# HowTo 4 - Using a Behaviour

A guide to enhance your GSE project with behaviours

| | |
|---|---|
| Product: | Guiliani Streaming Editor (GSE) |
| Release version: | 2.2 |
| Release date: | August 31, 2018 |

## Disclaimer

**Guiliani, guiliani.de and GSE are products of**

TES Electronic Solutions GmbH
Hanauer Landstrasse 328-330
60314 Frankfurt am Main
Germany

| | |
|---|---|
| Email: | support@guiliani.de |
| Website: | http://www.guiliani.de |
| Forum: | https://www.guiliani.de/forum/ |
| HelpDesk: | https://guiliani.on.spiceworks.com/portal |

## Table of contents

## Table of figures

# 1. Introduction

This document explains step by step how to use Behaviours in a GSE project.

## 1.1. Assumed knowledge

- Basic handling of GSE
- It is recommended to read “How to 1 - build a project step by step”, first.
- For a better understanding on how the project works, we suggest reading “How to 2 - using Commands in GSE” and “How to 3 - using a DataPool in GSE”, too.
  Knowledge about both topics will be needed here.

## 1.2. Prerequisites

- Unpacked Guiliani SDK including GSE
- Your project “step by step” created in “How to 3” (or the “step by step” project inside the folder called “*How to 3 - sample solution*”)

## 1.3. Documentation conventions

Whenever you can use keys from your computer’s keyboard, these will be displayed in square brackets (e.g.,” To run your project press [Ctrl] + [r].”).

Menu commands or file path used in this document will be shown in *italic*.

Text that appears in the software on controls will be printed in **bold and blue**.

▶ Whenever the reader of this document has to do something in his project, the text will start with this triangle.

➡ Results will be shown using this arrow.

In this document, we use icons whenever we will warn the user or will give him additional or important information.

The speech bubble icon will show additional helpful information.

Whenever a text begins with an exclamation mark icon, it contains important information that is essential for the current chapter.

A warning sign icon signals serious issues and potential risks that require your full attention.

## 1.4. Shortcuts

In the documents, we often select a command from the window. These can be selected by the following short cuts, too:

| | |
|---|---|
| *File* ➔ *New Project…* | [Ctrl] + [Shift] + [n] |
| *File* ➔ *New Dialog…* | [Ctrl] + [n] |
| *File* ➔ *Save Project…* | [Ctrl] + [s] |
| *File* ➔ *Run Simulation…* | [Ctrl] + [r] |

## 1.5. This document’s goal

At the end of this HowTo you will have learned

- how to use Behaviours
- send information between different controls
- using two wheels
- and more

You will be able to save the heating state of two burners.
The main dialog will look like this:

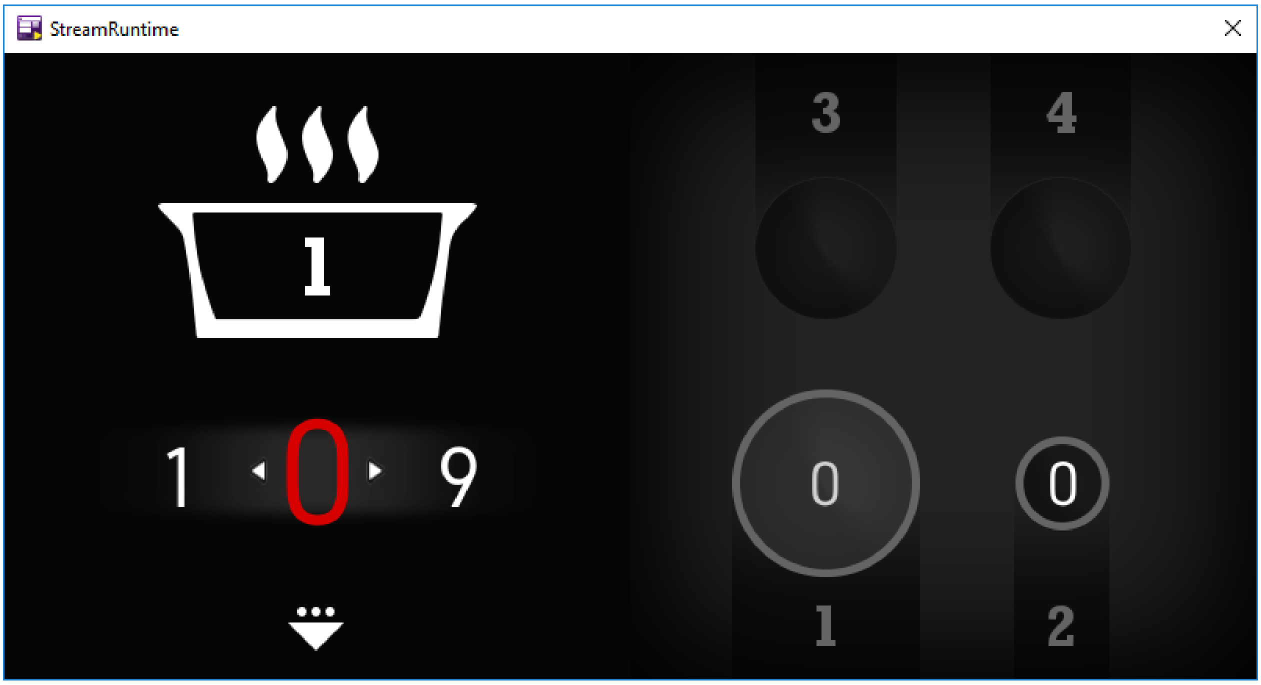


**Fig. 1 This “How To” result**

# 2. Using a Behaviour without programming

## 2.1. Step 1: Load project “step by step”

- Start the GSE and load the created Step by Step project from HowTo 3 (step_by_step.gpr).

## 2.2. Step 2: Preparations

Before we will use the Behaviour, we have to add some images to our project.
We need to add another wheel for burner 2 and we have to display the burner.
For this HowTo we will use burner 1 and 2, only.

First we will add the wheel.

### 2.2.1. Adding wheel for burner 2

In the last HowTo (HowTo 3 - using a DataPool in GSE) we connected the wheel with the burner 1 using a DataPool.
So, if we like to use burner 2, we cannot connect to the present wheel, because it shows the data for burner 1. We have to use a new wheel which we have to connect to burner 2.

We do not have to add a new wheel control like we have done it in the chapters 2.6 to 2.12 in the document “How to 1 - build a project step by step”.
Just copy the existing one, because we need exactly the same wheel at the same position.

- Select “**AID_WHEEL_1**” in the “**Object Hierarchy**” window.
- Copy “**AID_WHEEL_1**” (*Edit*➔*Copy*).
- Paste the copied wheel to the main dialog (*Edit*➔*Paste*).

Now we have an exact copy of our wheel – with even the same **ObjectID**.

- Change the **ObjectID** of the topmost “**AID_WHEEL_1**” to “**AID_WHEEL_2**” by clicking onto the triangle behind the Attribute “**ObjectID**” and choose “**new object-ID…**”. Then enter “AID_WHEEL_2”.
- Save your project and run it.

You will notice two things: Both wheels will be displayed and a change of the wheel's value will not lead to a change in burner 1.

If we check the attribute "**Invisible**" of **AID_WHEEL_2**, we can use the first wheel for burner 1 again.

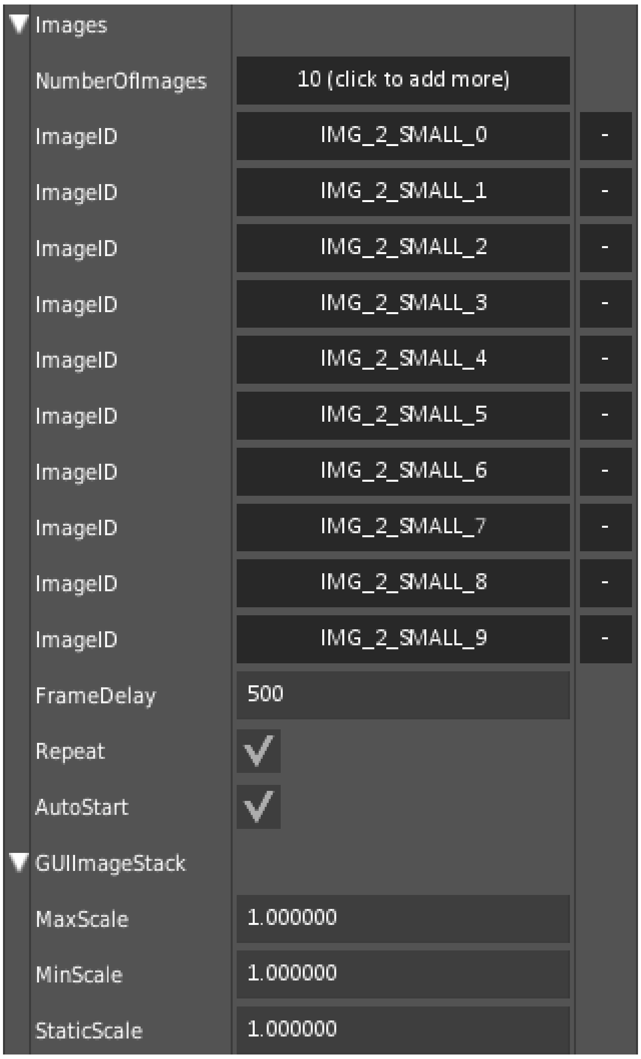
But how can we use our new wheel for burner 2?
You will get the answer soon, but first let us add some images to burner 2, so it will look similar to burner 1.

Like burner 1, the ring of burner 2 should be "glow" according to the heating intensity (shown as the number inside the wheel).

