

Fibaro HCL Cookbook Guide

Table of Content

1	Introduction.....	2
1.1	Motivation.....	2
1.2	Things I would like to document	2
1.3	Abbreviations	2
1.4	Definitions	2
1.5	References.....	2
1.6	Disclaimer	2
2	Introduction.....	2
2.1	Recommended Update Procedure by chaicka	3
3	Forcing Brackets in Block Scenes	4
4	Virtual Device calling a Specific Scene	4
4.1	Definition of the Global Variables	4
4.2	Definition of the scenes	4
4.3	Definition of the Virtual Device	5
4.4	Overview and Function	6
5	Virtual Device Setting Global Variable	6
5.1	Definition of the Global Variables	6
5.2	Definition of the Virtual Device	6
5.3	Overview and Function	9
6	Virtual Device Setting Global Variables using Scenes.....	10
6.1	Definition of the Global Variables	10
6.2	Definition of the Virtual Device	10
6.3	Definition of the scenes	12
6.4	Overview and Function	15
7	Virtual Device and Scenes to Globally Activate/Deactivate Scene Execution.....	16
7.1	Definition of the Global Variables	16
7.2	Definition of the Virtual Device	17
7.3	Definition of the scenes	19
7.4	Overview and Function	21
7.5	Using the Conditions to execute Scenes.....	21
8	Restarting HCL with a VD and Scene with Bootup Scene	22
8.1	The VD initiating the restart	22
8.2	Scene to call the VD to restart	25
9	Smart heating, using Scenes Instead of the Panel.....	26
9.1	Global Variables	27
9.2	Scene definition	27
9.3	VD	33
10	Robomow® RM615 Telling Operational State.....	36
10.1	The VD representing the mower.....	36
10.2	Global Variable	37
10.3	Scenes to evaluate the mower state	37
11	Downgrading HCL/HC2.....	41

1 Introduction

1.1 Motivation

A cookbook guide to the HCL for this and that. Helping newbies to avoid frustration.

Thanks to all the contributors in the Fibaro Forum (<https://forum.fibaro.com/>), without these I would have not been able to get this all working as intended.

Keep in mind:

Do NOT copy scenes as they will most probably not work as intended, open a second window and rewrite it with the necessary modifications.

If a global variable used in a scene is changed, the scene must most probably be rewritten, as it will not be possible to change the variable in the scene unless all the commands below the variable are deleted. The same applies if a VD is renamed.

Try to have items where a modification is very likely all at the bottom of the IF and the THEN, because this allows removing and adding without deleting most of the scene.

It is recommended to use Internet Explorer® to access the web interface as firefox and chrome show a slow performance when updating the pages. At least this is the observation on computers running Windows®.

1.2 Things I would like to document

- Writing an event from a Scene or VD to the the event log (not SMS or e-mail, only an event)
- Clearing the event log from a scene or VD
- Writing more than one variable from a VD
- Calling more than one scene from a VD not using InvokeScene, but calling with the Scene ID

1.3 Abbreviations

1.4 Definitions

1.5 References

1.6 Disclaimer

All the proposals are tested on the available HCL and seem to work fine on this system. If they perform the same way on other system is unknown.

2 Introduction

For all examples, the following user account is used:

Username (superuser or administrator): admin

Password for this account: password

To make this account usable in all the examples, where the authentication is necessary the string admin:password is to be converted to a base64 string.

Human readable admin:password

base64 encoded YWRtaW46cGFzc3dvcmQ=

This can be done using the following web service: <https://www.base64encode.org/>

Good to know:

Built in documentation: <http://HCL-IP-Address/docs>

All commands can be tested and work fine here, but how this is to be translated into HCL VD code I have no clue.

Getting info on a device: [http:// HCL-IP-Address/api/devices/ID](http://HCL-IP-Address/api/devices/ID)

Getting info on a scene: [http:// HCL-IP-Address/api/scenes/ID](http://HCL-IP-Address/api/scenes/ID)

All the rooms: [http:// HCL-IP-Address/api/rooms](http://HCL-IP-Address/api/rooms)

2.1 Recommended Update Procedure by chaicka

1. Perform a configuration backup,
2. Reboot HC2/HCL (logic: flush out non-visible stuffs such as abnormalities, cache, hung processor threads, etc.),
3. Sanity Check for 5-15 minutes (logic: reboot usually boots up fine but just in case of unforeseeable abnormal behavior),
4. Proceed to firmware update,
5. Clear web browser cache,
6. Terminate web browser,
7. Relaunch web browser,
8. Repeat step 5-7 if firmware update still in progress,
9. Change Freeze for 1 hour (logic: allows non-visible backend scripts/processes to complete any conversions, system changes, etc. that the new firmware has to perform).

3 Forcing Brackets in Block Scenes

The strategy when brackets are automatically inserted in block scenes is a mystery.

Brackets are only automatically inserted, when a physical device is added to the IF, AND or OR condition.

To force the block scene editor to add brackets, start the IF or immediately after the AND or OR with a physical device. Then add all the rest of the items necessary for the decision. To make the scene work properly, the physical device must be available in the system.

4 Virtual Device calling a Specific Scene

This virtual device calls a specific scene, which is incrementing a global variable.

Very basic using one button, but showing how it is done. The very same approach is used if several buttons are defined in the VD calling different scenes.

Attention: the THEN of the scene is executed without any evaluation of the IF. So, even the IF is FALSE, the THEN will be executed. Might be wanted in some situations, but might not be wanted in others. If the IF needs to be evaluated before the THEN is executed, the approach as described in chapter 6 is to be used.

4.1 Definition of the Global Variables

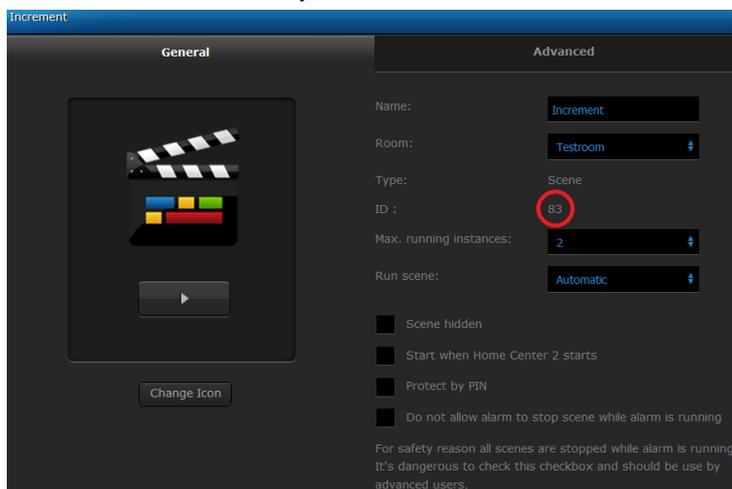
Create a global variable, which will then be modified by the scene called by the VD

Variables:

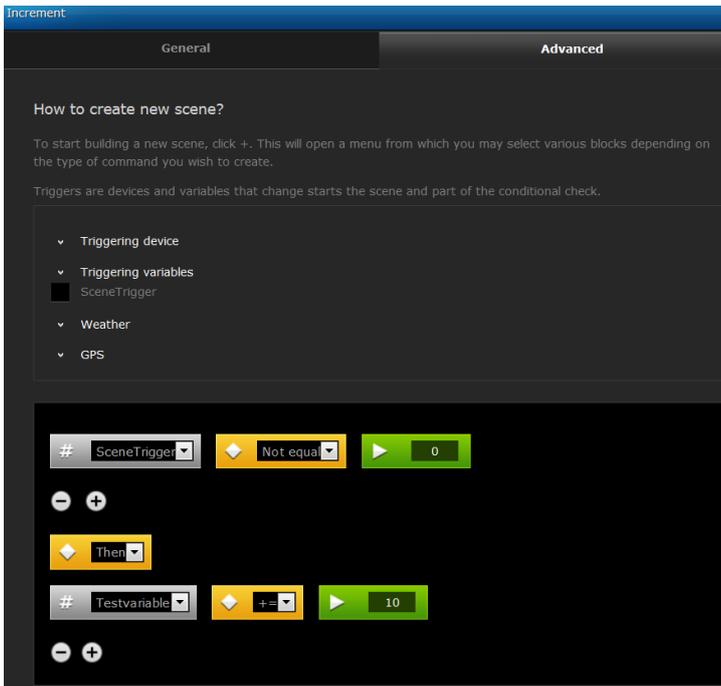
Variable	Value	Delete
Testvariable	0	

4.2 Definition of the scenes

The scene to be called by the VD must be created first, as the ID of the scene must be known in the VD.



In this example the scenes gets the ID 83 after it is saved.



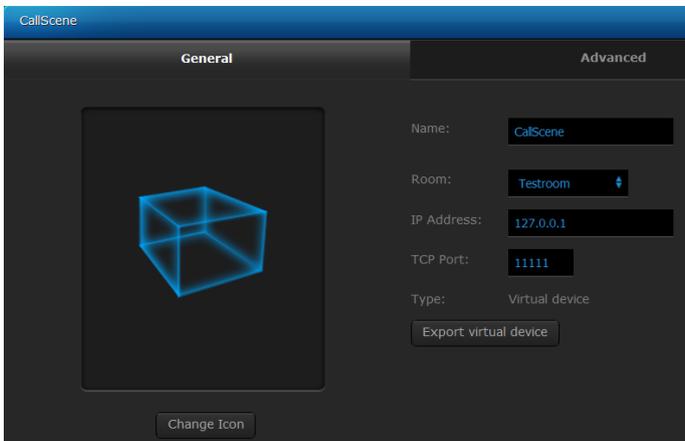
Even the scene will be called explicitly, there is an IF condition needed. Another variable is used to make the IF being true. Without an IF the scene does not work.

Even a scene without an IF can be defined and does not lead to an error message when saved, the THEN will never be called, even the scene is explicitly called by the VD or with the  button in the scene overview

Would be very nice if the current value of the variable could be shown in the label. However, I do not know how this might work.

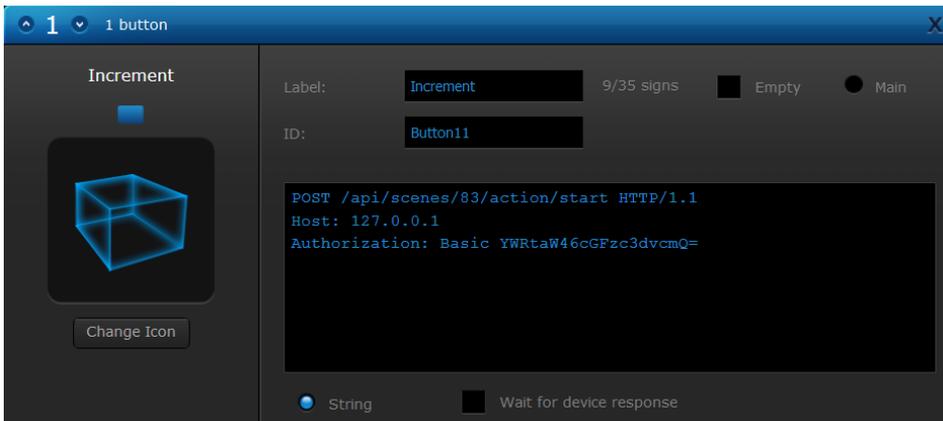
4.3 Definition of the Virtual Device

This virtual device calls a specific scene.



Create the VD with one button.

Make sure to put in the correct ID of the scene. Scene ID is 83 in this example (red circle in above scene definition).

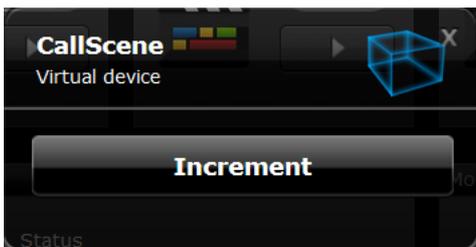


Explanation of the code-lines in the buttons:

POST /api/scenes/83/action/start HTTP/1.1 # start the scene with ID 83
 Host: 127.0.0.1 # the local host of the HCL
 Authorization: Basic YWRtaW46cGFzc3dvcmQ= # admin:password coded base64

4.4 Overview and Function

In the house overview there is now the VD in the Testroom.



