



Sketchify Tutorial

Properties and Variables

sketchify.sf.net

Željko Obrenović

z.obrenovic@tue.nl



Properties and Variables

- Properties of active regions and sketches can be given directly, or indirectly through variables
 - Directly specify the value
 - 90
 - Indirectly specify the value through variables
 - Using formulas
 - =variable
 - =a + (b-c) / 50
 - Using string templates
 - You said <%=text%>



Why Variables?

- Benefits of indirect specification
 - One variable can control several properties
 - Through variables, objects in sketches can communicate with each other and with external services (such as Wii, text-to-speech service)

[YouTube Video](#)



Interface for Working with Variables

- Variables can be accessed through a spreadsheet-like interface, making all data immediately visible and manipulatable.
- A designer can directly observe and update variables; useful to explore and play with the functionality.

Filter

Sort

Variables

sort by creation

Variable Name	Value	Descr
trajectory_position	1.0	
trajectory_position_2	1.0	
trajectory_position_3	0.0	
query		[in, trig
babelfish-status	ready	[out] S
test	this	
hallo		
r		
r+rob+		
babelfis		[in, trig
babelfis		[in] Inp
babelfis		[in] Inp
babelfis		[in, trig

Edit...

Derive New Variable(s)

Copy Variable Names

Copy Spreadsheet Formulas

Copy Script Expressions

Remove

Set count filter...

+

Delete selected variables

Derived variables

Remote updates

Count filter

12

Disable variables' updates



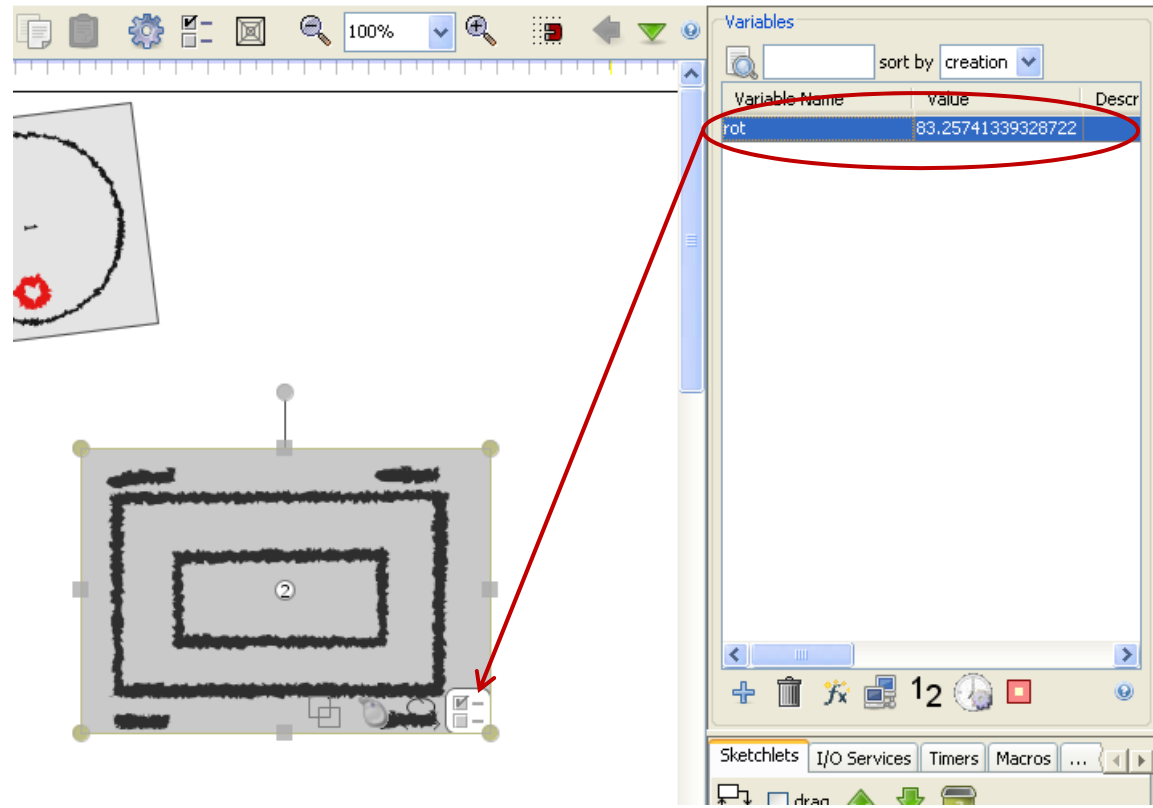
Connecting Variables and Properties

- Drag-and-Drop
 - You can drag a variable and drop it on the region or sketch to set its properties
- Specifying property value in region setting
 - With expressions and formulas
 - With string templates

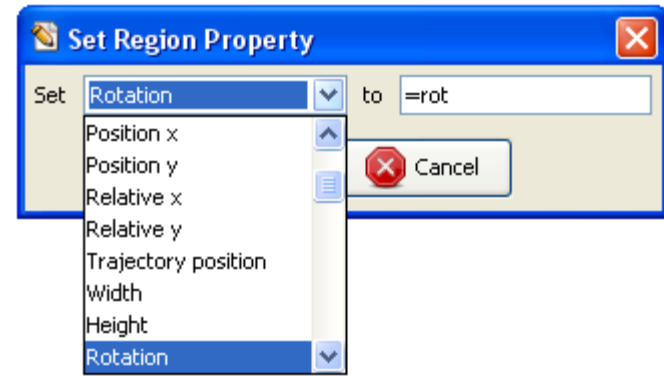


Drag-and-Drop Connection between Variables and Properties

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



[YouTube Video](#)





