

Sketchlet Tutorial

Defining Interaction and Logic in Sketchlet

sketchlet.sf.net

Željko Obrenović

obren.info/



Events and Actions

- Event – what happens
- Action – what to do when something happens
- Logic – connecting actions with events



Defining Interaction in Sketchlet

- **Events**
 - Mouse Events, Keyboard Events, Regions Overlap and Interaction, Variable Events
- **Actions**
 - Page Transition, Variable Actions, Timer Actions, Action Actions
- **Connecting Actions and Events**
 - Drag-and-drop
 - Templates and Formulas



Defining Interaction in Sketchlet

- **Events**
 - Mouse Events, Keyboard Events, Regions Overlap and Interaction, Variable Events
- **Actions**
 - Page Transition, Variable Actions, Timer Actions, Action Actions
- **Connecting Actions and Events**
 - Drag-and-drop
 - Direct Specifying



Events

- Region Events:
 - Mouse Events
 - Keyboard Events
 - Region Overlap Events
- Page-level Events
 - Mouse Events
 - Keyboard Events
 - Page Entry/Exit Events
 - Variable Update Events



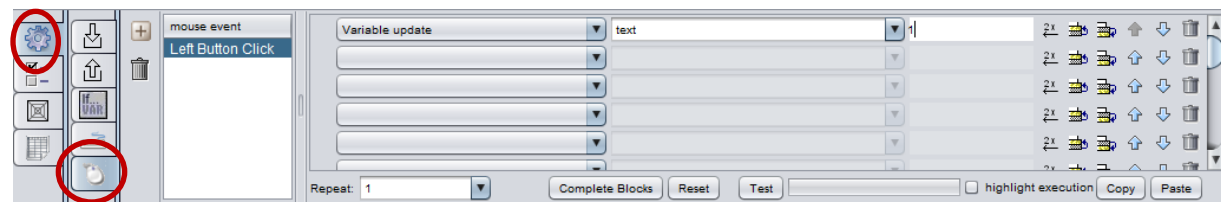
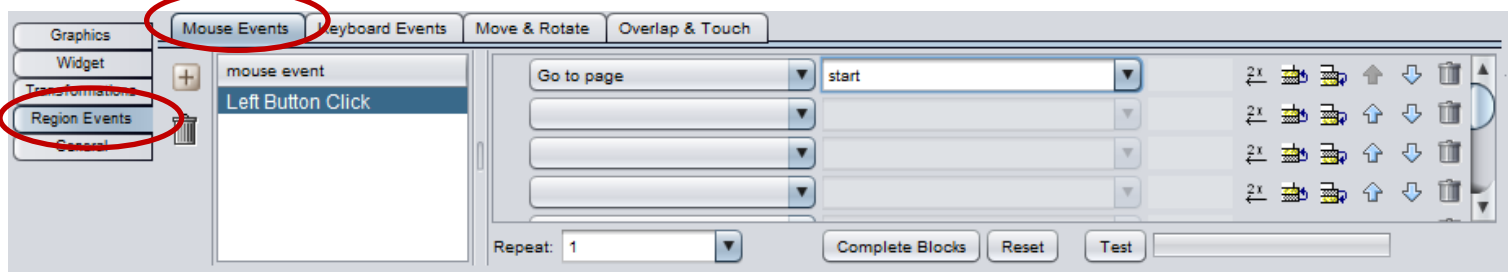
Mouse Events

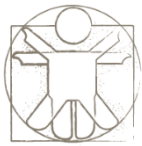
- Mouse Events are captured by active regions or by the page
- Two types of active region mouse events
 - Continuous motions
 - Discrete mouse events
- Page supports only discrete mouse events



Discrete Mouse Events

- 14 types of mouse events
 - Left Button Click/Press/Release, Right Button Click/Press/Release, Middle Button Click/Press/Release, Double Click, Mouse Entry/Exit, Mouse Wheel Up/Down
- Discrete Mouse Events are defined in the properties of active regions and pages





Continuous Mouse Events

- When user drags the active region, you can connect the variables with several elements:
 - position x, position y,
 - rotation (when dragged with right mouse button),
 - speed of dragging, and
 - trajectory position (if trajectory is defined)

The screenshot shows a software interface with several tabs: Graphics, Widget, Transformations, Region Events, and General. The 'Region Events' tab is selected and circled in red. The 'Move & Rotate' tab is also circled in red. The interface includes sections for 'Enable motion', 'Trajectory', and 'Mapping motion to variable updates'.

