

Roku OS Release Notes

Table of Contents

- SDK Update Release Notes: version 9 - 10/23/2018
- SDK Update Release Notes: version 8.2 - 09/26/2018
- SDK Update Release Notes: version 8.1 - 05/1/2018
- SDK Update Release Notes: version 8.0 - 10/2/2017
- SDK Update Release Notes: version 7.7 - 06/20/2017
- SDK Update Release Notes: version 7.6 - 04/11/2017
- SDK Update Release Notes: version 7.5 - 11/01/2016
- SDK Update Release Notes: version 7.2 - 06/21/2016
- SDK Update Release Notes: version 7.1 - 04/05/2016
- SDK Update Release Notes: version 7.0 - 10/06/2015
- SDK Update Release Notes: version 6.2 - 04/09/2015
- SDK Update Release Notes: version 6.1 - 12/04/2014
- SDK Update Release Notes: version 5.4 - 04/14/2014
- SDK Update Release Notes: version 5.3 - 12/17/2013
- SDK Update Release Notes: version 5.2 - 10/08/2013
- SDK Update Release Notes: version 5.1 - 08/05/2013
- SDK Update Release Notes: version 5.0 - 06/05/2013
- SDK Update Release Notes: version 4.9 - 11/12/2012
- SDK Update Release Notes: version 4.8 - 07/03/2011
- SDK Update Release Notes: version 4.1 - 12/21/2011
- SDK Update Release Notes: version 3.0 - 04/18/2011
- SDK Update Release Notes: version 2.9 - 03/31/2011
- SDK Update Release Notes: version 2.8 - 11/18/2010
- SDK Update Release Notes: version 2.7 - 10/07/2010
- SDK Update Release Notes: version 2.6 - 06/28/2010
- SDK Update Release Notes: version 2.4 - 12/17/2009

SDK Update Release Notes: version 9 - 10/23/2018

Roku OS 9 introduces a new manifest flag which adds drastic improvements to the memory footprint and launch times of Roku SceneGraph (RSG) channels. All RSG channels should set the "rsg_version=1.2" manifest entry to run the new, higher performing SceneGraph update. This update also adds memory profiling capabilities to the BrightScript Profiler tool, enabling developers to better identify memory leaks in their channels.

Roku TV models are expected to receive the update in early 2019.

Below is a list of new APIs, tools, performance enhancements, and media updates for developers.

SceneGraph enhancements

- **New manifest flag for "rsg_version=1.2"** — By adding this new manifest entry, developers enable tremendous memory savings and app launch time improvements in RSG channels. The "rsg_version=1.2" manifest entry enables a new internal mechanism for processing component `<script>` tags that optimizes the resulting compiled script code. This results in a major reduction in the channel's initial startup time and uses dramatically less memory while preserving total compatibility. Initial startup time is typically reduced by approximately 30% when this manifest flag is set. Memory savings range from 10 - 20MiB in a moderately complex channel, with up to 40MiB saved in a complex channel with extensive component hierarchies.
 - **eval() is no longer supported for "rsg_version=1.2"** — In order to realize the load time performance and memory benefits enabled by the "rsg_version=1.2" entry, eval() can no longer be supported. Any use of eval() will cause compilation and runtime errors if "rsg_version=1.2" is in the channel's manifest.

- **Faster AssociativeArray access** — Read and write access into and out of AssociativeArray fields on nodes, and retrieval of AssociativeArrays using `roSGNodeEvent.getData()` has been made 5x faster. The semantics of AssociativeArray fields are copy on read and write, and copy on `roSGNodeEvent.getData()`. Channels that heavily use AAs can expect dramatic speed increases that directly translate to user-perceivable operations.
- **Faster rendezvous** — This change enables an auxiliary thread to handle rendezvous operations when the render thread is occupied with tasks that don't need rendezvous protection. The result is dramatic speed-ups for rendezvous operations, particularly on Roku devices with lesser processing power.
- **Faster component creation** — Node creation times have been improved in Roku OS 9.
- **ZoomRowList component** — `ZoomRowList`, a `RowList` component where the focused item is zoomed in, is now available for developers.
- **roSGNode.threadInfo() method** — A new diagnostic function, `threadInfo()`, has been added to the `ifSGNodeField` interface. With few exceptions, this function may be called from any component, on any thread, in any function, at any time.

New BrightScript Profiler features

The [BrightScript Profiler](#) now includes a memory profiling tool that can be used to help developers identify memory leaks or memory that is allocated during channel operation without ever being freed. Roku OS 9 also adds support for streaming of profiling data to a local network host.

Media, DRM, and content meta-data updates

- **Dash EventStream** - Dash EventStream elements are supported in Roku OS 9.
- **[BETA] Opening Widevine DRM CBCS decryption mode** — Roku OS 9 adds CBCS decryption support for [Widevine DRM](#) in DASH and HLS streams. At this stage, Widevine support is considered in beta on the Roku platform.
- **[BETA] DASH XLink support** - Roku OS 9 adds DASH XLink support, including EventStream events that are surfaced to the app through the `timedMetaData` field.
- **Widevine key rotation** - Roku OS 9 also adds support for Widevine key rotation and license renewal for Dash and HLS content.

Miscellaneous

- **New manifest entry for a "game" channel** — Developers can now set the `"game"` manifest entry (`game=1`) to avoid audio delays in their game channels.
- **roList indexing is no longer disturbed by array index access** — The behavior of `roList` indexing has been modified to yield more intuitive results and resolve bugs. As of Roku OS 9, when implementing `roList`, `ifEnum` enumeration and `ifList` enumeration will use a separate and independent internal 'pointers' to the current element, instead of a "current integer index" variable. This has been done to ensure the `ifList` index position does not corrupt the `roList` index when the array read/write operators are used.
- **Programmatic access to BIFs through BrightScript** — Two new fields, `getNearestFrame` and `nearestFrame`, have been added to the `BifDisplay` component so that developers can now access BIFs programmatically in their channels. Previously a BrightScript channel had to use the built-in trickplay UI to access these BIFs.

For an overview of the consumer features added in Roku OS 9, [visit the Roku Blog](#).

SDK Update Release Notes: version 8.2 - 09/26/2018

Roku OS 8.2 is a firmware update focused primarily on firmware optimizations and bug fixes, along with new features that will enable Roku TV models to work seamlessly with the upcoming Roku TV Wireless Speakers. This update introduces no new features, APIs, console logs, or tooling about which developers should be mindful.

For our consumer release notes, [see here](#).

SDK Update Release Notes: version 8.1 - 05/1/2018

Media, DRM, and content meta-data updates

- **PlayReady 3 Update** — All Roku devices with MStar chips update to the [PlayReady 3](#) library with Roku OS 8.1. Previously they included PlayReady 2.5.
- **[BETA] Opening Access to Widevine DRM** — Roku OS 8.1 adds support for [Widevine DRM](#) for DASH streams. At this stage, Widevine support is considered in beta on the Roku platform.
- **Digital Rights Management (DRM) control attributes** — [Content metadata control attributes](#) for DRM have been added to the firmware.
 - **Passing custom HTTP headers to licensing requests** — Developers looking to pass custom HTTP headers with a licensing request can now set those headers using the [ifHttpAgent](#) interface methods on the [Video](#) node.
- **Media Player content metadata updates** — Two content metadata attributes of the Media Player have been updated and three new attributes have been added:
 - [PlayDuration](#) is no longer used by the media player.
 - [BookmarkPosition](#) is being deprecated.
 - The existing [PlayStart](#) attribute should be used instead as it has been modified to allow channels to seek to positions prior to [PlayStart](#). The firmware will continue to support [BookmarkPosition](#) to maintain the backward compatibility, but channels should plan a migration to use [PlayStart](#).
 - A new content meta-data attribute, [ClipStart](#), sets the clip start position.
 - A new content meta-data attribute, [ClipEnd](#), sets the clip end position.
 - A new content meta-data attribute, [LiveBoundsPauseBehavior](#), allows a channel to customize Media Player behavior on live streams when playing in the earliest part of a DVR buffer.

