Content Meta-Data

Table of Contents

Content Meta-Data describes a viewable title that will be shown to the user. Content may be any supported type of video and the meta-data is used by the UI to format and display the title to the user. Some attributes (e.g. ContentType) affect how the title is displayed on screen, other attributes (e.g. SDPosterURL) specify where to fetch artwork to display with the content and other attributes (e.g. Title) are just rendered as text.

The content meta-data is stored in an associative array by the script and provided to the various screen objects as needed for display. In some cases an array of content meta-data may be provided so that the screen can render multiple items as a list. The attribute names are critical and used as the key to look up the attribute at run time. The following table details the attributes in use. Certain attributes are recognized by particular screens, while others are more globally applicable. If the attribute is not a generally recognized attribute, the table below specifies where the attributes are recognized.

Keep in mind that there are two ways to specify stream content metadata, **data.Stream** and **data.Streams**:

- **data.Stream**: This is used when there is one stream URL, typically an HLS or smooth streaming manifest URL.
- **data.Streams**: This is used when you have a set of fixed bitrate streams. This is typically the case for non-adaptive MP4 streams, in which case multiple variants are specified to simulate true adaptation.

Descriptive Attributes

Descriptive meta-data attributes can be used to describe the content item to the user, in a user interface element that allows the user to select the item.

Attribute	Type	Values	Example

<table>
<tr><td>ContentType</td><td>String</td><td>Although <code>ContentType</code> accepts type <code>String</code>, the return value is of type <code>roInt</code>. See table below.

<table><tr><th>ContentType</th><th>Return Value</th></tr><tr><td>not set or not supported</td><td>0</td></tr><tr><td>movie</td><td>1</td></tr><tr><td>series</td><td>2</td></tr><tr><td>season</td><td>3</td></tr><tr><td>episode</td><td>4</td></tr><tr><td>audio</td><td>5</td></tr></table></td><td>"movie"</td></tr>
<tr><td>Title</td><td>String</td><td>Content title:
movie title for films,
episode title for TV series</td><td>"The Dark Knight"</td></tr>
<tr><td>TitleSeason</td><td>String</td><td>Season title for TV series</td><td>"Battlestar Galactica Season 5"</td></tr>
<tr><td>Description</td><td>String</td><td>Description of content</td><td>"Batman, Gordon and Harvey Dent are forced…"</td></tr>
<tr><td>SDPosterUrl</td><td>String</td><td>URL for SD content artwork</td><td>http://www.myco.com/img/sd1932.jpg</td></tr>
<tr><td>HDPosterUrl</td><td>String</td><td>URL for HD content artwork</td><td>http://www.myco.com/img/hd1932.jpg</td></tr>
<tr><td>FHDPosterUrl</td><td>String</td><td>URL for FHD content artwork</td><td>http://www.myco.com/img/fhd1932.jpg</td></tr>
<tr><td>ReleaseDate</td><td>String</td><td>Formatted Date String</td><td>"3/31/2009"</td></tr>
<tr><td>Rating</td><td>String</td><td>Selects an icon to be displayed for the corresponding MPAA or TV rating, that is, the value will display as an icon artwork.

See Rating Attribute Icons for a list of the acceptable values and the corresponding icon.</td><td>"PG-13"</td></tr>
<tr><td>StarRating</td><td>Integer</td><td>Specifies the star rating to display as red star icon artwork, as a number from 1 to 100:

20 displays one star
40 displays two stars
60 displays three stars
80 displays four stars
100 displays five stars

Numbers not divisible by 20 are displayed as a fractional star. (A number of 30 will display one and a half stars.)</td><td>80</td></tr>
<tr><td>UserStarRating</td><td>Integer</td><td>Specifies the user star rating to display as yellow star icon artwork, as a number from 1 to 100:

20 displays one star
40 displays two stars
60 displays three stars
80 displays four stars
100 displays five stars

Does not display fractional stars for numbers not divisible by 20.</td><td>80</td></tr>
<tr><td>ShortDescriptionLine1</td><td>String</td><td>Line 1 of Poster Screen Description</td><td>"The Dark Knight"</td></tr>
<tr><td>ShortDescriptionLine2</td><td>String</td><td>Line 2 of Poster Screen Description</td><td>"Rent $1.99, Buy $14.99"</td></tr>
</table>