### 2.2.2. Adding an Image Stack for burner 2

- Add an "**image stack**" control to the main window.
- Set the image stack's attributes **MaxScale**, **MinScale** and **StaticScale** to 1 and **XPos** to 646, **YPos** to 245 and **Width** and **Height** to 60.
- Scroll down the attribute list to **Image**. Here you see the corresponding attributes for the handling of the images to be shown.
- Click onto the button behind "**NumberOfImages**". The text will change from "**0 (click to add more)**" to "**1 (click to add more)**" and another attribute appears, called "**ImageID**".
- Click onto the button behind "**ImageID**". Now you can select an image. Add the image "2_small_0.png" from inside the folder "*How to 4 – resources/burne2*" of your SDK's documentation folder.
- Repeat this nine times, adding the images "2_small_1.png" to "2_small_9.png".

The attributes should be set like this:



| | | |
|---|---|---|
| ▼ Images | | |
| NumberOfImages | 10 (click to add more) | |
| ImageID | IMG_2_SMALL_0 | - |
| ImageID | IMG_2_SMALL_1 | - |
| ImageID | IMG_2_SMALL_2 | - |
| ImageID | IMG_2_SMALL_3 | - |
| ImageID | IMG_2_SMALL_4 | - |
| ImageID | IMG_2_SMALL_5 | - |
| ImageID | IMG_2_SMALL_6 | - |
| ImageID | IMG_2_SMALL_7 | - |
| ImageID | IMG_2_SMALL_8 | - |
| ImageID | IMG_2_SMALL_9 | - |
| FrameDelay | 500 | |
| Repeat | ✓ | |
| AutoStart | ✓ | |
| ▼ GUIImageStack | | |
| MaxScale | 1.000000 | |
| MinScale | 1.000000 | |
| StaticScale | 1.000000 | |

**Fig. 2 Added images to the stack**

Save your project and run it.

➡ The project should look like this:

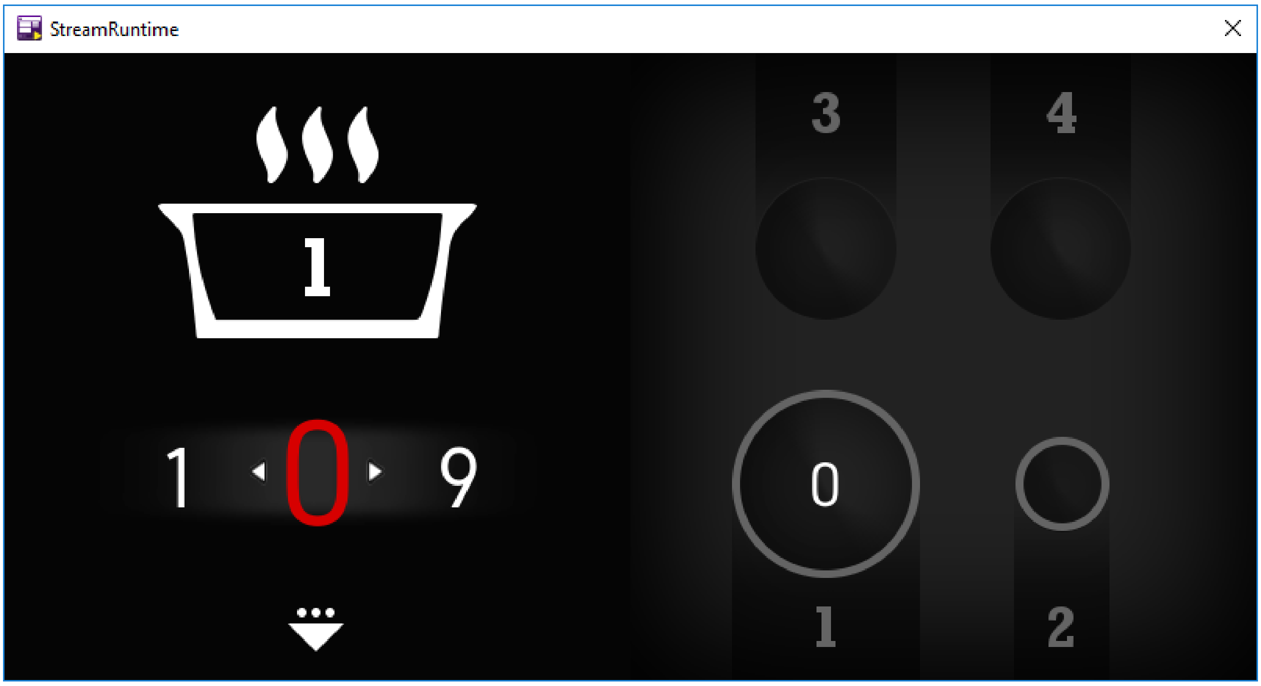


**Fig. 3 Run project - with burner rings**

### 2.2.3. Select burner 1 – using a behaviour

Now we could show two wheels with the temperature state for the appropriate burner (1 or 2). But how can we select the burner (shown as a number in the pot)?

First, we have to add a text field into the burner 2.

▶ Add a text field to the main dialog.

▶ Set **XPos** to 651, **YPos** to 252, **Width** to 50 and **Height** to 50.

> **!** Make sure, that the **TextWidth** and **TextHeight** attributes are similar to the **Width** and **Height** attributes (all 50).

▶ The color for the text should be white, so you have to set all attributes beginning with "**TextColor**" to white (0XFFFFFFFF).

▶ Replace "**text field**" with "0" in the text field behind the attribute "**Text**".

▶ The text is too small, so we have to change the text font to "Gidole-Regular" 40.

> **!** If you have problems at this point, please take a look into the document "How to 1 - build a project step by step" and read through chapter 2.12. Setting a control's font.

▶ Save your project and run it.

➡ The project should look like this:

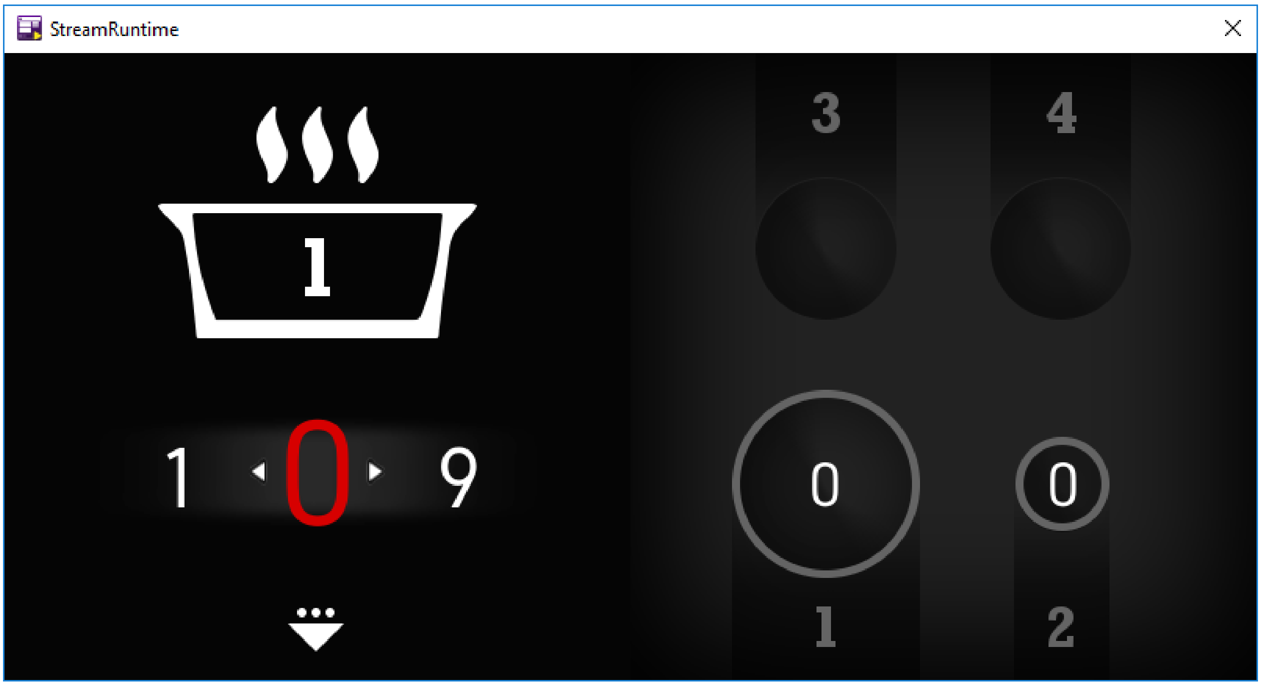


**Fig. 4 Run project - with two burners**

We should use an image to show if a burner is selected or not.
So we will add a picture to burner 1 and 2, showing a selected state.

- Add an “**image**” control to the main window.
- Set the image’s attributes **XPos** to 465, **YPos** to 215 and **Width** and **Height** to 120.
- Click onto the button behind “**ImageID**”. Add the image “1_big_selected.png” from inside the folder “*How to 4 – resources*” of your SDK’s documentation folder.
- Add an “**image**” control to the main window.
- Set the image’s attributes **XPos** to 646, **YPos** to 245 and **Width** and **Height** to 60.
- Click onto the button behind “**ImageID**”. Add the image “2_small_selected.png” from inside the folder “*How to 4 – resources*” of your SDK’s documentation folder.
- Because only one burner is selected at the same time, check “**Invisible**” for this image (**AID_IMAGE_4**).
- Save your project and run it.