Deprecated APIs

Five [roDeviceInfo](#) methods are being deprecated and replaced with similar APIs (as seen in the chart below).

The deprecated APIs will remain in the firmware and continue to work for one year; they will be removed from the firmware in the Spring 2019 OS update. **Developers must update their channels to use the new APIs within the next year.**

Below is a complete list of the APIs deprecated as of Roku OS 8.1.

Deprecated API	Replacement API
GetDrmInfo()	GetDrmInfoEx()
GetAdvertisingId()	GetRIDA()
IsAdIdTrackingDisabled()	IsRIDADisabled()
GetClientTrackingId()	GetChannelClientId()
GetDeviceUniquelId()	N/A — Use GetChannelClientId()

General API Enhancements

- **Memory-level notification** — This release adds a memory-level notification API ([generalMemoryLevel](#)) to [roDeviceInfoEvent](#) to fire notifications to the channel. Channels can also query the memory level directly using [two new methods](#) of the [roDeviceInfo](#) component.
- **GetUserCountryCode() API** — To determine the country associated with a user's Roku account, a new method [GetUserCountryCode\(\)](#) as String was added to [roDeviceInfo](#).
- **roRegex.MatchAll()** — A new method [MatchAll\(\)](#), has been added to [roRegex](#). This adds the ability to return all matches of a specific regular expression pattern in the target string.
- **Getting captions mode from device event** — The developer can now call [GetInfo\(\)](#) on a device event to know the current global setting

for closed caption mode property using `isCaptionModeChanged()`.

- **API to determine if a Roku TV is Energy Star Compliant** — Developers can determine if a Roku powered TV is Energy Star Compliant by using the `roDeviceInfo.HasFeature("energy_star_compliant")` API.
- **StoreChannelCredData API** — This OS update introduces a new `roChannelStore` method, `StoreChannelCredData()`, that allows channels to store information in our backend which they can retrieve later using `GetChannelCred()`. Calling this new API allows developers to store OAuth tokens for a signed-in user so that when the channel is launched on a new Roku device, the channel can request the stored information, identify their customer, and automatically sign them in.

SceneGraph Updates

- **ReplaceChildren() ignores extra items in the replace list** — When using `replaceChildren()` to update the content of each item in a `markupGrid`, if the developer supplies more items than there are in the original list (going from 4 items to 5), the 'extra' items are ignored and not added as children.
- **Mobile or ECP keypress events now appear in onKeyEvent()** — Literal key keypress events (such as keyboard letters, and so forth) that are sent to the channels via the mobile app or ECP keydown/keyup commands, now go to the `onKeyEvent()` handler. Previously, only keys that corresponded to remote keys went to the `onKeyEvent` handler.
- **SimpleLabel** - Firmware version 8.1 introduces `SimpleLabel` which is a lightweight complement node to the `Label` node. It supports simplified font style specification and is more memory efficient than the `Label` node.

SDK Update Release Notes: version 8.0 - 10/2/2017

Performance & Optimization

- **fps_display command** — A new command, `fps_display`, has been added to Telnet port 8080 to display frames-per-second and free memory on-screen. Developers can leverage this tool to optimize their channel UI.
- **Registry ReadMulti and WriteMulti APIs** — `roRegistrySection` adds two new APIs, **WriteMulti** and **ReadMulti** — to allow channels to read/write multiple keys at a time.
- **[BETA] New file system for data caching** — A new file system, `cacheFs`, has been introduced to allow applications to cache data to volatile or persistent storage. Users who extend the persistent storage available on their device by adding an SD card will see the biggest benefit as application data will survive reboots and benefit from additional cache space to improve performance. Users without extended storage will also benefit from the use of a shared in-memory cache that is automatically managed by the system to optimize for the most recently used assets.
- **RSG platform performance improvements** — Many improvements have been built into the firmware itself, enabling better support for low-end devices. All channels automatically inherit these benefits, with no action required from the developer.
 - `<script>` include files no longer incur an expensive copy for each component that includes it.
 - The time penalty for rendezvous has been reduced.
 - The per-node memory penalty has been significantly reduced.
 - Image caching has been added for all channels.

SceneGraph Updates

- **Support for RSG 1.0 functionality is deprecated** — Starting with Roku OS 8, support for the `"rsg_version=1.0"` manifest flag is deprecated. This deprecation means that the 1.0 features continue to work in Roku OS 8, but will no longer be supported (and thus should not be expected to work) starting with our next major firmware release. Channels affected by the change in Roku's `observer callback model` introduced in Roku OS 7.5 should be updated accordingly.
- **Video node updates** — Many new fields have been added to the `Video` node:
 - **captionStyle** allows channels to style closed captions.
 - **contentBlocked** determines whether the current content is blocked.
 - **supplementaryAudioVolume** sets the volume of the description tracks separately from the main audio track.
 - **availableAudioTracks** has been updated to return/include audio description tracks, which are typically seen on broadcast TV.

- **itemHasFocus field for item components** — A new optional field "itemHasFocus" has been added for RSG item components: [MarkupList](#), [MarkupGrid](#), [RowList](#) and [TargetGroup](#). It stores a boolean value that indicates whether the item component currently is the focused item. Only one item component of any of the nodes should have itemHasFocus set to true.
- **ParentalControlPinPad** — Roku OS 8 contains a new node, [ParentalControlPinPad](#). It is a variant of the PinPad component, but with a few key differences:
 - The pin, pinLength, and secureModefields are made private.
 - If the user enters the correct pin, a 2-hour override of content blocking begins, similar to the system behavior on Roku TV.
 - If the user enters an incorrect PIN, the text fields are cleared automatically.
 - A new field, pinSuccess, exists for blocking content.
- **Rectangle node blendingEnabled support** — A blendingEnabled field has been added to the RSG [Rectangle](#) component that specifies if the rectangle should be alpha blended with the nodes behind it.

System Overlay & Closed Caption Updates

- **Improvements to the system overlay** — The behavior of the Roku system overlay has been modified, such that the system overlay now slides in whenever the * button is pressed, the Video node is in focus, and the app does not have its OnKeyEvent() handler fired. When the Video node is not in focus, the system overlay does not slide in and the OnKeyEvent() handler is fired.
- **System overlay notification event** — A new notification has been added to [roDeviceInfo](#). Channels can get notified when a system overlay is displayed.
- **roDeviceInfoEvent update** — A new event, isCaptionModeChangedEvent, has been added to [roDeviceInfoEvent](#) to enable developers to check if the user changes the closed caption mode or track.
- **Closed caption track selection** — It is no longer necessary for a channel to partake in the CC track selection, apart from adding any tracks to the list of available tracks. The firmware now selects a CC track based on the preferred caption language selection in the system preferences. When the selected language is not available, it defaults to the system's UI language.

Miscellaneous

- **Case-preserving quoted keys in Associative Arrays** — The quoted keys in [Associative Array](#) literals are now case-preserving. This change improves the readability of your code and is compatible with JSON usage.
- **CEC status events** — A [roCECStatusEvent](#) has been added for set-top-boxes to determine their active display source status. Channels subscribing to the event will be notified when the active-source status of the device changes per the CEC message traffic.

