

Framework

AccessoryTransportExtension Beta

Exchange sensitive information with a connected accessory.

iOS 26.2+ Beta | iPadOS 26.2+ Beta | Mac Catalyst 26.2+ Beta

Overview

Use the Accessory Transport Extension framework to send sensitive information to accessories you've discovered and set up with [AccessorySetupKit](#). You can use the [Wi-Fi Infrastructure](#) framework if you need to share a WiFi network to the accessory.

Note

The Accessory Transport Extension framework is available for iOS. For apps built with Catalyst for macOS, and for iOS apps running on visionOS or on Mac computers with Apple silicon, calls to the framework do nothing.

You implement this functionality in an app extension that conforms to the [AccessoryTransportAppExtension](#) protocol. To add an accessory transport app extension, select your project from the Xcode project navigator, create a new target with File > New > Target, and choose the "Accessory Transport" template.

The framework calls your extension's [accept\(sessionRequest:\)](#) method when the accessory is ready to begin a transport session. Implement this method to inspect the received [AccessoryTransportSession.Request](#) and then call [accept\(_:\)](#) or [reject\(error:\)](#) on the request.

After you accept a session from an accessory, your extension is responsible for directly connecting to the accessory and delivering data to it. The code included in the Accessory Transport template provides an event handler that creates an [ASAccessorySession](#) and responds to accessory activation by setting up a [WINetworkSharingProvider](#) for the accessory. You can use this to send data to the accessory.

Important

You may develop and test Accessory Transport Extension apps on devices in all regions. People using your app must have an account registered in the European Union (EU), and their device must be located within the EU.

Topics

App extensions

`protocol` `AccessoryTransportAppExtension`

A protocol that defines the behavior of the app extension and how it handles requests.

`protocol` `AccessoryTransportExtensionConfiguration`

An interface you use to configure and manage communication between the extension and the host process.

Transport sessions

`class` `AccessoryTransportSession`

A class that manages a session between the extension and host process.

Beta Software

This documentation contains preliminary information about an API or technology in development. This information is subject to change, and software implemented according to this documentation should be tested with final operating system software.

[Learn more about using Apple's beta software >](#)