Architecture Guide*). Core Text is the technology underlying those text systems, so they share in its speed and efficiency. In addition, Text Kit and the Cocoa text system provide rich text editing, full-featured page layout engines, and other infrastructure that your app would otherwise need to provide if it used Core Text alone.



## At a Glance

Core Text is for apps that need a low-level text-handling technology correlating with the Core Graphics framework (Quartz). If you work directly with Quartz and you need to draw some text, use Core Text. If, for example, you have your own page layout engine—you have some text and you know where it needs to go in your view—you can use Core Text to generate the glyphs and position them relative to each other with all the features of fine typesetting, such as [kerning](#), [ligatures](#), [line-breaking](#), [hyphenation](#), and [justification](#).

## Core Text Lays Out Text

Core Text generates glyphs (from character codes and font data) and positions them relative to each other in glyph runs. It breaks glyph runs into lines, and it assembles lines into multiline frames (such as paragraphs). Core Text also provides glyph- and layout-related data, such as glyph locations and measurement of lines and frames. It handles character attributes and paragraph styles, including various types of tab styles and positioning.

**Relevant Chapters:** Core Text Overview, Common Text Layout Operations

## You Can Manage Fonts With Core Text

The Core Text font API provides fonts, font collections, font descriptors, and easy access to font data. It also provides support for multiple master fonts, font variations, font cascading, and font linking. Core Text provides an alternative to Quartz for loading your own fonts into the current process, that is, font activation.

**Relevant Chapters:** Common Font Operations

## Prerequisites

To make good use of this document, you should have an understanding of text systems and issues, and you should know how to use Core Foundation opaque types. For information about Core Foundation, see *Core Foundation Design Concepts*.

## See Also

In addition to this document, there are several that cover more specific aspects of Core Text or describe the software services used by Core Text.

- [Core Text Reference Collection](#) provides complete reference information for the Core Text layout and font API.
- *CoreTextPageViewer* (in the iOS Developer Library) shows how to use Core Text to display large bodies of text.
- *DownloadFont* (in the iOS Developer Library) demonstrates how to download fonts on demand.
- *CoreTextRTF* (in the Mac Developer Library) shows how to use Core Text to lay out and draw RTF content in a window of a Cocoa application.
- *Drawing Along a Path Using Core Text with Cocoa* (in the Mac Developer Library) shows how to use Core Text to lay out and draw glyphs along a curve.
- *Core Foundation Design Concepts* and *Core Foundation Framework Reference* describe Core Foundation, a framework that provides abstractions for common data types and fundamental software services used by Core Text.

The following chapters (in the iOS Developer Library) describe Text Kit in iOS:

- Drawing and Managing Text in *Text Programming Guide for iOS* describes the app-level text handling system in iOS.
- For information about typographic concepts relevant to Core Text and other text systems, see *Typographical Concepts* in *Text Programming Guide for iOS*.

The following documents (in the Mac Developer Library) provide entry points to the documentation describing the Cocoa text system in OS X:

- *Cocoa Text Architecture Guide* gives an introduction to the Cocoa text system.
- *Text Layout Programming Guide* describes the Cocoa text layout engine.