SDK Update Release Notes: version 7.7 - 06/20/2017

Roku OS 7.7 focuses mainly on bug fixes and firmware optimizations to increase performance of Roku SceneGraph (RSG) channels.

How to update

All current-generation Roku streaming players and Roku TV models will update to Roku OS 7.7.

SceneGraph additions and modifications

- **New event added for DASH manifest updates** — A new field, manifestData, has been added to the [Video node](#) to detect the periods in a DASH manifest before they are played back. One major use case for this is to display ad markers in the trickplay progress bar.
- **New field to reflect current design resolution** — A read-only field, [currentDesignResolution](#), has been added to Scene nodes to determine which of the supported design resolutions is currently being used by RSG.
- **UI changes to the RowList & ArrayGrid components** — Two new fields have been added to the [RowList](#) and [ArrayGrid](#) components to

provide greater control over the UX:

- **RowList** — `rowCounterRightOffset`
Used to specify the location of the right edge of the row counter relative to right edge of the RowList's clipping rectangle.
- **RowList** — `showRowCounterForShortRows`
Determines whether the row counter is shown for all rows.
- **ArrayGrid** — `fadeFocusFeedbackWhenAutoScrolling`
Determines whether to fade the focus feedback indicator while scrolling multiple items.
- **ArrayGrid** — `currFocusFeedbackOpacity`
Provides access to the current opacity of the focus feedback indicator.
- **New field to play animations in reverse** — A "reverse" boolean field has been added to the `FloatFieldInterpolator`, `ColorFieldInterpolator`, and `Vector2DFieldInterpolator` RSG nodes to allow for interpolated values to be computed in reverse.
- **Component compile time optimizations** — Roku OS 7.7 includes several BrightScript compile time optimizations that significantly improve channel launch times. In particular, RSG channels defining many components with the same script files will benefit from the largest channel launch time enhancements.

These optimizations are in the firmware and require no action from the developer.

Additional updates

- **Manifest addition for confirming channel launches** — An optional field, "confirm_partner_button", has been added to the manifest to confirm channel launches before leaving the current channel after a partner button was pressed on the Roku remote. Use this feature to minimize the number of unintended channel launches after a user accidentally hits a partner button while fast forwarding or rewinding content.
- **Manifest entry for overriding network connectivity HUD** — Roku OS 7.7 introduces a system-level display for indicating when media playback is interrupted due to network connection failures. However, channels that have designed their own error dialogue for these interruptions can suppress this pop-up HUD by including a new flag in their channel manifest. The manifest entry to override the HUD is "suppress_unconnected_hud=1".

For more information on the **connectivity HUD**, please read the related [support article](#).

- **New logTypes added to ifSystemLog** — `ifSystemLog` now supports a new logType: "http.complete". When enabled, the "http.complete" events will be sent to Roku after an http transfer is completed for adaptive streams. This event consolidates information related to a cURL transfer such as:
 - DNS look up time,
 - connection latency,
 - transfer speed
 - and number of bytes.

While Roku OS 7.7 is focused almost entirely on bug fixes and developer optimizations, it does include a few new consumer features.

For our consumer release notes, visit the [Roku Blog](#).

SDK Update Release Notes: version 7.6 - 04/11/2017

SceneGraph Additions and Changes

- **TVOD APIs** — New APIs have been added to the RSG `ChannelStore` component to support in-channel purchases. (`roChannelStore`)
- **New function for cloning an entire node tree** — The `clone()` function has been added to `ifSGNodeDict` to clone an entire node tree, or just the node depending on the boolean value passed.
- **Convert structures containing arrays into an RSG node tree** — An "update" function has been added to `ifSGNodeChildren`. Use

`ifSGNodeChildren.update()` to convert objects with arrays and/or associative arrays into node trees.

- **Improved rotation animation** — Fixed a rotation [animation](#) issue for the following device model groups: 2450X, 2500X, 27XXX, 37XXX, and 5000X. ([FloatFieldInterpolator](#))

BrightScript Additions and Changes

- **Microphone APIs** — Added a [roMicrophone](#) component and [roMicrophoneEvent](#) for capturing voice input from the user. Note that channels in the Kids & Family category can not use these APIs.
- **HTTP/2 support** — [roUrlTransfer](#) objects now support HTTP/2. Among the benefits, HTTP/2 support allows for connection sharing, which is accomplished by establishing all [roUrlTransfers](#) that could share a connection from the same Task thread.
- **Conditional compilation** — BrightScript now supports [conditional compilation](#), which allows blocks of code to be run only while set to true.
- **API to count node impressions** — Two new fields — `enableRenderTracking` and `renderTracking` — have been added to [RenderableNode](#) to check whether a node is fully or partially rendered within the bounding rectangle of the screen. Developers can use this information to track node impressions.
- **Import BrightScript files using relative URIs** — Developers can now specify URIs relative to the XML component's file for the `<script>` element's "uri" attribute.
- **New functions added to [ifDeviceInfo](#):**
 - `GetGraphicsPlatform` queries the device to see if it supports OpenGL or DirectFB. The function takes no arguments. The return is either "opengl" or "directfb" as a string.
 - `GetClientTrackingId` is an alias for the existing `GetPublisherId` function. The alternative function name is meant to help developers remember to use publisher ID instead of the device ESN.

BrightScript Debugger Changes

- **BrightScript profiler tool**— A BrightScript profiler tool has been added to Roku OS to gather important metrics such as CPU usage and function "wall-clock" times. Use this tool to analyze performance and increase code efficiency. ([BrightScript profiler documentation](#); [visualization tool](#))
- **Rendezvous logging** — Information on the thread `rendezvous` can now be accessed from within the developer console. To toggle this log setting, telnet to the device port 8080 and enter "enhanced_dev_log rendezvous [on|off]". ([Testing and debugging Roku channels](#))
- **Node operation performance metrics** — Debug command "sgperf" has been added to port 8080 to track `roSGNode` operations by thread, whether it's a create- or a subsequent-operation on an existing node, and whether it involved a rendezvous. ([Testing and debugging Roku channels](#))
- **Shorthand debugger commands** — Shorthand alternatives can now be used to replace many debugger commands. ([Testing and debugging Roku channels](#))

New manifest entry

- `splash_rsg_optimization=1`: Remove flicker between the splash screen and initial screen during channel launch of SceneGraph channels

UI Watchdog

UI watchdog was re-enabled in version 7.6.

The UI Watchdog is a timer that runs on the Roku system and ensures overall stability of the platform by forcing a reboot of the system if it appears to be hung. During normal system operation the timer (watchdog timeout value) is periodically reset to ensure that a reboot is not triggered. In the case of the UI Watchdog, the firmware will give a 3 second grace period for a side-loaded channel locking/over-taxing the Render thread before a system reboot is triggered (for a channel-store installed channel the firmware gives a 10 second grace period).