➡ The project should look like this:

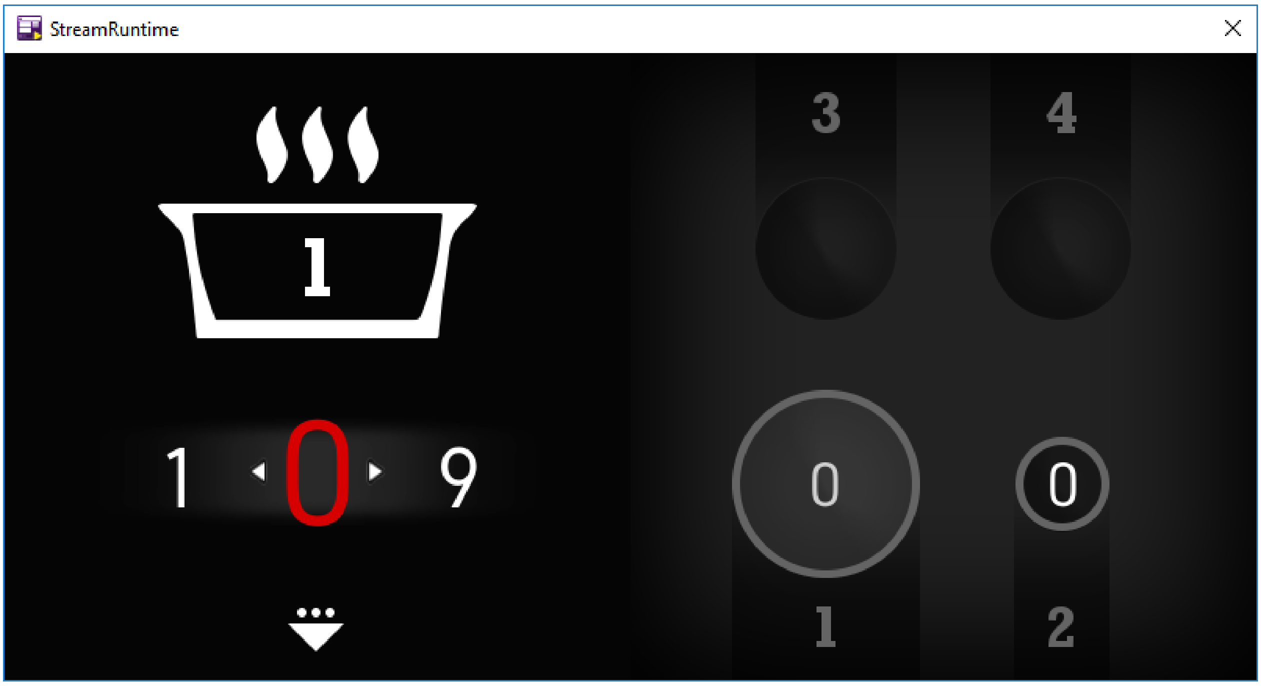


**Fig. 5 Run project - burner 1 selected**

Right now, burner 1 is selected. So, if the user clicks onto burner 2, it should be selected and burner 1 deselected. Simultaneously the right temperature state should be shown in the wheel. That means, wheel 1 should be invisible and wheel 2 visible.
Let's do exactly this, step by step, using behaviours.
But the question is: where to add the behaviour for the burner?
The user should click everywhere in the space defined by the background for burner 1.
We have to add a control that covers this space and will have the appropriate behaviour.

- Add an image to the main dialog (**AID_IMAGE_5**).
- Set **XPos** to 465, **YPos** to 215, **Width** to 120 and **Height** to 185.
- Choose "**DUMMY_IMAGE**" as **ImageID**.
- Add another image to the main dialog (**AID_IMAGE_6**).
- Set **XPos** to 646, **YPos** to 245, **Width** to 60 and **Height** to 155.
- Choose "**DUMMY_IMAGE**" as **ImageID**.

So, what should happen, if the user clicks onto the burner 1 (the image **AID_IMAGE_5**)?

### 2.2.3.1. Wheel 1 should be displayed

- Select AID_IMAGE_5 in the "Object Hierarchy" window.

- Click onto the triangle behind the attribute BehaviourClassID and choose "BEHAVIOUR_SINGLE_CMD".

- The label GUIBehaviour changes to GUISingleCmdBehaviour and a new attribute "GUICommand" appears.

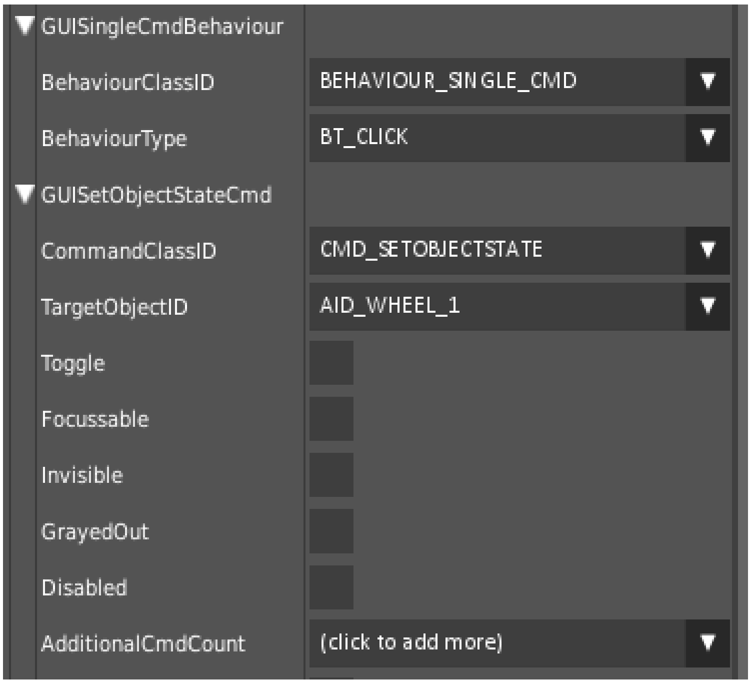
- Click onto the button behind the attribute "BehaviourType" and choose "BT_CLICK".

- Below the attribute GUICommand you will find its attribute "CommandClassID". Click onto the button behind this attribute and choose "CMD_SETOBJECTSTATE".

Here, we will choose wheel 1. Do not change anything else. We want to have wheel 1 visible. And therefore Invisible must be left unchecked. The command will set it correctly.

- Select AID_WHEEL_1 as TargetObjectID.

- The attributes should look like this:



| | |
|---|---|
| ▼ GUISingleCmdBehaviour | |
| BehaviourClassID | BEHAVIOUR_SINGLE_CMD ▼ |
| BehaviourType | BT_CLICK ▼ |
| ▼ GUISetObjectStateCmd | |
| CommandClassID | CMD_SETOBJECTSTATE ▼ |
| TargetObjectID | AID_WHEEL_1 ▼ |
| Toggle | □ |
| Focussable | □ |
| Invisible | □ |
| GrayedOut | □ |
| Disabled | □ |
| AdditionalCmdCount | (click to add more) ▼ |

**Fig. 6 First command for burner 1 set**

### 2.2.3.2. Wheel 2 should be hidden

▶ Keep **AID_IMAGE_5** selected in the "**Object Hierarchy**" window.

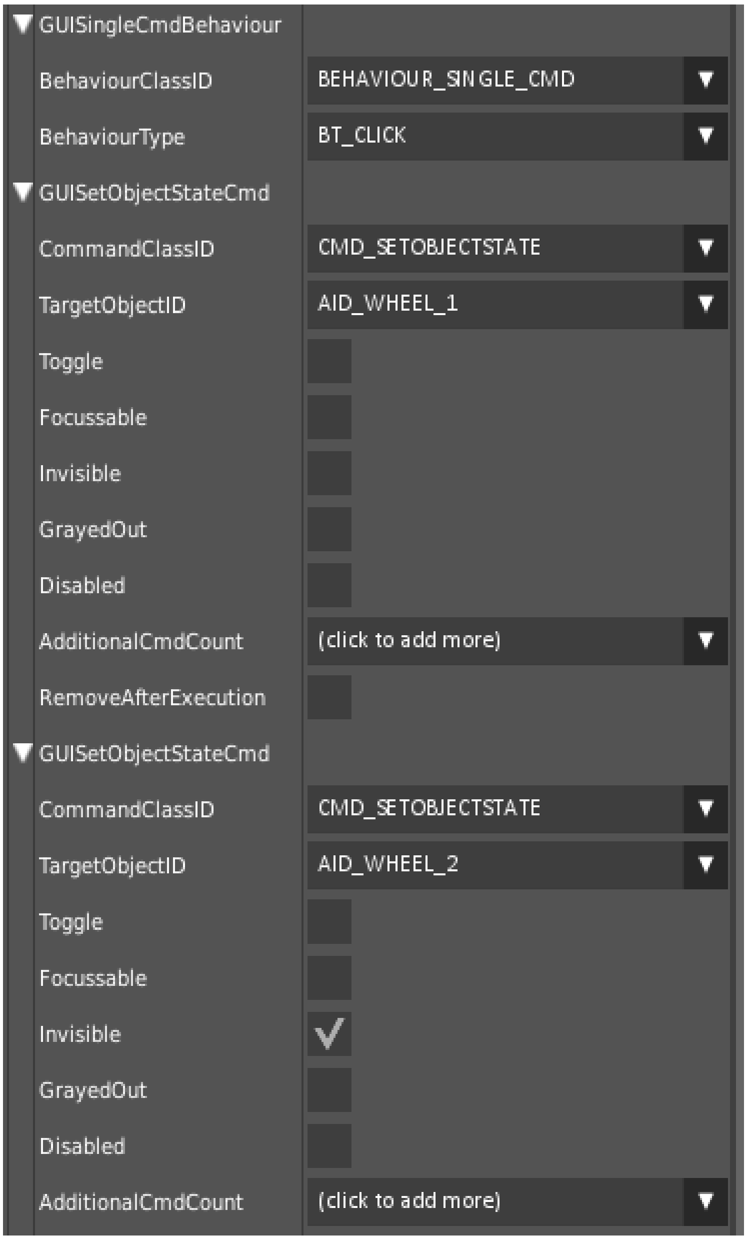
▶ Click onto the button behind the attribute "**AdditionalCmdCount**" (labeled with "**(click to add more)**" and choose "**CMD_SETOBJECTSTATE**".