EpisodeNumber	String	Episode Number	"1"
NumEpisodes	Integer	Number of episodes for a "season" or "series" contentType	40
Actors	roArray	List of Actor Names	["Brad Pitt", "Angelina Jolie"]
Actors	String	Individual Actor Name	"Brad Pitt"
Directors	roArray	List of Director Names	["Joel Coen", "Clint Eastwood"]
Director	String	Individual Director Name	"Christopher Nolan"
Categories	roArray	List of Category/Genre Names	["Comedy", "Drama"]
Categories	String	Individual Category/Genre Name	"Comedy"
Album	String	roSpringboard audio style uses this to display the album	"Achtung"
Artist	String	roSpringboard audio style uses to show artist	"U2"
TextOverlayUL	String	roSlideShow displays this string in Upper Left corner of slide	"Joe's Photos"
TextOverlayUR	String	roSlideShow displays this string in Upper Right corner of slide	"3 of 20"
TextOverlayBody	String	roSlideShow displays this string on the bottom part of slide	"Wanda's 40'th Birthday"

Digital Rights Management (DRM) Control Attributes

Available since firmware version 8.1

Digital rights management (DRM) content meta-data control attributes are available in the firmware through the `drmParams` parameter of type roAssociativeArray. The table below enumerates all usable attributes of `drmParams`.

Note: Not all attributes are required, and may not have the same semantic meaning when applied to different DRM systems.

Attribute	DRM System	Type	Value	
KeySystem	Required for all	String	"playready", "widevine", "aaxs" (Adobe Access) or "verimatrix". This value is case-insensitive. The default is an empty string.	"widevine"
licenseServerURL	Playready: Optional Widevine: Required	String	A URL location of a license server. This URL may include CGI parameters.	"https://hos
serializationURL	Verimatrix: Required Playready, Widevine, Aaxs: Optional	String	A server address used for device provisioning	"https://hos
licenseRenewURL	Widevine: Optional	String	A URL location for sending license renewal requests. If not specified, the Roku OS would send renewal requests to the URL specified in the licenseServerURL.	"https://hos 2"
appData	Playready, Widevine, Aaxs, Verimatrix: Optional	String	Special meaning per DRM system. If supplied, expected to be a base64 encoded string.	"SGF2ZSB

authDomain	Aaxs: Optional	String	Used in conjunction with appData to form authentication block to be passed with a licensing request	`"Roku"`
encodingType	Playready: Required	String	Specifies the encoding scheme for PlayReady DRM, by setting to one of the following values: • `"PlayReadyLicenseAcquisitionUrl"` • `"PlayReadyLicenseAcquisitionAndChallenge"` `Note, this is the same value that used to be specified directly in Content Metadata structure`	`"PlayRea`
encodingKey	Playready: Required	String	Specifies the PlayReady license acquisition data, in format depending on the `EncodingType` attribute value specified: • when `encodingType="PlayReadyLicenseAcquisitionUrl"`, the `EncodingKey` attribute contains the PlayReady license acquisition URL • when `encodingType="PlayReadyLicenseAcquisitionAndChallenge"`, the `EncodingKey` attribute contains the PlayReady license acquisition URL plus additional custom license acquisition request data in format "URL%%%<customdata>" Note, this is the same value that used to be specified directly in Content Metadata structure	`"http://`

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.

Example of configuring a dash stream with Widevine DRM

```
contMeta = {
    HDPosterUrl:"pkg:/images/BigBuckBunny.jpg"
    SDPosterUrl:"pkg:/images/BigBuckBunny.jpg"
    ShortDescriptionLine1:"Parking Wars(VOD)"
    ShortDescriptionLine2:"dash | widevine"
    Streamformat:"dash"
    SwitchingStrategy:""
    MinBandwidth:500
    VideoUrl: "http://dev.domain.com/mm/dash/vod/173850/85768039/TG_W_WIFI.mpd"
    drmParams: { ' setting up DRM config
        keySystem: "Widevine"
        licenseServerURL: "http://msfrn-ci-cp-dev.mobitv.com/widevine/get_license"
    }
}
```

Playback Configuration Attributes

Playback configuration meta-data attributes are used to configure the playback of the content item.

