

# EB GUIDE tutorial

Modeling a path gesture

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: Modeling a path gesture

Path gestures are shapes drawn by a finger on a touch screen or entered by some other input device.

The following instructions guide you through the process of modeling a path gesture.

Approximate duration: 10 minutes



## Adding widgets and configuring default widget properties

Prerequisite:

- The **Main** state machine contains an initial state and a view state.
- The initial state has a transition to the view state.
- The content area displays a view.

### Step 1

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

### Step 2

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

The label is added as a child widget to the rectangle.

The **Properties** panel displays the properties of the label.

### Step 3

In the **Properties** panel, enter 500 in the `width` text box.

### Step 4

Select the rectangle.

The **Properties** panel displays the properties of the rectangle.

### Step 5

Enter 500 in the `width` text box.

### Step 6

In the **Properties** panel, go to **fillColor**, and select red.

You added two widgets and configured default widget properties.



## Adding widget features to a rectangle

To enable the user to enter a shape starting on the widget, you add the widget feature **Path gesture** to the rectangle. The shape is matched against a set of known shapes and, if a match is found, a gesture is recognized.

Prerequisite:

- You completed the previous instruction.

### Step 1

Select the rectangle.

The **Properties** panel displays the properties of the rectangle.

### Step 2

In the **Properties** panel, go to **Widget feature properties**, and click **Add/Remove**.

The **Widget features** dialog is displayed.

### Step 3

Under **Available widget features**, expand the **Gestures** category, and select `Path gestures`.

The **Touched** widget feature is automatically selected, as it is required for the **Gestures** widget feature.

### Step 4

Click **Accept**.

The related widget feature properties are added to the rectangle and displayed in the **Properties** panel.

### Step 5

For the **Path gestures** widget feature edit the following properties:

#### Step 5.1

Next to the `onPath` property, click **Edit**.

#### Step 5.2

Enter the following EB GUIDE Script:

```
function (v:gestureId::int)
{
  v:this->"Label 1".text = "recognized path gesture #"
  + f:int2string(v:gestureId);
}
```

#### Step 5.3

Click **Accept**.

#### Step 5.4

Next to the `onPathStart` property, click **Edit**.

Step 5.5

Enter the following EB GUIDE Script:

```
function ()
{
  v:this->"Label 1".text = "path gesture start";
}
```

Step 5.6

Click **Accept**.

Step 5.7

Next to the `onPathNotRecognized` property, click **Edit**.

Step 5.8

Enter the following EB GUIDE Script:

```
function()
{
  v:this->"Label 1".text = "shape not recognized";
}
```

Step 5.9

Click **Accept**.

Step 6

To start the simulation, click  in the command area.

The simulation and EB GUIDE Monitor start. To see a reaction, draw a shape with the mouse inside the rectangle.