Each time the button is pressed, the Scene with ID 83 is called and the testvariable is incremented by 10.

5 Virtual Device Setting Global Variable

5.1 Definition of the Global Variables

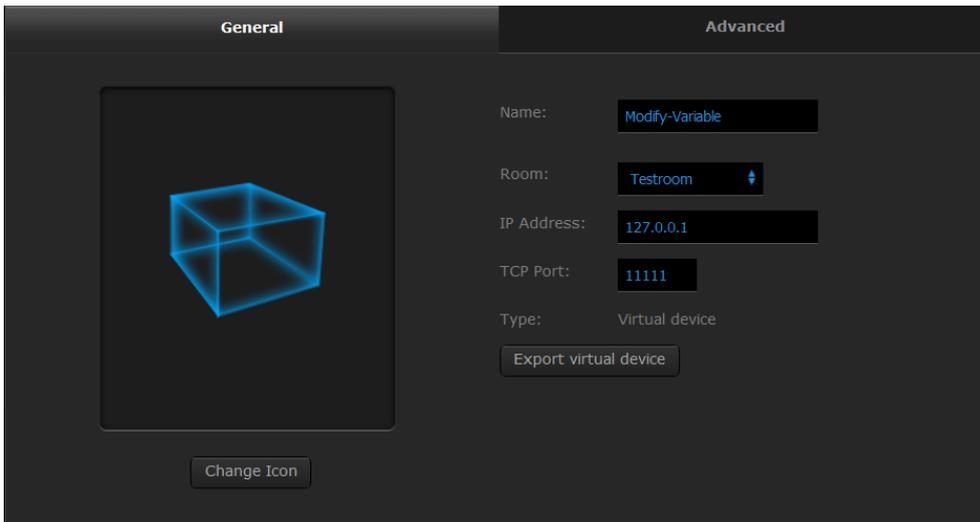
Create a global variable, which will then be modified by the Virtual Device

Variables:

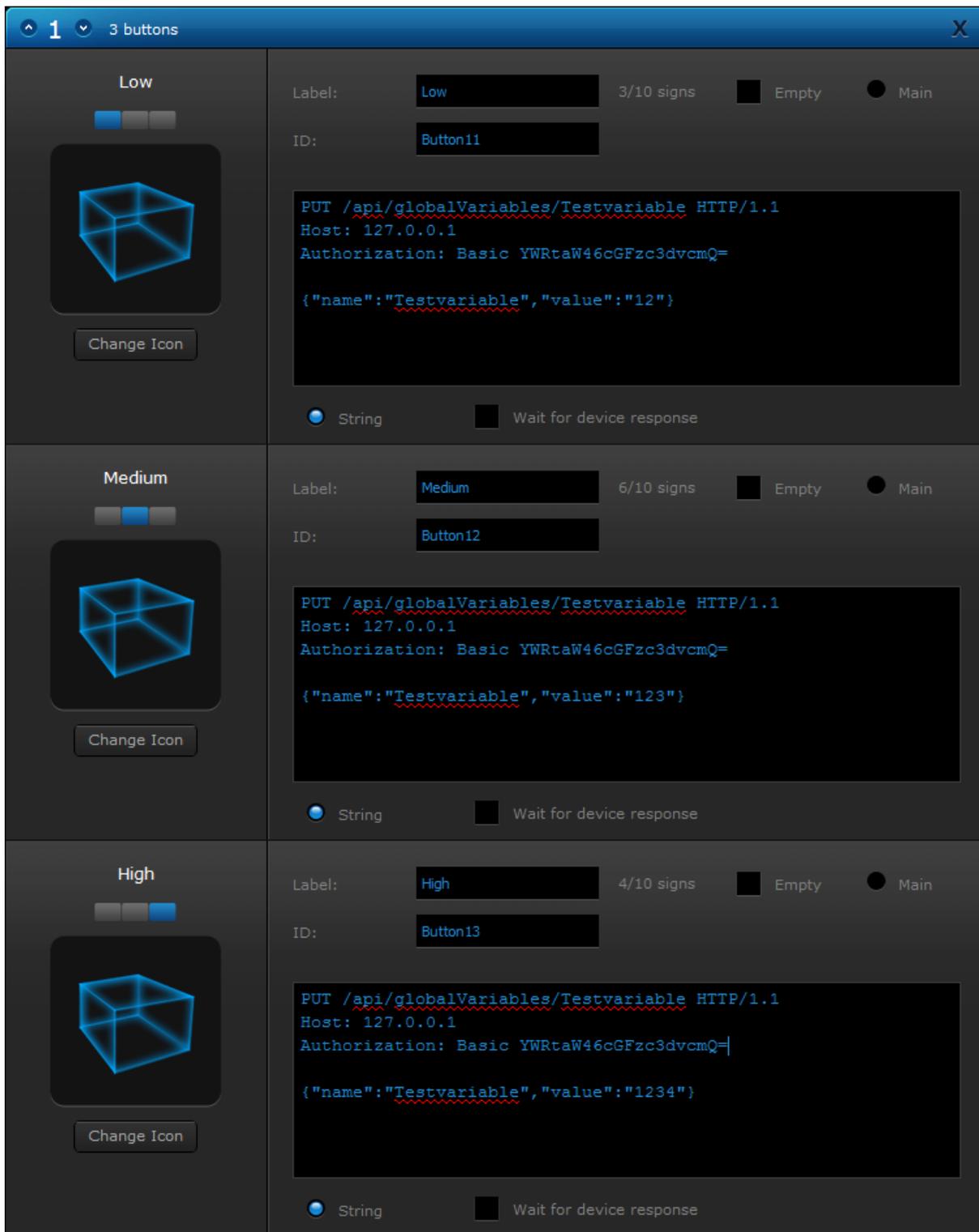
Variable	Value	Delete
Testvariable	0	

5.2 Definition of the Virtual Device

This virtual device will modify the variable.



In the advanced section, create a VD having 3 buttons:

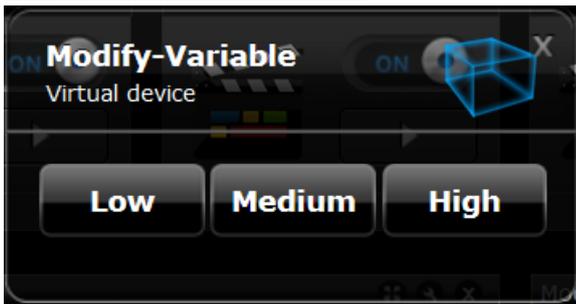


Explanation to the code-lines in the buttons:

PUT /api/globalVariables/Testvariable HTTP/1.1	# command to be executed, including the name of the variable to be modified
Host: 127.0.0.1	# the local host of the HCL
Authorization: Basic YWRtaW46cGFzc3dvcmQ=	# admin:password coded base64
	# empty line between header and body
{"name": "Testvariable", "value": "12"}	# body, setting the Testvariable to the desired value

5.3 Overview and Function

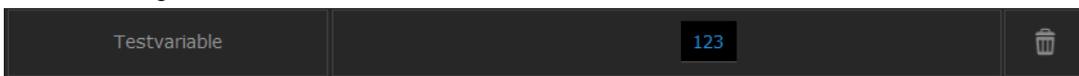
In the house overview there is now the VD in the Testroom.



Clicking the buttons Low, Medium or High leads to the values as below in the global variable
When clicking Low:



When clicking Medium:



When clicking High:



This does not make the current value of the variable visible in the VD. If this is wanted, use the scene approach as described in chapter 6. Unfortunately, the value to be shown in the Label of the VD has to be hard wired as the content of the variable cannot be used directly (or I do not know how).

6 Virtual Device Setting Global Variables using Scenes

Use this approach to test scenes, as the execution of a scene (▶) overrides all the IF conditions and directly executes the THEN. Using this approach makes sure the IF condition is properly checked. It should be easily possible to add the global variable as an AND condition at the end of the IF, so it can be removed when the scene is tested and considered good.

By clicking the Play (▶) button in the scene overview the THEN is executed without any evaluation of the IF, so the THEN can be tested, even the IF is not TRUE.

6.1 Definition of the Global Variables

This variable is only used to trigger scenes:

Variable	Value	Delete
SceneTrigger	1	

Some real variables modified by the scene, when triggered by the virtual device:

Variable	Values	Edit	Delete
VDTestVar	AUS		

For the examples used, this variable has three predefined values as shown below:

Predefined Variable

Variable may hold multiple values depending on Your needs. To add further value to variable click +

Variable Name:
VDTestVar

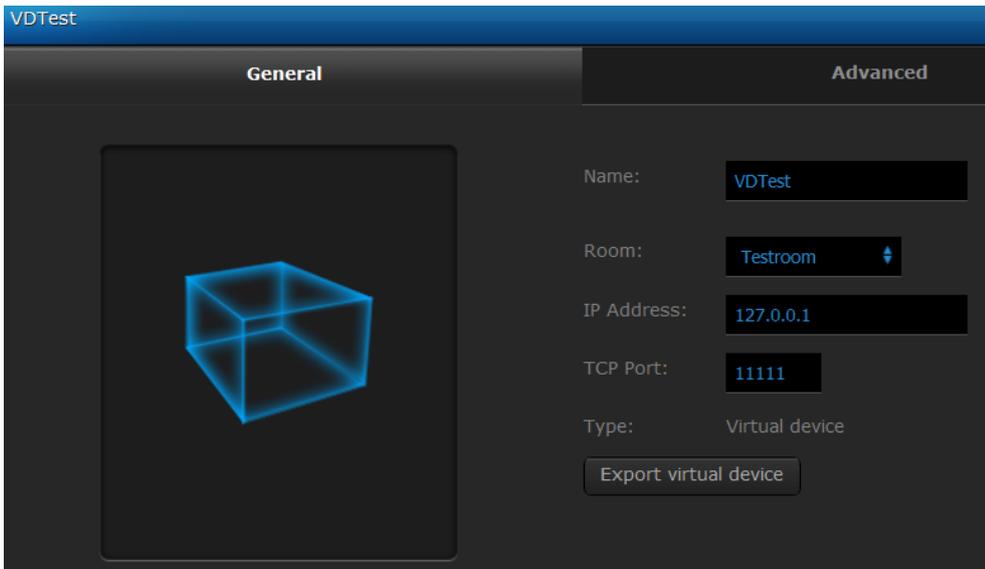
Value
AUS

Value
EIN

Value
NEU

6.2 Definition of the Virtual Device

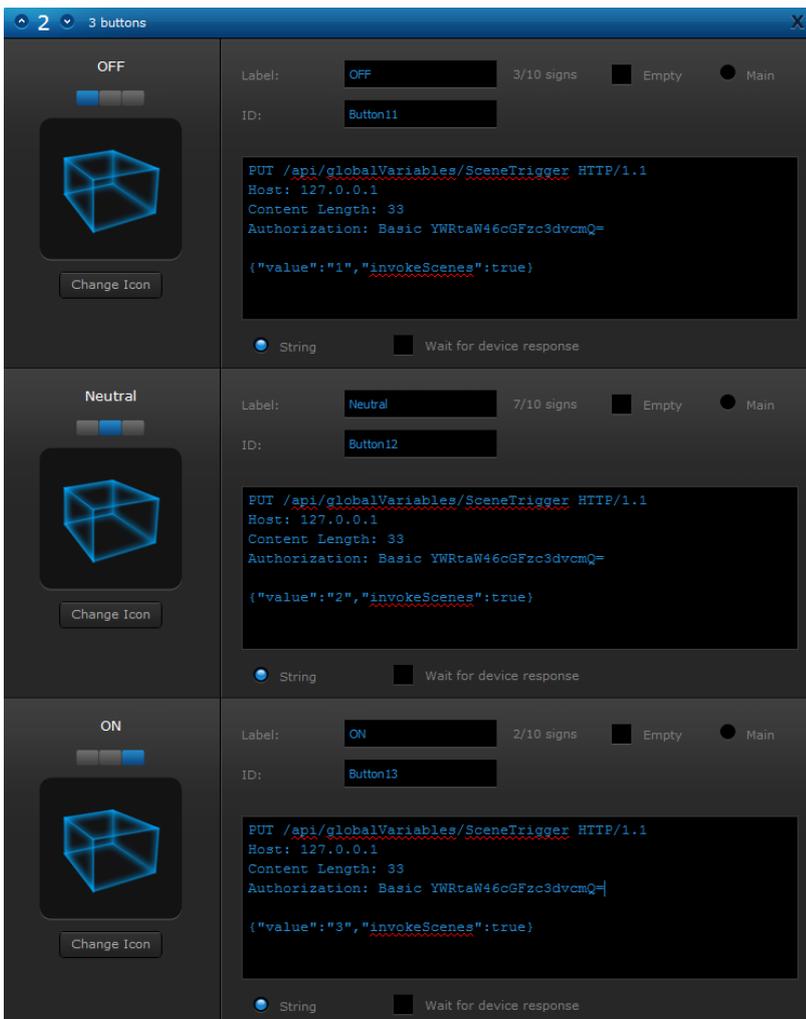
The first step is to define a virtual device, which will be used to modify the SceneTrigger and check the scenes, using this trigger in the IF condition and the THEN action is performed when the condition is fulfilled (TRUE).