<table>
<tr><th>Attribute</th><th>Type</th><th>Values</th><th>Example</th></tr>
<tr><td>Live</td><td>Boolean</td><td>Optional flag indicating video is live. Replaces time remaining in progress bar to display "Live". Default is false.</td><td>True</td></tr>
<tr><td>Url</td><td>String</td><td>Image URL for roSlideShow or roImageCanvas, stream URL for Scene Graph Video node</td><td>http://www.myco.com/img/vacation.jpg</td></tr>
<tr><td>SDBifUrl</td><td>String</td><td>BIF URL for SD trick mode</td><td>http://www.myco.com/bif/sd1932.bif</td></tr>
<tr><td>HDBifUrl</td><td>String</td><td>BIF URL for HD trick mode</td><td>http://www.myco.com/bif/hd1932.bif</td></tr>
<tr><td>FHDBifUrl</td><td>String</td><td>BIF URL for FHD trick mode</td><td>http://www.myco.com/bif/fhd1932.bif</td></tr>
<tr><td>Stream</td><td>roAssociativeArray</td><td>Supported by roVideoPlayer and roVideoScreen, but not the Roku Scene Graph Video node. For the Video node, use the top level url, streamformat, etc. attributes. The exception is cases where you don't have adaptive streams (typically MP4) and need to specify different bitrate variants separately. For this use case use the Streams attribute.

roAssociativeArray that has parameters representing the stream settings that were set as individual roArrays in previous firmware revisions. The old method is still supported and descriptions of the parameters can be found under those content-meta data entries. For <code>url</code> please see <code>StreamUrls</code>, for <code>quality</code> it is now a Boolean that is true for HD quality.

<table><tr><th>Key</th><th>Type</th></tr><tr><td>url</td><td>String</td></tr><tr><td>bitrate</td><td>Integer</td></tr><tr><td>quality</td><td>Boolean</td></tr><tr><td>contentid</td><td>String</td></tr><tr><td>stickyredirects</td><td>Boolean</td></tr></table></td><td>{
url : "http://me.com/big.m3u8",
quality : true
contentid : "big-hls"
}</td></tr>
<tr><td>Streams</td><td>roArray of roAssociativeArrays</td><td>Used by roVideoPlayer and roVideoScreen to specify the content metadata for a set of fixed bitrate streams. Each array item specifies the URL, bitrate, etc. for one stream variant.

Please note that setting stream content metadata using the Streams value is recommended for non-adaptive video (such as MP4 progressive download) only. For adaptive streaming, use the Stream metadata value.

<table><tr><th>Key</th><th>Type</th></tr><tr><td>url</td><td>String</td></tr><tr><td>bitrate</td><td>Integer</td></tr><tr><td>quality</td><td>Boolean</td></tr><tr><td>contentid</td><td>String</td></tr><tr><td>stickyredirects</td><td>Boolean</td></tr></table></td><td>[
{
url : "http://me.com/x-384.mp4",
bitrate : 384
quality : false
contentid : "x-384"
},
{
url : "http://me.com/x-2500.mp4",
bitrate : 2500
quality : true
contentid : "x-1500"
}
]</td></tr>
</table>

StreamBitrates	roArray	Array of bitrates in kbps for content streams used. Please note that setting stream bitrates using this value is recommended for non-adaptive video (such as MP4 progressive download) only. Must be used in conjunction with `StreamUrls` and `StreamQualities`.	[384, 500, 1000, 1500]
StreamUrls	roArray	Array of URL's for content streams. Please note that setting stream urls using this value is recommended for non-adaptive video (such as MP4 progressive download) only. Must be used in conjunction with `StreamBitrates` and `StreamQualities`.	["http://www.myco.com/vid/1932-1.mp4", "http://www.myco.com/vid/1932-2.mp4", "http://www.myco.com/vid/1932-3.mp4", "http://www.myco.com/vid/1932-4.mp4"]
StreamQualities	roArray	Array of Strings quality indicators identifying a stream as "SD" or "HD" Must be used in conjunction with `StreamBitrates` and `StreamUrls`.	["SD", "SD", "SD", "HD"]
StreamContentIDs	roArray	Array of String values logged in Roku logs to identify stream and bitrate played.	["myco-19321-384.mp4", "myco-19321-500.mp4", "myco-19321-1000.mp4", "myco-19321-1500.mp4"]
StreamStickyHttpRedirects	roArray	Array of Boolean values indicating if the HTTP endpoint should be sticky and not subject to change on subsequent requests. Default is false.	[false, false, false, false]
StreamStartTimeOffset	Integer	Optional. Default is 0. The offset into the stream which is considered the beginning of playback. Time in seconds.	3600 (one hour)