➡ A new (second) attribute "**GUISetObjectStateCmd**" appears. And a new (second) attribute "**CommandClassID**" will be seen as its attribute.

We do not have to change the "**CommandClassID**", because the same command which has been chosen before will be selected: "**CMD_SETOBJECTSTATE**".

▶ Select **AID_WHEEL_2** as **TargetObjectID** (labeled with "**NO_HANDLE**" at the moment) and check "**Invisible**".

The attributes should look like this:



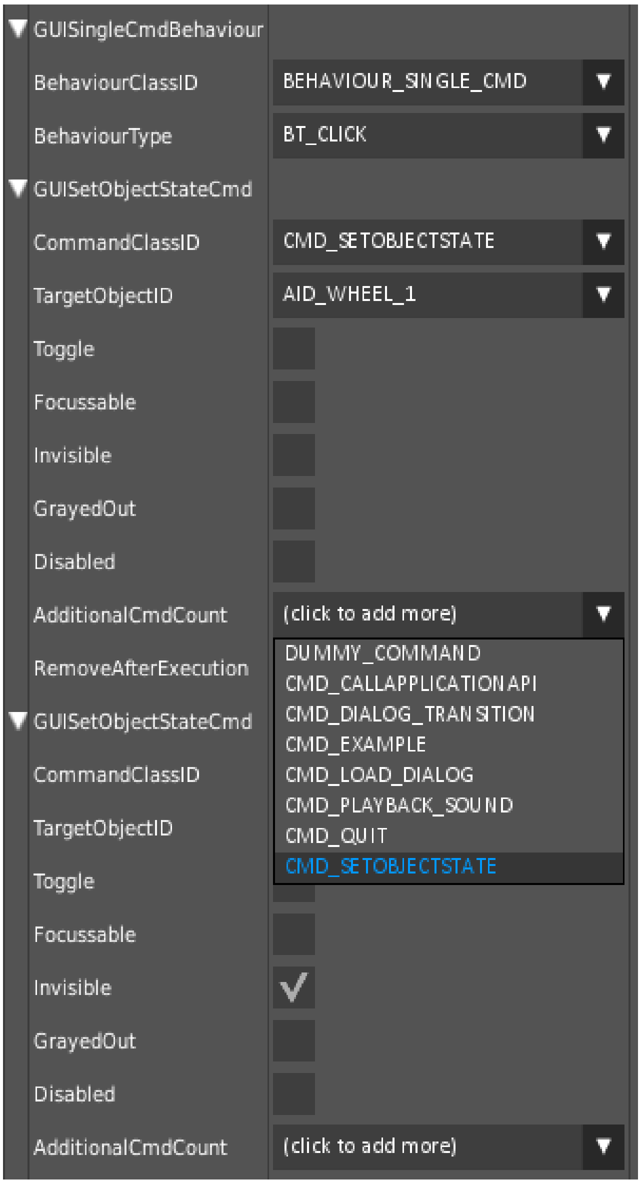
| | |
|---|---|
| ▼ GUISingleCmdBehaviour | |
| BehaviourClassID | BEHAVIOUR_SINGLE_CMD ▼ |
| BehaviourType | BT_CLICK ▼ |
| ▼ GUISetObjectStateCmd | |
| CommandClassID | CMD_SETOBJECTSTATE ▼ |
| TargetObjectID | AID_WHEEL_1 ▼ |
| Toggle | ☐ |
| Focussable | ☐ |
| Invisible | ☐ |
| GrayedOut | ☐ |
| Disabled | ☐ |
| AdditionalCmdCount | (click to add more) ▼ |
| RemoveAfterExecution | ☐ |
| ▼ GUISetObjectStateCmd | |
| CommandClassID | CMD_SETOBJECTSTATE ▼ |
| TargetObjectID | AID_WHEEL_2 ▼ |
| Toggle | ☐ |
| Focussable | ☐ |
| Invisible | ☑ |
| GrayedOut | ☐ |
| Disabled | ☐ |
| AdditionalCmdCount | (click to add more) ▼ |

**Fig. 7 Second command for burner 1 set**

### 2.2.3.3. Burner 1 should be shown as selected

- We still work on **AID_IMAGE_5**.

- Click onto the button behind the second attribute "**AdditionalCmdCount**" and choose "**CMD_SETOBJECTSTATE**".



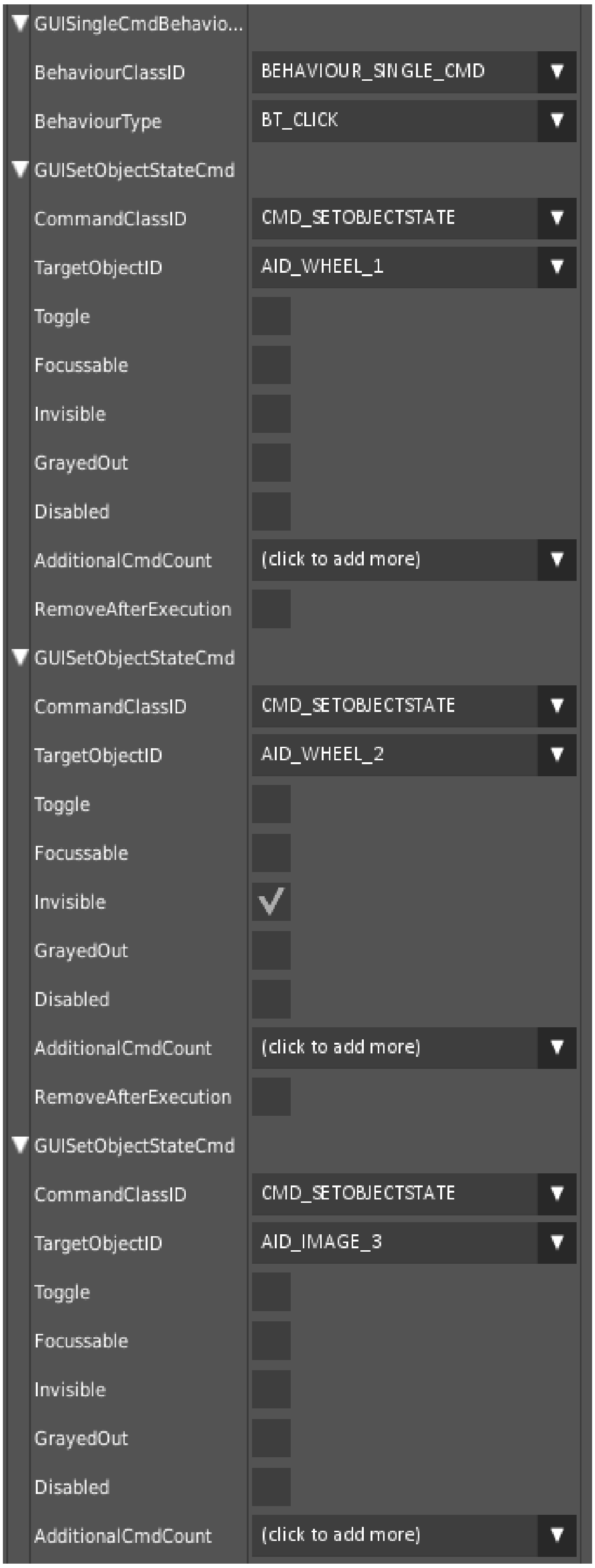
| | |
|---|---|
| ▼ GUISingleCmdBehaviour | |
| BehaviourClassID | BEHAVIOUR_SINGLE_CMD ▼ |
| BehaviourType | BT_CLICK ▼ |
| ▼ GUISetObjectStateCmd | |
| CommandClassID | CMD_SETOBJECTSTATE ▼ |
| TargetObjectID | AID_WHEEL_1 ▼ |
| Toggle | □ |
| Focussable | □ |
| Invisible | □ |
| GrayedOut | □ |
| Disabled | □ |
| AdditionalCmdCount | (click to add more) ▼ |
| RemoveAfterExecution | DUMMY_COMMAND<br>CMD_CALLAPPLICATIONAPI<br>CMD_DIALOG_TRANSITION<br>CMD_EXAMPLE<br>CMD_LOAD_DIALOG<br>CMD_PLAYBACK_SOUND<br>CMD_QUIT<br>CMD_SETOBJECTSTATE |
| ▼ GUISetObjectStateCmd | |
| CommandClassID | |
| TargetObjectID | |
| Toggle | |
| Focussable | □ |
| Invisible | ☑ |
| GrayedOut | □ |
| Disabled | □ |
| AdditionalCmdCount | (click to add more) ▼ |

**Fig. 8 Additional command**

- A new (third) attribute "**GUISetObjectStateCmd**" appears below the other two. And a new (third) attribute "**CommandClassID**" will be seen.

- Select **AID_IMAGE_3** as **TargetObjectID**. Leave all other attributes unchecked.