Enable motion

- Enable moving by mouse
- Enable rotating by mouse

Trajectory

- Stick to Trajectory
- Control Orientation
- clear trajectory

Mapping motion to variable updates

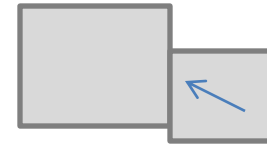
Dimension	Variable	Start value	End value	Format
position x	x	0	100	

Buttons: Delete, Move Up, Move Down, Duplicate, Edit

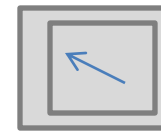


Region Overlap Events

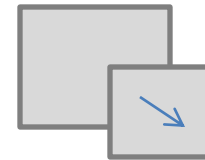
- Detected in active regions
- Define what will happen when two regions overlap
- Four types of events:
 - touches, inside, outside, completely outside



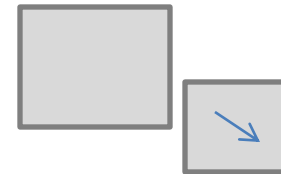
touches



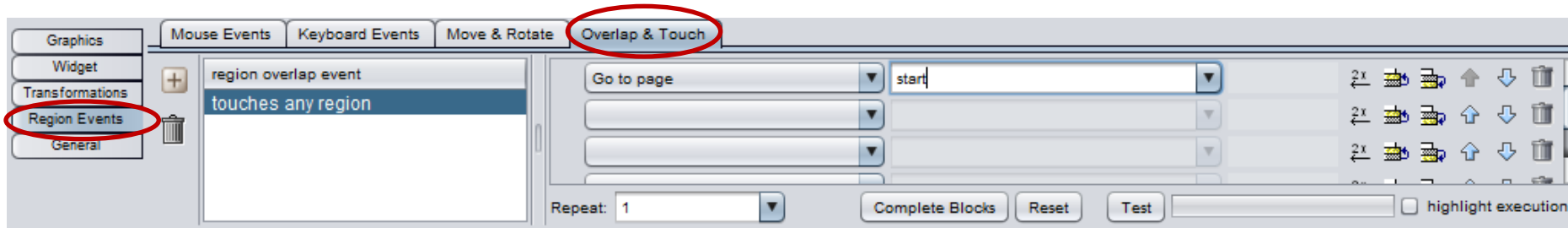
inside



outside



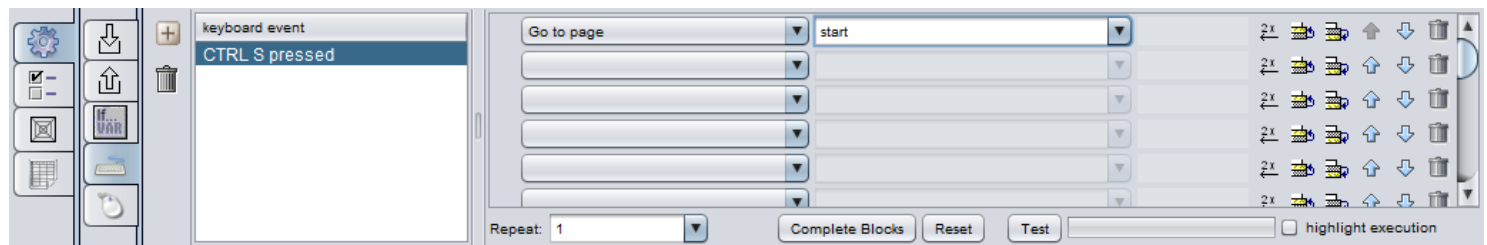
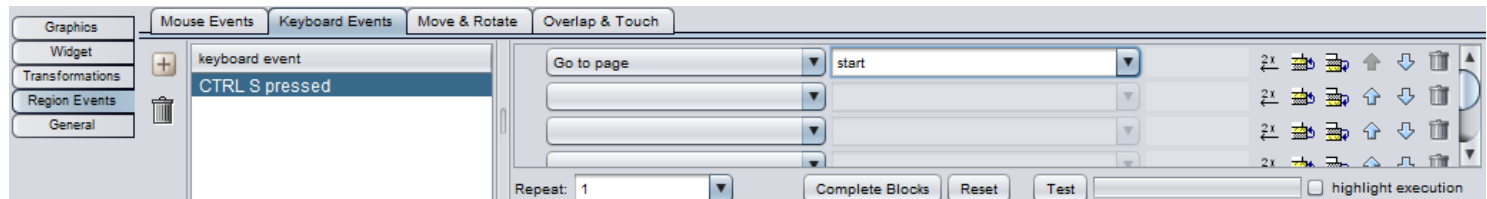
completely outside





