API Collection

App execution

Control app launch, execution, and termination.

Overview

Your app interacts with the system during normal execution by calling system APIs. However, you need to communicate information about how to execute your app before you have access to these API calls. For example, you may need to specify under what conditions your app can launch, the environment that it should launch into, and what should happen when it terminates. You add keys to your app's Information Property List file to manage its execution.

Topics

Launch

NSPrincipalClass

The name of the bundle's main executable class.
Name: Principal class

CLKComplicationPrincipalClass

The name of the class that implements the complication data source protocol.
Name: ClockKit Complication - Principal Class

CFBundleExecutable

The name of the bundle's executable file.
Name: Executable file

LSEnvironment

Environment variables to set before launching the app.
Name: Environment variables

UIApplicationShortcutItems

Launch conditions

UIRequiredDeviceCapabilities

The device-related features that your app requires to run.
Name: Required device capabilities

LSMultipleInstancesProhibited

A Boolean value indicating whether more than one user can launch the app simultaneously.
Name: Application prohibits multiple instances

LSArchitecturePriority

An array of the architectures that the app supports, arranged according to their preferred usage.
Name: Architecture priority

LSRequiresNativeExecution

A Boolean value that indicates whether to require the execution of the app's native architecture when multiple architectures are available.
Name: Application requires native environment

WKPrefersNetworkUponForeground

A Boolean value that indicates whether an app requires network access on launch.

WKRunsIndependentlyOfCompanionApp

A Boolean value indicating whether the user can install and run the watchOS app independently of its iOS companion app.
Name: App can run independently of companion iPhone app

WKWatchOnly

A Boolean value indicating whether the app is a watch-only app.
Name: App is only available as a standalone watchOS app

PUICAutoLaunchAudioOptOut

A Boolean value that indicates whether a watchOS app should opt out of automatically launching when its companion iOS app starts playing audio content.
Name: Opt out of Auto-launch Audio App (Watch)

~~CLKComplicationSupportedFamilies~~

The complication families for which the app can provide data.
Name: ClockKit Complication - Supported Families

`Deprecated`

Extensions and services

NSExtension

The properties of an app extension.

NSServices

The services provided by an app.
Name: Services

WKExtensionDelegateClassName

The name of your watchOS app's extension delegate.

UIApplicationShortcutWidget

The bundle ID of the widget that's available as a Home screen quick action in apps that have more than one widget.
Name: Home Screen Widget

App clips

NSAppClip

A collection of keys that an App Clip uses to get additional capabilities.
Name: App Clip

Background execution

UIBackgroundModes

Services provided by an app that require it to run in the background.
Name: Required background modes

WKBackgroundModes

The services a watchOS app provides that require it to continue running in the background.
Name: Required background modes (Watch)

BGTaskSchedulerPermittedIdentifiers

An array of strings containing developer-specified task identifiers in reverse URL notation.

Name: Permitted background task scheduler identifiers

`LSBackgroundOnly`

A Boolean value indicating whether the app runs only in the background.
Name: Application is background only

Endpoint security

`NSEndpointSecurityEarlyBoot`

`NSEndpointSecurityRebootRequired`

Plug-in support

`NSDockTilePlugIn`

The name of the app's plug-in bundle.
Name: Dock Tile plugin path

Plug-in configuration

`CFPlugInDynamicRegisterFunction`

The function to use when dynamically registering a plug-in.
Name: Plug-in dynamic registration function name

`CFPlugInDynamicRegistration`

A Boolean value indicating whether the host loads this plug-in.
Name: Plug-in should be registered dynamically

`CFPlugInFactories`

The interfaces supported by the plug-in for static registration.
Name: Plug-in factory interfaces

`CFPlugInTypes`

One or more groups of interfaces supported by the plug-in for static registration.
Name: Plug-in types

`CFPlugInUnloadFunction`

The name of the function to call to unload the plug-in code from memory.
Name: Plug-in unload function name

Termination

`LSGetAppDiedEvents`

A Boolean value indicating whether the app is notified when a child process dies.

Name: Application should get App Died events

`NSSupportsSuddenTermination`

A Boolean value indicating whether the system may terminate the app to log out or shut down more quickly.

Name: Application can be killed immediately after launch

~~UIApplicationExitsOnSuspend~~

A Boolean value indicating whether the app terminates, rather than moves to the background, when the app quits.

Name: Application does not run in background

Deprecated

Logging

`OSLogPreferences`

A dictionary that defines the logging configuration profile for the app.

See Also

Core settings

≔ Bundle configuration

Define basic characteristics of a bundle, like its name, type, and version.

≔ User interface

Configure an app's scenes, storyboards, icons, fonts, and other user interface elements.