In the advanced section, define a device using a label and three buttons:



Select Main and tick Favorite to have the label visible in the overview



Explanation of the code-lines in the buttons:

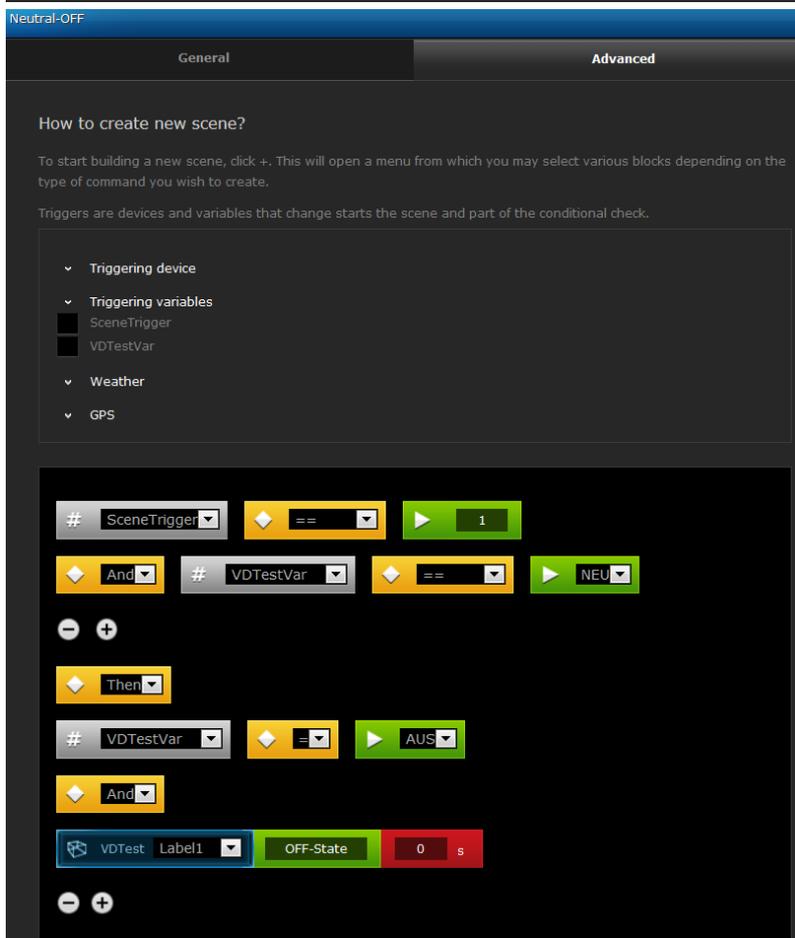
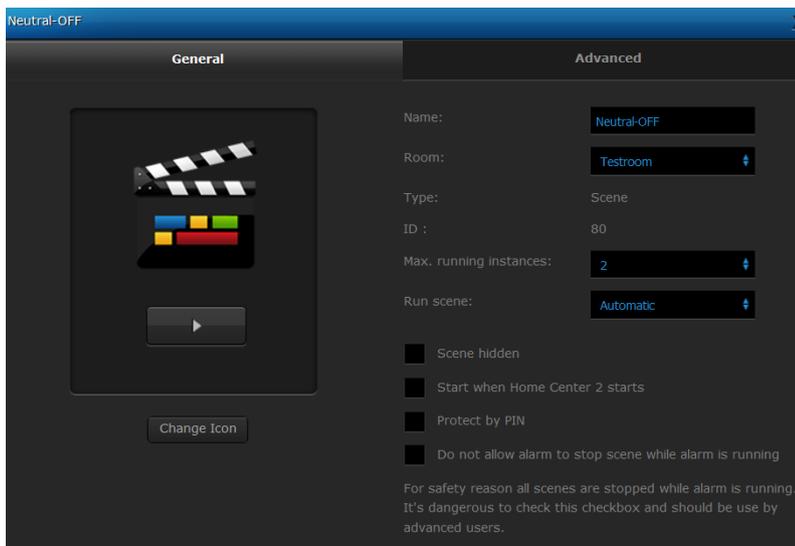
PUT /api/globalVariables/SceneTrigger HTTP/1.1	# command to be executes
Host: 127.0.0.1	# the local host for the device
Content Length: 33	# the exact length of the string below
Authorization: Basic YWRtaW46cGFzc3dvcmQ=	# admin:password coded base64
	# empty line between header and body
{"value":"1","invokeScenes":true}	# body with {}, having 33 characters as defined above
	# set the SceneTrigger to the value 1 and invokeScenes
	# to check all IF conditions of scenes using the SceneTrigger variable

6.3 Definition of the scenes

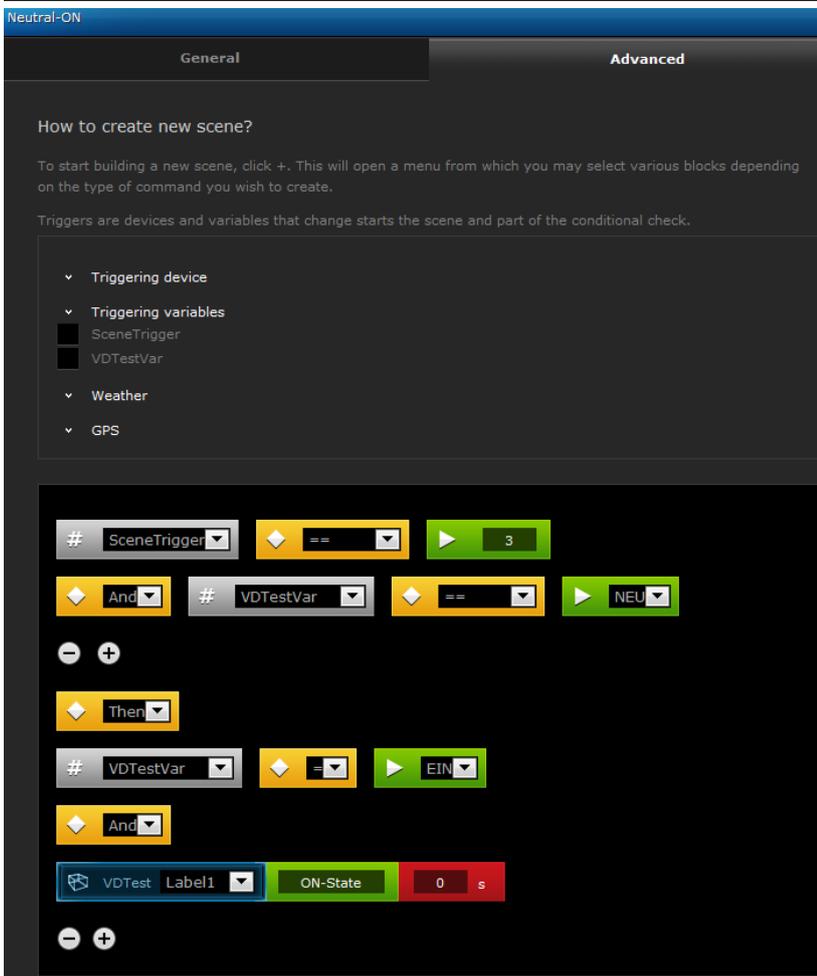
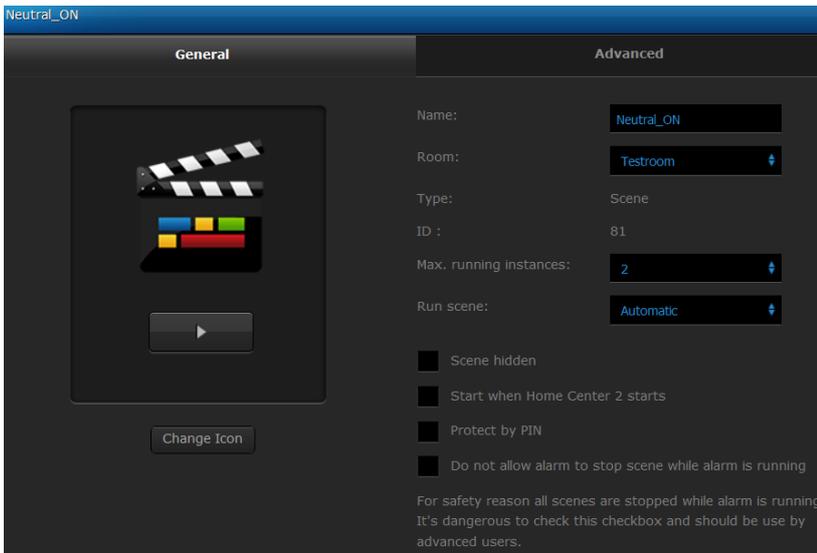
There are three scenes necessary to fulfill the task. The scenes are made in such a way that the direct change from ON to OFF and vice versa is not possible. It is only allowed to switch from Neutral to ON or OFF.

If the condition is true, the scene body modifies the global variable to the desired value and writes a human readable text to the Label of the VD. Make sure not to violate the max label length.

