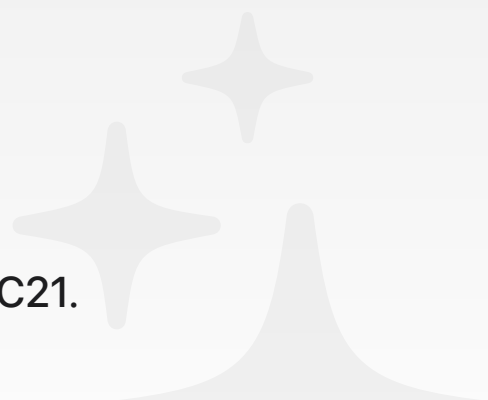


[Updates](#) / WWDC21

# WWDC21

Highlights of new technologies introduced at WWDC21.



## Overview

Newer documentation highlights are available in [WWDC22](#). This page is an archive from WWDC21.

Check out a selection of documentation for new technologies, frameworks, and APIs introduced at WWDC21. Existing frameworks have added significant functionality, and you'll find new ways to enhance your apps targeting the latest platform release.

## Topics

### Xcode Cloud



#### Xcode Cloud

Automatically build, test, and distribute your apps with Xcode Cloud to verify changes and create high-quality apps.



#### About continuous integration and delivery with Xcode Cloud

Learn how continuous integration and delivery with Xcode Cloud helps you create high-quality apps and frameworks.



#### Configuring your first Xcode Cloud workflow

Set up your project or workspace to use Xcode Cloud and adopt continuous integration and delivery.

### SwiftUI

`{}` Building a great Mac app with SwiftUI

Create engaging SwiftUI Mac apps by incorporating side bars, tables, toolbars, and several other popular user interface elements.

`{}` Add rich graphics to your SwiftUI app

Make your apps stand out by adding background materials, vibrancy, custom graphics, and animations.

```
struct TimelineView<Schedule, Content> where Schedule : TimelineSchedule
```

A view that updates according to a schedule that you provide.

```
struct AsyncImage<Content> where Content : View
```

A view that asynchronously loads and displays an image.

```
@frozen @propertyWrapper struct FocusState<Value> where Value : Hashable
```

A property wrapper type that can read and write a value that SwiftUI updates as the placement of focus within the scene changes.

```
struct Table<Value, Rows, Columns> where Value == Rows.TableRowValue, Rows : TableRowContent, Columns : TableColumnContent, Rows.TableRowValue == Columns.TableRowValue
```

A container that presents rows of data arranged in one or more columns, optionally providing the ability to select one or more members.

```
struct Canvas<Symbols> where Symbols : View
```

A view type that supports immediate mode drawing.

```
struct Material
```

A background material type.

```
nonisolated func swipeActions<T>(edge: HorizontalEdge = .trailing, allowsFullSwipe: Bool = true, @ViewBuilder content: () -> T) -> some View where T : View
```

Adds custom swipe actions to a row in a list.

```
nonisolated func badge(_ key: LocalizedStringKey?) -> some View
```

Generates a badge for the view from a localized string key.

```
nonisolated func searchable(text: Binding<String>, placement: SearchFieldPlacement = .automatic, prompt: Text? = nil) -> some View
```

Marks this view as searchable, which configures the display of a search field.

```
nonisolated func listRowSeparatorTint(_ color: Color?, edges: VerticalEdge.Set = .all) -> some View
```

Sets the tint color associated with a row.

```
nonisolated func previewInterfaceOrientation(_ value: InterfaceOrientation) -> some View
```

Overrides the orientation of the preview.

```
nonisolated func symbolVariant(_ variant: SymbolVariants) -> some View
```

Makes symbols within the view show a particular variant.

```
nonisolated func symbolRenderingMode(_ mode: SymbolRenderingMode?) -> some View
```

Sets the rendering mode for symbol images within this view.

## SharePlay and Group Activities



### Group Activities

Create app-specific activities your users can share and experience together.

## DocC



### DocC

Produce rich API reference documentation and interactive tutorials for your app, framework, or package.

## Notifications



### User Notifications

Push user-facing notifications to the user's device from a server, or generate them locally from your app.

## WatchKit



### Interacting with Bluetooth peripherals during background app refresh

Keep your complications up-to-date by reading values from a Bluetooth peripheral while your app is running in the background.

## Accessibility



### Audio graphs

Define an accessible representation of your chart for VoiceOver to generate an audio graph.



### Hearing device support

Access information about paired hearing aid devices and streaming status.

## Extensions



### MailKit

Secure, customize, and act on email messages that users send and receive.



### Safari web extensions

Create web extensions that work in Safari and other browsers.

```
class EKVirtualConferenceProvider
```

An object that associates virtual conferencing details with an event object in a user's calendar.



### Network Extension

Customize and extend core networking features.

## App Store



### StoreKit

Support In-App Purchases and interactions with the App Store.