Specifying in the Properties Tab

- You can also specify properties by going to the properties tab of the region
- Instead of giving the value directly, you can use the expression “=variable”
- This means that value will be taken from the variable

Property	Value	Description
Position		
position x		horizontal position (lef...
position y		vertical position (top, ...
relative x		relative horizontal pos...
relative y		vertical position (0.0 t...
trajectory position		0.0 to 1.0
Size		
width		region width
height		region height
Orientation		
rotation	=rot	angle
Transparency		
transparency		0.0 to 1.0
Visible area		
visible.area x		



Variables and Templates

- Templates are simple way to define a text with some part of it will be replaced with the variable value:
“Rotation is <%=rot%> degrees”

The screenshot shows the Sketch 19 software interface. The main canvas displays the text "Rotation is 95 degrees" in a handwritten font. The text is highlighted with a grey background. Below the canvas, the "Text" tool is active, and the text input field shows the template "Rotation is <%=rot%> degrees". The "Variables" panel on the right shows a table with one variable named "rot" with a value of "95".

Variable Name	Value	Descr
rot	95	



Variables and Formulas

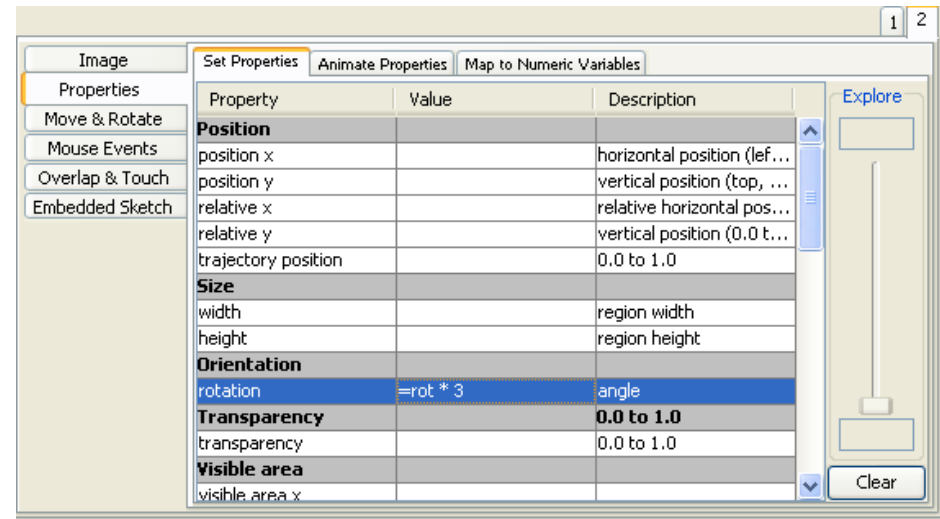
- You can also use more complex formulas to derive values using diverse operators and function

- Examples

$$=\text{sqrt}(a^2 + b^2)$$

$$=\text{rot} * 3$$

$$=100 + 50 * \sin(\text{rot})$$



- Expression with formulas have to start with “=”
- NOTE:** If the variable name contains operator, such as “-”, you have to put the name within apostrophes, for example **“'movement-intensity' / 2”**. [YouTube Video](#)



Formulas

- All common arithmetic operators are supported. Boolean operators are also fully supported.
- You can also derive the value conditionally using the **if** command.
 - For example, the formula "**if(a > -0.1 && a < 0.1, b, 1.0)**" will return the value of variable **b** if the variable **a** is within **-0.1** and **0.1**, or **1.0** otherwise. Boolean expressions are evaluated to be either 1 or 0 (true or false respectively).



Operators

Power	\wedge
Boolean Not	!
Unary Plus, Unary Minus	+x, -x
Modulus	%
Division	/
Multiplication	*
Addition, Subtraction	+, -
Less or Equal, More or Equal	=, >=
Less Than, Greater Than	, >
Not Equal, Equal	!=, ==
Boolean And	&&
Boolean Or	

Functions

Sine	sin(x)
Cosine	cos(x)
Tangent	tan(x)
Arc Sine	asin(x)
Arc Cosine	acos(x)
Arc Tangent	atan(x)
Arc Tangent (with 2 parameters)	atan2(y, x)
Hyperbolic Sine	sinh(x)
Hyperbolic Cosine	cosh(x)
Hyperbolic Tangent	tanh(x)
Inverse Hyperbolic Sine	asinh(x)
Inverse Hyperbolic Cosine	acosh(x)
Inverse Hyperbolic Tangent	atanh(x)
Natural Logarithm	ln(x)
Logarithm base 10	log(x)
Exponential (e^x)	exp(x)
Absolute Value / Magnitude	abs(x)
Random number (between 0 and 1)	rand()
Modulus	mod(x,y) = x % y
Square Root	sqrt(x)
Min	min(x,y)
Max	max(x,y)
Sum	sum(x,y,z)
If	if(cond,trueval,falseval)
Str (number to string)	str(x)
Binomial coefficients	binom(n,i)