SDK Update Release Notes: version 7.5 - 11/01/2016

SceneGraph Additions and Changes

- [Thread rendezvous](#) no longer timeout and will wait indefinitely.
- New components:
 - [TargetGroup](#)
 - [TargetList](#)
 - [TargetSet](#)

- **SoundEffect** node
- New and updated fields:
 - **<interface>**: added `roArray`, `rect2D`, and `rect2DArray` types
 - **LayoutGroup**: "custom" alignment value for `horizAlignment` and `vertAlignment` to explicitly set translation values for each child layout
 - **Poster** node: `bitmapMargins` - set to an associative array containing margin information for 9-patch images
 - **Animation**: optional - set to true to skip animations on lower performing devices
 - **Functional Fields** - procedural functions that can be called on components directly
 - **Overhang** - added `optionsText` and `optionsMaxWidth` to customize the text next to the options (★) symbol
 - **RowList**: `rowTitleComponentName` - specify an XML component to render text in place of the row label
 - `findNode()` arguments and `nodeType` fields are now case insensitive
 - **ChannelStore** node commands have been serialized to ensure that one command finishes before the next begins
 - Components in **Component Libraries** can be extended
 - **Component Libraries** can be unsigned if delivered over HTTPS
 - New **Dialog** fields: `titleColor`, `titleFont`, `numberedBullets`, `bulletText`, `graphicWidth`, `graphicHeight`, `width`, `maxHeight`
 - New **ifSGNodeField** methods: `observeFieldScoped()`, `unobserveFieldScoped()`

BrightScript Additions and Changes

- New components:
 - `roAudioGuide` / `ifAudioGuide`
- Updated components:
 - **ifChannelStore**: `GetIdentity()` - returns a unique number for this object that can be used to identify whether events originated from this object by comparing with `roChannelStoreEvent.GetSourceIdentity()`.
 - **ifTuner**: `GetChannelInfo().delivery_system`, `source_data.network_id`, `source_data.transport_stream_id`
 - **ifAppManager**: `SetAutomaticAudioGuideEnabled(enabled as Boolean)` - enable/disable automatic Audio Guide and override any manifest setting
 - **Content Meta-Data - Playback Configuration Attributes**:
 - `ForwardQueryStringParams` - controls whether query string parameters from initial HLS stream manifest requests are forwarded to subsequent segment download requests
 - `IgnoreStreamErrors` - continue playback when encountering any streaming related errors
 - `AdaptiveMinStartBitrate` - minimum startup bitrate to start streaming with a variant equal to or greater than the value specified
 - `AdaptiveMaxStartBitrate` - maximum startup bitrate to start streaming with a variant less than or equal to the value specified
 - `PlayStart` - cannot be overridden by a seek operation
 - `BookmarkPosition` - can be overridden by a seek operation; this value takes precedence over `PlayStart`
 - **roVideoPlayerEvent**:
 - `isStreamSegmentInfo.GetMessage()` - supports segment information for HLS, DASH, and Smooth streams
 - `isRequestFailed().GetInfo()`:
 - `ClipIdx` - The zero starting index of the item in the content list this event is related to
 - `Ignored` - true if the error was ignored and the player skipped to the next item in the content list
 - `isPlaybackPosition().GetInfo()`:
 - `ClipIdx` - The zero starting index of the item in the content list this event is related to
 - `ClipPos` - player position relative to the start of the clip in milliseconds
 - **ifVideoPlayer**:
 - `Play()` - starts playback at the seek position if seek was called prior to play. If seek was not called, the player advances its current position to the next item in the content list and starts playing that item
 - `Stop()` - stops playback and resets the seek position, keeping the player's current position unchanged
 - `SetContentList()` - Resets the current player position, the next time `Play()` is called playback will start at the first item of the content list (Unless `Seek()` is called prior); prefetching updates
 - **ifDeviceInfo**:
 - `EnableScreensaverExitedEvent()` - set to true to enable events to indicate when the user has exited the screensaver
 - `IsAudioGuideEnabled()` - returns true if Audio Guide is enabled on a supported device
 - `EnableAudioGuideChangedEvent()` - set to true to receive Audio Guide events
 - **roDeviceInfoEvent**:
 - `GetInfo().audioGuideEnabled`
 - `GetInfo().exitedScreensaver`

- [ifStringOps](#): [Escape\(\)](#), [Unescape\(\)](#), [EncodeUri\(\)](#), [DecodeUri\(\)](#), [EncodeUriComponent\(\)](#), [DecodeUriComponent\(\)](#)
- [ifTextToSpeech](#): [GetRate\(\)](#), [SetRate\(\)](#), [GetPitch\(\)](#), [SetPitch\(\)](#)
- [ifAssociativeArray](#): [Items\(\)](#) - Returns an array containing the associative array key/value pairs in lexicographical order of key. Each item in the returned array is an associative array with 'key' and 'value' fields.

New manifest entries:

- [automatic_audio_guide_disabled](#): disable Audio Guide within a channel
- [usb_media_handler](#): auto launch channels when a USB device is inserted
- [rsg_version](#): change the type of [observer callback model](#) used

BrightScript Debugger Changes

- Port 8085 now provides context for all threads and port 8089-8093 will no longer be used.
 - STOP, breaks, and continue will stop/resume all threads. Prior to 7.5, only one thread would stop/resume.
 - See the overview on [Debugging SceneGraph applications](#) for more details.
- New 8085 commands:
 - [threads](#): list all current executed suspended threads
 - [thread <id>](#): select a suspended thread to debug
- New 8080 commands:
 - [loaded_textures](#): display the current set of images loaded into texture memory
 - [sgversion](#): change the [observer callback model](#)

External Control Protocol Additions

- [Roku TV commands](#): [query/tv-channels](#), [query/tv-active-channel](#), [launch/tvinput.dtv](#)
- [search commands](#): [query](#) and [launch](#) Roku Search-driven content

Media Player Updates

- Fast Video Start can now prebuffer from non-zero positions, allowing for prebuffering of content in "Continue Watching" lists
- Live DASH streaming support
- The Roku MediaPlayer now automatically detects and plays a video even if the file type is not specified. All major file types — HLS, DASH, Smooth and MP4 — are detected
- Video player automatically scales when changing to a different stream with different aspect ratios

SDK Update Release Notes: version 7.2 - 06/21/2016

This release adds two notable features. A text to speech feature has been added to allow all Roku applications to provide audible spoken versions of the user interface. Also, an option to buffer a video stream without actually playing it is now included, to provide a "fast start" video playback capability to your applications. You can use this option to begin buffering the video stream while a user is reading a description of the video, and then start the actual playback when the user selects it. Roku SceneGraph also supports this new option, as well as built-in support for Audio and Video node playlists, improved HTTPS support, improved debugging, and several other new features.

SceneGraph Changes

- A prebuffer option has been added to the control field of the Audio and Video nodes to allow buffering of media playback prior to the user starting the media item ([Audio](#), [Video](#), [Playing Videos](#)).
- Screensavers can now be created in SceneGraph ([Screensavers](#)).
- New debugging commands are available ([Debugging SceneGraph Applications](#), [ifSGNodeChildren](#)).
- The ChannelStore node class has been added for in-channel purchase support in SceneGraph applications ([ChannelStore](#)).
- The Task node has been modified ([Task](#), [SceneGraph Threads](#)).
- A bufferingStatus field has been added to the Audio node ([Audio](#)).
- Timed meta-data is now supported for both Audio and Video node playback ([Audio](#), [Video](#)).
- Audio and Video nodes now have built-in support for playlists that can play several media items in sequence ([Audio](#), [Video](#)).
- HTTPS support is now available for all SceneGraph nodes ([roHttpAgent](#)).
- A MaxVideoDecodeResolution field has been added to the Video node ([Video](#)).
- New fields have been added to the Video node to allow customizing the internal ProgressBar node ([Video](#)).
- New fields have been added to the Video node to allow customizing the internal TrickPlayBar node ([Video](#)).

- The order of field setting of component-based lists and grids has changed ([MarkupList](#), [MarkupGrid](#), [RowList](#)).
- Several new methods have been added to the `ifSGNodeChildren` interface ([ifSGNodeChildren](#)).