StreamFormat	String	Type of content Default: H.264/AAC in .mp4 Container **Valid values:** "mp4", "wma", "mp3" *Note: mp4 will also accept .mov and .m4v files.* "hls" "ism" (smooth streaming) "dash" (MPEG-DASH) "mkv", "mka", "mks" **Deprecated:** "wmv"	
Length	Integer	Movie Length in Seconds Length zero displays at 0m, Length not set will not display.	3600 (one hour)
BookmarkPosition	Integer	BookmarkPosition sets the default start position, in seconds, for this content. The player will start playback at this position in the content unless an explicit seek position was set. An explicit seek position can be set by calling seek on the player or after a user has selected a starting point via the trickplay UI. Users are allowed to seek to positions prior to BookmarkPostion in the content. This value takes precedence over the PlayStart value. Starting from firmware version 8.1, this attribute is made obsolete. The existing `PlayStart` attribute does the same thing, and there is no need to have two attributes to set the play start time. The firmware will continue to support `BookmarkPosition` to maintain the backward compatibility, but channels should use `PlayStart`.	1950 (32 minutes, 30 seconds)
PlayStart	Integer	PlayStart defines the start position of the content, in seconds. The player is not allowed to move to a position prior to this point. Any seek operation prior to this point will be clipped to PlayStart. Channels can use PlayStart and PlayDuration to split one content piece into multiple clips and insert these clips with other content (typically advertisements) into one content list. Starting from firmware 8.0, content metadata supports negative PlayStart values. This feature allows the media players to start playbacks distanced from the edge of the live stream.	0
PlayDuration	Integer	Optional Playback Max Duration in Seconds PlayDuration has been deprecated is is no longer used by the media player starting from firmware version 8.1.	120 (two minute preview)
ClosedCaptions	Boolean	Boolean indicating if CC icon should be displayed.	True
HDBranded	Boolean	Boolean indicating if HD branding should be displayed	True
IsHD	Boolean	Boolean indicating if content is HD	True

SubtitleColor	String	Theme metadata attribute that specifies the color to use when rendering subtitle text	"#FF0000"
SubtitleConfig	roAssociativeArray: { TrackName: *String* }	Specifies the caption settings for content playback. TrackName sets the name of the caption track to render. This string is a concatenation of the track source and track id, separated by a "/". Valid track sources are: "ism", "mkv", "eia608" and "dvb". The track id must match the track identifier in the smooth ormkvmanifest. Forexampleif anmkvfile has a caption track called “english1” the TrackName to select this track is “mkv/english1”. When the track source is "dvb", the track id is the three-letter language code, with "_sdh" appended for subtitles for the deaf and hard of hearing. For example, "dvb/eng_sdh" are English subtitles for the deaf and hard of hearing and "dvb/nor" are normal Norwegian subtitles. For sideloaded caption tracks, the TrackName is the url from where the caption track can be downloaded. Sideloaded caption formats can include srt,ttml, anddfxp. Specifying eia608/1 will trigger the firmware to search for embedded 608/708 captions in the stream. In the 8.0 firmware, automatic track selection based on a preferred caption language setting is introduced. Omit setting a URL here to avoid interfering with the automatic track selection. It is sufficient to add the URLs to SubtitleTracks.	{ TrackName : "mkv/english1" }
SubtitleTracks	roArray of roAssociativeArray: [{ Language: *String* Description: *String* TrackName: *String* },...]	SubtitleTracks sets the list of available caption tracks available to the stream. This list is added to the track list in the closed caption configuration dialog that is displayed during video playback when the user presses the Roku remote control * button. The captions from the selected track are then displayed on the screen. Language specifies the ISO 639.2B 3 character language code. This string is used to match the proper caption track with the audio language. Description specifies the text that will be shown for the corresponding track in the closed caption configuration dialog. For sideloaded caption tracks the TrackName is the URL from where the caption track can be downloaded. Sideloaded caption formats can include srt, ttml, and dfxp. The SubtitleTracks metadata is generally only used for side loaded captions. The Roku firmware detects in-stream captions and thus specifying SubtitleTracks in this case is not necessary.	
SubtitleUrl	String	Specifies the path to an SRT or TTML formatted file used to render subtitles or closed captions, respectively. This is supported on roVideoScreen only. See Closed Caption Support for additional details.	"http:// www.myco.com/vid/1932.srt" "http://www.myco.com/vid/1932.xml"
VideoDisableUI	Boolean	If set to true, hides the Scene Graph Video node trick play UI If set to false (the default) shows the Scene Graph Video node trick play UI	video = createObject("roSGNode", "Video") video.content.VideoDisableUI = true

