App Store

* インスタンスのプロパティTransactionとrenewalInfoがjwsRepresentationになっています。
* メソッドのProduct名前がに変更されましたproducts(for:)。
* 「Transaction」、「Product.SubscriptionInfo.Status」および「Storefront」のリスナータイププロパティが更新されました。 入れ子になったAsyncSequence適合構造は、それぞれTransactions、Statuss、Storefrontsになります。 「TransactionSequence」と「TransactionListener」の両方が「TransactionSequenceとTransactionListenerはTransactionsです。
* StoreKitError.userDidNotAuthenticate使用できなくなりました; 代わりにStoreKitError.userCancelled 使用してください。
* Subscription更新は、AppStoreサンドボックス環境のStoreKit2APIを介してアプリに自動的にプッシュされることはありません。Transactionまた、更新情報がすぐに利用できない場合があります。

Audio Units

* 「AudioUnits」は、「AudioUnit」ホストがiOSに表示できるカスタムビューを提供します。 「AUAudioUnit」にユーザインタフェースがあるかどうかを確認するには、「providesUserInterface」プロパティを使用します。 「requestViewController(completionHandler:)」メソッドを使用して、「ビュー」の「AUViewController」を取得します。 ユーザー定義ビューは、「tintColor」属性によるビューの色調設定をサポートします。 これは、ビューの色をトラックごとに異なる色に設定したり、アプリの外観と一致させるために使用できます。

Core Haptics

* 「 CHHapticEventTypeAudioContinuous 」、「 CHHapticEventTypeHapticContinuous 」、および「 CHHapticEventTypeAudioCustom 」タイプのイベントは、一時停止された「 CHHapticAdvancedPatternPlayer 」が再開されると、イベント中に再生が再開されます。 「シーク（toOffset:）」が特定のタイムオフセットでプレーヤーを起動する場合、これらのイベントはイベントの途中で開始されません。
* これで、ボリュームエンベロープを「type」リソースに適用するかどうかを制御できます。 基本的に、これらのリソースは、クリックを避けるために、信号を最初にランプインし、最後にランプアウトする内蔵ボリュームエンベロープで再生されます。
  + オーディオリソースIDを登録してカスタムオーディオ資産をインポートする場合、システムが「 registerAudioResource(\_:options:). 」に渡す新しいキー値引数「 CHHapticAudioResourceKeyUseVolumeEnvelope 」を使用してこの動作を指定できます。
  + AHAPファイルまたは「 CHHapticPattern 」の「 initWithDictionary:error: 」を使用してオーディオ資産を参照する場合、「 CHHapticPatternKeyEventWaveformUseVolumeEnvelope 」パターンキーを使用してこの動作を制御できます。

Create ML

* CreateMLフレームワークは現在iOSとiPadOS15ベータで使用可能であり、オンデバイスMLを活用する動的アプリ環境を構築する新しい機会を開拓する。
* オーディオ機能プリントベースのMLSoundClassifierアルゴリズムは、より高い精度、低い遅延、および小さいモデルサイズで、サウンド分類器モデルをより速くトレーニングします。 これがCreateMLのMLSoundClassifierのデフォルトアルゴリズムオプションになりました

Foundation

* foundation には自動文法合意エンジンが搭載されている。 これにより、コードが簡素化され、複数化、文法的な性別の同意、およびユーザのアドレス条件への同意を説明するためにローカライズされた文字列が自動的に屈折することによって、提供されるローカライズされた文字列の数が削減されます。 それは英語とスペイン語で利用できる.
* フォーマットAPIが使用可能になり、フォーマットに重点を置き、フォーマットインスタンスを作成、設定、およびキャッシュする必要がなくなります。 各フォーマッタタイプには、フォーマットされた機能があります。 これらの関数には、スタイルの設定とカスタマイズを可能にする引数があります。
* JSONSerialization とJSONDecoder は現在JSON5からのデコードをサポートしている。
* SortDescriptor、KeyPathComparator、およびSortComparatorAPIは、値をソートするためのアーカイブ可能な規則を表現するSwiftインターフェイスを提供します。

