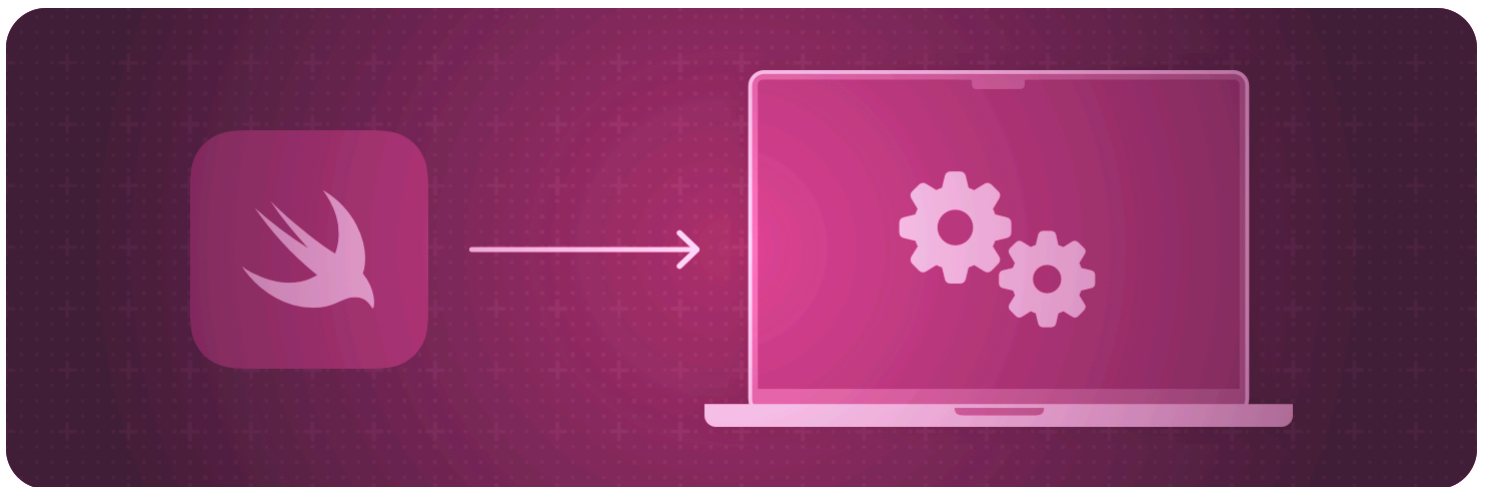API Collection

AppKit integration

Add AppKit views to your SwiftUI app, or use SwiftUI views in your AppKit app.

Overview

Integrate SwiftUI with your app's existing content using hosting controllers to add SwiftUI views into AppKit interfaces. A hosting controller wraps a set of SwiftUI views in a form that you can then add to your storyboard-based app.

You can also add AppKit views and view controllers to your SwiftUI interfaces. A representable object wraps the designated view or view controller, and facilitates communication between the wrapped object and your SwiftUI views.

For design guidance, see Designing for macOS in the Human Interface Guidelines.

Topics

Displaying SwiftUI views in AppKit

📄 **Unifying your app's animations**

Create a consistent UI animation experience across SwiftUI, UIKit, and AppKit.

`class` `NSHostingController`

An AppKit view controller that hosts SwiftUI view hierarchy.

`class` `NSHostingView`

An AppKit view that hosts a SwiftUI view hierarchy.

`class` `NSHostingMenu`

An AppKit menu with menu items that are defined by a SwiftUI View.

`struct` `NSHostingSizingOptions`

Options for how hosting views and controllers reflect their content's size into Auto Layout constraints.

`class` `NSHostingSceneRepresentation`

An AppKit type that hosts and can present SwiftUI scenes

`struct` `NSHostingSceneBridgingOptions`

Options for how hosting views and controllers manage aspects of the associated window.

Adding AppKit views to SwiftUI view hierarchies

`protocol` `NSViewRepresentable`

A wrapper that you use to integrate an AppKit view into your SwiftUI view hierarchy.

`struct` `NSViewRepresentableContext`

Contextual information about the state of the system that you use to create and update your AppKit view.

`protocol` `NSViewControllerRepresentable`

A wrapper that you use to integrate an AppKit view controller into your SwiftUI interface.

`struct` `NSViewControllerRepresentableContext`

Contextual information about the state of the system that you use to create and update your AppKit view controller.

Adding AppKit gesture recognizers into SwiftUI view hierarchies

`protocol NSGestureRecognizerRepresentable`

A wrapper for an `NSGestureRecognizer` that you use to integrate that gesture recognizer into your SwiftUI hierarchy.

`struct NSGestureRecognizerRepresentableContext`

Contextual information about the state of the system that you use to create and update a represented gesture recognizer.

`struct NSGestureRecognizerRepresentableCoordinateSpaceConverter`

A structure used to convert locations to and from coordinate spaces in the hierarchy of the SwiftUI view associated with an <u>NSGestureRecognizerRepresentable</u>.

See Also

Framework integration

☰ UIKit integration

Add UIKit views to your SwiftUI app, or use SwiftUI views in your UIKit app.

☰ WatchKit integration

Add WatchKit views to your SwiftUI app, or use SwiftUI views in your WatchKit app.

☰ Technology-specific views

Use SwiftUI views that other Apple frameworks provide.