EncodingType	String	Specifies the encoding scheme for PlayReady DRM, by setting to one of the following values: PlayReadyLicenseAcquisitionUrl If specified, the EncodingKey attribute contains the PlayReady license acquisition URL PlayReadyLicenseAcquisitionAndChallenge If specified, the EncodingKey attribute contains the PlayReady licen se acquisition URL plus additional custom request data	
EncodingKey	String	Specifies the PlayReady license acquisition URL, and additional custom request data, determined by the EncodingType attribute value specified: PlayReadyLicenseAcquisitionUrl TheEncodingKey attribute contains the PlayReady license acquisition URL PlayReadyLicenseAcquisitionAndChallenge The EncodingKey attribute contains the PlayReady license acquisition URL. It also contains additional custom license acquisition URL request data. In this case the EncodingKey string uses the format "URL%%%<customdata>. For example: EncodingKey = "http://ipaddress/mylicense%%%data"	
SwitchingStrategy	String	roVideoPlayer or roVideoScreen: Specify different stream switching algorithms to be used in HLS adaptive streaming. Only has an effect on HLS streams. "full-adaptation" Uses measured bandwidth and buffer fullness to determine when to switch. This strategy requires that segments align across variants as the HLS spec requires. This is the new default.	"full-adaptation"
Watched	Boolean	Flag indicating if content has been watched	True
ForwardQueryStringParams	Boolean	Controls whether query string parameters from initial HLS stream manifest requests are forward to subsequent segment download requests. The default value is set to true for backwards compatibility. *Available since firmware version 7.5*	True
IgnoreStreamErrors	Boolean	When set to true the media player will not stop playback when it runs into a streaming related error for this content. Instead, it will skip to the next item in the content list. If this was the last item in the content list the media player will send a regular completion event (like isFullResult). Channels are still notified of any errors via an isRequestFailed notification but a new attribute in the event's GetInfo object tells the channel the error was ignored. See the changes related to isRequestFailed for more information. The default value is false. *Available since firmware version 7.5*	video_details = { streamFormat: "mp4" ignoreStreamErrors: true streams: [{bitrate: 537, height: 360, width: 640, url: "https://..."}] }
AdaptiveMinStartBitrate	Integer	Minimum startup bitrate specified in kbps. Streaming will start with a variant equal to or greater than this value. If this value is not set or if it's set to zero, any minimum start bitrate will be ignored. *Available since firmware version 7.5*	5000

AdaptiveMaxStartBitrate	Integer	Maximum startup bitrate specified in kbps. Streaming will start with a variant less than or equal to this value. If this value is not set, it will default to 2500 kbps. *Available since firmware version 7.5*	2000
filterCodecProfiles	Boolean	Filters out any video profile/codec level combinations that the specified hardware cannot play. The default value is false, in which case no filtering occurs. Note that this currently only works for DASH streams. *Available since firmware version 8.0*	True
LiveBoundsPauseBehavior	String	Allows a channel to customize Media Player behavior on live streams when playing in the earliest part of a DVR buffer. The stream remains paused even though it is playing in the earliest part of the buffer of a live stream when the value of the attribute is set to "pause." This enables the firmware to distinguish between live streams and live streams that eventually transition to video on demand. The possible values of this attribute are "resume", "stop", "pause", with resume being the default value. Currently, this attribute will work only with Smooth and Dash streams. *Available since firmware version 8.1*	Resume
ClipStart	Float	ClipStart sets the clip start position of the playback. The unit of ClipStart is seconds. *Available since firmware version 8.1*	00.0
ClipEnd	Float	ClipEnd sets the clip end position. The unit of ClipEnd is seconds. *Available since firmware version 8.1*	00.0
preferredaudiocodec	String	Specifies the audio codec that should be used during playback. The Media Player will select and report to the channel only those audio renditions that are encoded with the specified codec. Renditions that are encoded with a different codec are ignored. Possible values of this attribute are "aac", "ac3" and "eac3". *Available since firmware version 9.0*	aac