Logging

* Swiftのos\_signpost(\_:dso:log:name:signpostID:)は、すべてのプラットフォームのフレームワークOSの一部です
  + サブシステムとカテゴリ、既存の「OSLog」オブジェクト、または既存の「Logger」オブジェクトを使用して、「OSsignposter」をインスタンス化します。
  + "OSsignposter"APIは標識柱を放出する方法を提供する。 "beginInterval(\_:id:)"は開始標識ポストを、"endInterval(\_:\_:)"は終了標識ポストを、"emitEvent(\_:id:)はイベント標識ポストを、それぞれ放射する。 これらは、Stringおよびvargsに基づいて既存のos\_signpostコールを置き換えます
  + APIはメタデータパラメータの文字列補間をサポートします。 文字列補間は、LoggerAPIで受け入れられたものと同じです。
  + OSsignposterAPIは、以前はos\_signpost関数によって提供されていたすべてのフォーマットおよびプライバシーオプションをサポートし、LoggerAPIと同じ構文に従います
  + APIはレガシーAPIに比べて性能が向上します。
  + OSSignposterタイプは、コードブロックを開始および終了サインポストで囲むための新しいスコープAPIを提供します。withIntervalSignpost(\_:id:\_:around:)

\*\* これらのAPIはiOS14およびiPadOS14以前では使用できませんが、既存のos\_signpostAPIは使用可能です。

AVFoundation

* iPadOSアプリは、複数のウィンドウを表示し、画面上の唯一のアプリケーションでありながら、カメラを引き続き使用できるようになりました。
* recommendedVideoSettings(forVideoCodecType:assetWriterOutputFileType:)Objective-Cではなくnullable NSDictionary<NSString \*, id>
* cgImageRepresentation()そして previewCGImageRepresentation()が Unmanaged<CGImage>?ではなくCGImage?を返されます。
* recommendedAudioSettingsForAssetWriter(writingTo:)Objective-Cではなく nullable NSDictionary<NSString \*, id>

マップ

* MKPinAnnotationViewは非推奨としてマークされています。

ネットワーキング

* 現在、ProxyAutomaticConfiguration（PAC）のクリアテキストHTTPURLスキームのサポートは廃止されています。 PACにはHTTPSURLスキームだけを使用します。 これは、[Settings]、[SystemPreferences]、[Profile]、connectionProxyDictionaryやCFNetworkなどのURLSessionAPIを介して設定された設定を含む、すべてのPAC設定に影響します。ExecuteProxyAutoConfiguration（プロキシ自動設定の実行）URL（\_:\_:\_:\_:\_:）。 クリアテキストのHTTPPACURLを設定すると、PACファイルのロード中にHTTPSにアップグレードされる場合があります。 DNSを介したWebProxyAuto-Discovery（WPAD;Webプロキシ自動検出）プロトコルは影響を受けません。 DynamicHostConfigurationProtocol（DHCP）Option252WPADは、PACファイルのロード中にクリアテキストHTTPURLをHTTPSにアップグレードしようとする場合があります。
* 「URLSession」が送信するデフォルトのAccept-Languageヘッダーには、複数のロケールに対して更新された形式と修正された値があります。 ヘッダーには、優先言語の他に、現在のシステム言語が優先言語と異なる場合のフォールバックとして含まれています。 この動作は、「macOS12」、「iOS15」、「tvOS15」、「watchOS 8 SDKs」に対してリンクするアプリケーションに影響します。
* URLSessionに非同期機能が追加されました

Privacy

* AppPrivacyレポートのアプリケーションコンテンツを表示するファイルをダウンロードするには、[Settings]>[Privacy]>[RecordAppActivity]を選択します。

SKAdNetwork

* 開発者が当選ポストバックを受け取ることを選択すると、デバイスは広告されたアプリの開発者に当選ポストバックのコピーを送ることができます

Swift

* 新しいSwift値タイプAttributeStringが、Swift文字列と同じ文字カウント動作で使用できるようになりました。 完全にローカライズ可能で、マークダウン、コード化可能、強力に入力されたアトリビュートなどのサポートも含まれています
* NotificationCenterには、非同期待機を使用して通知を受信するための新しい「AsyncSequenceAPI」が含まれています。