Component Changes

- `roVideoPlayer` and `roVideoScreen` interface `Prebuffer()` method added to allow buffering of video playback prior to the user starting the video ([ifVideoPlayer](#), [ifVideoScreen](#), [Fast Video Start](#)).
- A text-to-speech component has been added to support audible spoken versions of the user interface (`roTextToSpeech`, [ifTextToSpeech](#), `roTextToSpeechEvent`, [Text to Speech](#)).
- New methods have been added to `roUniversalControlEvent` that improve distinguishing between remote control and keyboard key presses, and the key press and release events ([roUniversalControlEvent](#)).

BrightScript Debugger Changes

- Commands to step over and out of functions have been added ([Debugging Your Application](#)).
- Special commands to debug SceneGraph applications have been added ([Debugging SceneGraph Applications](#)).

SDK Update Release Notes: version 7.1 - 04/05/2016

The 7.1 firmware release incorporates several BrightScript and SceneGraph component improvements. There is now support for playing broadcast and cable content on a Roku TV which includes a tuner. SceneGraph has numerous improvements to the Video node, and new capabilities such as passing global data between components, passing parameters to callback functions, localization, and downloading libraries of SceneGraph components.

SceneGraph Changes

- Libraries of SceneGraph components can now be loaded and used at the start of a SceneGraph application ([ComponentLibrary](#))
- SceneGraph components can now be extended from other custom components ([Creating Custom Components](#)).
- A `uri` field type has been added to better support URL resolution and features like certificates and cookies ([Content Meta-Data](#)).
- Support for localization/internationalization string translations and automatic localized graphic image insertion ([SceneGraph Localization](#)).
- Automatic scaling of image files to a specified size on download, with aspect ratio preserving options ([Poster](#), [PosterGrid](#)).
- Parameters can now be passed to observer callback functions ([ifSGNodeField](#)).
- Global application data can now be more easily shared between components, using an `m.global` object reference ([SceneGraph Data Scoping](#)).
- Support for node identity comparison ([ifSGNodeDict](#)).
- Support for dynamic additions to interface fields for all nodes, allowing all node fields to be observed ([ifSGNodeField](#), [Node](#)).
- Video node class includes several new fields to configure trick play and other playback features ([Video](#)).
- Focus indicators for list and grids can now be customized by blending the indicator colors ([ArrayGrid](#)).
- New XML markup component interfaces:
 - `onChange` takes an associative array and a function name ([<interface>](#))
 - calling functions from an associative array is now supported ([<interface>](#))

BrightScript Language Additions

- added increment (`++`) and decrement (`--`) operators to allow integer increment and decrement operations to have effect on a variable ([Expressions, Variables, and Types](#))
- added the following assignment operators to support mathematical and bitshift operations with numeric operands ([Expressions, Variables, and Types](#)):
 - `+=`
 - `-=`
 - `*=`
 - `/=`
 - `\=`
 - `<<=`
 - `>>=`
- `ReadAsciiFile()` now supports UTF-16 files ([Global Utility Functions](#))

BrightScript Language Fixes

- Print now always explicitly prints the component type for enumerable objects. Previously, it would just print the contents of enumerable objects, and did not identify the container object itself, which could lead to confusion ([Program Statements](#)).
- Print and FormatJSON no longer have side effects on enumeration state when accessing enumerable objects (associative array, array, list, and so forth) ([Program Statements](#), [Global Utility Functions](#))

Example:

```
aa={a:2,b:1,c:3} : for each x in aa : print x;" from ";aa : end for
```

Previously:

```
a from ' ...
```

Now:

```
a from ' ...
```

```
c from ' ...
```

```
b from ' ...
```

Component Changes

- roSlideShow SetLoop added ([roSlideShow](#))
- roTextureManager ifHttpAgent added ([roTextureManager](#))
- roUrlTransfer GetToString(), AsyncGetToString() now support UTF-16 files ([roUrlTransfer](#))
- roAppManager/ifAppManager SetUserSignedIn() method added to indicate that a user has signed into the channel ([roAppManager](#))
- roArray/ifArraySort Sort() method was added ([roArray](#))
- roArray/ifArraySort SortBy() method was added ([roArray](#))
- roArray/ifArraySort Reverse() method was added ([roArray](#))
- roString/ifStringOps Split() method was added ([roString](#))

SDK Update Release Notes: version 7.0 - 10/06/2015

SceneGraph XML API

A new user interface programming API has been added. Information on this new API can be found in:

- [SceneGraph XML Guide](#)
- [SceneGraph Reference](#)
- [SceneGraph XML Tutorial](#)

Roku Search and Follow

Users can now follow content from the Roku homescreen **Search** in addition to **My Feed**. This allows you to have users who choose to follow a particular search term (a movie, director, or an actor) to receive automatic updates when your latest content includes that term.

Roku 4 Support

Support is now included for the Roku 4 Streaming Media Player. This Roku Player features greatly enhanced video resolution output up to 2160p (ultra-high definition, or UHD). This Roku Player also allows the creation and use of user interfaces with 1080p (full high-definition, or FHD) resolution. The Roku 4 Streaming Media Player includes a much more powerful quad core ARM processor and decoding support for the HEVC (high-efficiency video code) codec to allow efficient streaming of 2160p video content.

ECP Install Command

An install command has been added to the ECP to allow deep-linking to uninstalled channels.

BrightScript language

- Associative array literals can now specify key names as string literals (quoted strings).

- added `aa.Keys` to return the keys for an associative array
- added a 64-bit `LongInteger` type

BrightScript components

roAssociativeArray

- added `Function Keys() As Object`

roDeviceInfo

- added Functions `CanDecodeVideo()`, `CanDecodeAudio()`, and `GetDrmInfo()` to `ifDeviceInfo` interface.

SDK Update Release Notes: version 6.2 - 04/09/2015

Roku Ad Framework

- added the [Roku Advertising Framework](#) to natively integrate advertising capabilities

BrightScript language

- now supports embedded quotation mark characters in string literals

BrightScript components

ifDateTime changes

- added `GetDayOfWeek()` as Integer
- added `GetTimeZoneOffset()` as Integer
- added `ToISOString()` as String

ifDraw2D changes

- added `DrawPoint(x as Integer, y as Integer, size as Float, rgba as Integer) as Void`

SDK Update Release Notes: version 6.1 - 12/04/2014

BrightScript language

- now supports integer division operator `\`
- now supports integer bitshift operators `<<` and `>>`
- `roAssociativeArray Count` function was added
- parameter validation diagnostics have been added to core components
- `roString Replace` method was added.
- `ParseJSON` bug fixes
- `FormatJSON` bug fixes

- global function FindMemberFunction was added.
- floats are now auto-boxed for method calls
- global function Val has a new overload for parsing from hexadecimal strings etc.
- global function StrI has a new overload for formatting hexadecimal strings etc.