Scene Graph Certificate Attributes

The Scene Graph certificate meta-data attributes are used to configure the use of HTTP certificates and cookies by the Audio and Video nodes. Please note that when setting any of the following four attributes on a Video or Audio node, you need to be careful that the values are set on the correct HTTPAgent. If the node does not have its own HTTPAgent, set by explicitly calling setHttpAgent() on the node, the Roku OS will traverse up the scene graph hierarchy until it finds the first node in the Video or Audio node's ancestry that has set an HTTPAgent. If none is found, the values will be set on the global HTTPAgent which is always guaranteed to exist. Therefore if you expect the header, etc. values set to only apply to your Audio and Video nodes, create a unique instance of roHttpAgent for them and assign it directly. For example, for a Video node you should do the following:

```
'Assume video is a valid Video node instance
httpAgent = CreateObject("roHttpAgent")
video.setHttpAgent(httpAgent)
```

Attribute	Type	Values	Example
HttpCertificatesFile	uri	If set, the Scene Graph Audio or Video node loads this public certificate bundle, to authenticate the server. The protocol must be https for this to have any effect. When used with a Scene Graph Audio or Video node, the node or global HttpAgent is found, as explained elsewhere in this documentation. When playing this content, the agent is updated in the following manner: • If this attribute is defined, the file URI is set into the HttpAgent instance. However, if this attribute is specified and the value is the empty string (""), then no changes will be made to the HttpAgent. • If this attribute is not defined, the behavior depends upon whether the Content Meta-Data (CMD) contains secure (https) URLs: • If no secure URLs exist in the meta-data, then no certificates file path is set into the agent. • If a secure URL does exist, the platform's default certificates are set into the agent.	
HttpCookies	array of strings	If set, the Scene Graph Audio or Video node send the cookies to the server. Each cookie must have the following syntax: `dom=domain;path=path;name=name;val=value;` When used with a Scene Graph Audio or Video node, the node or global HttpAgent is found, as explained elsewhere in this documentation. When this Content Meta-Data is played and this attribute is set, all HTTP cookies in the agent are cleared and replaced with the cookies defined by this attribute.	
HttpHeaders	array of strings	If set, the Scene Graph Audio or Video node sends these headers to the server. Each string must be of the format `"name:value"` When used with a Scene Graph Audio or Video node, the node or global HttpAgent is found, as explained elsewhere in this documentation. When this Content Meta-Data is played and this attribute is set, all HTTP headers in the agent are cleared and replaced with the headers defined by this attribute.	
HttpSendClientCertificate	Boolean	If true, the Scene Graph Audio or Video node sends the client device certificate to the server, for client authentication. The protocol must be https for this to have any effect. When used with a Scene Graph Audio or Video node, the node or global HttpAgent is found, as explained elsewhere in this documentation. When this Content Meta-Data is played and this attribute exists, the value of this attribute (true or false) is set into the HttpAgent.	

Playback Control Attributes

The playback control meta-data attributes are used to control the playback parameters for the content item.

Attribute	Type	Values	Example
MinBandwidth	Integer	roVideoPlayer or roVideoScreen: Will only select variant streams with a bandwidth higher than this minimum bandwidth. Units are kbps. By default Wowza servers set streams to 64 kbs, so you might want to set this parameter to something smaller than 64 when first testing Wowza streams. You will eventually want to specify the Wowza bitrates with a smil file (Please see the encoding guide.)	48
MaxBandwidth	Integer	roVideoPlayer or roVideoScreen: Will only select variant streams with a bandwidth less than this maximum bandwidth. Units are kbps.	2500

AudioPIDPref	Integer	roVideoPlayer or roVideoScreen If the specified preferred PID audio stream exists, it will be chosen. Otherwise the last audio stream will be chosen. This is valid only for HLS streams. This attribute is deprecated. Use AudioLanguageSelected instead for channels running on 4.8 and later firmware.	483
FullHD	Boolean	roVideoPlayer or roVideoScreen Specify that this stream was encoded at 1080p resolution.	true
FrameRate	Integer	roVideoPlayer or roVideoScreen Specify the 1080p stream was encoded at 24 or 30 fps.	24

Track ID Attributes

Attribute	Type	Values	Example
TrackIDAudio	String	roVideoPlayer or roVideoScreen: Used in SmoothStreaming (StreamFormat = "ISM") to specify Set the TrackIDAudio field to the desired track's *StreamIndex.Name* attribute from the manifest file.	"Spanish"
TrackIDVideo	String	roVideoPlayer or roVideoScreen: Used in SmoothStreaming (StreamFormat = "ISM") to specify Set the TrackIDVideo field to the desired track's *StreamIndex.Name* attribute from the manifest file.	"h264video"
TrackIDSubtitle	String	roVideoPlayer Used to specify a closed caption track in a video stream that supports 608/708 captions.	"eia608/1"
AudioFormat	String	roSpringboardScreen If set to "dolby-digital", will display a "5.1))" icon in the lower left of a movie style springboard screen.	"dolby-digital"
AudioLanguageSelected	String	roVideoPlayer or roVideoScreen: An ISO-639 3-letter language code. If multiple language tracks are available in the content, this specifies the one that should be used.	"eng"

