

[AppKit](#) / [Writing Tools](#) / Enhancing your custom text engine with Writing Tools

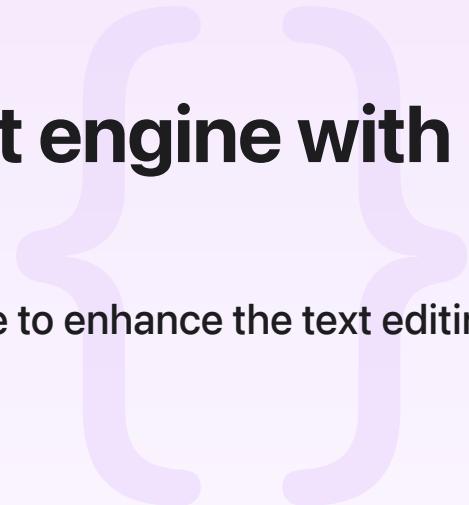
Sample Code

Enhancing your custom text engine with Writing Tools

Add Writing Tools support to your custom text engine to enhance the text editing experience.

[Download](#)

macOS 15.4+ | Xcode 16.4+



Overview

The systems provide easy-to-use and highly capable APIs for text editing, such as [NSTextView](#), [UITextView](#), and [SwiftUI TextEditor](#). These APIs handle text rendering, text input, and user interactions, support multiple languages, and provide many features like spell checking and Writing Tools. Apps generally use these APIs to implement text editing.

In some cases, apps may desire to build a custom text editing experience by implementing a custom text engine and integrating the editor with system-provided features, such as Writing Tools. The sample app demonstrates how to implement a basic [NSTextInputClient](#) view with Writing Tools support.

Note

This sample code project is associated with WWDC25 session 265: [Dive deeper into Writing Tools](#).

Configure the sample code project

To configure the sample code project, do the following in Xcode:

1. Open the sample with the latest version of Xcode.
 2. Set the developer team to let Xcode automatically manage the provisioning profile. For more information, see [Set the bundle ID](#) and [Assign the project to a team](#).
-

See Also

Writing Tools for custom views

- Supporting Writing Tools via the pasteboard
Adopt a simplified version of the Writing Tools experience in a custom view using the pasteboard and macOS services.
- Adding Writing Tools support to a custom AppKit view
Integrate Writing Tools support, including support for inline replacement animations, to your custom text views on macOS.

`class NSWritingToolsCoordinator`

An object that manages interactions between Writing Tools and your custom text view.

`protocol Delegate`

An interface that you use to manage interactions between Writing Tools and your custom text view.

`class Context`

A data object that you use to share your custom view's text with Writing Tools.

`class AnimationParameters`

An object you use to configure additional tasks or animations to run alongside the Writing Tools animations.