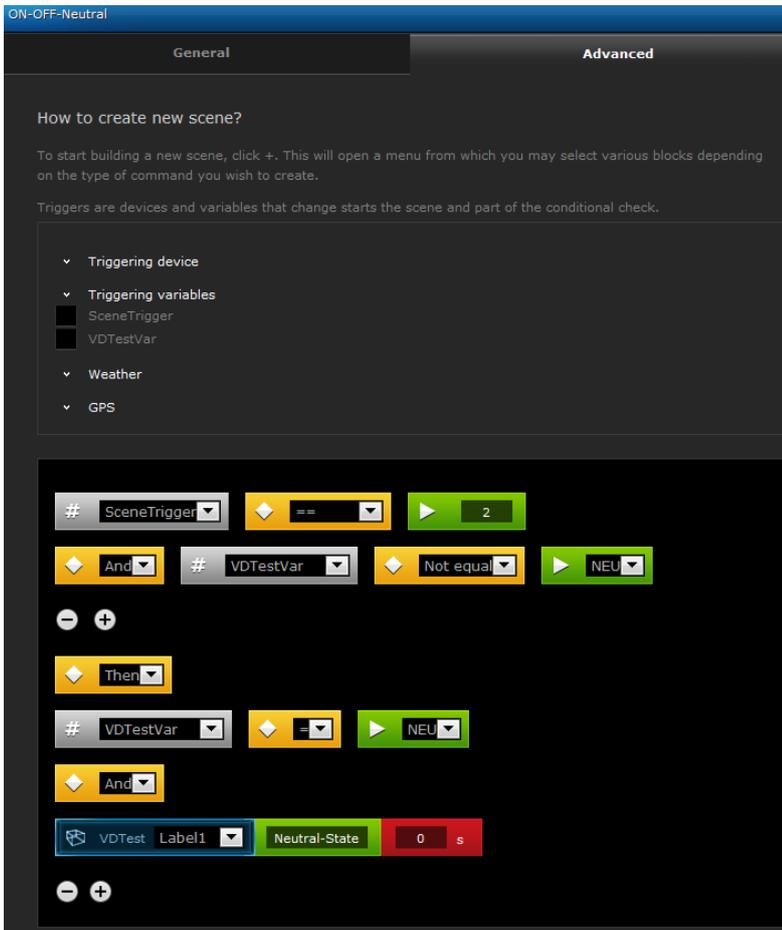
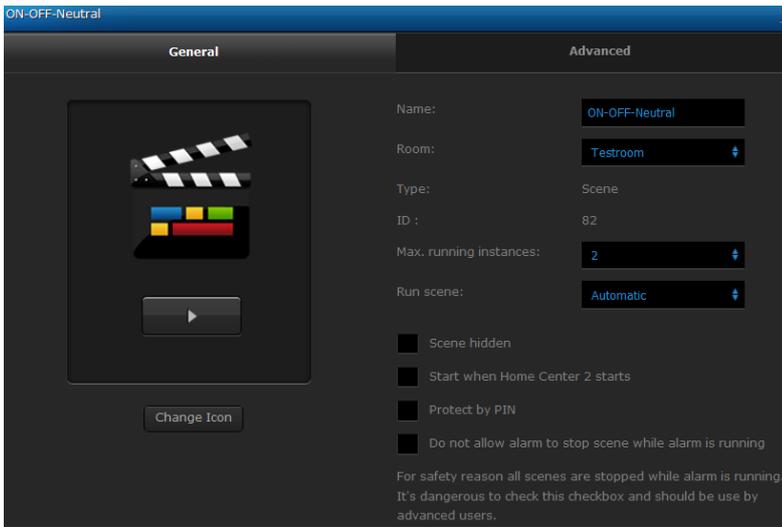
The scene switching from Neutral to OFF:



The scene switching from Neutral to ON:

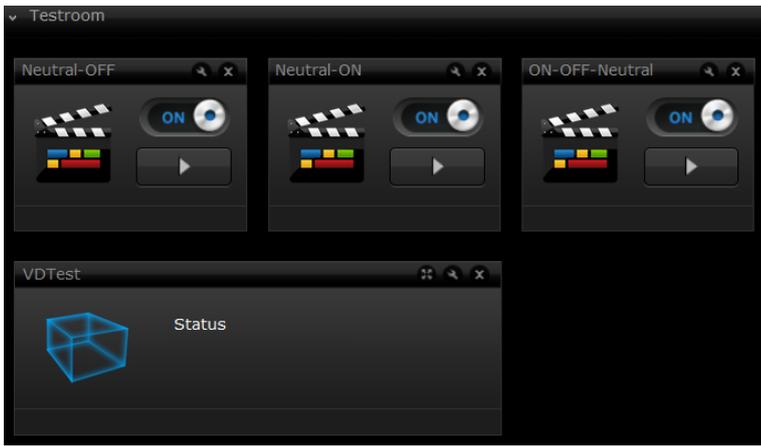


The scene switching from ON or OFF to Neutral:

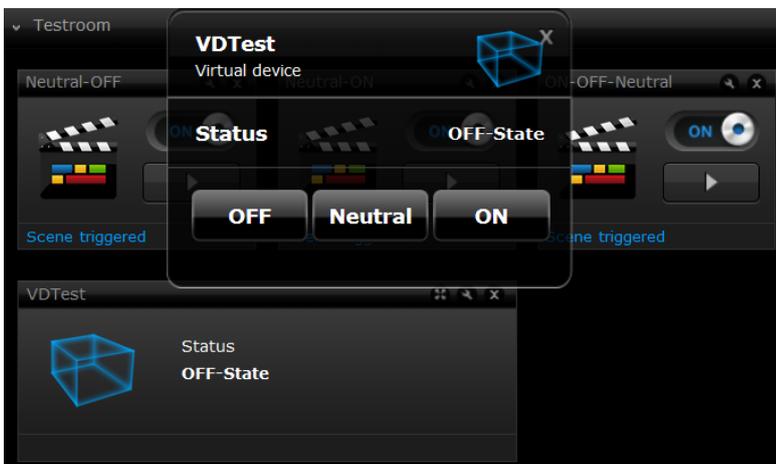


6.4 Overview and Function

In the house overview, there are now these items in the Testroom. As the label of the VD has not been set, it is still an empty field in the overview.



Open the VD and click any button to initialize the Status. In the background, the running instances of the scenes are visible.



The status of the VD is updated as defined in the scenes.

7 Virtual Device and Scenes to Globally Activate/Deactivate Scene Execution

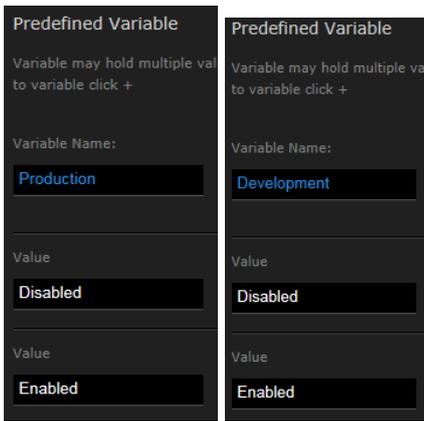
Use this approach to have global criteria to enable and disable scenes and hinder scenes still being under test to be executed, while not finished. This approach also overcomes the drawback of the Play (▶) button in the scene overview as the THEN is executed without any evaluation of the IF, which makes sense to test the THEN, even the IF is not TRUE.

7.1 Definition of the Global Variables

Two predefined variables are defined in the Variables Panel.

Production Disabled would then hinder all Scenes to run. Production Enabled would allow their normal execution based on the IF condition.

Development Disabled would hinder all Scenes being still under development from running. Development Enable would allow them to run and evaluate their IF and execute the THEN.

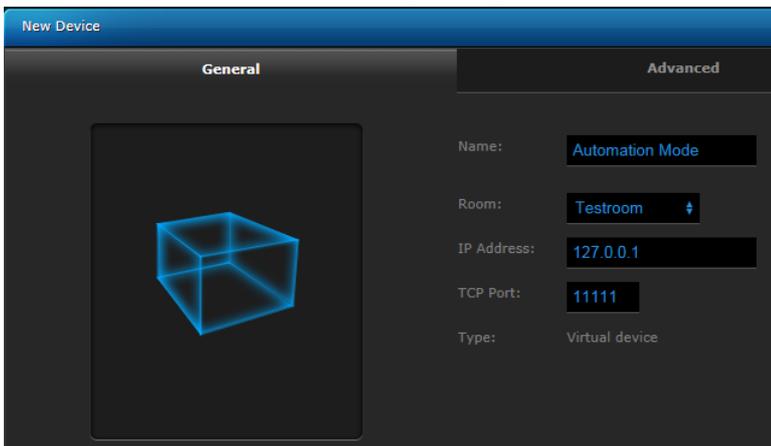


7.2 Definition of the Virtual Device

As the VDs need to know the ID of the scenes and the scenes need to know label information of the VD, the development of the two parts can not be done independently. So first a 'stub' of the VD is created and later completed by adding the Scene ID of the scenes created. The example here shows the final VD, where the ID of the scene is already correct.

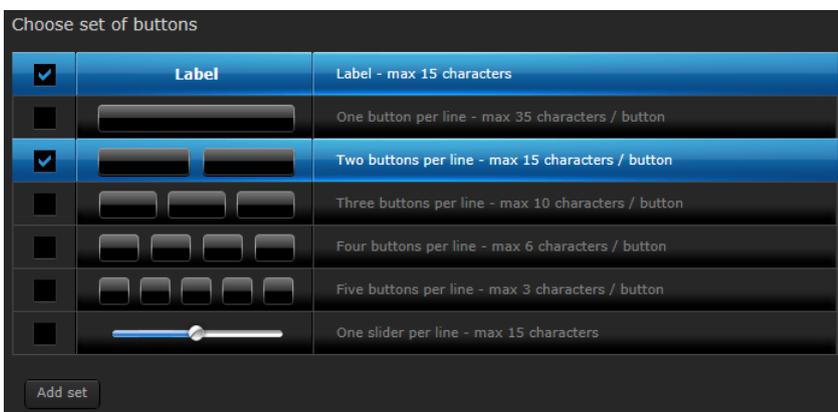
Additional explanation of the VD 'code' is not given, as this is available in previous chapters, where the basics of the interaction between VDs and Scenes are shown.

Create a new VD with the following settings in the General tab

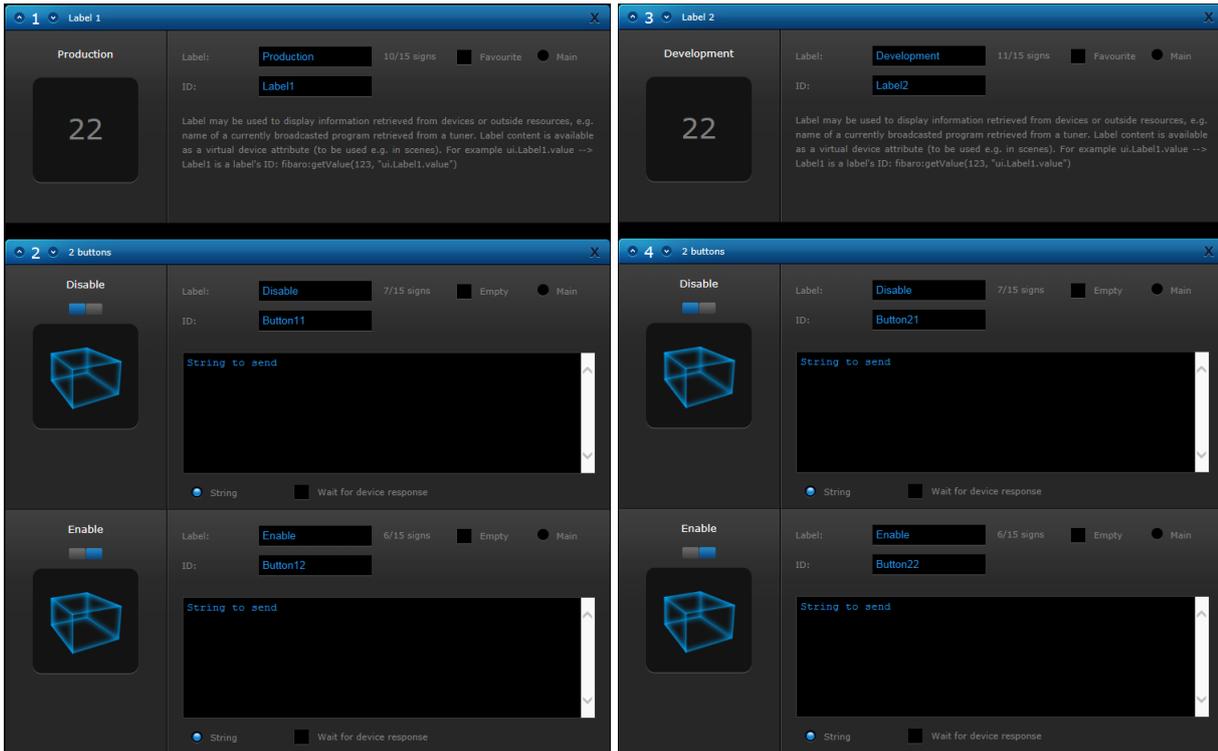


Within the Advanced tab a VD having two labels and twice two buttons is created. The label will later show the state of the global variable, while the buttons call the scenes to change the value of the global variable and the label value.