roListScreen Attributes

Attribute	Type	Values	Example
SDBackgroundImageUrl	String	roListScreen: URL for the SD background image	http://www.myco.com/images/bg1_sd.jpg
HDBackgroundImageUrl	String	roListScreen: URL for the HD background image	http://www.myco.com/images/bg1_hd.jpg

roImageCanvas Attributes

Attribute	Type	Values	Example

SourceRect	Associative Array: { x: Integer y: Integer w: Integer h: Integer }	roImageCanvas: The rectangle from the image that should be drawn. The default is the entire image at the origin. *Note the values must be of type Integer. Other numeric types such as Float will be ignored and treated as 0.* *Note that BrightScript returns a Float by default when doing division.* *You can convert calculated values to integer values using the Int() or Fix() functions, for example, if needed.*	{ x : 100, y : 100, w : 200, h : 200 }
TargetRect	Associative Array: { x: Integer y: Integer w: Integer h: Integer }	roImageCanvas: The rectangle where the text/or image should be drawn. The default is the entire image at the origin. Can also be used with roFontMetrics to provide a target rect for the desired font size. *Note the values must be of type Integer. Other numeric types such as Float will be ignored and treated as 0.* *Note that BrightScript returns a Float by default when doing division.* *You can convert calculated values to integer values using the Int() or Fix() functions, for example, if needed.*	{ x : 400, y : 200, w : 200, h : 200 }
TargetTranslation	Associative Array: { x: Integer y: Integer }	roImageCanvas: The amount to translate the coordinate system prior to drawing the image and/or text. Translation is done before rotation. Default: {0, 0} *Note the values must be of type Integer. Other numeric types such as Float will be ignored and treated as 0.* *Note that BrightScript returns a Float by default when doing division.* *You can convert calculated values to integer values using the Int() or Fix() functions, for example, if needed.*	{ x : 100, y : 100 }
TargetRotation	Float	roImageCanvas: The angle (in degrees) to rotate the coordinate system prior to drawing the image and/or text. Default: 0	90.0
CompositionMode	String	roImageCanvas: Either "**Source**" (where source pixels completely replace destination pixels) or "**Source_Over**" (where source pixels are alpha blended with destination pixels).	"Source_Over"
Text	String	roImageCanvas: The text is drawn into the TargetRect.	"Hello ImageCanvas"
TextAttrs	Associative Array	See **TextAttrs Attribute Keys** for descriptions of the array keys.	{ Color : "#FFCCCCCC", Font : "Medium", HAlign : "HCenter", VAlign : "VCenter", Direction : "LeftToRight" }

TextAttrs Attribute Keys

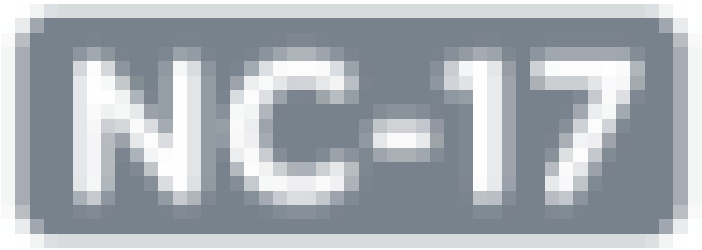
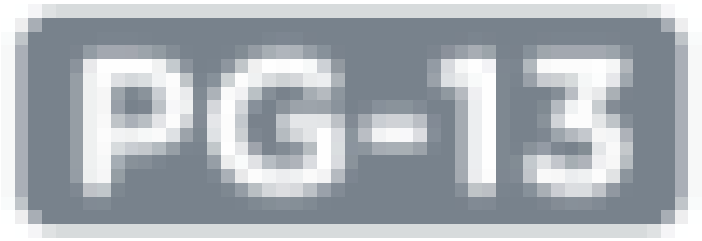
Key	Type	Description	Example

