Example

Playback - Set fade-in time

by:	Sebastian Beutel
published:	August 2018
description:	Set a playback's fade-in time
remarks:	

This is also an example how `time:x` is parsed into a time value.

playback, fade-in, time

functions

- Handles.ClearSelection
- ActionScript.SetProperty
- Playbacks.Editor.EnsurePlaybackCueSelected

affected properties

- Playbacks.Editor.SelectedPlayback
- Handles.SourceHandle
- Playbacks.Editor.Times.CueFadeInTime

control structures

- step condition

Code

SetPBFadeInTime.xml

```
<?xml version="1.0" encoding="utf-8"?>
<avolites.macros>
  <macro id="Wiki.Macros.Fadeintime3" name="PB Fade-In Time 3s">
    <sequence>
      <step>Handles.ClearSelection()</step>
<step>ActionScript.SetProperty("Playbacks.Editor.SelectedPlayback",
null)</step>

      <step>Handles.SetSourceHandle("PlaybackWindow", 0)</step>
      <step condition="Playbacks.IsCueHandle(Handles.SourceHandle)">
        ActionScript.SetProperty("Playbacks.Editor.SelectedPlayback",
Handles.SourceHandle)</step>
      <step>Playbacks.Editor.EnsurePlaybackCueSelected()</step>
```

```
<step>ActionScript.SetProperty("Playbacks.Editor.Times.CueFadeInTime",
time:3)</step>

      <step>Handles.SetSourceHandle("PlaybackWindow", 1)</step>
      <step condition="Playbacks.IsCueHandle(Handles.SourceHandle)">
        ActionScript.SetProperty("Playbacks.Editor.SelectedPlayback",
Handles.SourceHandle)</step>
      <step>Playbacks.Editor.EnsurePlaybackCueSelected()</step>
<step>ActionScript.SetProperty("Playbacks.Editor.Times.CueFadeInTime",
time:3)</step>

      <step>Handles.ClearSelection()</step>
<step>ActionScript.SetProperty("Playbacks.Editor.SelectedPlayback",
null)</step>
    </sequence>
  </macro>
</avolites.macros>
```

Explanation

This explains the functional steps within the sequence. For all the other XML details please refer to Formats and syntax

- The first and last two lines `Handles.ClearSelection()` and `ActionScript.SetProperty("Playbacks.Editor.SelectedPlayback", null)` make handle and playback selection empty so that no other handle/playback is affected by accident.
- `Handles.SetSourceHandle("PlaybackWindow", 0)` selects a playback as source handle
- `<step condition="Playbacks.IsCueHandle(Handles.SourceHandle)">`: only if this is a cue playback handle (and not maybe a macro or group handle which would break the macro)...
- ... `ActionScript.SetProperty("Playbacks.Editor.SelectedPlayback", Handles.SourceHandle)` derive the playback from the given handle (the playback holds the time, not the handle)
- `Playbacks.Editor.EnsurePlaybackCueSelected()` is required for single-cue playbacks to make sure the cue itself is selected/in the editor
- `ActionScript.SetProperty("Playbacks.Editor.Times.CueFadeInTime", time:3)` finally sets the fade-in time to the value. `time:3` tells the parser what to do with the number 3, i.e. parse it as time value.

Note that this requires the first playback to be a single cue. If it is e.g. a cue list or a macro then the property `Playbacks.Editor.SelectedPlayback` isn't set and the macro fails.

How to use it

- make this macro available

- while it's well possible to create some macros to set some playbacks to a given fade-in time, it does make more sense to integrate this into a bigger context like Create Workspaces in order to setup a known show environment.

From:
https://avosupport.de/wiki/ - **AVOSUPPORT**

Permanent link:
https://avosupport.de/wiki/macros/example/setplaybackfadeintime?rev=1619021658

Last update: **2021/04/21 16:14**