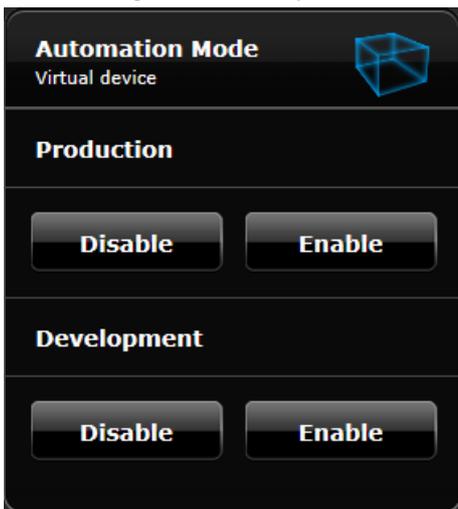
Add twice a Label and two buttons:



Within the created Label and Button panels, put some reasonable name for the label and the buttons:

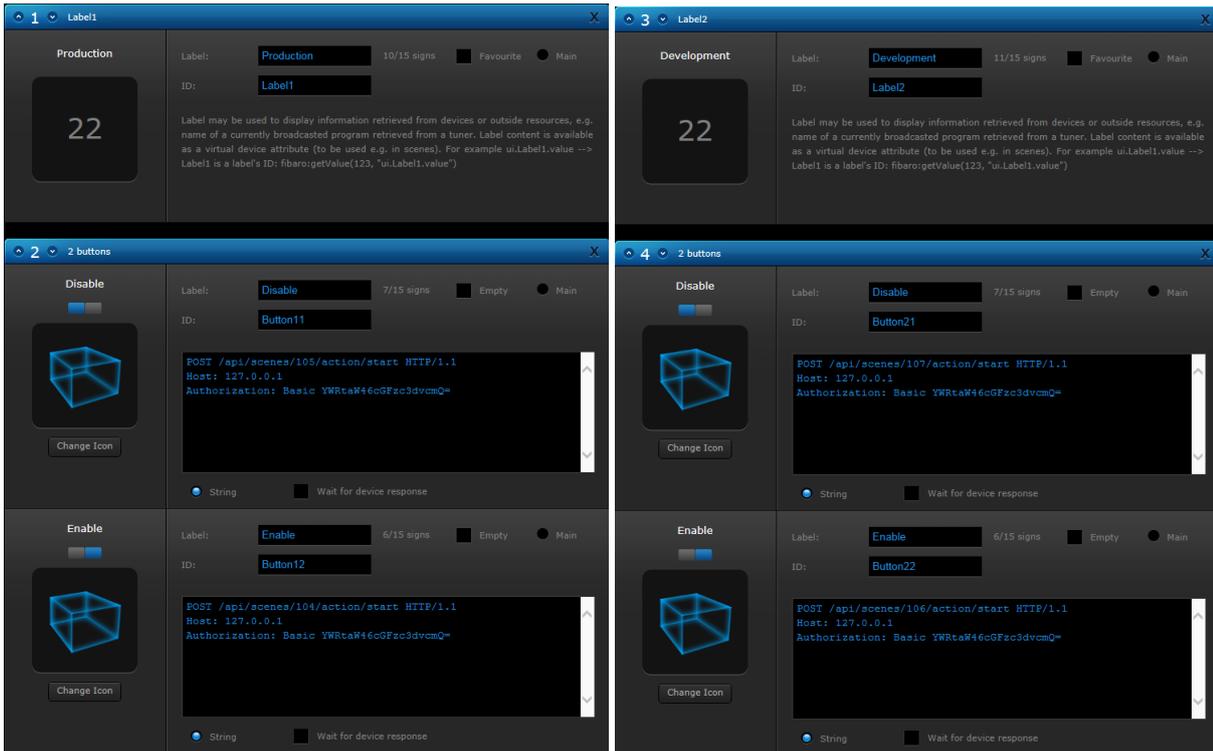


After saving, the basic layout of the VD is already available:



As now all the information from the VD is available, the scenes should be created, so the IDs of the Scenes are known to complete the code of the VD.

The VD with the necessary code to call the scenes:

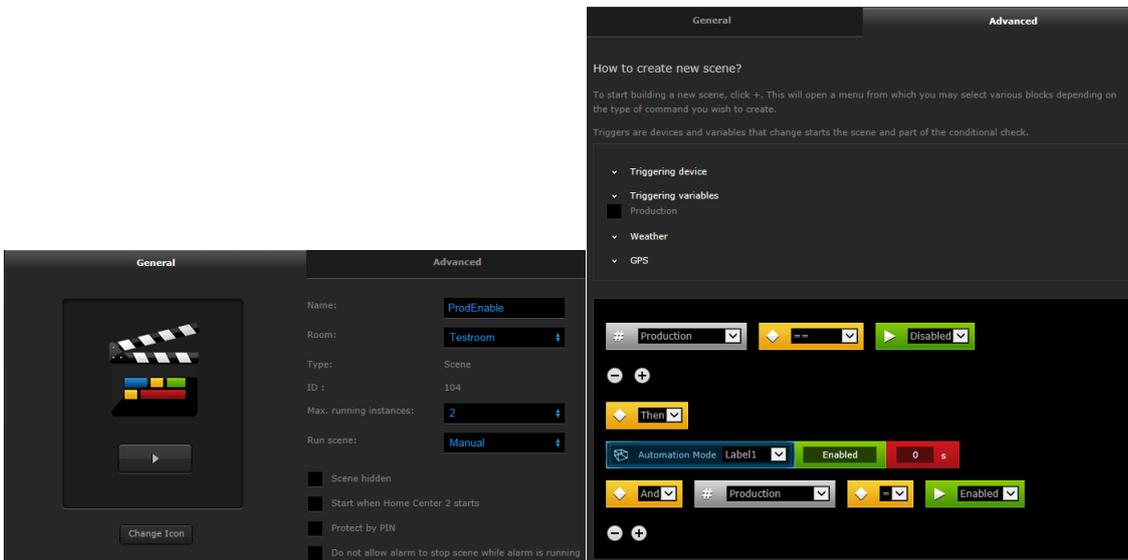


7.3 Definition of the scenes

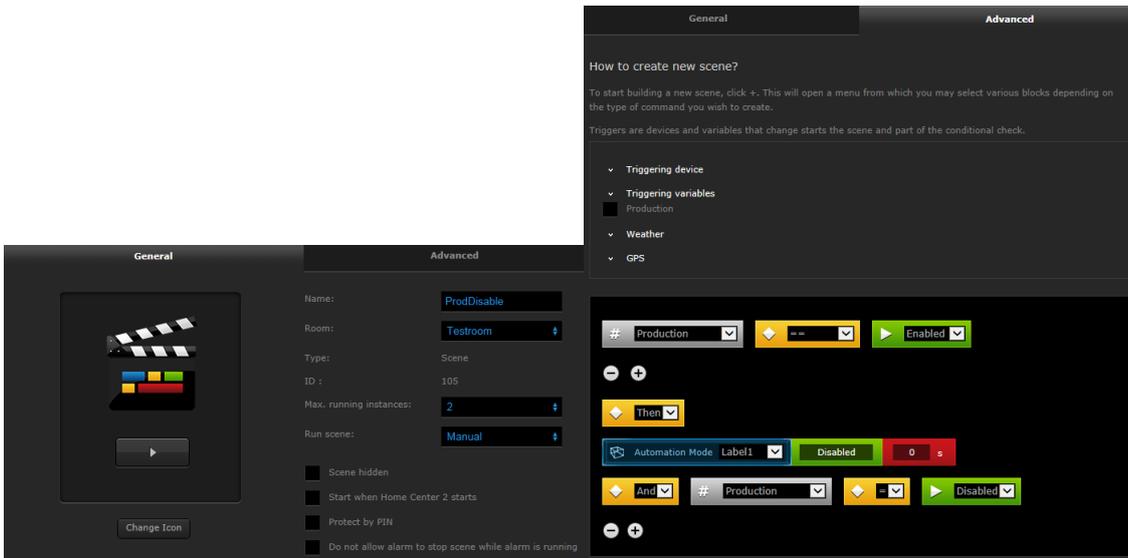
There are four scenes necessary to fulfill the task. The scenes put a human readable text to the label of the VD and change the global variable to the desired value.

Make sure all scenes are set to manual and have no triggering variable as they will be called by the VD.

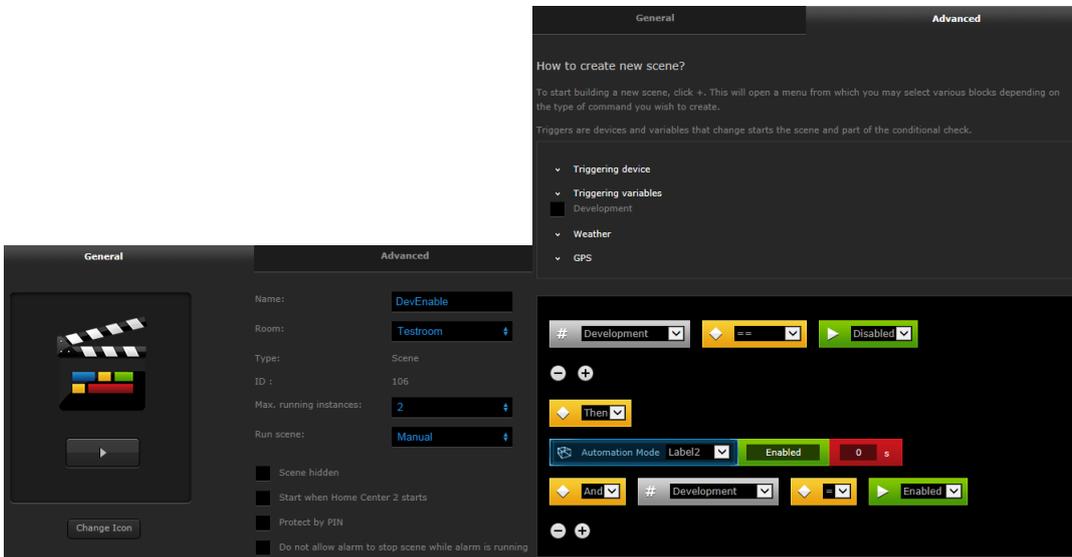
Enable the production mode:



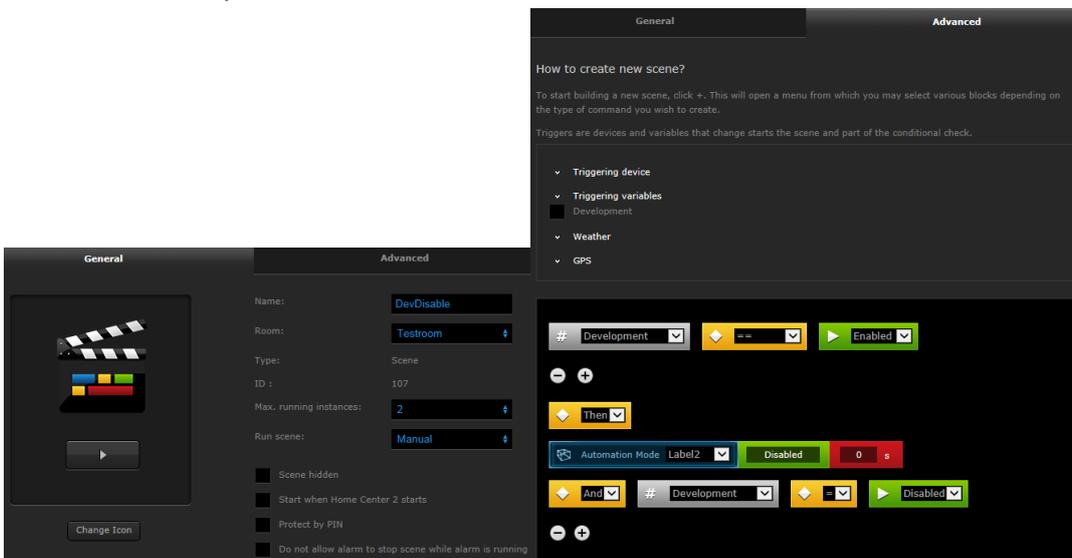
Disable the production mode:



