Example

Double chase speed

by:	Gregory Haynes, Feb 2016
published:	http://forum.avolites.com/viewtopic.php?f=20&t=3744#p15848
description:	double current chase's BPM rate
remarks:	The problem with this code is that it requires the chase to be set as the selected playback in the editor (used for the Edit Times menu). This means that the Playback View which could be showing a cue list switches to show the chase instead, to get around this I have stored the previous selected playback and restore that afterwards but this does cause the display to flicker.

functions

- ActionScript.SetProperty
- ActionScript.SetProperty.Float
- Math.Min
- Math.IsEqual
- SimpleMath

affected properties

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

This macro works perfectly well in Titan 9.1, does misbehave in 10.0 (always sets speed to 1 BPM), and throws an error in 10.1 (variable not registered).

Code

doublechasespeed.xml

```
<?xml version="1.0" encoding="utf-8"?>
<avolites.macros xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="Avolites.Menus.xsd">
 <macro name="Chase Speed Double"
id="Avolites.Macros.ChaseSpeedDouble">
    <description>Double the speed of the currently connected
chase.</description>
    <sequence>
      <step>ActionScript.SetProperty("Handles.SourceHandle",
Playbacks.Editor.SelectedPlayback)</step>
<step>ActionScript.SetProperty("Playbacks.Editor.SelectedPlayback",
Chases.ConnectedHandle)</step>
      <step>Math.Min("Playbacks.Editor.Times.ChaseSpeed",
```

```
Playbacks.Editor.Times.ChaseSpeed * 2, 3600.0)</step>
      <step condition="Math.IsEqual(Playbacks.Editor.Times.ChaseSpeed,
0.0)">ActionScript.SetProperty.Float("Playbacks.Editor.Times.ChaseSpeed
", 1.0)</step>
<step>ActionScript.SetProperty("Playbacks.Editor.SelectedPlayback",
Handles.SourceHandle)</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

- `ActionScript.SetProperty("Handles.SourceHandle", Playbacks.Editor.SelectedPlayback)`
  temporarily stores the currently selected playback into 'Handles.SourceHandle' from where it will be retrieved when the macro is finished, see Gregory's explanation above
- `ActionScript.SetProperty("Playbacks.Editor.SelectedPlayback", Chases.ConnectedHandle)`
  this selects the currently connected chaser for editing
- `Math.Min("Playbacks.Editor.Times.ChaseSpeed", Playbacks.Editor.Times.ChaseSpeed * 2, 3600.0)`
  sets the speed of the chaser to its double, or 3600.0, whatever the smaller value - this way it will never be faster than 3600 BPM
- `<step condition="Math.IsEqual(Playbacks.Editor.Times.ChaseSpeed, 0.0)">ActionScript.SetProperty.Float("Playbacks.Editor.Times.ChaseSpeed", 1.0)</step>`
  if the speed equals 0.0: sets speed to 1.0
- `ActionScript.SetProperty("Playbacks.Editor.SelectedPlayback", Handles.SourceHandle)`
  restores the selected playback from before the macro was run

How to use it

This macro works perfectly well in Titan 9.1, does misbehave in 10.0 (always sets speed to 1 BPM), and throws an error in 10.1 (variable not registered).

- make this macro available
- with a chase running and connected (Speed and Cross Fade on the wheels), triggering this macro doubles the speed of the chase