

Example

# Playback - Set fade-out time

<b>by:</b>	Sebastian Beutel, July 2020
<b>published:</b>	here
<b>description:</b>	sets some playbacks' fade-out time also this is a good example for various ways to select playback handles
<b>remarks:</b>	

[playback](#), [fade-out](#), [time](#), [handles](#)

## functions

- [Handles.ClearSelection](#)
- [ActionScript.SetProperty](#)
- [Handles.SetSourceHandle](#)
- [Playbacks.IsCueHandle](#)
- [Playbacks.Editor.EnsurePlaybackCueSelected](#)

## affected properties

- [Playbacks.Editor.SelectedPlayback](#)
- [Handles.SourceHandle](#)
- [Playbacks.Editor.Times.CueFadeOutTime](#)

## control structures

- [step condition](#)

## Code

[SetPBFadeOutTime.xml](#)

```
<?xml version="1.0" encoding="utf-8"?>
<avolites.macros>
  <macro id="Wiki.Macros.FadeOutTime3" name="PB Fade-Out Time 3s">
    <!-- sets the Fade-Out time for playbacks in slots 1 and 2 on the
    current page of the playbacks window -->
    <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.CueFadeOutTime",
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.CueFadeOutTime",
time:3)</step>

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

<macro id="Wiki.Macros.FadeOutTime1.5" name="PB Fade-out Time 1.5s">
    <!-- sets the Fade-Out time for playbacks in slots 1 and 2 on page 1
of the playbacks window -->
    <sequence>
        <step>Handles.ClearSelection()</step>
<step>ActionScript.SetProperty("Playbacks.Editor.SelectedPlayback",
null)</step>

<step>ActionScript.SetProperty("Playbacks.Editor.SelectedPlayback",
handle:"Location=Playbacks,1,1")</step>
<step>ActionScript.SetProperty("Playbacks.Editor.Times.CueFadeOutTime",
time:1.5)</step>

<step>ActionScript.SetProperty("Playbacks.Editor.SelectedPlayback",
handle:"Location=Playbacks,1,2")</step>
<step>ActionScript.SetProperty("Playbacks.Editor.Times.CueFadeOutTime",
time:1.5)</step>

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

<macro id="Wiki.Macros.FadeOutTime0" name="PB Fade-out Time 0s">
    <!-- sets the Fade-Out time for playbacks with user numbers 101 and
102 -->
    <sequence>
        <step>Handles.ClearSelection()</step>
<step>ActionScript.SetProperty("Playbacks.Editor.SelectedPlayback",
null)</step>
```

```

<step>ActionScript.SetProperty("Playbacks.Editor.SelectedPlayback",
handle:"playbackHandleUN=101")</step>
<step>ActionScript.SetProperty("Playbacks.Editor.Times.CueFadeOutTime",
time:0)</step>

<step>ActionScript.SetProperty("Playbacks.Editor.SelectedPlayback",
handle:"playbackHandleUN=102")</step>
<step>ActionScript.SetProperty("Playbacks.Editor.Times.CueFadeOutTime",
time:0)</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](#)

This example shows various ways to select playback handles: by index on the current page, by location, and by user number.

- the first and last two lines `Handles.ClearSelection()` and `ActionScript.SetProperty("Playbacks.Editor.SelectedPlayback", null)` always make sure that there is no playback selected from previous actions
- in the first macro the playbacks are selected using `Handles.SetSourceHandle("PlaybackWindow", 0)`. This refers to the index of the button on the current page of this windows. however we cannot be sure that an actual playback is there in this slot - and if there is not then the macro will cause an error. Thus we need to make sure that an actual playback is selected as source handle:
- `<step condition="Playbacks.IsCueHandle(Handles.SourceHandle)">` checks that this really is a cue handle
- only if this check yields a positive result `ActionScript.SetProperty("Playbacks.Editor.SelectedPlayback", Handles.SourceHandle)` sets this as source handle for the next operations
- `Playbacks.Editor.EnsurePlaybackCueSelected()` makes sure a cue is selected
- `ActionScript.SetProperty("Playbacks.Editor.Times.CueFadeOutTime", time:3)` finally sets the fade-out time

In the other two macros the playbacks are directly selected using either the location (`ActionScript.SetProperty("Playbacks.Editor.SelectedPlayback", handle:"Location=Playbacks,1,1")`) or the user number (`ActionScript.SetProperty("Playbacks.Editor.SelectedPlayback",`

handle: "playbackHandleUN=101" ). Thus it's possible to omit some of the checks.

## How to use it

- [make this macro available](#)
- it makes much sense to organise the playbacks in a certain way so that you know which playbacks are affected by your macros.
- fire the macros to you liking in order to set your playbacks' fade-out time

From:

<https://avosupport.de/wiki/> - **AVOSUPPORT**

Permanent link:

<https://avosupport.de/wiki/macros/example/setplaybackfadeouttime>

Last update: **2021/07/21 10:22**