BrightScript debugger

- bscs 'Summarize BrightScript Component instances' command was added
- brkd 'Break on BrightScript diagnostics' command was added

BrightScript components

roDeviceInfo changes

- added Function GetModelDetails() As Object
- added Function GetFriendlyName() As String
- added Function GetCreationTime(Void) As String
- added Function GetAudioDecodeInfo(Void) As Object
- added Function GetVideoDecodeInfo(Void) As Object
- added Function GetAdvertisingId(Void) As String
- added Function IsAdIdTrackingDisabled(Void) As Boolean
- added Function GetPublisherId(Void) As String
- added Function GetRandomUUID(Void) As String

roMessageDialog changes

- added Function UpdateButton(id As Integer, title As String) As Boolean

SDK Update Release Notes: version 5.4 - 04/14/2014

New in this release

- Two new content metadata structures for controlling closed captions: SubtitleConfig and SubtitleTracks. Details of how to use these to control captions [can be found here](#).
- Two new caption renderer functions for retrieving all caption tracks in a stream and for setting the current track: ifCaptionRenderer.GetSubtitleTracks() and ChangeSubtitleTrack(). Details about these two new functions [can be found here](#).
- Bug fix: On the 2450X and 2500X platforms, the ifChannelStore.DoOrder() function was always returning false even if the corresponding purchase was successful.

SDK Update Release Notes: version 5.3 - 12/17/2013

Version 5.3 introduced user configurable closed caption settings. The Roku settings UI now includes controls for turning captions on or off at the system level. These settings control closed caption behavior for all channels. Closed captions on instant replay are also controlled from these settings.

SDK Update Release Notes: version 5.2 - 10/08/2013

Version 5.2 introduced a new security model for side-loading channels. Accessing the web interface for the device now requires a userid and password to log in. Please see the article here for more details: <http://blog.roku.com/developer/2013/10/08/security-enhancements-added-to-channel-development>

New in this release

- Security enhancements to channel side-loading
- `roListScreen.SetupBehaviorAtTopRow()`
- Closed Captioning support added to `roVideoPlayer`, `roCaptionRenderer` added
- HLS ID3 events added to `BrightScript`

SDK Update Release Notes: version 5.1 - 08/05/2013

New in this release

- In-stream 608 captions support
- updates to `roVideoScreen` events
- FLAC container support (local playback)

SDK Update Release Notes: version 5.0 - 06/05/2013

Version 5.0 of the Roku OS was released to all second and later generation devices. The home screen introduces a completely new look and feel for navigating my channels, the channel store, and settings. This release does not introduce any new screens or templates to the SDK.

New in this release

- Updated home screen user interface
- Increase Brightscript function limit beyond 1024
- Miscellaneous fixes for gaming remotes
- Memory improvements to the OS
- Improved ECP security

SDK Update Release Notes: version 4.9 - 11/12/2012

New in this release

- Roku billing - in-app subscription support in `BrightScript`
- SMPTE TT CC (MP4, HLS, Smooth Streaming)
- `RoTextureManager` 2D component

SDK Update Release Notes: version 4.8 - 07/03/2011

We've had our heads down working diligently on new firmware features and supporting new hardware. Hello Streaming Stick! In v4.8, we have some new developer SDK features to share with you. We are continuing our firmware version numbering scheme with v3.1 currently released to all "Classic" or "Roku1" models and major version 4.8 coming soon to all Roku2 models.

We are maturing as a company, and with that comes new legal requirements. In v4.8, we are asking our development community to agree to the development terms directly on the box before it can side-load and channels. As soon as your developer mode box updates to v4.8, you may find you are unable to side-load channels. You will need to re-enable developer mode by entering the following remote control sequence: 3x Home, 2x Up, Right, Left, Right, Left, Right. After entering "Enable Installer" on the secret screen, it will present you with the developer terms that you must read and scroll through and then click down to "I Agree". The box will then reboot with the side-loading installer enabled.

The Roku Team

New in this release

- Internationalization and Localization support. (Currently localizing to en_US, fr_CA, es_ES, de_DE)
- Smooth Streaming with PlayReady Support
- In-channel purchasing of new content, upgrades, features.
- Native, fast `ParseJSON()` function.

Compatibility Issues

Remember to re-enable developer mode in v4.8!

SDK Update Release Notes: version 4.1 - 12/21/2011

With v4.1, Roku is now supporting the "Roku 2" generation of boxes. We've tried to keep the Roku SDK compatible across all the different models and have highlighted hardware and SDK differences in a new Section 1.4 of the Developer Guide. Our firmware version numbering requires a little explanation as major version 3 is currently released to all "Classic" models and major version 4 is release to all "Currently Selling" models. We continue to have minor releases planned on both of these major release branches. All features from the v3.0 beta have made it into the final v3.0 and 4.1.

We understand that we haven't exposed all the functionality the Roku 2 platform has to offer to the entire Roku Developer Community. Roku has a good track record of taking security issues seriously. There are many such issues to work through before we can expose low level OpenGL interfaces in a secure manner that the content owners in our community are also comfortable with. We are working diligently toward that end.

We appreciate all the hard work the Roku Developer Community has put into developing some great applications on the Roku platform. We are happy that the Roku 2 has been well received by the market and are looking forward to continued shared success with the Roku Developer Community and all the great new channels you can create.

The Roku Team

New in this release

- Roku 2 platform support.
- Brightscript Plugin for Eclipse
- Updated videoPlayer sample application
- All v3.0 beta features are now fully supported on classic models running v3.0 and Roku 2 models running v4.1:

- BrightScript v3.0 that includes:
 - performance improvements
 - typed values in function parameters and returns
 - improved auto-boxing and type promotion
 - explicit programmer controlled garbage collection
 - better statement stepping in source level debugger
 - collections can include intrinsic values rather than only objects
 - 2D Graphics APIs:
 - roScreen
 - roBitmap
 - roRegion
 - roCompositor
 - roSprite
 - roAudioResource
 - roFont
 - roFontRegistry
 - roFontMetrics
- New Platform Components
 - roSocketAddress
 - roStreamSocket
 - roDataGramSocket
- New ECP "input" command to pass user defined input parameters to your channel.

Compatibility Issues

Please note that we have deprecated support for Macrovision and WMV video. We still have support for CGMS (Copy Guard Management System) protection for analog outputs and HDCP protection for digital output. There has been very little use of WMV video, and if anyone has content out there it can be converted to MP4 with several transcoding applications, including ffmpeg (please see our encoding guide for example usage).

SDK Update Release Notes: version 3.0 - 04/18/2011

We're making the 3.0 Beta firmware available to all developers who request it. If your box isn't already part of the developer beta group, send a private developer forum message to RokuKevin. Include the serial number of the Roku units you would like to run the 3.0 beta on. It is very important that all developers regression test their applications on version 3.0 and note any incompatibilities. You may need to publish an update to your application to make it compatible with v3.0.

There are some exciting new features in the v3.0 SDK that will support developers building 2D games for the Roku box. We encourage all game developers to support both HD and SD modes. About half of the Roku's out there still run in SD mode. We've added screen level scaling to support developers using one set of graphics assets and a single game engine running in both HD and SD modes. Although v3.0 has not been released to end users yet, we are now encouraging discussion about v3.0 features on the developer forum.

The Roku Team

New in this release

- BrightScript v3.0 that includes:
 - performance improvements
 - typed values in function parameters and returns
 - improved auto-boxing and type promotion
 - explicit programmer controlled garbage collection
 - better statement stepping in source level debugger
 - collections can include intrinsic values rather than only objects
 - 2D Graphics APIs:

- roScreen
- roBitmap
- roRegion
- roCompositor
- roSprite
- roAudioResource
- roFont
- roFontRegistry
- roFontMetrics
- New Platform Components
 - roSocketAddress
 - roStreamSocket
 - roDataGramSocket
- New ECP “input” command to pass user defined input parameters to your channel.

The new 2D Graphics components are considered part of the core BrightScript language and are documented in the BrightScript Reference. The new platform components are documented in the Component Reference. New v3.0 components have their own sections and new methods on previously existing components are called out with their own “Since v3.0” sections.