Keyboard Events

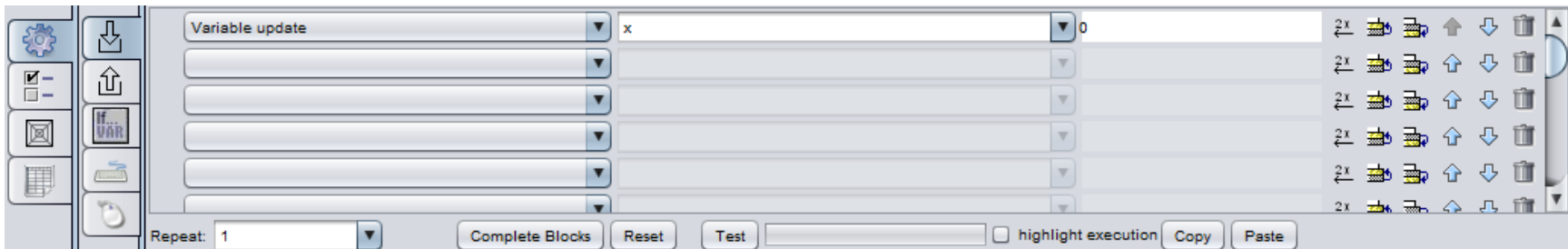
- Keyboard events are defined within the active region or page settings
- You can capture “pressed” or “released” events for any keyboard key





Page Events

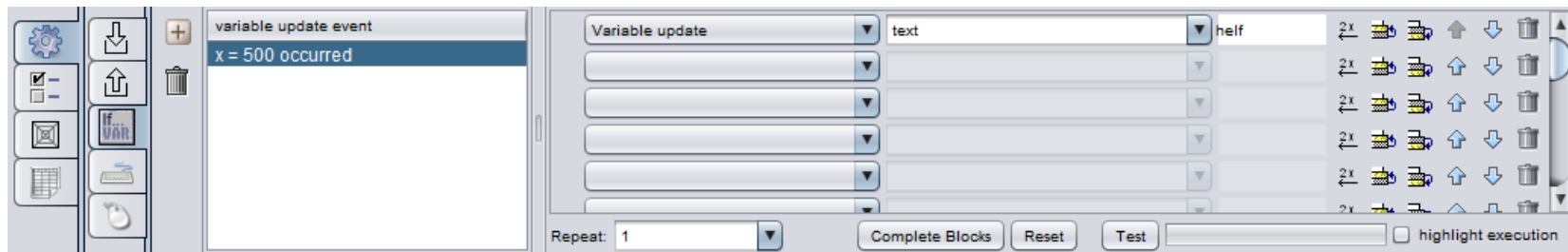
- Two types of page events
 - On page entry
 - Defines what will happen when you open the page
 - On page exit
 - Defines what will happen when you close the page (or go move to another page)





Variable Events

- Variable events are defined in page settings
- Define what will happen when variable is updated or has particular value





Defining Interaction in Sketchlet

- **Events**
 - Mouse Events, Keyboard Events, Regions Overlap and Interaction, Variable Events
- **Actions**
 - Page Transition, Variable Actions, Timer Actions, Action Actions
- **Connecting Actions and Events**
 - Drag-and-drop
 - Direct Specifying



