

EB GUIDE tutorial

Making an ellipse move across the screen

Copyright © 2017 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 an ellipse move across the screen

NOTE**Default window layout**

All instructions and screenshots of this user manual use the default window layout. If you want to follow the instructions, we recommend to reset the EB GUIDE Studio window to default layout by selecting **Layout > Reset to default layout**.

The following instructions guide you through the process of animating an ellipse 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 content area displays the **Main** state machine.
- 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 an ellipse from the **Toolbox** into the view.

Step 3

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

Step 4

In the **Navigation** component, 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 **Navigation** component and drop it so that it becomes a child widget of the animation.

Now, if you start the simulation, an ellipse is displayed in a view. The ellipse 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 ellipse. With the conditional script property, rendering the ellipse during simulation starts the animation.

Prerequisite:

- You completed the previous instruction.

Step 1

Select the ellipse.

Step 2

In the **Properties** component, go to the **User-defined properties** category, and click **+**.

A menu expands.

Step 3

In the menu, click `Conditional script`.

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

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** component, go to the `target` property, and click the  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 ellipse, and select its `x` property.

Step 5

Click **Accept**.

The dialog closes. The  button 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 ellipse changes from zero to the width of the view. Thus the ellipse 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 ellipse continually moves from the left side of the view to the right side of the view.