

Bookmarking Playback Position

Table of contents:

- [Overview](#)
 - [Retrieving playback position](#)
 - [Storing timestamps for cross-platform retrieval](#)
 - [Storing timestamps on-device for local retrieval](#)
-

Overview

When it comes to streaming media, content bookmarking refers to the idea of recording a user's playback position on the content in your channel so that later on they will be able to continue watching from precisely where they left off.

It is recommended that all services with content longer than 15 minutes build bookmarking functionality into their service, as it will enhance the end-user's experience — not just on Roku, but across all interactions they have with your service on any platform. For example, if your service implements bookmarks, then users who begin watching a television show on Roku before switching over to their mobile device can pick up where they left off.

This guide provides simple instructions on two different ways to bookmark media content, either in your service's backend or locally in the device's registry.

Retrieving playback position

It is important to know that video playback position (or "timestamp") can be retrieved via the position field in the video node.

```
m.video = m.top.findNode("MyVideo")
TimeStamp = m.video.position
```

Storing timestamps for cross-platform retrieval

It is best practice to store the timestamp of a user's bookmark position in the service's backend, so that it can be retrieved on any platform, not just Roku.

To do this, the channel must first retrieve the timestamp as outlined above, then make a request to store the timestamp on the service's backend. This ensures that when starting media playback on other platforms, the developer can load the previously-saved timestamp.

It is recommended that the channel makes the request to store this timestamp on the backend once every 30 seconds, but the frequency can be increased on devices with more memory. This concept is very similar to beacons fired by the Roku Ad Framework. The best way to approach this is through `roUrlTransfer`.

```
url = ('url with timestamp to send to developer end')
curl = createObject("roUrlTransfer")
curl.setUrl(url)
curl.postFromString(TimeStamp as String)
```

This should be done on a 30 second timer to ensure functionality across all devices.

Storing timestamps on-device for local retrieval

While it is ideal to store timestamps in your backend service, it is also possible to store it locally on a Roku device's registry. With this approach, a user will only be able to resume watching from their last playback position if they use the same Roku device. The timestamp won't be accessible on other platforms.

To write to the registry, use the `roRegistrySection` component.

```
sec = createObject("roRegistrySection", "MySection")
if sec.Exists("PlaybackBookmark")
    BookmarkTime = sec.Read("PlaybackBookmark")
end if
```

If the roku device has a previously stored value that matches the `PlaybackBookmark` key, then it will return the value stored inside the registry. The function below shows how to create a key value pair to store the timestamp of a bookmark. The timestamp must be done in seconds.

```
TimeStamp = 360
sec = createObject("roRegistrySection", "MySection")
sec.Write("PlaybackBookmark", TimeStamp)
sec.Flush()
```

This will save the media playback position inside the registry and the `Flush()` method will save it to persistent storage in the case of a reboot. Note that if you are running this multiple times on a timer, it will overwrite any previous value associated with the same key. Once this is done, all that's left is to find run the `seek()` function from the video node to resume playback from the last point.

```
m.video = m.top.findNode("MyVideo")
m.video.content = videoContent
m.video.control = "play"
m.video.seek = BookmarkTime
```