Actions

- Page Transitions
- Variable Actions
- Timer Actions
- Action Actions



Sketchlet Actions

Action	Param1	Param2
Go To Page	page-name	not used
Variable update	variable-name	new value
Variable increment	variable-name	number to add
Variable append	variable-name	string to append
Variable glide	variable-name	[start] end duration-in-sec
Start timer	timer-name	not used
Pause timer	timer-name	not used
Stop timer	timer-name	not used
Start action	action-name	not used
Stop action	action-name	not used
Start sequence	command sequence	command sequence



Page Transitions

- Closes current page and opens another
- In interface defined as
“Go To Page” <page-name>
- When this actions occurs:
 - On exit event occurs on the old page
 - On entry event occurs on the new page



Variable Actions

- Three types of variables actions
 - **Variable update** – sets the variable to a new value
 - **Variable increment** – increments current value
 - “122” increment with “12” results in change of variable value to “134”
 - If a variable does not contain numeric value nothing happens
 - If variable is empty (or does not exists), 0 value is assumed
 - **Variable append** – appends string to existing variable content
 - For example, if variable has the value “abc”, append of “de” will result in change of variable value to “abcde”
 - **Variable glide** – animates variable updates from start to end, with given duration
 - If start is not given, current variable value is used



Timer Action

- Three types of timer actions
 - Start timer
 - Pause timer
 - Stop timer



Action Actions

- Two types of action actions
 - Start action
 - Stop action



Defining Interaction in Sketchlet

- **Events**
 - Mouse Events, Keyboard Events, Regions Overlap and Interaction, Variable Events
- **Actions**
 - Page Transition, Variable Actions, Timer Actions, Action Actions
- **Connecting Actions and Events**
 - Drag-and-drop
 - Direct Specifying



Events and Actions

- Event – what happens
- Action – what to do when something happens
- **Logic – connecting actions with events**



