# **EB GUIDE tutorial**

Making a rectangle move across the screen Copyright © 2016 Elektrobit Automotive GmbH

Legal notice

Confidential and proprietary information.

ALL RIGHTS RESERVED. No part of this publication may be copied in any form, by photocopy, microfilm, retrieval system, or by any other means now known or hereafter invented without the prior written permission of Elektrobit Automotive GmbH.

ProOSEK®, tresos®, and street director® are registered trademarks of Elektrobit Automotive GmbH.

All brand names, trademarks and registered trademarks are property of their rightful owners and are used only for description.

# **1. Tutorial: Making a rectangle move across the screen**

The following instructions guide you through the process of animating a rectangle so that it continually moves across the screen when the simulation starts.

Approximate duration: Five minutes.



Adding widgets

In the following steps, you add three widgets to the view and organize the hierarchy of the widgets.

Prerequisite:

- The Main state machine contains an initial state and a view state.
- The initial state has a transition to the view state
- Step 1

In the content area, double-click the view state.

The view is displayed in the content area.

Step 2

Drag a rectangle from the **Toolbox** into the view.

Step 3

Drag an animation from the **Toolbox** into the rectangle.

Step 4

In the navigation area, click the animation, and press the F2 key. Rename the animation to MyAnimation.

### Step 5

Drag a linear interpolation integer widget from the **Toolbox** into the rectangle.

#### Step 6

In the navigation area, move the linear interpolation integer widget in the hierarchy so that it becomes a child widget of the animation.

Now, if you start the simulation, a rectangle is displayed in a view. The rectangle does not move yet.



Adding a user-defined property of type conditional script

As a next step, you add a user-defined property to the rectangle. With the conditional script property, rendering the rectangle during simulation starts the animation.

#### Prerequisite:

You completed the previous instruction.

Select the rectangle.

<u>Step 2</u> In the **Properties** panel, go to the **User-defined properties** category, and click  $\blacksquare$ .

A menu expands.

<u>Step 3</u> In the menu, click **Conditional script**.

A user-defined property of type Conditional script is added to the rectangle.

Step 4 Rename the property to startAnimation.

Step 5 Next to the startAnimation property, click Edit....

A script editor opens in the content area.

Step 6 Enter the following EB GUIDE Script:

```
function(v:arg0::bool)
{
    f:animation_play(v:this->MyAnimation)
    }
```



Making the animation visible

The following instructions guide you through the process of making the animation visible.

Prerequisite:

You completed the previous instruction.

Step 1

Select the linear interpolation integer widget.

#### Step 2

In the **Properties** panel, go to the target property, and click the **D** button next to the property.

A menu expands.

#### Step 3

In the menu, click Add link to widget property.

#### A dialog opens.

Step 4

In the dialog, go to the rectangle, and select its x property.

width 80 height 48 x 0 y 0 • Rectangle 1 width height x y • MyAnimati repeat • Linear in	0 100 100 111 154 ion 1			
height 48 x 0 y 0 V Rectangle 1 width height x y MyAnimati repeat V Linear in	100 100 111 154 ion			į
x 0 y 0 Rectangle 1 width height x y MyAnimati repeat t Linear in	100 100 111 154 ion			į
y 0 • Rectangle 1 width height × y • MyAnimati repeat • Linear in	100 100 111 154 ion			è
<ul> <li>Rectangle 1</li> <li>width</li> <li>height</li> <li>x</li> <li>y</li> <li>MyAnimati</li> <li>repeat</li> <li>Linear in</li> </ul>	100 100 111 154 ion			ė
width height y MyAnimati repeat tinear in	100 100 111 154 ion			ė
height x y ✓ MyAnimati repeat ✓ Linear in	100 111 154 ion			ė
x y YMyAnimati repeat Linear in	111 154 ion 1			
y MyAnimati repeat	154 ion 1			
✓ MyAnimati repeat	ion 1			
repeat	1			
🔫 Linear in				
	nterp	olation integer	1	
delay		0		
durati	ion	1000		
repea	t	1		
start		0		
end		1		
targe	t	0		

Figure 1. Linking between widget properties

#### Step 5 Click Accept.

The dialog closes. The <u>button</u> is displayed next to the target property.

#### Step 6

Link the end property to the view's width property.

With these settings, when the animation starts, the x property of the rectangle changes from zero to the width of the view. Thus the rectangle moves from the left boundary to the right boundary of the view.

#### Step 7

To make the animation run in infinite repetitions, enter 0 in the repeat property.

## Step 8

Save the project.

#### Step 9

To start the simulation, click 🕨 in the command area.

Result:

The rectangle continually moves from the left side of the view to the right side of the view.