Enable the development mode:

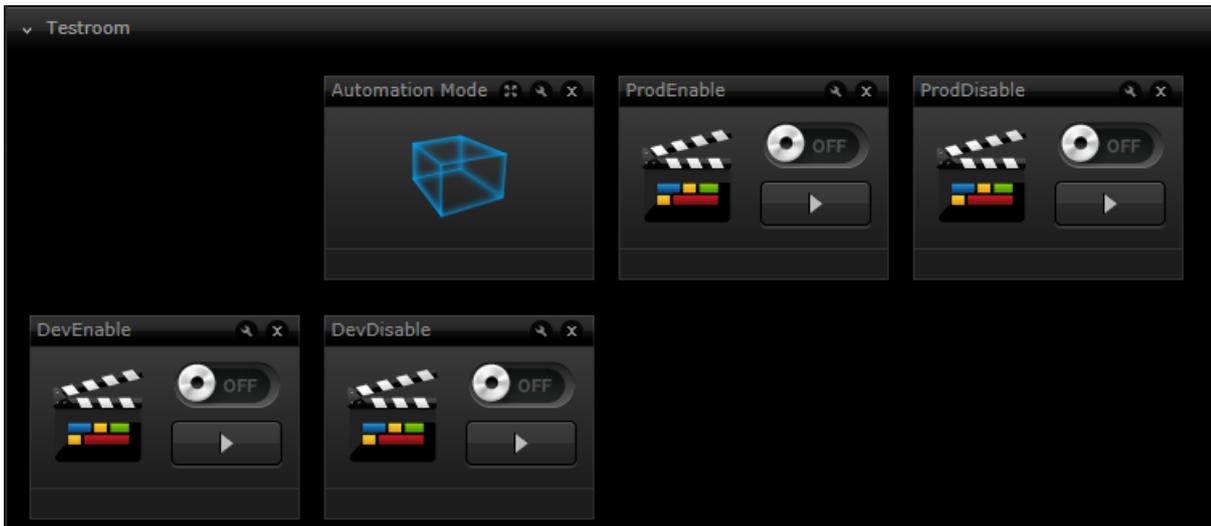


Disable the development mode:



7.4 Overview and Function

In the house overview, there is now the VD and the scenes in the Testroom. As there is none of the labels the main label the VD is just an empty block.



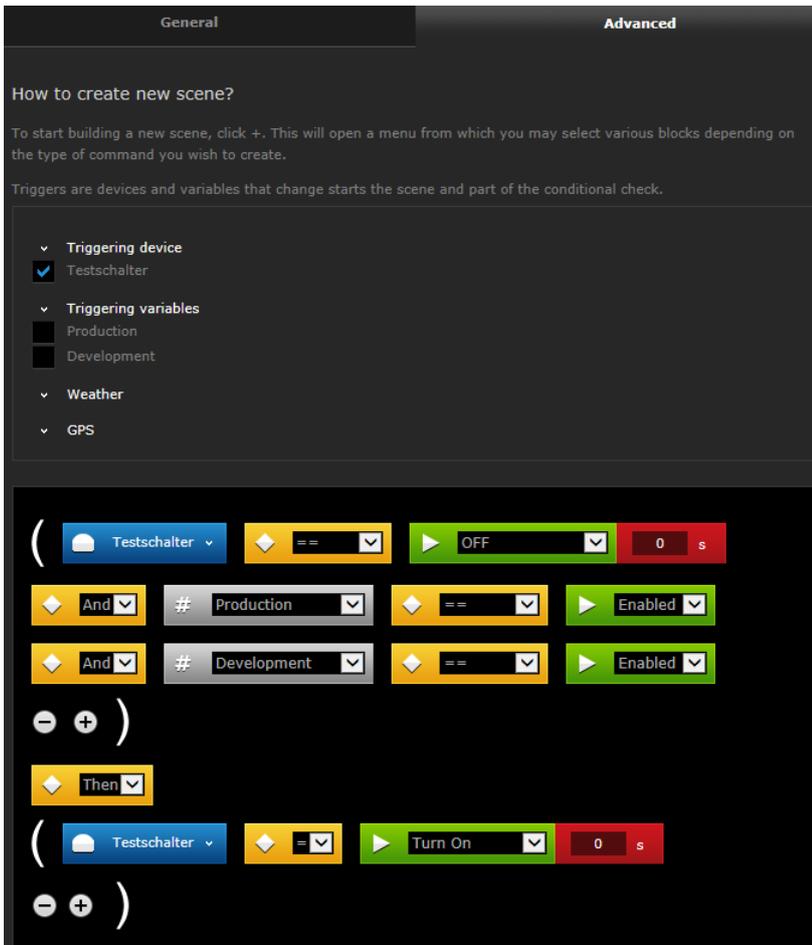
Open the VD and click any buttons to initialize the Status. In the background, the running instances of the scenes are visible.

The status of the VD is updated as defined in the scenes.



7.5 Using the Conditions to execute Scenes

Any scene can now be equipped with the check for the value of the production and while still in development the development variable like shown below.



Make sure to neither define the Production nor the Development variable status as a trigger condition. In addition the Scene trigger could be added as shown in earlier chapters to allow a test of the scene with the entire IF conditions.

Major disadvantage is the fact that individual And or Or statements in the IF cannot be removed, so it is required to remove all the conditions bottom up and add them again top down.

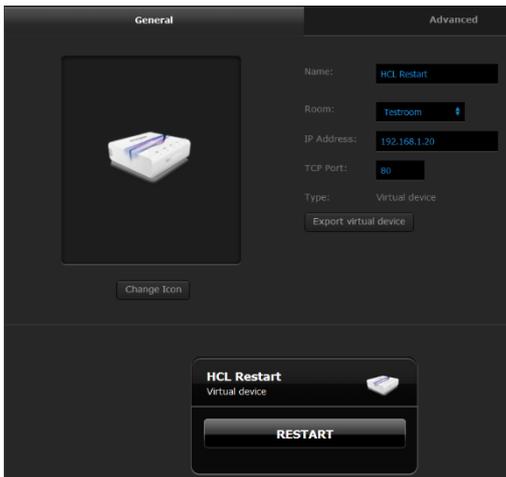
8 Restarting HCL with a VD and Scene with Bootup Scene

This VD together with the scene restarts the HCL when the IF of the Scene is TRUE.

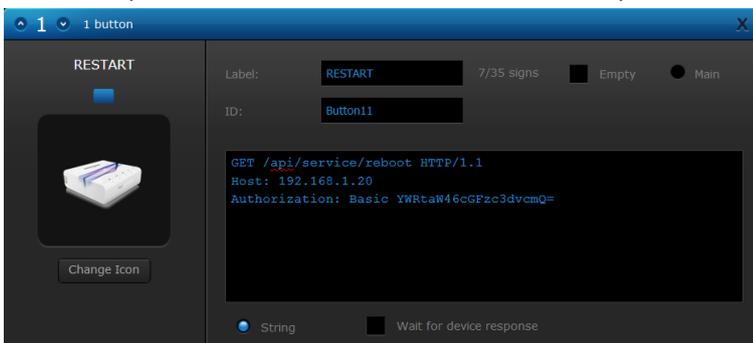
This VD and scene might ONLY work with SW Version 4.120 (This is the only SW version this has been tested on)

8.1 The VD initiating the restart

Create a VD with one button.



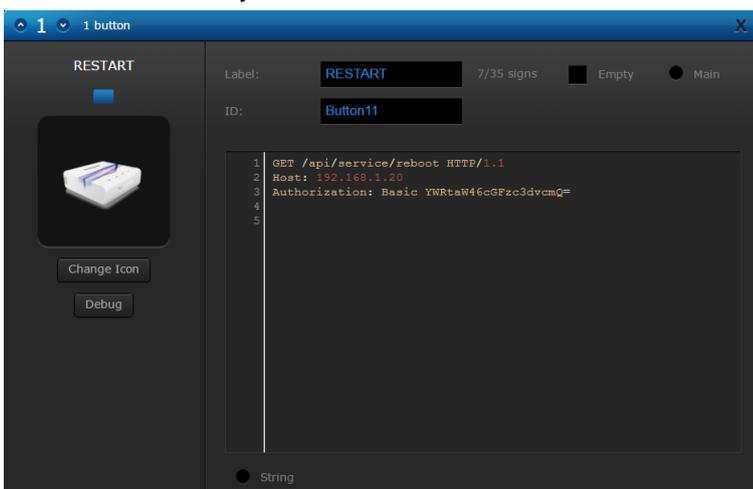
The IP address must be set to the IP address of the HCL, the local address 127.0.0.1 does not work in this case. The port is to be set to 80, also here the local port 11111 does not work.



```
GET /api/service/reboot HTTP/1.1      # command to be executes
Host: 192.168.1.20                   # the local host for the device
Authorization: Basic YWRtaW46cGFzc3dvcmQ= # admin:password coded base64
                                        # empty line
                                        # empty line
```

The VD does not work, if the two empty lines are not added to the end. More than two lines have no effect on the behavior.

Best is to double check the VD in the 'non-string' view, then it should look like this, the lines 4 and 5 should be there but without any text shown:



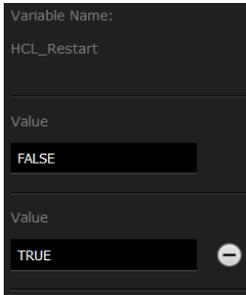
For older SW version it might be needed to exchange the GET with a POST and service with settings, so the first line would look like. This is untested:

```
GET /api/service/reboot HTTP/1.1      # command to be executes
```

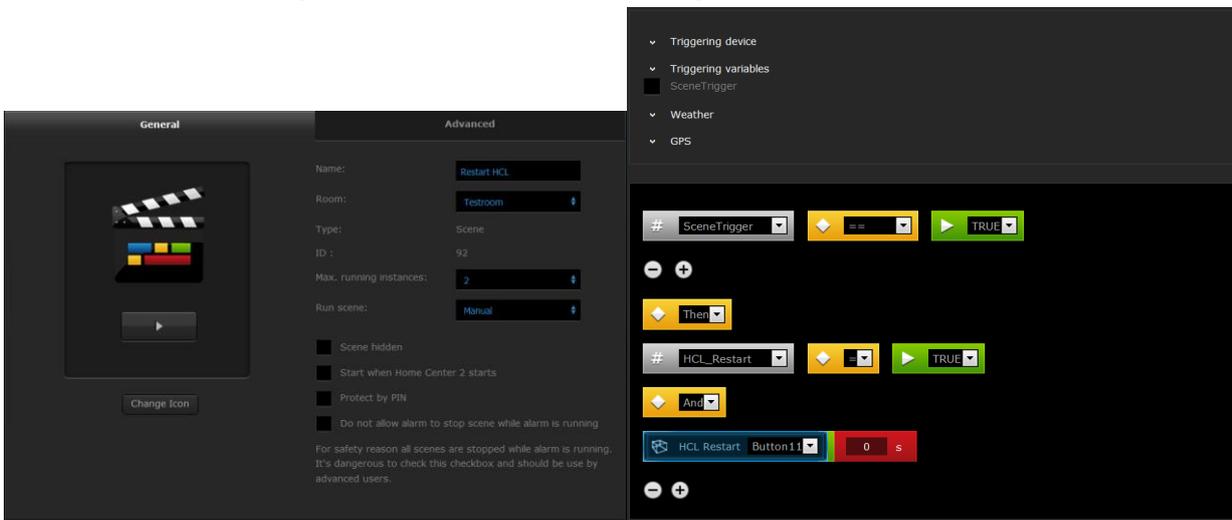

8.2 Scene to call the VD to restart

The Scene is using a global variable for the example. There might be another trigger condition, e.g. time based. Make sure to choose a trigger condition at a point in time where no scene is running, which has a delayed switch on or off or alike as these timers will not work anymore.