The attributes should look like this:



| | |
|---|---|
| ▼ GUISingleCmdBehavio... | |
| BehaviourClassID | BEHAVIOUR_SINGLE_CMD ▼ |
| BehaviourType | BT_CLICK ▼ |
| ▼ GUISetObjectStateCmd | |
| CommandClassID | CMD_SETOBJECTSTATE ▼ |
| TargetObjectID | AID_WHEEL_1 ▼ |
| Toggle | □ |
| Focussable | □ |
| Invisible | □ |
| GrayedOut | □ |
| Disabled | □ |
| AdditionalCmdCount | (click to add more) ▼ |
| RemoveAfterExecution | □ |
| ▼ GUISetObjectStateCmd | |
| CommandClassID | CMD_SETOBJECTSTATE ▼ |
| TargetObjectID | AID_WHEEL_2 ▼ |
| Toggle | □ |
| Focussable | □ |
| Invisible | ☑ |
| GrayedOut | □ |
| Disabled | □ |
| AdditionalCmdCount | (click to add more) ▼ |
| RemoveAfterExecution | □ |
| ▼ GUISetObjectStateCmd | |
| CommandClassID | CMD_SETOBJECTSTATE ▼ |
| TargetObjectID | AID_IMAGE_3 ▼ |
| Toggle | □ |
| Focussable | □ |
| Invisible | □ |
| GrayedOut | □ |
| Disabled | □ |
| AdditionalCmdCount | (click to add more) ▼ |

**Fig. 9 Third command for burner 1 set**

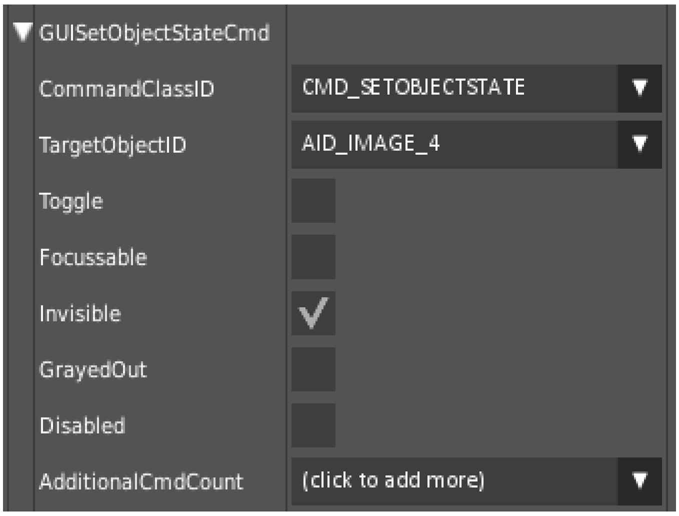
### 2.2.3.4. Burner 2 should be shown as deselected

▶ Click onto the button behind **AID_IMAGE_5**’s last (third) “**AdditionalCmdCount**” attribute and choose “**CMD_SETOBJECTSTATE**”.

➡ A new (fourth) attribute “**GUISetObjectStateCmd**” appears. And a new (fourth) attribute “**CommandClassID**” will be seen.

▶ Select **AID_IMAGE_4** as **TargetObjectID** and check **Invisible**.

➡ The attributes should look like this:



▼ GUISetObjectStateCmd

CommandClassID: CMD_SETOBJECTSTATE ▼

TargetObjectID: AID_IMAGE_4 ▼

Toggle: ☐

Focussable: ☐

Invisible: ☑

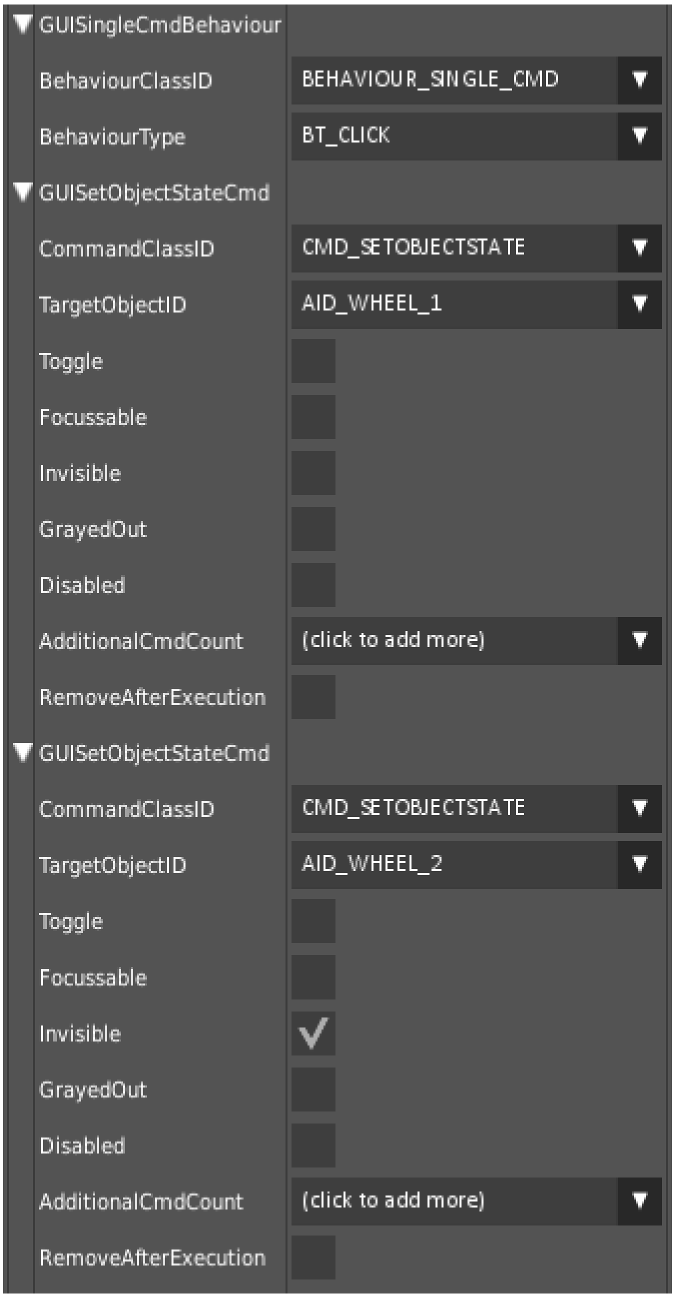
GrayedOut: ☐

Disabled: ☐

AdditionalCmdCount: (click to add more) ▼

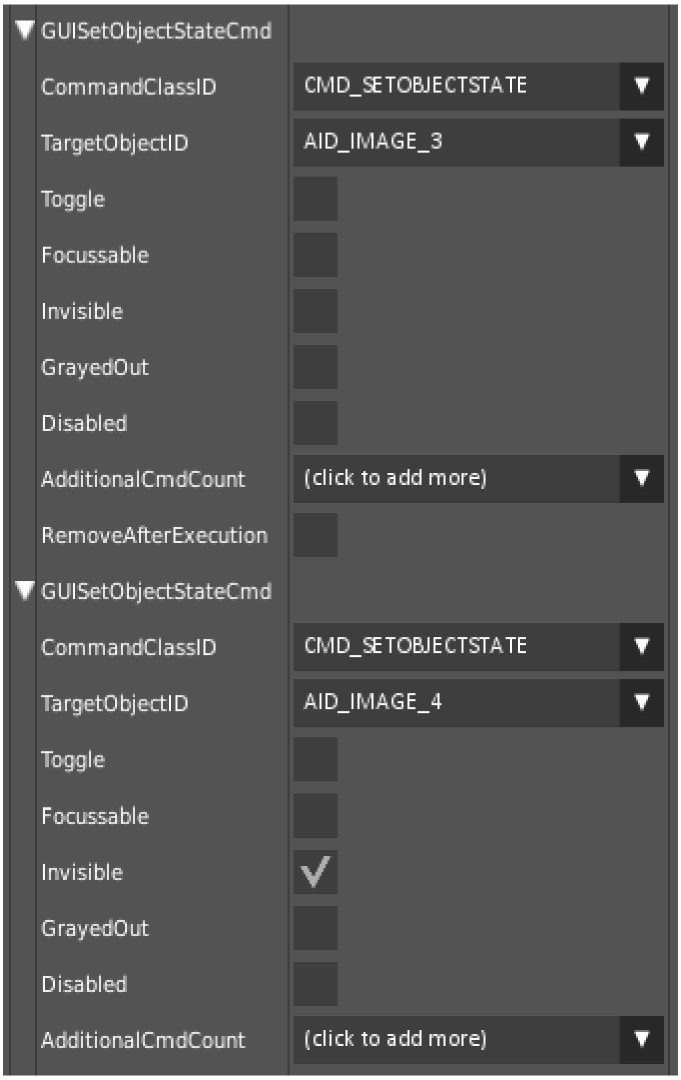
**Fig. 10 Fourth command for burner 1 set**

The whole behaviour/attributes list should look like this:



| | |
|---|---|
| ▼ GUISingleCmdBehaviour | |
| BehaviourClassID | BEHAVIOUR_SINGLE_CMD ▼ |
| BehaviourType | BT_CLICK ▼ |
| ▼ GUISetObjectStateCmd | |
| CommandClassID | CMD_SETOBJECTSTATE ▼ |
| TargetObjectID | AID_WHEEL_1 ▼ |
| Toggle | □ |
| Focussable | □ |
| Invisible | □ |
| GrayedOut | □ |
| Disabled | □ |
| AdditionalCmdCount | (click to add more) ▼ |
| RemoveAfterExecution | □ |
| ▼ GUISetObjectStateCmd | |
| CommandClassID | CMD_SETOBJECTSTATE ▼ |
| TargetObjectID | AID_WHEEL_2 ▼ |
| Toggle | □ |
| Focussable | □ |
| Invisible | ☑ |
| GrayedOut | □ |
| Disabled | □ |
| AdditionalCmdCount | (click to add more) ▼ |
| RemoveAfterExecution | □ |

**Fig. 11 All commands set for burner 1 - part 1**



| | |
|---|---|
| ▼ GUISetObjectStateCmd | |
| CommandClassID | CMD_SETOBJECTSTATE ▼ |
| TargetObjectID | AID_IMAGE_3 ▼ |
| Toggle | □ |
| Focussable | □ |
| Invisible | □ |
| GrayedOut | □ |
| Disabled | □ |
| AdditionalCmdCount | (click to add more) ▼ |
| RemoveAfterExecution | □ |
| ▼ GUISetObjectStateCmd | |
| CommandClassID | CMD_SETOBJECTSTATE ▼ |
| TargetObjectID | AID_IMAGE_4 ▼ |
| Toggle | □ |
| Focussable | □ |
| Invisible | ☑ |
| GrayedOut | □ |
| Disabled | □ |
| AdditionalCmdCount | (click to add more) ▼ |

**Fig. 12 All commands set for burner 1 - part 2**

Now we have to add a behaviour to start the actions, which take place, if the user clicks onto the burner 2 (the image **AID_IMAGE_6**).

