

EB GUIDE tutorial

Modeling button behavior with EB GUIDE Script

Version 6.12.0.211022102259

Copyright © 2021 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.

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

This document contains links to websites owned and operated by third parties. Elektrobit has no control of the content of any third party website and therefore takes no warranty nor liability for damage or loss caused in connection with the use or reliance on any information, material, products or services contained or accessed through any such linked website.

1. Tutorial: Modeling button behavior with EB GUIDE Script

TIP



Default window layout

All instructions and screenshots use the default window layout. If you want to follow the instructions, we recommend to set the EB GUIDE Studio or EB GUIDE Monitor window to default layout by selecting **Layout > Reset to default layout**.

With EB GUIDE Script you can express property values, actions, or conditions and evaluate them during run-time.

The following instructions guide you through the process of using EB GUIDE Script to model the behavior of a button. The button increases in size when it is clicked and shrinks back to its original size when it reaches a defined maximum size. For best results, work through the steps in the order presented.

Approximate duration: 10 minutes.



Adding widgets

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 the View.

Step 1

Drag a Rectangle from the **Toolbox** into the View.

Step 2

In the **Navigation** component, select the Rectangle, press the **F2** key, and rename the Rectangle to `Back-ground`.

Step 3

Drag a Rectangle from the **Toolbox** into the **Navigation** component. Place it as a child widget to the `Back-ground` Rectangle.

Step 4

In the **Navigation** component, select the new Rectangle, press the **F2** key, and rename the Rectangle to `Button`.

Step 5

Drag a Label from the **Toolbox** into the **Navigation** component. Place the Label as a child widget to the `Button` rectangle.

Step 6

In the **Navigation** component, select the Label, press the **F2** key, and rename the label to `Button text`.

Your widget hierarchy now looks as follows.

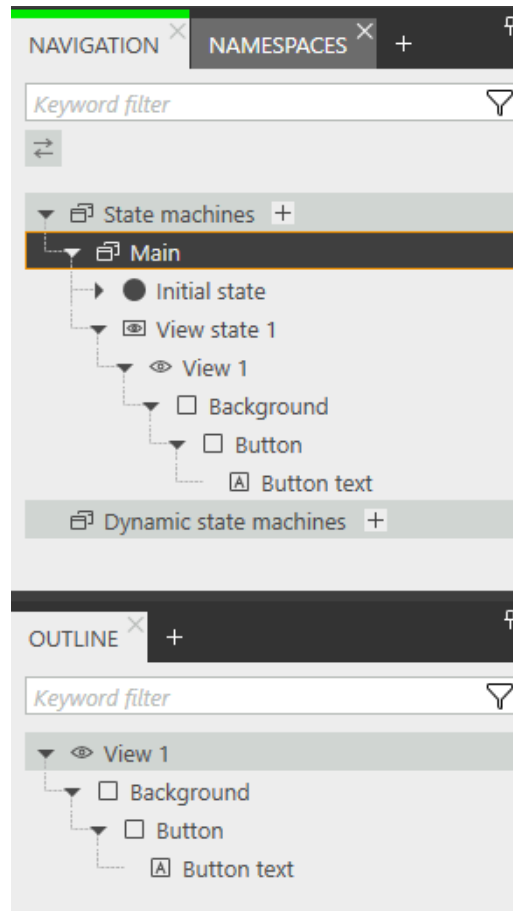


Figure 1. Widget hierarchy



Configuring the background

Prerequisite:

- You completed the previous instruction.

Step 1

In the **Navigation** component, click the `Background Rectangle`, and go to the **Properties** component.

Step 2

Next to the `width` property, click the  button.

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 view, and select its `width` property.

Step 5

Click **Accept**.

The dialog closes. The  button is displayed next to the `width` property.

Step 6

Link the `height` property of the `Background Rectangle` to the `height` property of the `View`.

Step 7

Link the `x` property of the `Background Rectangle` to the `x` property of the `View`.

Step 8

Link the `y` property of the `Background Rectangle` to the `y` property of the `View`.

The `Background Rectangle` covers the exact size and position of the `View`.



Defining the maximum button width

A datapool item holds the value for the maximum width of the button. It can be changed during run-time.

Prerequisite:

- You completed the previous instruction.

Step 1

In the **Datapool** component, click **+**.

A menu expands.

Step 2

In the menu, click **Integer**.

A new datapool item of type `Integer` is added.

Step 3

Rename the datapool item to `Maximum width`.

Step 4

In the `Value` text box, enter 400.



Configuring the button

Prerequisite:

- You completed the previous instruction.

Step 1

In the **Navigation** component, click the `Button Rectangle`, and go to the **Properties** component.

Step 1.1

Enter 50 in the `height` text box.

Step 1.2

Enter 350 in the `x` text box.

Step 1.3

Enter 215 in the `y` text box.

Step 1.4

Select blue for the `fillColor` property.

The button is now colored blue.

Step 2

In the **Widget feature properties** category, click **Add/Remove**.

The **Widget features** dialog is displayed.

Step 3

Under **Available widget features**, expand the **Input handling** category, and select the **Touch pressed** widget feature.

Step 4

Click **Accept**.

The related widget feature properties are added to the `Button Rectangle` and displayed in the **Properties** component.

Step 5

Next to the `touchPressed` property, click .

An EB GUIDE Script editor opens.

Step 6

Replace the existing EB GUIDE Script with the following code:

```
function(v:touchId::int, v:x::int, v:y::int, v:fingerId::int)
{
    if (v:this.width > dp:"Maximum width") // If the button has grown
                                           // beyond its maximum size...
    {
```

```
// ...reset its dimensions to the default values.
v:this.height = 50
v:this.width = 100
v:this.x = 350
v:this.y = 215
    }
    else // Otherwise...
    {

// ... increase button size...
v:this.width += 80
v:this.height += 40

// ...and move the button to keep it centered.
v:this.x -= 40
v:this.y -= 20
    }
    false
}
```

Step 7

Click **Accept**.

You configured the `Button Rectangle` and wrote an EB GUIDE Script which changes the size of the `Button Rectangle` in run-time.



Configuring the button text

Prerequisite:

- You completed the previous instruction.

Step 1

In the **Navigation** component, click the `Button text Label`, and go to the **Properties** component.

Step 2

Enter `grow!` in the `text` text box.

Step 3

Link the `width` property of the `Button text label` to the `width` property of the `Button rectangle`.

Step 4

Link the `height` property of the `Button text Label` to the `height` property of the `Button Rectangle`.

Step 5

Enter `0` in the `x` text box.

Step 6

Enter `0` in the `y` text box.

Step 7

Next to the `horizontalAlign` property, select `center(1)`.

Now the `Button` text `Label` and the `Button` `Rectangle` are equal in size and position.



Saving and testing the EB GUIDE model

Prerequisite:

- You completed the previous instruction.

Step 1

To save the project, click  in the command area.

Step 2

To start the simulation, click  in the command area.

Result:

The simulation starts the EB GUIDE model you created. It behaves as follows.

1. First, it displays a grey screen with a blue button in its center. The screen looks as follows.



Figure 2. Result

2. Whenever you click the button, it increases in size but keeps its position at the center of the screen.
3. As soon as the button width reaches the value of the `Maximum width` datapool item, it shrinks back to its original size and position.