SwiftUI

* これからTextをFormatStyleで初期化できます。
* 検索中に、searchCompletion(\_:)修飾子を使用する提案をタップすると、選択した単一の提案を表示するのではなく、提案リストが表示されなくなります。
* 検索可能な修飾子が以前のタイトルパラメータではなくpromptパラメータを使用して設定する検索フィールドのプロンプトをカスタマイズします。
* textSelection 修飾子を支援します。
* Function（Fn）ショートカット修飾子は非推奨になり、システムで使用するために予約されています.
* LocalizedStringKeyにマークダウン構文を含めることができるようになりました。 文字列リテラルで作成されたテキストビューを含め、ローカライズされた文字列キーからテキストビューを作成すると、マークダウン文字列が解析されます。 システムは、マークダウン構造に従ってテキストをスタイル指定します
* [AttributeString]構造からテキストを作成できます。 テキストはSwift内の属性を通じて提供するスタイルを尊重します。UI属性範囲。これらのスタイルは、ビュー修飾子を使用して提供するスタイルよりも優先されます
* 特定の種類のアニメーションがメインスレッドで実行されるようになったため、新しいスレッド安全要件があります。
  + 次のプロトコルに準拠するすべての方式およびタイプのアクセス
    - AlignmentID, Animatable, EnvironmentKey, EnvironmentValues, Equatable, GeometryEffect, Hashable, Identifiable, PreferenceKey, Shape, VectorArithmetic.
  + 次のタイプと関数に渡す「closure」は、作成したビューにObservableObjectタイプへの参照がない場合のみです。
    - ForEach, GeometryReader, backgroundPreferenceValue(\_:\_:), overlayPreferenceValue(\_:\_:), transformPreference(\_:\_:), anchorPreference(key:value:transform:), transformAnchorPreference(key:value:transform:), transformEnvironment(\_:transform:), transaction(\_:).
* TextFieldはNSFormatterがユーザタイプに応じてバインディングを更新することを提供した。

NSFormatterは、ユーザがフィールドを送信するとき、またはフォーカスがフィールドから離れるときに、フィールドのテキストをフォーマットします。

* 行をタップすると、DisclosureGroupの展開が切り替わります
* デフォルトのListStyleはinsetGroupedです
* 「TextField labels」は、フィールドの横にフォームで表示されません。 promptパラメータを使用して、フィールドの明示的なプレースホルダを指定します。

Tabular Data

* TabularDataは、表形式データを分析および操作するために使用する新しいSwiftフレームワークです。 DataFrameを使用すると、CSVファイルとJSONファイルを読み取り、データの参加、グループ化、および集計を行うことができます。

UIKit

* iOS15ベータSDKに対抗してコンパイルされたアプリの場合、キーコマンドはテキストビューとテキストフィールドに入力する間、テキスト入力とテキスト編集コマンドをこれ以上遮断しない。 たとえば、Deleteキーを押すと、常に文字が削除され、Deletekeyコマンドが存在する場合はトリガーされません。 keyコマンドでテキスト入力を代行受信するには、keyコマンドのwantsPriorityOverSystemBehaviorプロパティをtrueに設定します。 これはまた、キーコマンドが矢印キーやタブキーなどのフォーカスキーボードナビゲーションコマンドよりも優先されるようにするためにも必要です。
* iOS14およびiPadの場合OS14以前では、「 autocorrectionType 」が「 UITextAutocorrectionTypeNo 」に設定されている場合、QuickTypeバーはディセーブルになっている。 iOS15とiPadOS15以降に連動するアプリの場合、QuickTypeバーが有効になり、スペルチェック候補が表示されます。 新しい動作が使用例に適していない場合は、SpellCheckingTypeをUITextSpellCheckingTypeNoに設定してQuickTypeバーを非表示にします
* iOS15ベータSDKを使用してコンパイルすると、いくつかのキーウィンドウ関連のプロパティ、方法、および通知が変更されます
  + isKeyWindowは、ウィンドウがアプリケーションではなく、そのシーンのキーである場合にtrueを返します。
  + becomeKeyWindowは、ウィンドウがアプリケーションではなく、そのシーンでキーになると呼び出されます。
  + didBecomeKeyNotificationは、ウィンドウがアプリケーションではなくそのシーンのキーになったときに投稿されます。
  + resignKeyWindowは、ウィンドウがアプリケーションではなく、そのシーンでキーウィンドウのステータスを放棄するときに呼び出されます。
  + didResignKeyNotificationは、ウィンドウがアプリケーションではなく、そのシーンのキーウィンドウステータスを放棄すると投稿されます。