### In-App Purchase

Offer content and services in your app across Apple platforms using a Swift-based interface.

```
struct Transaction
```

Information that represents the customer's purchase of a product in your app.



### App Store Connect API

Automate the tasks you perform on the Apple Developer website and in App Store Connect.



### App Store Server Notifications

Monitor In-App Purchase events in real time and learn of unreported external purchase tokens, with server notifications from the App Store.



## App Store Server API

Manage your customers' App Store transactions from your server.

# Graphics



## Metal

Render advanced 3D graphics and compute data in parallel with graphics processors.



## Media Player

Find and play songs, audio podcasts, audio books, and more from within your app.

`class` `AVCaption`

An object that represents text to present over a time range.

`class` `AVCaptureDevice`

An object that represents a hardware or virtual capture device like a camera or microphone.



## Recording and Streaming Your macOS App

Share screen recordings, or broadcast live audio and video of your app, by adding ReplayKit to your macOS apps and games.

# Audio and Haptics



## MusicKit

Integrate your app with Apple Music.



## AudioDriverKit

Develop drivers for audio devices.



## Classifying Live Audio Input with a Built-in Sound Classifier

Detect and identify hundreds of sounds by using a trained classifier.



## Core Haptics

Compose and play haptic patterns to customize your iOS app's haptic feedback.

# Screen Time API



## ManagedSettings

Access and change settings with your app while maintaining user privacy and control.



## ManagedSettingsUI

Define and configure the appearance of shielding views.



## DeviceActivity

Monitor device activity with your app extension while maintaining user privacy.



## FamilyControls

Authorize your app to provide parental controls on a device.

# AppKit



## TextKit

Manage text storage and perform custom layout of text-based content in your app's views.

# UIKit

## Catalyst

```
@MainActor class UISheetPresentationController
```

A presentation controller that manages the appearance and behavior of a sheet.

```
struct Configuration
```

A configuration that specifies the appearance and behavior of a button and its contents.

```
func prepareForDisplay(completionHandler: @escaping (UIImage?) -> Void)
```

Decodes an image asynchronously and provides a new one for display in views and animations.

```
func prepareThumbnail(of size: CGSize, completionHandler: @escaping (UIImage?) -> Void)
```

Creates a thumbnail image at the specified size asynchronously on a background thread.



## CoreLocationUI

Streamline access to users' location data through a standard, secure UI.

```
enum UIBehavioralStyle
```

Constants that indicate how a control behaves in apps built with Mac Catalyst.

```
UIApplicationSupportsPrintCommand
```

A Boolean value that indicates whether the app supports the Command-P keyboard shortcut.

## UIApplicationSupportsTabbedSceneCollection

A Boolean value indicating whether an app built with Mac Catalyst supports automatic tabbing mode.

```
@MainActor var subtitle: String { get set }
```

A string that the app displays in the title bar of a window when running in macOS.

## Security and Privacy

### Public-Private Key Authentication

Register and authenticate users with passkeys and security keys, without using passwords.

### Customizing the notarization workflow

Notarize your app from the command line to handle special distribution cases.

```
@MainActor class LAAuthenticationView
```

A graphical representation of the state of biometric authentication.

### Exposure Notification

Implement a COVID-19 exposure notification system that protects user privacy.

## iCloud

### Shared Records

Share one or more records with other iCloud users.

```
@NSCopying var encryptedValues: any CKRecordKeyValueSetting & Sendable  
{ get }
```

An object that manages the record's encrypted key-value pairs.

### Integrating a Text-Based Schema into Your Workflow

Define and update your schema with the CloudKit Schema Language.

## Core Data

```
class NSPersistentCloudKitContainer
```

A container that encapsulates the Core Data stack in your app, and mirrors select persistent stores to a CloudKit private database.

```
var allowsCloudEncryption: Bool { get set }
```

A Boolean value that determines whether to encrypt the attribute's value.

```
class NSCoreDataCoreSpotlightDelegate
```

A set of methods that enable integration with Core Spotlight.

```
@MainActor @propertyWrapper @preconcurrency struct SectionedFetchRequest<SectionIdentifier, Result> where SectionIdentifier : Hashable, Result : NSFetchRequestResult
```

A property wrapper type that retrieves entities, grouped into sections, from a Core Data persistent store.

## Machine Learning



TabularData

Import, organize, and prepare a table of data to train a machine learning model.

```
struct MLShapedArray<Scalar> where Scalar : MLShapedArrayScalar
```

A machine learning collection type that stores scalar values in a multidimensional array.



Applying Matte Effects to People in Images and Video

Generate image masks for people automatically by using semantic person-segmentation.

```
class VNGeneratePersonSegmentationRequest
```

An object that produces a matte image for a person it finds in the input image.

## Foundation

```
@dynamicMemberLookup struct AttributedString
```

A value type for a string with associated attributes for portions of its text.



Data Formatting

Convert numbers, dates, measurements, and other values to and from locale-aware string representations.

```
struct Morphology
```

A description of the grammatical properties of a string.

