

Panel Builder Test Procedure

Version 1.0

General Test Procedure

ACTION: Start a new project and give it any name you want. Create a panel using the provided tools which contains four (4) buttons of various sizes aligned in a 2 x 2 grid. The panel should have a background. Double click the text on each to edit the labels of the four buttons from the top left in a "Z" order, "first" "second" "third" "next panel".

EXPECTED RESULT: At this point the panel should be visually close to what an end product panel may appear like. The panel should have a background and contain four buttons each labeled with text.

ACTION: Create a second panel following the instructions of the first action, only this time add only two (2) buttons and label them "fourth" and "previous panel".

EXPECTED RESULT: The project should now contain two visually acceptable panels and six adequately labeled buttons, four on the first panel, and two on the second panel.

ACTION: Double click the button labeled "first" to edit it and go to the Scripts/Conditional section. For the button's State 1 script, enter or copy and paste the following:

```
this.state = state2  
this.text = first2  
$clientside = first2
```

In its State 2 script enter or paste the following:

```
this.state = state1  
this.text = first  
$clientside = first
```

Do not edit the button's conditional. Then save the button.

EXPECTED RESULT: When you reopen the "first" button's State 1 and State 2 script, they should read as what you saved before.

ACTION: Double click the button labeled "second" to edit it and go to the Scripts/Conditional section. For the button's State 1 script, enter or copy and paste the following:

```
if ($clientside == first2){  
    this.text = clientside is first2  
}  
if ($clientside == first){  
    this.text = clientside is first  
}
```

Do not change the button's State 2 script. Instead in the button's Conditional enter the following:

```
if ($$serverside == 5){  
    this.fontcolor = red  
}  
else {  
    this.fontcolor = black  
}
```

EXPECTED RESULT: When you reopen the "second" button's State 1 script and Conditional, they should read as what you saved before.

ACTION: Double click the button labeled "third" to edit it and go to the Scripts/Conditional section. For the button's State 1 script, enter or copy and paste the following:

```
$$serverside = 1  
this.state = state2  
this.text = serverside is 1
```

In its State 2 script enter or paste the following:

```
$$serverside = 5  
this.state = state1  
this.text = serverside is 5
```

Do not change the button's Conditional.

EXPECTED RESULT: When you reopen the "third" button's State 1 and State 2 script, they should read as what you saved before.

ACTION: Double click the button labeled "next panel" to edit it and go to the Scripts/Conditional section. For the button's State 1 script, enter or copy and paste the following:

```
if ($$serverside == 5 && $clientside == first2){  
    goto 2  
}
```

Do not change the button's State 2 script. Instead change the button's Conditional to read as follows:

```
if ($clientside == first2){  
    this.state = state1  
}  
else {  
    this.state = hidden  
}
```

EXPECTED RESULT: When you reopen the "next panel" button's State 1 script, it should read as what you saved before.

ACTION: Double click the button labeled "fourth" to edit it and go to the Scripts/Conditional section. For the button's State 1 script, try setting the button to execute a command that affects one or more SVSi units on your network.

Do not change the button's State 2 script. Instead write a conditional like the ones above but instead based on some property of an SVSi unit on your network that will determine whether the button is set to state1 or hidden.

EXPECTED RESULT: When you reopen the "fourth" button's State 1 script and Conditional, they should read as what you saved before.

ACTION: Double click the button labeled "previous panel" to edit it and go to the Scripts/Conditional section. For the button's State 1 script, enter or paste the following:

```
goto 1
```

Do not change the button's State 2 script or Conditional.

EXPECTED RESULT: When you reopen the "previous panel" button's State 1 script, it should read as what you saved before.

ACTION: Click the "File" menu and select the "Save" button. When the dialog appears, click the link to take you to your project/panels.

EXPECTED RESULT: A dialog should appear giving you a link to your project. The project should appear in the database. When the link is clicked, the browser should open a new tab or window and the panels that were created in the previous steps should be displayed.

ACTION: On the panels page, start clicking buttons and be very confused. Haha, no, don't worry, you have me here to explain to you what the panel should do now!

EXPECTED RESULT: The 'first' button sets a clientside variable to either 'first' or 'first2'. Before you press this button, the clientside variable is not set to anything, so any other conditional or script that you run which depends on this variable will return false for any checks against it equaling something. The 'first' button has no other function. However, the 'next panel' button's Conditional is set to check the state of the clientside variable and make itself set to hidden or state1 based on whether clientside is 'first' or 'first2'.

The 'second' button simply checks that variable and sets its own text to display the result of that check. If you press the 'second' button before any other buttons, then the button should not change at all because the clientside variable has not been set. If you click the 'first' button it will set the clientside variable. After that if you press the 'second' button, it should display the state of the clientside variable. The conditional on this button checks the serverside variable to see whether it equals 5 and turns the font red, otherwise it turns the font black. If this is the first time you have run this panel/project, then the serverside variable check should evaluate to false because the variable hasnt been set yet, and the text should be black.

The 'third' button when in State 1 sets a serverside variable to 1 and then changes its own text to let you know what it has done, and changes itself to State 2. In State 2 clicking the button sets the serverside variable to 5 and then switches itself back to State 1. Flipping this button back and forth

should change the font color of the 'second' button.

The 'next panel' button's script should move you to the next panel, however, it should only execute if the clientside variable is currently 'first2' and the serverside variable is set to 5.

ACTION: Mess around with the first panel to set the states of the variables and then when ready, move on to the second panel using the 'next panel' button (make sure that serverside equals 5 and clientside equals 'first2'.)

EXPECTED RESULT: The second panel is much simpler. Verify that the button which you added a custom script to executes the system command that you specified for the SVSi network units, and verify that the condition of the button's state represents the Conditional that you specified, then press the 'previous panel' button to take you back to the first panel and you are done.