

# Creating a Combination Puzzle Panel

## Prerequisites:

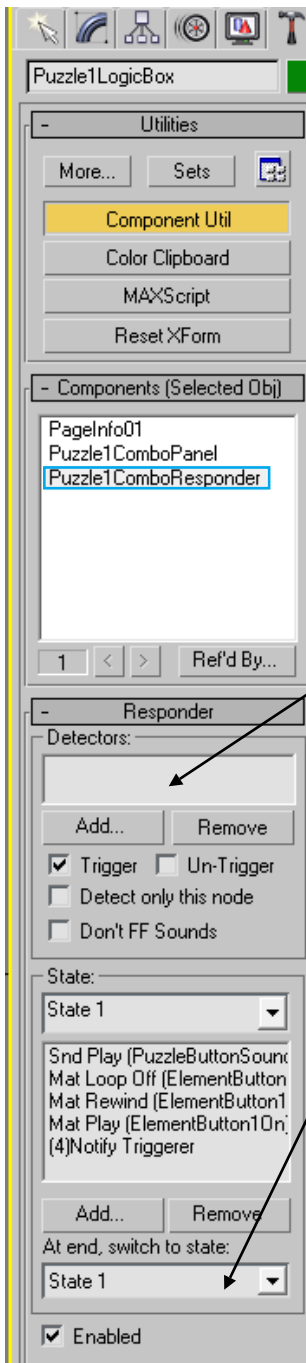
Have your Global Python set up for your Age.

Have your SDL created for the Age.

Have the xAgeSDLBoolActivatorComboSet.py and xAgeSDLBoolRespond.py files in your Age's Export>Python folder.

You will need to have modelled your panel with buttons.

Each button should have a clickable component attached to it.



Create a dummy object to attach the puzzle components. To this dummy object, add a responder with as many states as you have buttons.

Each responder state could have the button animation, sound effect and anything else to give feedback that the button has been pressed.

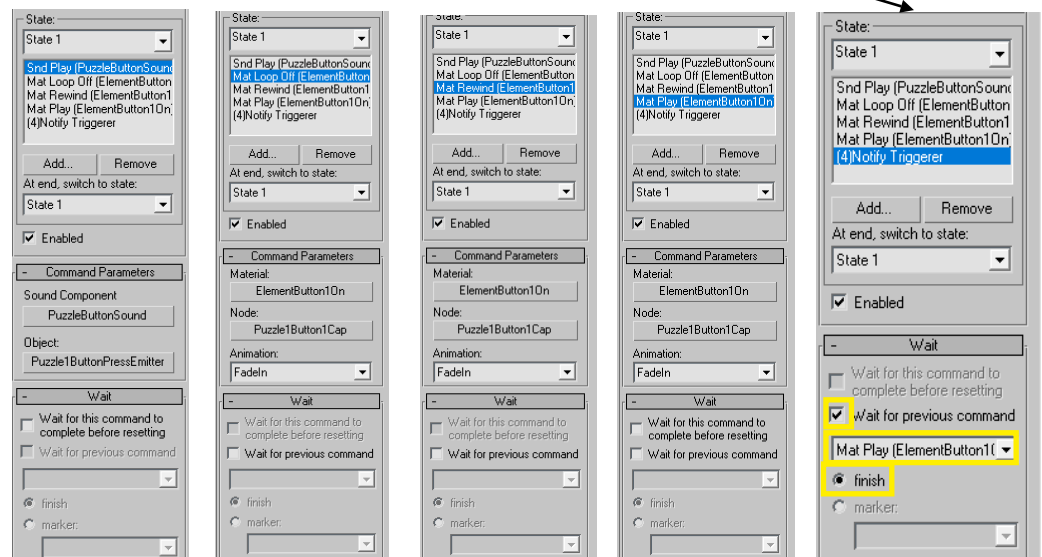
In this example, the Puzzle1ComboResponder has five states, one for each button.

Note that there is no detector for this responder as it will be controlled by the Python script which we will be adding.

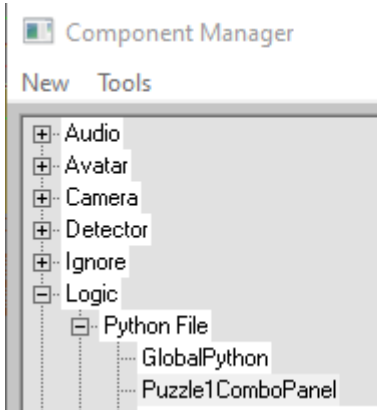
In this example there is a sound and a material animation to indicate the button has been pressed.

NB Each state does not progress to the next state as it will be controlled by the Python script which we will be adding.

Each state should have a "notify triggerer" as its last event that should wait for all previous commands to finish before activating.