## Developer Tools



MetricKit



Aggregate and analyze per-device reports on exception and crash diagnostics and on power and performance metrics.

## HealthKit

`class HKVerifiableClinicalRecord`

A sample that represents the contents of a SMART Health Card or EU Digital COVID Certificate.

## HomeKit



HomeKit

Configure, control, and communicate with home automation accessories.

## Siri



SiriKit

Empower users to interact with their devices through voice, intelligent suggestions, and personalized workflows.

## Games



GameKit

Enable players to interact with friends, compare leaderboard ranks, earn achievements, and participate in multiplayer games.



Game Controller

Support hardware game controllers in your game.

## Apple Pay

```
dictionary ApplePayLineItem { ApplePayLineItemType type; DOMString
label; DOMString amount; ApplePayPaymentTiming paymentTiming; Date
recurringPaymentStartDate; ApplePayRecurringPaymentDateUnit recurring
PaymentIntervalUnit; long recurringPaymentIntervalCount; Date recurring
PaymentEndDate; Date deferredPaymentDate; DOMString automaticReload
PaymentThresholdAmount;};
```

A line item in a payment request—for example, total, tax, discount, or grand total.

`boolean supportsCouponCode;`

A Boolean value that determines whether the payment sheet displays the coupon code field.

`DOMString couponCode;`

The initial coupon code for the payment request.

`ApplePayShippingContactEditingMode shippingContactEditingMode;`

A value that indicates whether the shipping mode prevents the user from editing the shipping address.

`attribute EventHandler oncouponcodechanged;`

An event handler called by the system when the user enters or updates a coupon code.

`boolean supportsCouponCode;`

A Boolean value that determines whether the payment sheet displays the coupon code field.

`DOMString couponCode;`

The initial coupon code for the payment request.

`ApplePayShippingContactEditingMode shippingContactEditingMode;`

A value that indicates if the shipping mode prevents the user editing the shipping address.

`interface PaymentMethodChangeEvent`

The Apple Pay extensions to the Payment Request payment change event.

`dictionary ApplePayModifier { ApplePayPaymentMethodType paymentMethodType; ApplePayLineItem total; sequence <ApplePayLineItem> additionalLineItems; sequence <ApplePayShippingMethod> additionalShippingMethods; sequence <ApplePayPaymentTokenContext> multiTokenContexts; ApplePayAutomaticReloadPaymentRequest automaticReloadPaymentRequest; ApplePayRecurringPaymentRequest recurringPaymentRequest; ApplePayDeferredPaymentRequest deferredPaymentRequest;};`

A dictionary that defines the Apple Pay modifiers for a payment type in the W3C Payment Request API.

`{}` Offering Apple Pay in Your App

Collect payments with iPhone and Apple Watch using Apple Pay.

`class PKDeferredPaymentSummaryItem`

An object that defines a summary item for a payment that occurs at a later date, such as a pre-order.

`class PKRecurringPaymentSummaryItem`

An object that defines a summary item for a payment that occurs repeatedly at a specified interval, such as a subscription.

`var supportsCouponCode: Bool { get set }`

A Boolean value that determines whether the payment sheet displays the coupon code field.

`var couponCode: String? { get set }`

The initial coupon code for the payment request.

`protocol PKPaymentAuthorizationControllerDelegate : NSObjectProtocol`

Methods that let you respond to user interactions with your payment authorization controller.

## Hardware



Nearby Interaction

Locate and interact with nearby devices using identifiers, distance, and direction.



Hypervisor

Build virtualization solutions on top of a lightweight hypervisor, without third-party kernel extensions.



SensorKit

Retrieve data and derived metrics from sensors on an iPhone, or paired Apple Watch.



DriverKit sample code

Explore projects that demonstrate how to write macOS device drivers with the DriverKit family of frameworks.

## ShazamKit



ShazamKit

Find information about a specific audio recording when a segment of it's part of captured sound in the Shazam catalog or your custom catalog.

## Photos



Delivering an Enhanced Privacy Experience in Your Photos App

Adopt the latest privacy enhancements to deliver advanced user-privacy controls.

 **Selecting Photos and Videos in iOS**

Improve the user experience of finding and selecting assets by using the Photos picker.

`struct PHPickerConfiguration`

An object that contains information about how to configure a picker view controller.

## Education

`class AEAssessmentConfiguration`

Configuration information for an assessment session.

## TVUIKit

 **TVUIKit**

Show common user interface elements from Apple TV in your native app.

## WidgetKit

 **Increasing the visibility of widgets in Smart Stacks**

Provide contextual information and donate intents to the system to make sure your widget appears prominently in Smart Stacks.

---

## See Also

### WWDC

 **WWDC25**

Highlights of new technologies introduced at WWDC25.

 **WWDC24**

Highlights of new technologies introduced at WWDC24.

 **WWDC23**

Highlights of new technologies introduced at WWDC23.

 **WWDC22**

Highlights of new technologies introduced at WWDC22.