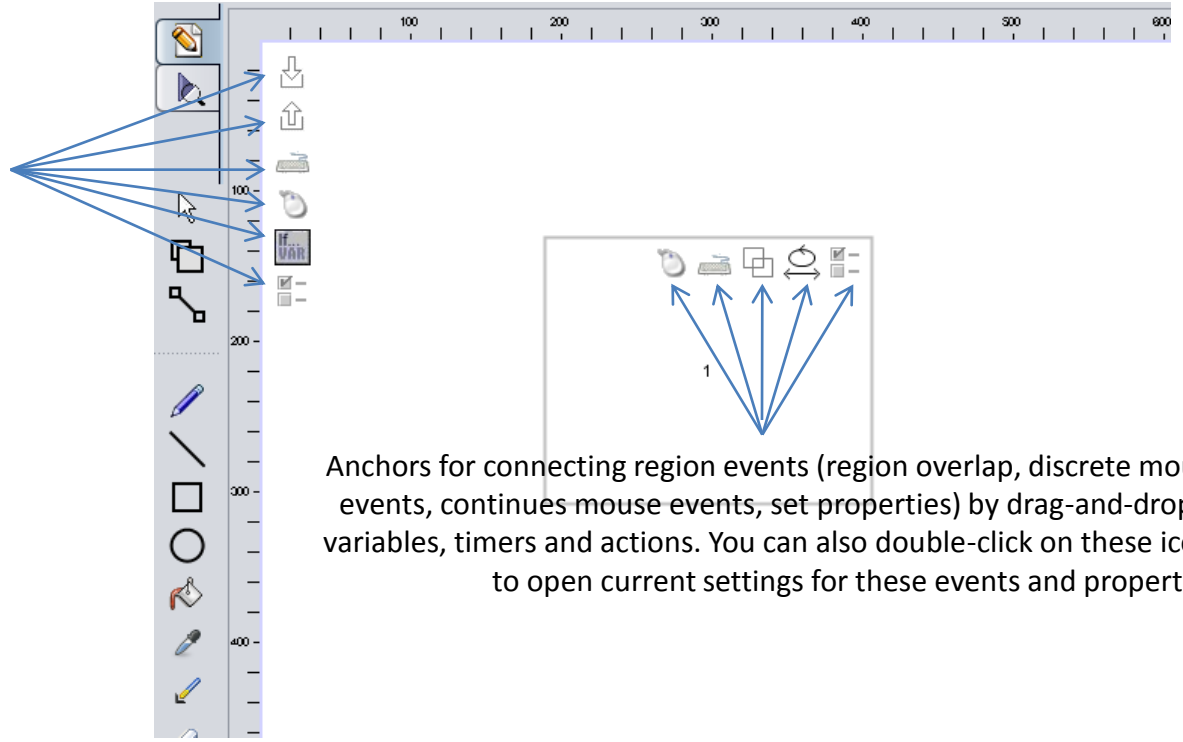
Connecting Actions and Events

- Drag-and-drop support
 - Drag variable, page, timer or action to connect it with various events
- Specify directly



Drop Event Anchors

Anchors for connecting page events (on entry, on exit, on variable update, on keyboard event, on mouse event, set property) by drag-and-drop of variables, timers and actions. You can also double-click on these icons to open current settings for these events and properties.



Anchors for connecting region events (region overlap, discrete mouse events, continues mouse events, set properties) by drag-and-drop of variables, timers and actions. You can also double-click on these icons to open current settings for these events and properties.



Example: Connecting Region Continuous Mouse Event to Variable

- Step 1: Creating variable “rot”

The screenshot shows a software interface for creating interactive content. A dialog box titled "Add New Variable" is open, with the text "rot" entered in the first field. The main workspace contains a square labeled "1". The right-hand panel has a table of variables:

Page Variable	Value	Forma
text	INSIDE	
number-of-faces	1	

Below this table is another table with columns "Global Variable" and "Value":

Global Variable	Value
text	INSIDE
number-of-faces	1



Example: Connecting Region Continuous Mouse Event to Variable

- Step 2: drag-and-drop the variable on the active region motion mapping icon

The screenshot shows a software interface with a workspace containing a circular region. A red arrow points from the 'on move or rotate' icon in the workspace to the 'rot' variable in the 'proxy' section of the right-hand panel. The interface includes a menu bar, a toolbar, and a right-hand panel with a 'Global Objects' section and a 'proxy' section. A red arrow points from the 'on move or rotate' icon in the workspace to the 'rot' variable in the 'proxy' section.