Compatibility Issues

BrightScript v3.0 is stricter in some ways than BrightScript v2.0. Areas in your app with variables that are used before initialized, or return statements that return a different type than specified in the function declaration may cause runtime errors in v.30 that may have run successfully in v2.0.

It is very important that all developers regression test their applications on version 3.0 and note any incompatibilities. You may need to publish an update to your application to make it compatible with v3.0.

SDK Update Release Notes: version 2.9 - 03/31/2011

With the release of firmware v2.9 build 1553, we are excited to share our latest SDK documentation. We have added several new features that developers can take advantage of. Many of you have already seen the new Premium Developer Program. With a premium developer account you can charge for applications in the Roku channel store and Roku will handle the billing for you. To sign up for a premium developer account you simply need to provide your tax id and agree to the new terms and conditions. Our new v2.9 SDK helps support paid applications with the ability to launch the channel store to purchase your application. You can create a free “Lite” app that includes a banner ad that up-sells to a premium app. When the user clicks on the ad, the “Buy” page is launched in the channel store. There is an example of this in the roPosterScreen section of the component reference.

Please refer to the Component Reference Section 7 for an introduction to the two new components in v2.9. The roAudioMetadata component gives you access to metadata included in many audio files. It recognizes ID3 tags and supports cover art. The roImageMetadata component gives you access to metadata in image files.

Version 2.9 updates the grid component by letting the developer control the “Up” button behavior and giving developers the ability to stack grid screens without a lot of extraneous code.

Thanks for all the channels you’ve developed so far. We’re looking forward to all the new ones you’re busy working on.

The Roku Team

New in this release

- MKV playback on local USB devices.
- Developer control of the Up button behavior on the top row of the Grid Screen.
- Stackable Grid Screens.
- Channel Store can be launched to purchase “Premium” channel from within “Lite” channel

- Get ID3 tags and other metadata from your audio files
- Get EXIF and other metadata from your image files

New v2.9 components have their own section in the reference and new methods on previously existing components are called out with their own "Since v2.9" sections.

SDK Update Release Notes: version 2.8 - 11/18/2010

It's been a short time since we last updated you, but we still have a few things worth sharing in this release of the SDK. We've added developer specified HLS stream switching strategies, new commands in the External Control API, and customizable GridScreen layouts.

We appreciate your support of the Roku platform and always welcome your feedback.
The Roku Team

New in this release

- Developer controlled HLS stream switching strategies for improved playback ability.
- Multiple Grid layouts: flat-movie, flat-portrait, flat-landscape, flat-square, and flat-16x9
- Developer customized Grid Focused Border image, and Grid Description callout box image.
- Use .png and .gif images as Grid posters.
- Ability to get the Roku's IP address within your channel. This will enable the External Control API to be utilized directly within your channel.

New v2.8 components have their own section in the reference and new methods on previously existing components are called out with their own "Since v2.8" sections.

Compatibility Issues

We have maintained backward compatibility with the v2.7 SDK. Some have experienced video playback issues with their HLS streams. The number one issue was incorrect aspect ratios and resolutions. We've addressed most of these issues with a new v2.8 build (1158), but we have requested that any other HLS playback regression issues from v2.7 be posted to a sticky thread on our forum. We will attempt to address them all in a timely manner.

SDK Update Release Notes: version 2.7 - 10/07/2010

This release brings several new features to the Roku SDK, as well as support for the additional hardware capabilities of the new Roku models. The new models add 1080p playback capability and a new 12 button remote control. The new remote control will come with the Roku XD and Roku XDS. The Roku HD will continue to ship a 9 button remote. However, the 12 button remote will be available for purchase separately. The 12 button remote will also work on the older Roku models, providing the installed base access to the new button capabilities with only a remote control purchase. While not all users will have the 12 button remote, we are strongly encouraging developers to add support for the three new remote control buttons.

The "Instant Replay", is implemented entirely in the firmware and will work on all channels during video playback without any changes to developer channels. The "Back" button works by default on most screens, but needs to be enabled on dialogs if the screen close behavior is desired. The "Info" button is intended to provide additional contextual information at various points within channels. It is up to developers to use the new features in the roMessageDialog and support the "Info" button usage pattern of launching context menus and dialogs. We have included a simpleinfo sample app in this release that illustrates the info button usage.

We are also introducing 1080p video playback support on the Roku XD and XDS models. Updated information on supported encoding resolutions is included in the new Encoding Guide. We encourage developers to add 1080p streaming content at bitrates below 4.0Mbps.

A much anticipated addition to the SDK is the new roGridScreen component which presents the user with a scrolling grid as an alternative to the

poster screen interface. The simplegrid sample app demonstrates how to use this new feature.

Below is the complete list of new features in this release.

We appreciate your support of the Roku platform and welcome your feedback.

The Roku Team

New in this release

- Support for new Remote Buttons - "Back", "Instant Replay", and "Info" buttons support ease of use, better trick play, and contextual menus.
- 1080p playback support – When the user sets his display to 1080p, any content with meta data parameters for FullHD and a FrameRate of 24 or 30 will playback at 1080p 24 fps or 1080p 30 fps.
- Grid Component – The grid screen enables users to easily navigate large collections of content.
- External Control Protocol – The ability to control the Roku over the network. It's now possible to create sophisticated iPhone, Android, and Blackberry apps.
- Paragraph Screen Default Menu Item – You can now control the button that is highlighted on the paragraph screen so that selection dialogs work as expected.
- Message Dialog Overlay Support – Expected to be popular when used in conjunction with the "Info" button to display contextual information.
- Content Meta-Data parameters – New parameters provide more control over video playback: min/max bandwidth, and audio stream selection.
- New Documentation Guides – The External Control Guide illustrates how to create remote control apps that work over the network. The Encoding Guide gives useful guidelines for encoding video content that is compatible with the Roku Streaming Player.
- Simpleinfo sample application - shows how to use the roMessageDialog with overlay on top of a roPosterScreen when the "Info" button is pressed.
- SimpleGrid sample application – shows how to use the roGridScreen component to display many rows of items.
- AudioApp sample application – adds an application screensaver.

New v2.7 components have their own section in the reference and new methods on v2.4 components are called out with their own "Since v2.7" sections.

Compatibility Issues

We have maintained backward compatibility with the v2.7 SDK. However, developers need to be aware of the new hardware models and update their channels to take advantage of the new features. If your back end services do any type of authentication based on model numbers, or you have any code that is based on model numbers you will need to update your code.

If any of your BrightScript code is enabling certain features based on model number, we have a new roDeviceInfo.HasFeature() method that will enable you to code this logic in a more forward-looking manner. If you don't account for the new "Back" and "Info" buttons in your code, users may think your channel does not behave correctly.

On most screens, the back button will automatically send an event that matches the isScreenClosed() predicate. However, this behavior would break many modal dialogs that do not expect to receive an isScreenClosed() event because they are waiting for a state change or reacting to user input that must be answered. We chose not to break these channels and instead did not enable the back button by default on the roMessageDialog component. We did provide a method to enable the back button on roMessageDialogs that can successfully handle an isScreenClosed() event. You should survey your channel for any roMessageDialog components that should enable the back button with EnableBackButton(true). The "Info" button will enable you to pop-up any screen of your choosing. There are two new events added to support this new button. Events matching isButtonInfo() return the button focus of any on screen buttons in when the Info remote key is pressed. Events matching isListItemInfo() return the index of the focused poster when the Info remote key is pressed. The simpleinfo sample application shows basic support for the "Info" button that also demonstrates an overlay dialog on a poster screen. When isRemoteKey() events are propagated to your script, the event.GetIndex() for the Info key is 10.