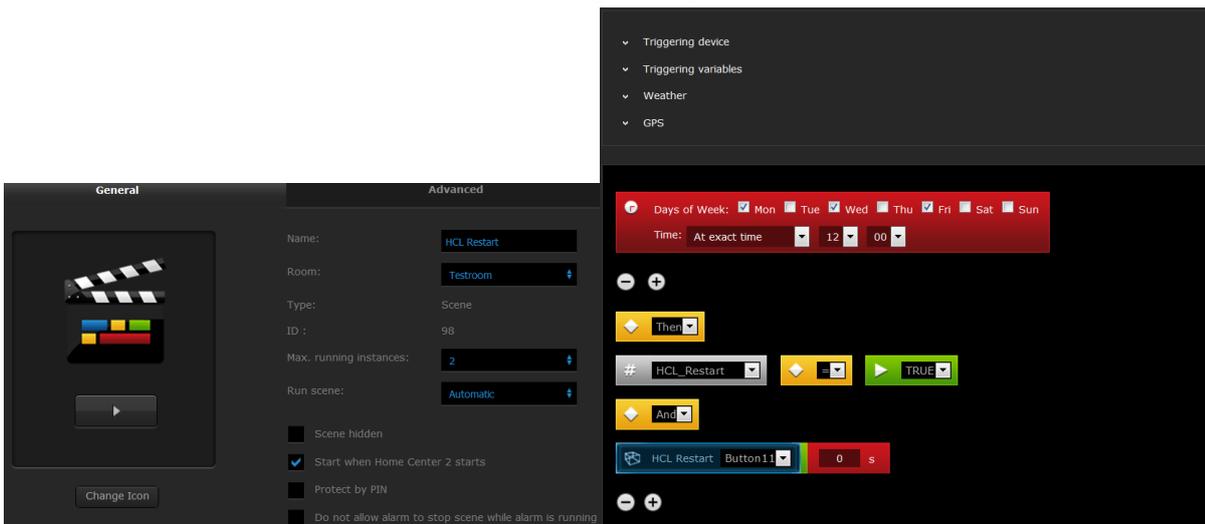
Define a global variable to save the restart:



The restart Scene, setting the variable to true and 'pressing' the button of the VD to initiate the restart.

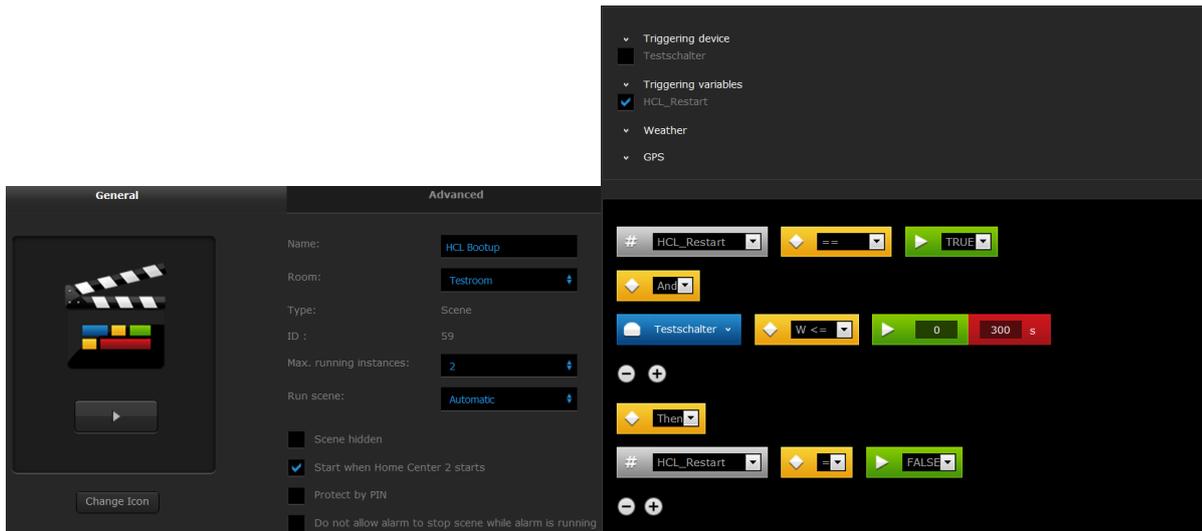


More reasonable would be a scene like this to trigger a regular restart of the HCL:



On Monday, Wednesday and Friday at noon the scene presses the button of the VD and restarts the HCL.

Create another scene, evaluating the value of the HCL_Restart variable:

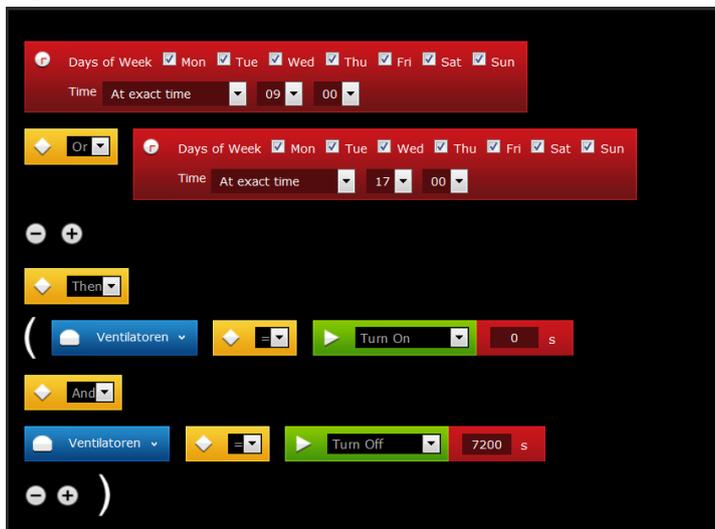


As this scene will trigger immediately when the variable is TRUE, something is needed to delay the setting of the variable to FALSE again. This delay must be longer than it takes the previous scene to press the restart button of the VD. In the example, an AEON wall switch is taken and the wattage of the switch is evaluated to be something bigger or equal to 0W after 300 Seconds. This is long enough for the restart to be initiated, before the variable is set to FLASE again. As far as tested it does even work when the AEON wall switch is not present in the network. To be on the safe side, there could be several of such silly conditions ORed.

About 5 minutes after the restart of the HCL, the variable is then set back to FLASE. Within this THEN anything can be done, necessary after a restart of the HCL.

Attention:

If there are scenes like the one shown below, the restart of the HCL must not be done between the on and Off time, as the restart will not 'return' to the scene and finish it, so the controlled switch "Ventilatoren" will remain on until the next time the scene is triggered and the "Turn Off" command is reached. So if the HCL is restarted e.g. at 10:00, the "Turn Off" which should happen at 09:00 + 7200 s (11:00) will not be performed.



9 Smart heating, using Scenes Instead of the Panel

The aim of the smart heating is to somewhat automate the heating with the aim to optimize it and maybe also save some energy. Similar approach can also be used for AC systems, partially controlling the temperatures the other way round.

The following situations are supported by the scenes and VD:

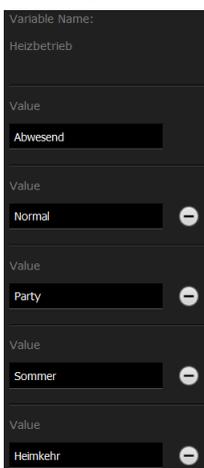
Normal: the heating should maintain a comfortable room temperature during the day and lower the temperature during the night.

- Away: the room temperature is lowered, as the house is empty; maybe just keep it from freezing
- Party: keep the temperature up to the comfortable level
- Daytime: the normal time during the day, when the comfortable room temperature should be kept
- Nighttime: the room temperature can be lowered somewhat
- Summer: the heating is off, so even the thermostats would regulate, there is no warm water flowing in the pipes. This could be triggered by a feedback from the central heating system, e.g. when the heating is off or the pump is shut down. Attention: most nowadays combined heating system shut off the heating pump, when the warm water is re-heated in a combined system.
- Returning: this could be an additional situation, forcing the heating system to intentionally overshoot. This might make sense as the thermal resistance of the air inside the house and all the furniture cannot be changed but the ΔT , so the overall time to get the house to comfortable temperature could be shortened.

9.1 Global Variables

It is the aim to use one predefined variable only having the following values:

Normal, Abwesend (Away), Party, Sommer (Summer), Heimkehr (Returning)



As some of the scenes are called explicitly and do not have an internal or automatic trigger condition an artificial scene trigger is needed as a scene does not work without a valid if condition.



The scene trigger is set to TRUE in the variable panel, but also allows easily to disable scenes when set to FALSE.

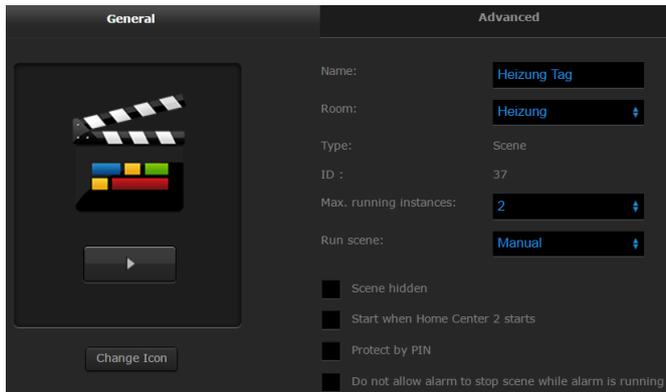
9.2 Scene definition

Some scenes do interact with the VD, so the minimal 'backbone' of the VD is created as described in the following chapter. As the thermostats do have a wakeup time, this should be considered in some of the scenes, especially when getting back to comfortable temperature, the 3 Min., 5 Min. or other predefined wakeup time could make the difference. Worst case, the content of the Scene is only reaching the thermostats after this wakeup time (delay).

9.2.1 Basic Temperature Setting Scenes

9.2.1.1 Daytime Temperature Setting

This scene is setting the comfortable temperature for all the valves

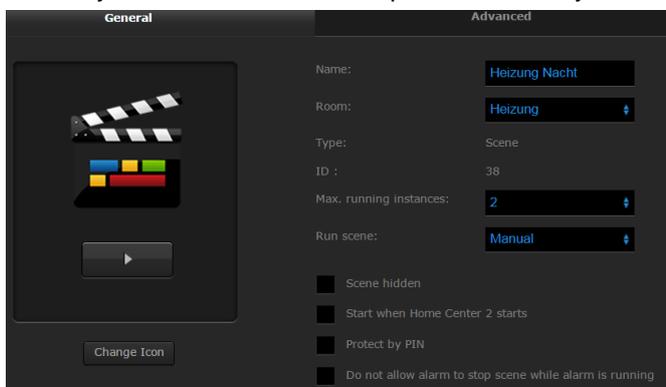


In the advanced section, the trigger condition and the temperatures for all the valves are defined.