## 2.2.4. Select burner 2 – using a behaviour

### 2.2.4.1. Wheel 2 should be displayed

▶ Select **AID_IMAGE_6** in the “**Object Hierarchy**” window.

▶ Click onto the triangle behind the attribute **BehaviourClasID** and choose “**BEHAVIOUR_SINGLE_CMD**”.

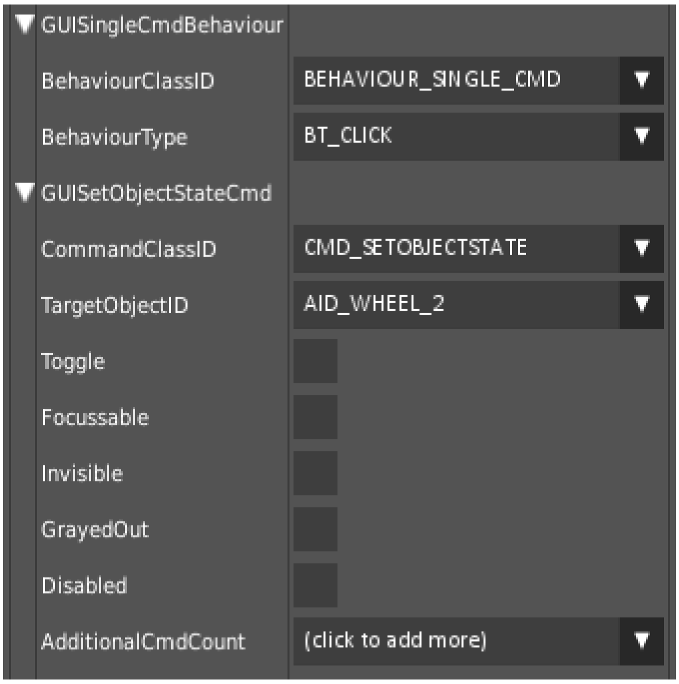
➡ The label **GUIBehaviour** changes to **GUISingleCmdBehaviour** and a new attribute “**GUICommand**” appears.

▶ Click onto the button behind the attribute “**BehaviourType**” and choose “**BT_CLICK**”.

▶ Below the attribute **GUICommand** you will find its attribute “**CommandClassID**”. Click onto the button behind this attribute and choose “**CMD_SETOBJECTSTATE**”.

▶ Select **AID_WHEEL_2** as **TargetObjectID**.

➡ The attributes should look like this:



| | |
|---|---|
| ▼ GUISingleCmdBehaviour | |
| BehaviourClassID | BEHAVIOUR_SINGLE_CMD ▼ |
| BehaviourType | BT_CLICK ▼ |
| ▼ GUISetObjectStateCmd | |
| CommandClassID | CMD_SETOBJECTSTATE ▼ |
| TargetObjectID | AID_WHEEL_2 ▼ |
| Toggle | □ |
| Focussable | □ |
| Invisible | □ |
| GrayedOut | □ |
| Disabled | □ |
| AdditionalCmdCount | (click to add more) ▼ |

**Fig. 13 First command for burner 2 set**

### 2.2.4.2. Wheel 1 should be hidden

▶ Keep **AID_IMAGE_6** selected in the “**Object Hierarchy**” window.

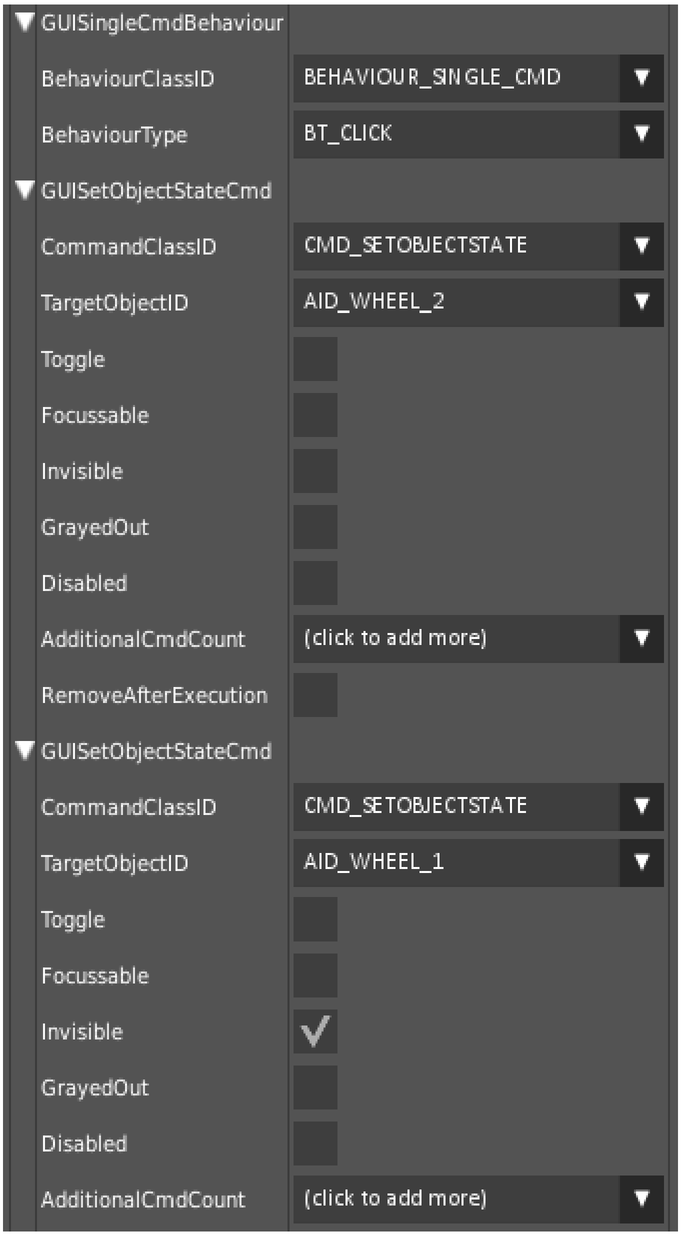
▶ Click onto the button behind the attribute “**AdditionalCmdCount**” (labeled with “**(click to add more)**” and choose “**CMD_SETOBJECTSTATE**”.

➡ A new (second) attribute “**GUISetObjectStateCmd**” appears. And a new (second) attribute “**CommandClassID**” will be seen as its attribute.

We do not have to change the “**CommandClassID**”, because the same command which has been chosen before will be selected: “**CMD_SETOBJECTSTATE**”.

▶ Select **AID_WHEEL_1** as **TargetObjectID** (labeled with “**NO_HANDLE**” at the moment) and check “**Invisible**”.

The attributes should look like this:



| | |
|---|---|
| ▼ GUISingleCmdBehaviour | |
| BehaviourClassID | BEHAVIOUR_SINGLE_CMD ▼ |
| BehaviourType | BT_CLICK ▼ |
| ▼ GUISetObjectStateCmd | |
| CommandClassID | CMD_SETOBJECTSTATE ▼ |
| TargetObjectID | AID_WHEEL_2 ▼ |
| Toggle | □ |
| Focussable | □ |
| Invisible | □ |
| GrayedOut | □ |
| Disabled | □ |
| AdditionalCmdCount | (click to add more) ▼ |
| RemoveAfterExecution | □ |
| ▼ GUISetObjectStateCmd | |
| CommandClassID | CMD_SETOBJECTSTATE ▼ |
| TargetObjectID | AID_WHEEL_1 ▼ |
| Toggle | □ |
| Focussable | □ |
| Invisible | ☑ |
| GrayedOut | □ |
| Disabled | □ |
| AdditionalCmdCount | (click to add more) ▼ |