SDK Update Release Notes: version 2.6 - 06/28/2010

The 2.6 SDK release adds a number of new components to the Roku Platform SDK and introduces a new user interface for developers on the Channel Store for managing your channels.

The Channel Store Developer Site now provides support for managing application dependencies based on minimum required firmware version or hardware capabilities. For example, if your application requires the features of a specific firmware release or a hardware feature, you can specify this dependency and ensure your application is only published to the correct systems. We recommend studying the Channel Packaging and Publishing Guide for more information about the Channel Store versioning support.

We have added a number of new components to the SDK and expanded the API's for some of our existing components. Be sure to check out the new `roImageCanvas` component, which allows much greater freedom for laying out custom types of screens. We've also added API's to open up the USB port on the Roku XR for developers, support for HTTP Live Streaming, plus many other new features.

We hope you enjoy these new features and we're anxious to see the new types of applications that you develop with these capabilities. Thanks again for your support.

The Roku Team

Compatibility Issues

The 2.6 SDK release adds additional events. Please be sure your event loops ignore unknown events. If your code is exiting event loops rather than just ignoring unknown events your application may have undefined behavior.

Important Notes

- Please review the Channel Packaging and Publishing guide for more information about the new versioning support and how it impacts your application deployments in the Channel Store.
- We have changed the behavior of the Home remote control key. In v2.6, it now immediately kills your application and returns to the home screen. If your app was previously relying on a graceful application exit to do cleanup and bookkeeping, you will need to modify your application. You will want to modify your app to periodically update playback positions, positions in poster screens, search results, etc. This should not wait until application exit, as that could happen to you at any time.

New in this release

- HTTP Live Streaming (HLS) – This is Roku's implementation of Apple's adaptive bitrate streaming solution. This feature provides support for both windowed, "live" and adaptive bit-rate VOD streaming capabilities.
- Image Canvas Screen – A clean slate for creating custom screens. This object will allow you to place text and graphics wherever on the screen you desire. You can see an example in the SDK clock sample app.
- Custom Font Support – Include your own TrueType (TTF) or Open Type (OTF) fonts in your application for use on the Image Canvas.
- CA-Cert Bundle – For applications that refer to many different feeds, we've included a collection of CA-Certs that is the trusted set from the Firefox browser in a common filesystem. Any application can now easily trust this same set without increasing the size of its package.
- Customizable Video Player – A video player that allows you play a video in a region on the screen and/or programmatically control playback. When used in conjunction with Image Canvas, you can combine video, images and text on a single screen.
- USB support – For those of you that have Roku XR boxes or want to develop applications that make use of USB storage. Basic support for USB hotplug events and automounting of USB volumes is included in v2.6. Support for a variety of USB devices and the VFAT, NTFS, HFS and HFS Plus file systems.
- File System access – You will be able to enumerate available file systems and access the content stored on them.
- SRT subtitle support – If the content you are playing has an available SRT file for subtitles, enabling subtitle display will be as simple as specifying a path to the SRT file.
- ScreenSavers – Create standalone screen saver applications that can run whenever the Roku Streaming Player is idle or create a custom screen saver for your Channel.
- Perl-Compatible Regular Expressions – String manipulation just got a whole lot easier!
- Application Logging Support – It's now easier to gather statistics in your application and send them back to your own logging server.

There are lots of other changes in this release and these are just some highlights that we wanted to specifically mention. Please check the Component Reference for additional details. New v2.6 components have their own section in the reference and new methods on v2.4 components are called out with their own "Since v2.6" sections.

SDK Update Release Notes: version 2.4 - 12/17/2009

This release unites our Channel Store development activities with the SDK. There are a few implications for developers during the transition that we'd like to highlight. Our legacy applications will not appear on the player until it syncs to our Channel Store. The Channel Store is not yet deployed in Production, so Netflix and Amazon will not appear on your player after the upgrade. These apps will eventually reappear on your players as the backend services are upgraded.

We've provided this firmware update early so that you have an opportunity to develop with this release before the Channel Store beta is available. After you update your software build, the only features that will be visible on the home screen will be Settings and any developer application that you've installed manually. We plan to distribute at least one more SDK release before we get to feature complete for the launch, so we're still adding more features and fixes on your behalf. We think that most of these enhancements are things that will be useful to a wide variety of developers.

Thanks again for your support!
The Roku Team

Compatibility Issues

There have been significant changes to our file system APIs. We've implemented these changes so that they are backward compatible for a limited time. Any deprecated API calls will be displayed in the debugger with a warning message. The samples applications were updated to use the new file specification format. Please refer to the BrightScript and Component Reference manuals for more information. Backward compatibility will be removed in the next release, so please update as soon as possible.

Update Instructions

The Roku player will automatically update to the new version within the next 1-2 days.

We hope you force your player to download the new version immediately by following these steps

- 1) Press the HOME button on the remote control.
- 2) Use the arrow keys to highlight the "settings" icon and press the SELECT button.
- 3) Press the right arrow several times until you see "player info" and press SELECT.
- 4) Highlight "check for update" and press SELECT.
- 5) SELECT "yes".
- 5) Wait for the software to download, and then SELECT "ok" to restart.

Note: If you already have the new version, a message will appear letting you know your version is current.

Important Notes

- This release includes new tools for packaging your application for deployment. Included in the SDK is the ChannelPackagingAndPublishing document. This document provides a step-by-step guide to the packaging process. Please be sure to review this document, since it will answer many questions about how applications will be deployed onto the Channel Store.
- The Channel Store linking screen is included in this release and there are a few situations you could encounter it and get stuck. If you reset your box to factory defaults or change your network settings, this could occur. We don't want you to link your box at this time, so we've provided the following instructions to help you bypass this screen if necessary.

Display the secret screen using the following key sequence:

Home 5x, FastForward 3x, Rewind 2x

After pressing this key sequence, the "secret screen" will appear. Select the "cycle channel store server" option until the text "<bypass account linking>" is displayed and then press the "back" button to exit the screen.

The software update options on this screen are controlled via the server and are not functional for developers. Just ignore these options, since they won't work anyway.

New in this release

- Documentation: There continue to be lots of changes to the documentation set, mostly to the Component Reference manual. The documentation has been reformatted with section numbers to make cross-referencing easier. New features have been added and additional details were provided on some video playback topics. The new file system changes are included as well as lots of API updates. There are several new docs in the set, such as the Channel Packaging and Publishing document mentioned above.
- More Theme Attributes – We added quite a few more attributes changing colors within the UI. The paragraph and registration screens now support color attributes, as well as font color changes to the buttons and filter banners. These still work the same way as they have in the past and allow you to set an HTML Hex color value to override the default color scheme.
- Flexible Image Scaling – We've added new scaling options for artwork that's used in the Poster Screen, Slide Show or Banner Ads. The new options allow you to specify how you'd like odd-sized artwork scaled to fill the destination area.
- Image Styles – We've added additional styles to several screens for 16x9 and square aspect ratio artwork. There are more changes in the works in this area, which will be coming soon. Many of you have different size and aspect ratio artwork and providing more types of frames, plus better scaling capabilities will help to take advantage of all the existing artwork out there.
- Display Mode API – Many developers have asked how they can find out the current display mode for the device. Check out the `roDeviceInfo` component for new APIs to access info about the users display settings.

There are lots of other changes in this release and these are just some highlights that we wanted to specifically mention. Please check the documentation for additional details.