Example

Palette - Fire Quick Palettes and Show Active

by:	Sebastian Beutel
published:	April 2025 (stems from 2023)
description:	fire some quick palettes and show active state
remarks:	useful to see which active palette is currently fired

quick, palette, active

functions

- ActionScript.SetProperty.Integer
- Palette.ApplyQuickPalette

control structures

- Variables
- Active Binding

Code

File with macros for some more palettes in two segments here:

firequickpalettes.xml

FireQuickPalettes_example.xml

```
<?xml version="1.0" encoding="utf-8"?>
<avolites.macros>

  <macro id="Wiki.Macros.Palettes.FireQuickPaletteInit" name="Init
Palette Vars">
    <description>Initializes variables for other macros.</description>
    <variables>
      <integer id="PaletteRange100" value="100"/>
      <integer id="PaletteRange110" value="110"/>
    </variables>
  </macro>

  <macro id="Wiki.Macros.Palettes.FireQuickPalette101" name="Fire
QuickPalette 101">
    <description>Fires QuickPalette 101 and shows as
active.</description>
    <active binding="{propertyLink
id='Wiki.Macros.Palettes.FireQuickPaletteInit.PaletteRange100'
converter='Math.EqualityConverter' converterParameter='101'}"/>
```

```
    <sequence>
<step>ActionScript.SetProperty.Integer("Wiki.Macros.Palettes.FireQuickPaletteInit.PaletteRange100", 101)</step>
      <step>Palette.ApplyQuickPalette(userNumber: 101, true)</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 macro simply initialises the variables `PaletteRange100` and `PaletteRange110` which are then used to hold which variable is currently active (you need to make sure this follow some logic…).

The second macro `Fire QuickPalette 101` does the work (you need to create some more such macros though):

- `<active binding=…` checks whether this macro should show as active, based on the number in the corresponding variable
- `ActionScript.SetProperty.Integer()` sets the variable to the number of this macro/palette
- `Palette.ApplyQuickPalette(userNumber: 101, true)` fires the corresponding palette as quick palette

How to use it

1. duplicate the second macro and adjust the numbers for the usernumbers you want to work with
2. make this macro available
3. adjust the usernumbers of the palettes you want to work with
4. instead of clickng the palette buttons now fire the macros