**Fig. 14 Second command for burner 2 set**

### 2.2.4.3. Burner 2 should be shown as selected

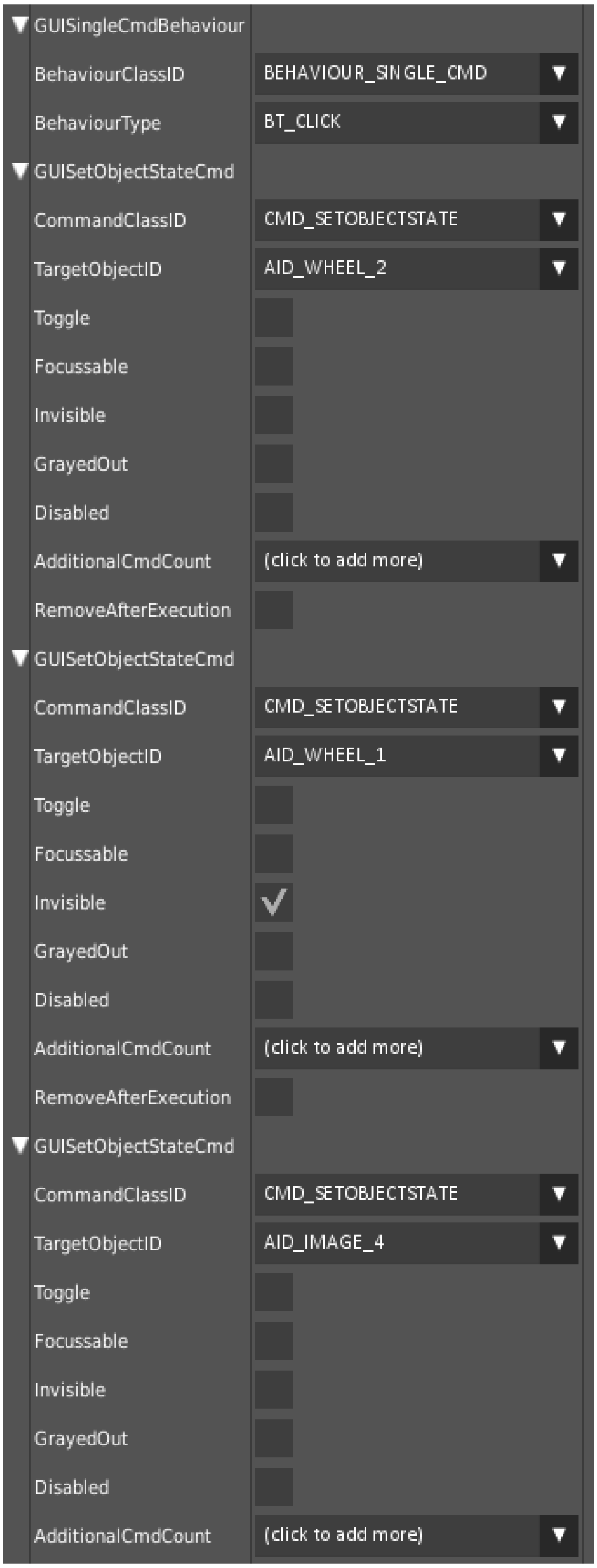
- ▶ We still work on **AID_IMAGE_6**.

- ▶ Click onto the button behind the second attribute “**AdditionalCmdCount**” and choose “**CMD_SETOBJECTSTATE**”.

- ➡ A new (third) attribute “**GUISetObjectStateCmd**” appears. And a new (third) attribute “**CommandClassID**” will be seen.

- ▶ Select **AID_IMAGE_4** as **TargetObjectID**. Leave all other attributes unchecked.

The attributes should look like this:



| | |
|---|---|
| ▼ GUISingleCmdBehaviour | |
| BehaviourClassID | BEHAVIOUR_SINGLE_CMD ▼ |
| BehaviourType | BT_CLICK ▼ |
| ▼ GUISetObjectStateCmd | |
| CommandClassID | CMD_SETOBJECTSTATE ▼ |
| TargetObjectID | AID_WHEEL_2 ▼ |
| Toggle | □ |
| Focussable | □ |
| Invisible | □ |
| GrayedOut | □ |
| Disabled | □ |
| AdditionalCmdCount | (click to add more) ▼ |
| RemoveAfterExecution | □ |
| ▼ GUISetObjectStateCmd | |
| CommandClassID | CMD_SETOBJECTSTATE ▼ |
| TargetObjectID | AID_WHEEL_1 ▼ |
| Toggle | □ |
| Focussable | □ |
| Invisible | ☑ |
| GrayedOut | □ |
| Disabled | □ |
| AdditionalCmdCount | (click to add more) ▼ |
| RemoveAfterExecution | □ |
| ▼ GUISetObjectStateCmd | |
| CommandClassID | CMD_SETOBJECTSTATE ▼ |
| TargetObjectID | AID_IMAGE_4 ▼ |
| Toggle | □ |
| Focussable | □ |
| Invisible | □ |
| GrayedOut | □ |
| Disabled | □ |
| AdditionalCmdCount | (click to add more) ▼ |

**Fig. 15 Third command for burner 2 set**

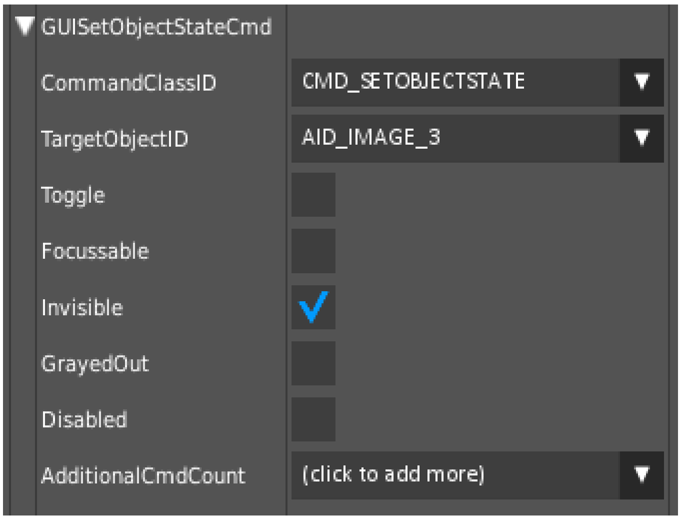
### 2.2.4.4. Burner 1 should be shown as deselected

▶ Click onto the button behind **AID_IMAGE_6**’s last (third) “**AdditionalCmdCount**” attribute and choose “**CMD_SETOBJECTSTATE**”.

➡ A new (fourth) attribute “**GUISetObjectStateCmd**” appears. And a new (fourth) attribute “**CommandClassID**” will be seen.

▶ Select **AID_IMAGE_3** as **TargetObjectID** and check **Invisible**.

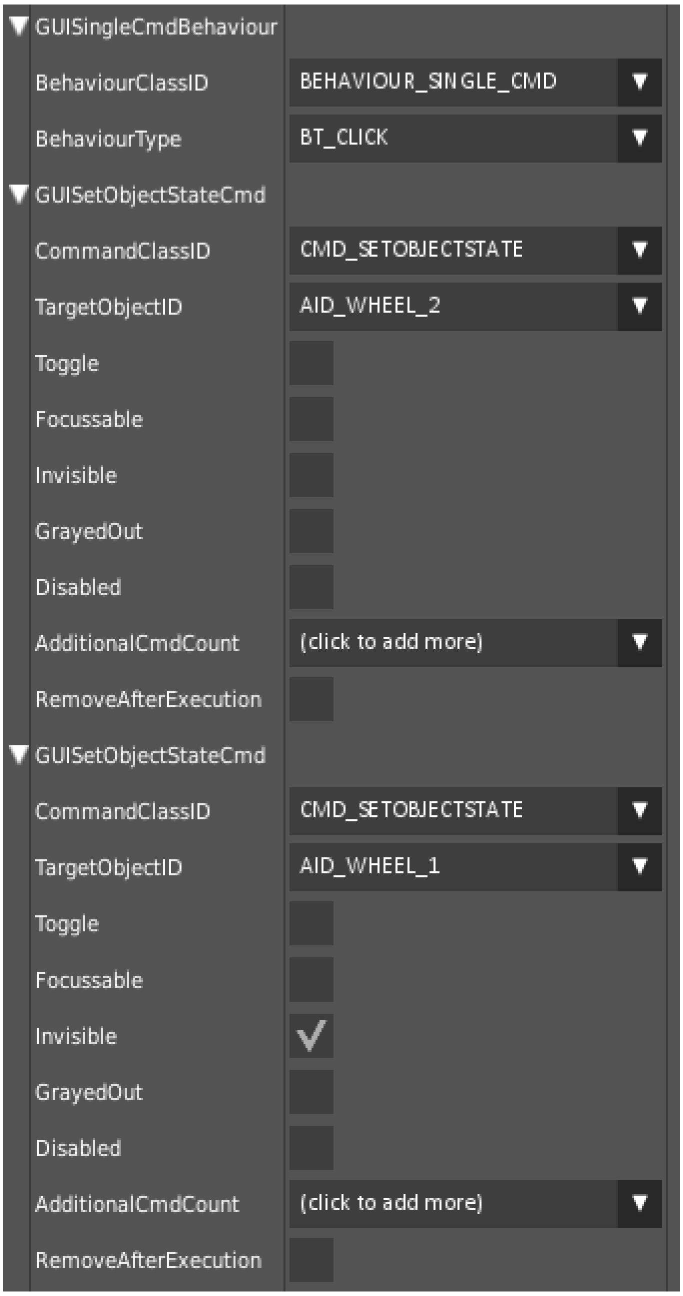
➡ The attributes should look like this:



| | |
|---|---|
| ▼ GUISetObjectStateCmd | |
| CommandClassID | CMD_SETOBJECTSTATE ▼ |
| TargetObjectID | AID_IMAGE_3 ▼ |
| Toggle | □ |
| Focussable | □ |
| Invisible | ☑ |
| GrayedOut | □ |
| Disabled | □ |
| AdditionalCmdCount | (click to add more) ▼ |