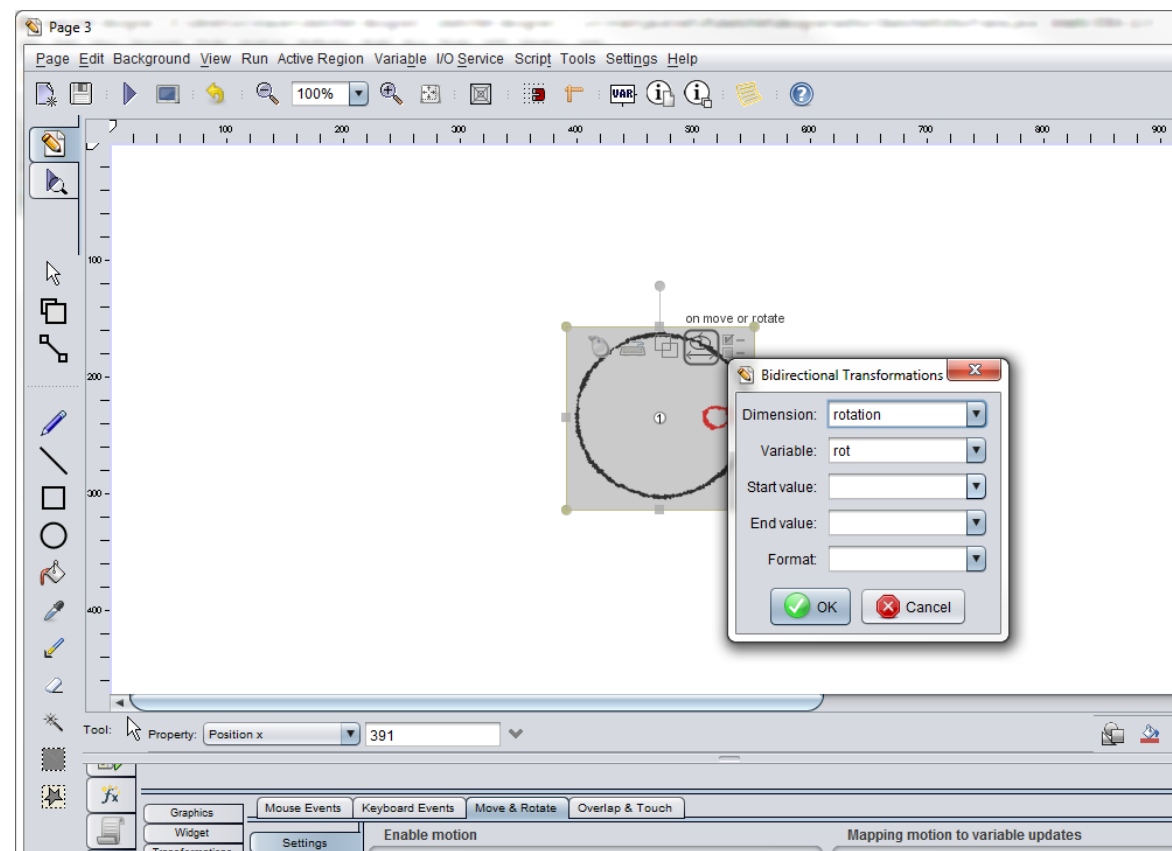
Page Variable	Value	Forma
rot		

Global Variable	Value
text	INSIDE



Example: Connecting Region Continuous Mouse Event to Variable

- Step 3: select properties of motion that you want to capture in the variable
- In this example it is “rotation”





Example: Connecting Region

Continuous Mouse Event to Variable

- Step 4: interact with the region and observe variables changes
- In this example, dragging the region with right mouse button will rotate it and update the variable “rot”

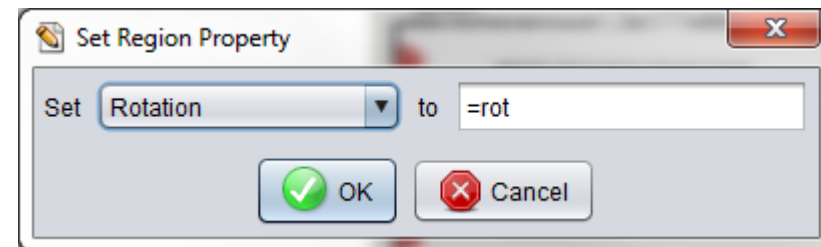
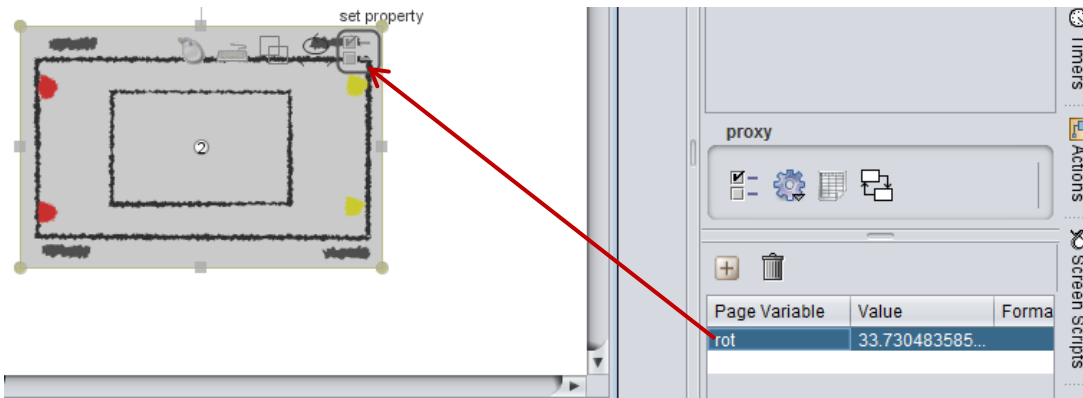
The screenshot displays a software interface with a 3D model on the left and a control panel on the right. The 3D model shows a diamond-shaped region with a circular hole. A red circle highlights a specific point on the inner boundary of the hole. A blue arrow points from this red circle to a table in the software's interface. The table has three columns: 'Page Variable', 'Value', and 'Forma'. The row for 'rot' shows a value of 33.730483585... and a blank 'Forma' cell.