Color	String	roImageCanvas (applies to text and other colors): A color can be specified as an opaque color 24-bit RGB value (#RRGGBB) or as a color with alpha 32-bit ARGB value (#AARRGGBB). AA (alpha), RR (red), GG (green), and BB (blue) are 8-bit values specified as hexadecimal values 00..FF. When specified as a 32-bit color, the alpha channel bits in the color value range from 00=transparent to FF=opaque. The alpha channel enables blending the background with the images. Default: "#FFFFFF" (opaque white)	"#FF0033FF"
Font	String	roImageCanvas (applies to text): "Small","Medium","Large", or "Huge". Default: "Medium" Also can use any fonts registered by the roFontRegistry Object and returned by its Get() method	"Large"
HAlign	String	roImageCanvas (applies to text): Controls the horizontal alignment of the text in the TargetRect. Options are: "Left", "HCenter" / "Center" / "Middle", "Right", "Justify" Default: "HCenter"	"Right"
VAlign	String	roImageCanvas (applies to text): Controls the vertical alignment of the text in the TargetRect. Options are: "Top", "VCenter" / "Center" / "Middle", "Bottom" Default: "VCenter"	"Bottom"
TextDirection	String	roImageCanvas (applies to text): "LeftToRight" or "RightToLeft" Default: "LeftToRight"	"RightToLeft"

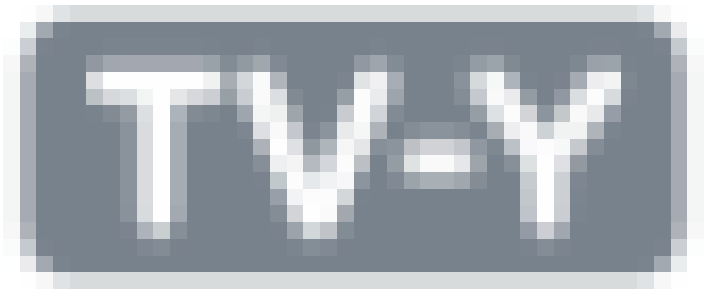
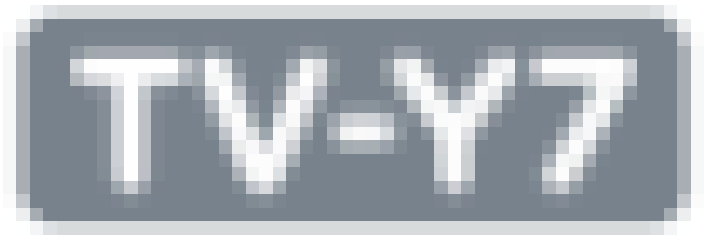
Rating Attribute Icons

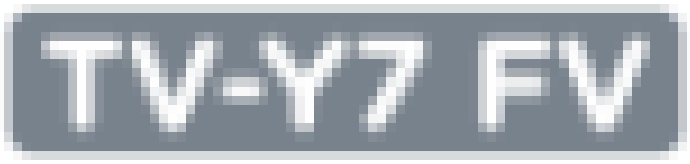
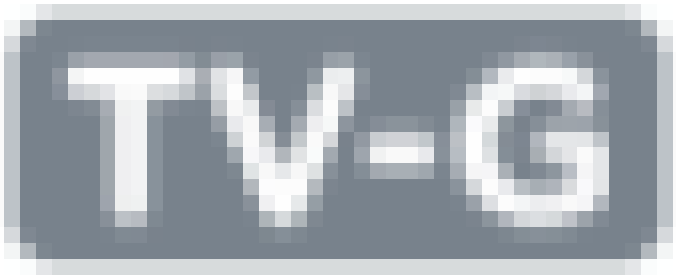
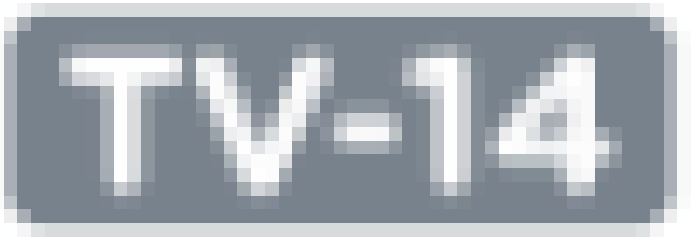
The Rating attribute contains the MPAA or TV rating stored as a string. At runtime, the ratings are shown with an icon instead of rendering the string as text. The following table shows the list of valid values for the Rating attribute, and the resulting icon that will be displayed for each value.

Value	Icon
G	

NC-17	
PG	
PG-13	

R	
UR	

UNRATED	
NR	
TV-Y	
TV-Y7	

TV-Y7-FV	
TV-G	
TV-PG	
TV-14	
TV-MA