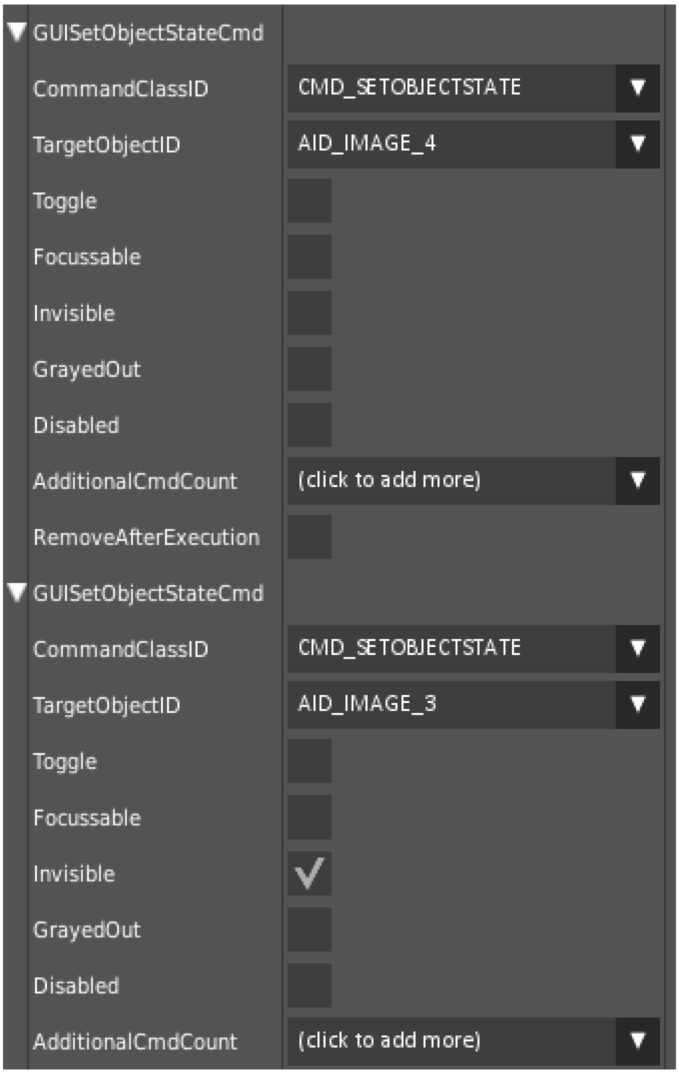
**Fig. 16 Fourth command for burner 2 set**

➡ The whole behaviour/attributes list should look like this:



| | |
|---|---|
| ▼ GUISingleCmdBehaviour | |
| BehaviourClassID | BEHAVIOUR_SINGLE_CMD ▼ |
| BehaviourType | BT_CLICK ▼ |
| ▼ GUISetObjectStateCmd | |
| CommandClassID | CMD_SETOBJECTSTATE ▼ |
| TargetObjectID | AID_WHEEL_2 ▼ |
| Toggle | □ |
| Focussable | □ |
| Invisible | □ |
| GrayedOut | □ |
| Disabled | □ |
| AdditionalCmdCount | (click to add more) ▼ |
| RemoveAfterExecution | □ |
| ▼ GUISetObjectStateCmd | |
| CommandClassID | CMD_SETOBJECTSTATE ▼ |
| TargetObjectID | AID_WHEEL_1 ▼ |
| Toggle | □ |
| Focussable | □ |
| Invisible | ☑ |
| GrayedOut | □ |
| Disabled | □ |
| AdditionalCmdCount | (click to add more) ▼ |
| RemoveAfterExecution | □ |

**Fig. 17 All commands set for burner 2 - part 1**



| | |
|---|---|
| ▼ GUISetObjectStateCmd | |
| CommandClassID | CMD_SETOBJECTSTATE ▼ |
| TargetObjectID | AID_IMAGE_4 ▼ |
| Toggle | □ |
| Focussable | □ |
| Invisible | □ |
| GrayedOut | □ |
| Disabled | □ |
| AdditionalCmdCount | (click to add more) ▼ |
| RemoveAfterExecution | □ |
| ▼ GUISetObjectStateCmd | |
| CommandClassID | CMD_SETOBJECTSTATE ▼ |
| TargetObjectID | AID_IMAGE_3 ▼ |
| Toggle | □ |
| Focussable | □ |
| Invisible | ☑ |
| GrayedOut | □ |
| Disabled | □ |
| AdditionalCmdCount | (click to add more) ▼ |

**Fig. 18 All commands set for burner 2 - part 2**

- Save your project and run it.

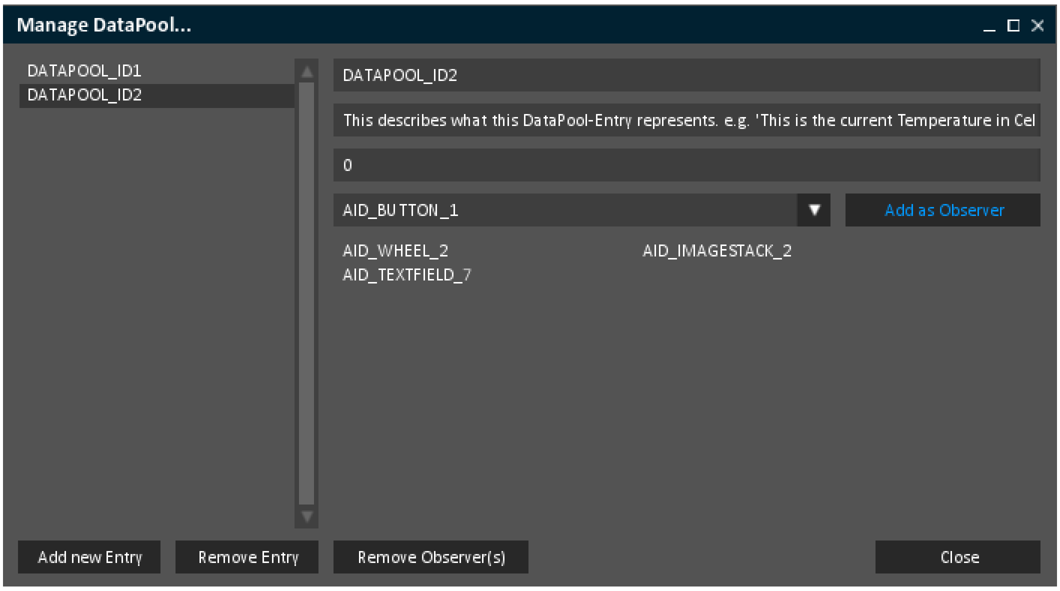
- Click alternately onto burner 1 and 2 and see what happens. And select different numbers for both of the two wheels.

You can see that the wheel, the status of the burners and their values change according to the burner you click on.
Burner 2 shows always 0, because it will not be updated.
So we have to add another DataPool like we have done in “How to 3 - using a DataPool in GSE”.
Can you do it on your own? If not, look at the next chapter.

## 2.2.5. Connect burner 2 with wheel 2 – using a DataPool

We have what we need: wheel 2, an image stack for burner 2 and a text field for burner 2.

- Select *Manage DataPool* from the menu *Resources*.
- Press the button "**Add new Entry**" and select **DATAPOOL_ID2**.
- Select "**AID_WHEEL_2**" from the pull down menu and press the button "**Add as Observer**".
- Select "**AID_IMAGESTACK_2**" from the pull down menu.
- Press the button "**Add as Observer**".
- Select "**AID_TEXTFIELD_7**" from the pull down menu.
- Press the button "**Add as Observer**".
- DataPool should look like this:



Manage DataPool...

DATAPOOL_ID1
DATAPOOL_ID2

DATAPOOL_ID2

This describes what this DataPool-Entry represents. e.g. 'This is the current Temperature in Cel

0

AID_BUTTON_1

AID_WHEEL_2
AID_IMAGESTACK_2
AID_TEXTFIELD_7

**Fig. 19 DataPool 2**

- Save your project and run it.
- Change the value of the wheel and look what happens with burner 2.

What a nice project you have, with a lot of functionality.
The chosen value from the wheel will be displayed in the burner and the burner's ring will be displayed according to the value. And you can change between two burners.

**And the best of all: this has been done without programming!**

What next?

We should connect the text field in the pot with the burners.
Every time a user clicks onto a burner's number, the chosen burner's number should be displayed in the pot.

Can we do it using a DataPool?
This will not work! Because every time the DataPool's value changes, all controls, which have been added as observers, will change their data. So clicking onto the text field of burner 1 will immediately change the text field in the pot (what we want to have) AND the number of burner 2 (what we don't want to have).

Can we use a behaviour?
Yes we can! Instead of a text field inside the pot, we could use four pictures with the numbers from 1 to 4. Then we could add three other commands to our behaviours, which set the image representing the burner's number to visible and the other three images to invisible.
That will be a lot of work and additional pictures.
But we all know, that images needs a lot of memory space and that is what we should avoid on an embedded system.

What can we use instead?
We could use APIs. But for that we need to program.
Read more about how to program and compile your project/the GSE in the next HowTos.

### 2.2.6. Show the wheel’s value in the burner 3 and 4

To show the values in the burners 3 and 4 you have to do just the same you have done for burners 1 and 2: use their own image stacks and wheels, add pictures for the state and the burner’s rings. And don’t forget to add the two needed DataPools.

And here you might have a problem: If you do not use the full version of the GSE, the DataPools are limited. So we have to look for an alternative.

Here, the only way would be to use APIs. But for that we need to program.
Read more about how to program and compile your project/the GSE in the next HowTos.

## 2.3. Step 3: How to continue?

### 2.3.1. Sample solution

Now you should be able to use the DataPool, commands and behaviour. With all of them you have worked in this HowTo.
If you encountered problems or wish to have the solution without creating the project on your own, we have added the sample solution into the folder called “*How to 4 - sample solution*” inside the documentation folder. Here you will find the GSE Project (step_by_step.gpr).

All resources you need for this HowTo will be found in the folder “*How to 4 - resources*”.

For Windows user inside the folder “temp” there is an executable StreamRuntime.exe.

### 2.3.2. Continuing HowTos

You will find an overview of continuing HowTos in the document “HowTo 0 - an overview of building GSE projects”.

Don’t forget to visit our homepage www.guiliani.de to get more information, demos, help, videos and the latest news about guiliani and GSE.

# 3. Index