Page Variable	Value	Forma
rot	33.730483585...	



Example: Connecting Region Continuous Mouse Event to Variable

- You can drag-and-drop the variable on the region properties icon, and select property





Example: Connecting Region Continuous Mouse Event to Variable

- In this example we are using variable “rot” to control rotation property of another region

The screenshot shows a software interface with a sketch area on the left and a 'Variables' panel on the right. The sketch area contains a rectangular region with a smaller inner rectangle and a red gear icon. The 'Variables' panel has a table with the following data:

Variable Name	value	Descr
rot	83.25741339328722	

A red oval highlights the 'rot' variable and its value. A red arrow points from this oval to a gear icon in the sketch area, indicating a connection between the variable and the region's rotation property.

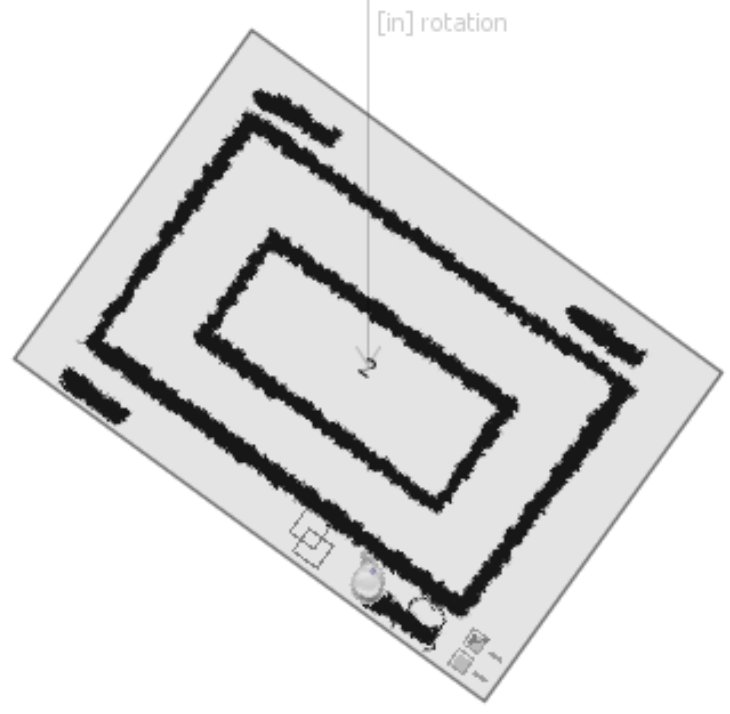
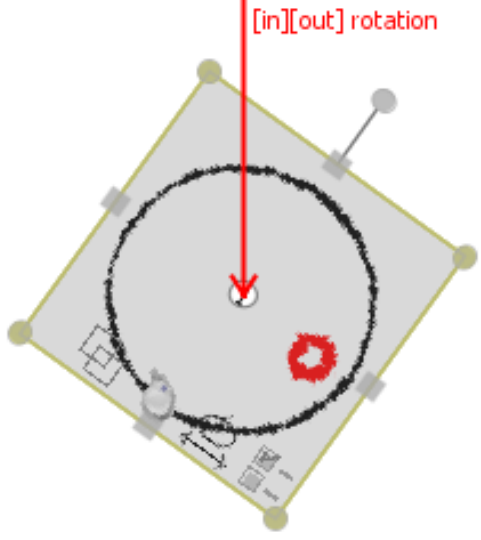
The 'Set Region Property' dialog box is shown with the following settings:

- Set: **Rotation** (selected from a dropdown menu)
- to: `=rot`
- Buttons: **Cancel** (with a red 'X' icon)



rot : 35.962867373302636 >

[YouTube Video](#)





More Examples

- **Page Transitions Based on Mouse Events**
 - [YouTube Video](#)



Directly Connecting Events and Actions

- For each event, you can directly open its settings, and there select actions