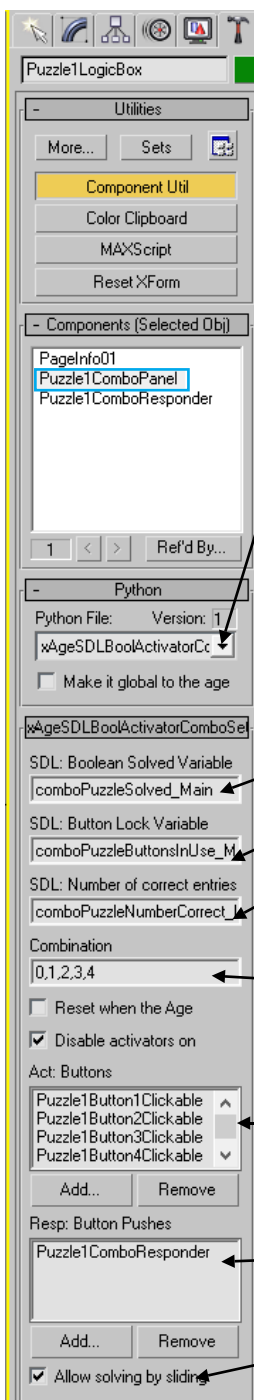
# Adding the Python Component



In the Component Manager, New>Logic>PythonFile

And rename it something useful and attach it to your Puzzle dummy object.

For example, Puzzle1ComboPanel



Select the Puzzle1ComboPanel in the component utility and from the Python file dropdown choose xAgeSDLBoolActivatorComboSet.py

The xAgeSDLBoolActivatorComboSet.py requires three SDL variables as follows:

```

1 #
2 #.State.Description.Language.for.TestAge
3
4 STATEDESC.TestAge
5 {
6     →VERSION.1
7
8     →VAR.BOOL.... comboPuzzleSolved_Main[1].... DEFAULT=0
9     →VAR.BOOL.... comboPuzzleButtonsInUse_Main[1].... DEFAULT=0
10    →VAR.INT.... comboPuzzleNumberCorrect_Main[1].... DEFAULT=0
11 }
12
    
```

Type the name of each variable in the correct box.

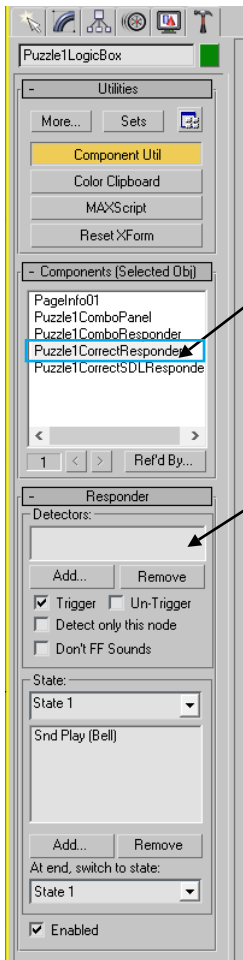
Type the solution to the button combination puzzle here. Remember to start at zero and put commas between each.

These are all the clickables assigned to your buttons and they must be added in the right order from the first to the last (1, 2, 3, 4, 5,...)

This is the ComboResponder you created earlier.

Allow solving by sliding must be checked.

# Add the PuzzleCorrect Responder

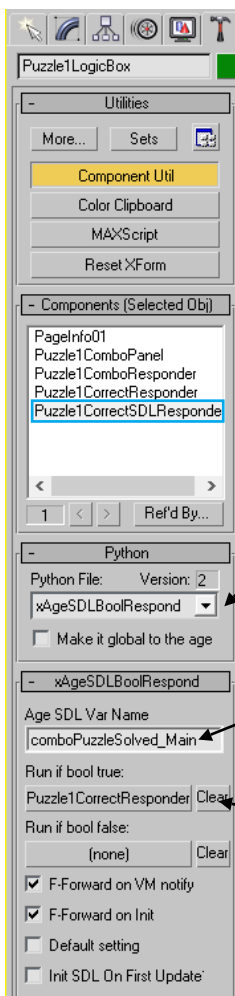


When the player has successfully entered the solution to the combination puzzle something needs to happen!

To do this you need to create a New>Logic>Responder in the Component Manager. Call it Puzzle1CorrectResponder. Attach it to the Puzzle dummy object.

In this example, the responder contains a single state that simply plays a sound. (You can add more events if needed.)

Note that it has no detectors as it will be triggered by another Python script which we will be adding.



To trigger the Puzzle1CorrectResponder, we need to add another Python component. This will respond to the change of the “SDL Boolean Solved” variable specified in the Puzzle1ComboPanel component.

New>Logic>Python. Rename it Puzzle1CorrectSDLResponder and add it to the Puzzle dummy object.

Select the Puzzle1CorrectSDLResponder in the component utility and from the Python file dropdown choose xAgeSDLBoolRespond.py

In the Age SDL Var Name box type the name of the variable which is in the “SDL Boolean Solved” variable specified in the Puzzle1ComboPanel component, in this example, the variable is called comboPuzzleSolved\_Main

For Run if bool true, select Puzzle1CorrectResponder.

Leave other settings as their defaults.