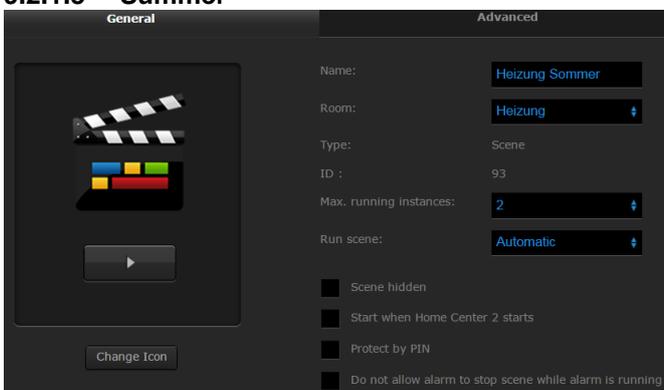
9.2.1.2 Nighttime Temperature Setting

The very same for the lower temperature, usually used during the night



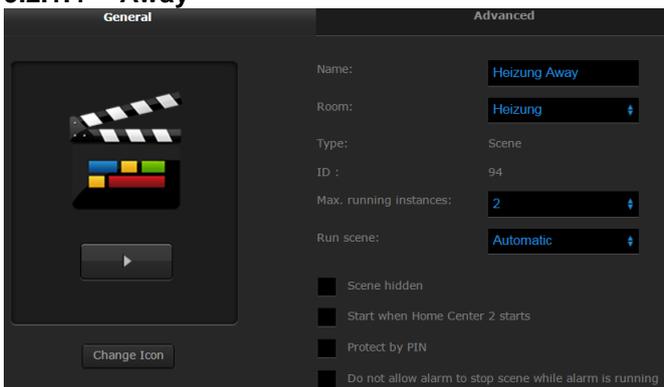


9.2.1.3 Summer



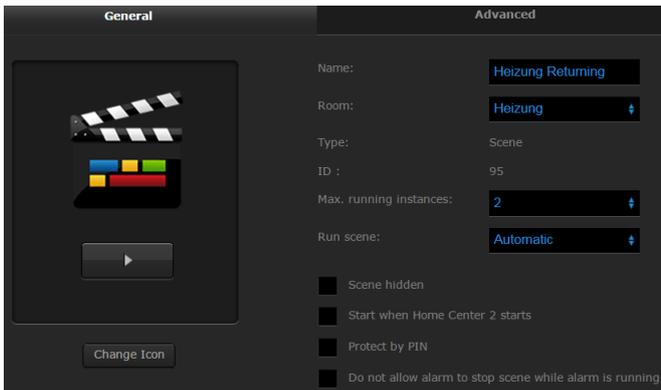
In the advanced section, it looks identical to the normal day or night temperature settings. My valves support a minimal temperature of 4°C, so the “Set target level” is set to 4.

9.2.1.4 Away



Here the temperature is set to 15°C. However, this could be same as the night scene for the room temperature.

9.2.1.5 Returning



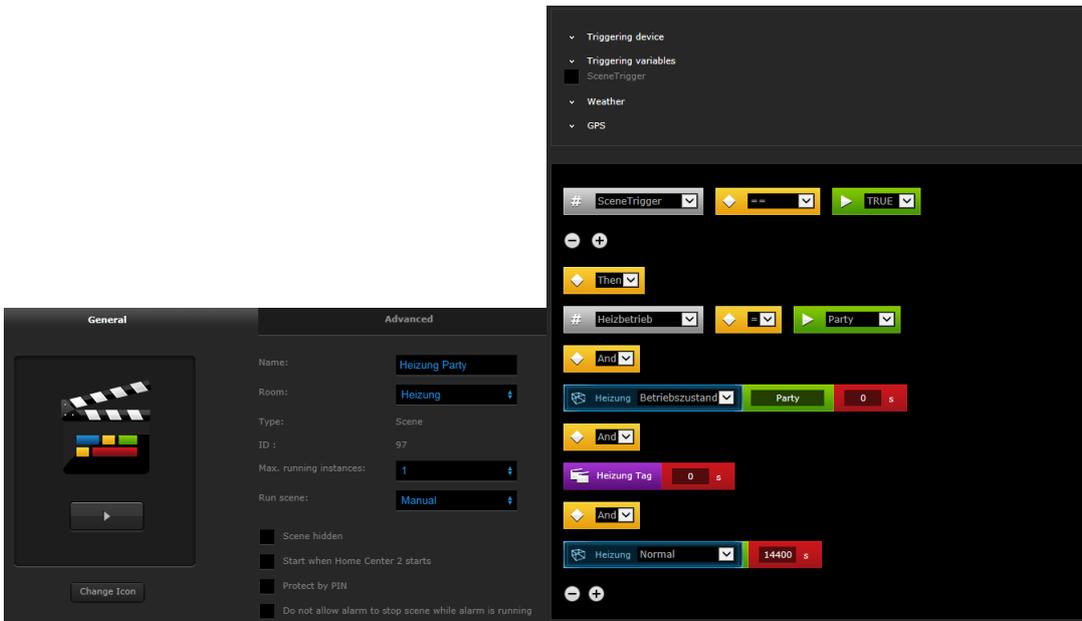
The temperature in the advanced section is set to 24°C for all the valves and at the very end of the scene after 4h, the button in the VD to return to normal operation mode is 'pressed'. The scene is also checking if the heating is in the away (Abwesend) mode.

For testing purposes the time is set to e.g. 10 Minutes to allow the valves to adjust to the returning temperature (24°C) and then to the normal operation mode again. Check the wakeup time of the valves and set the time for testing of e.g. 3 times the wakeup time to be sure the settings can be transferred to the valves.



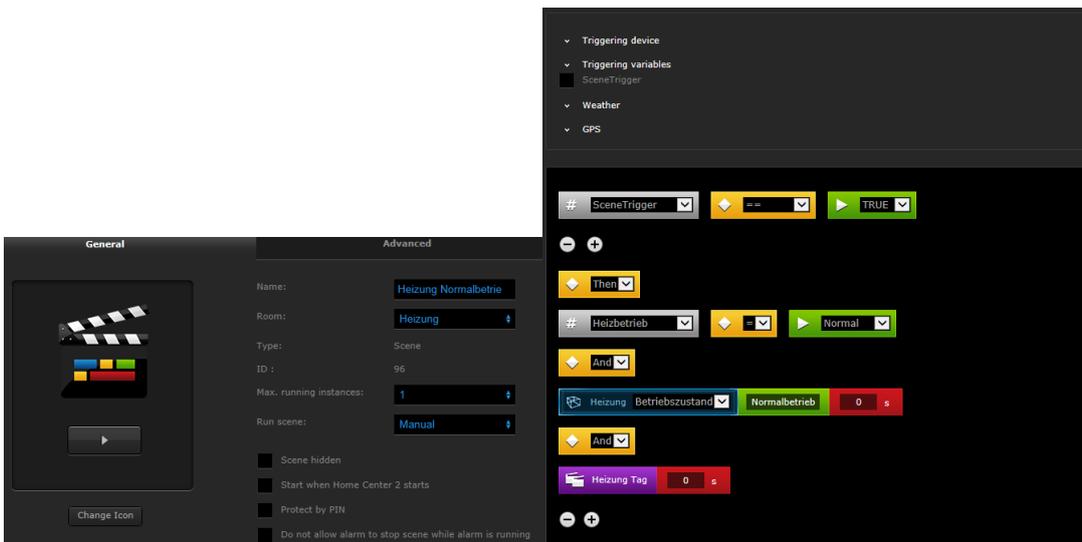
9.2.1.6 Party override

The scene for party to override especially the lowering of the temperature during the night:



9.2.1.7 Normal Operation

The scene to force the system into normal operation mode:



The scene could also be split up to allow returning to normal operation mode fir daytime and nighttime. In this example the scene will always put the system to the day mode (Heizung Tag). An IF-THEN-ELSE would be perfect to sort day and night time heating in one scene. An additional variable like daytime (TRUE/FALSE) can be used to have two scenes, sorting it out.

IF (SceneTrigger == TRUE)

THEN

 Heizbetrieb = Normal

 AND

 Betriebszustand = Normalbetrie

 AND

 CallScene(Heizung Tag)

 CallScene(Check Day or Night) #this scene would check if daytime is FALSE and call the Scene Heizung Nacht

9.2.2 Scenes for Normal Operation

The normal heating control is split up in four scenes controlling the day and night temperatures during the week and the weekends. These scenes are timer triggered and call the scenes where the valve temperature is set to a specific value.

9.2.2.1 Day Temperature from Monday to Friday

The screenshot shows the configuration for the 'Mo bis Fr Tagestemp' scene. The 'General' tab is active, displaying a clapperboard icon and a 'Change Icon' button. The 'Advanced' tab shows the following settings:

- Name: Mo bis Fr Tagestemp
- Room: Heizung
- Type: Scene
- ID: 25
- Max. running instances: 2
- Run scene: Automatic
- Scene hidden:
- Start when Home Center 2 starts:
- Protect by PIN:
- Do not allow alarm to stop scene while alarm is running:

The 'Triggering device' section is empty. The 'Triggering variables' section shows 'Heizbetrieb' with a checked checkbox. The 'Weather' and 'GPS' sections are empty. The 'Days of Week' section shows 'Mon', 'Tue', 'Wed', 'Thu', and 'Fri' selected. The 'Time' is set to 'At exact time' with '06' and '38' selected. The 'And' button is active, and the 'Then' button is active. The 'Normal' button is active. The 'Heizung Tag' scene is called with a duration of '0 s'.

9.2.2.2 Day Temperature on Saturday and Sunday

The screenshot shows the configuration for the 'Sa So Tagestemp' scene. The 'General' tab is active, displaying a clapperboard icon and a 'Change Icon' button. The 'Advanced' tab shows the following settings:

- Name: Sa So Tagestemp
- Room: Heizung
- Type: Scene
- ID: 36
- Max. running instances: 2
- Run scene: Automatic
- Scene hidden:
- Start when Home Center 2 starts:
- Protect by PIN:
- Do not allow alarm to stop scene while alarm is running:

The 'Triggering device' section is empty. The 'Triggering variables' section shows 'Heizbetrieb' with a checked checkbox. The 'Weather' and 'GPS' sections are empty. The 'Days of Week' section shows 'Sat' and 'Sun' selected. The 'Time' is set to 'At exact time' with '08' and '00' selected. The 'And' button is active, and the 'Then' button is active. The 'Normal' button is active. The 'Heizung Tag' scene is called with a duration of '0 s'.

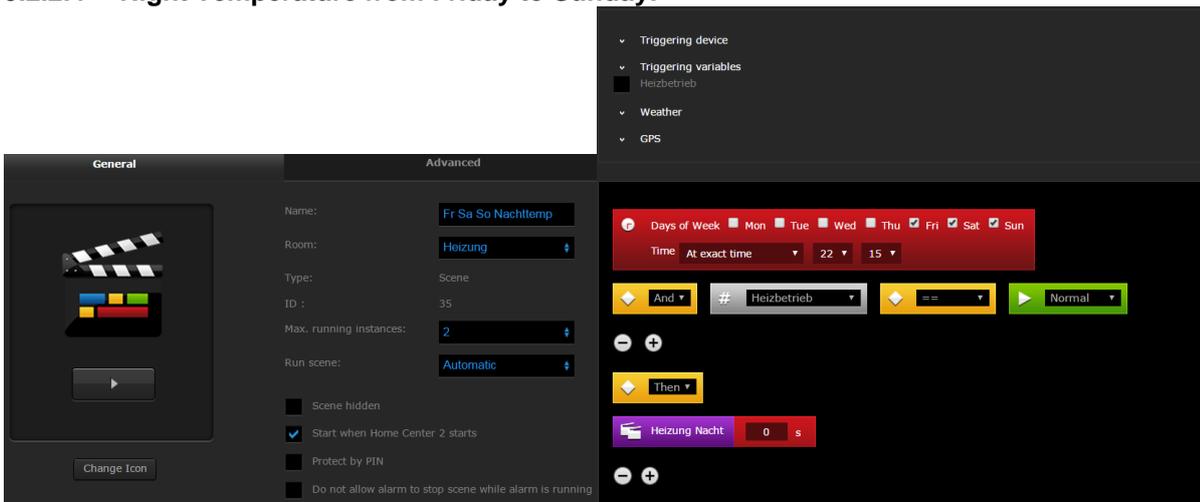
9.2.2.3 Night Temperature from Monday to Thursday:

The screenshot shows the configuration for the 'Mo bis Do Nachttemp' scene. The 'General' tab is active, displaying a clapperboard icon and a 'Change Icon' button. The 'Advanced' tab shows the following settings:

- Name: Mo bis Do Nachttemp
- Room: Heizung
- Type: Scene
- ID: 34
- Max. running instances: 2
- Run scene: Automatic
- Scene hidden:
- Start when Home Center 2 starts:
- Protect by PIN:
- Do not allow alarm to stop scene while alarm is running:

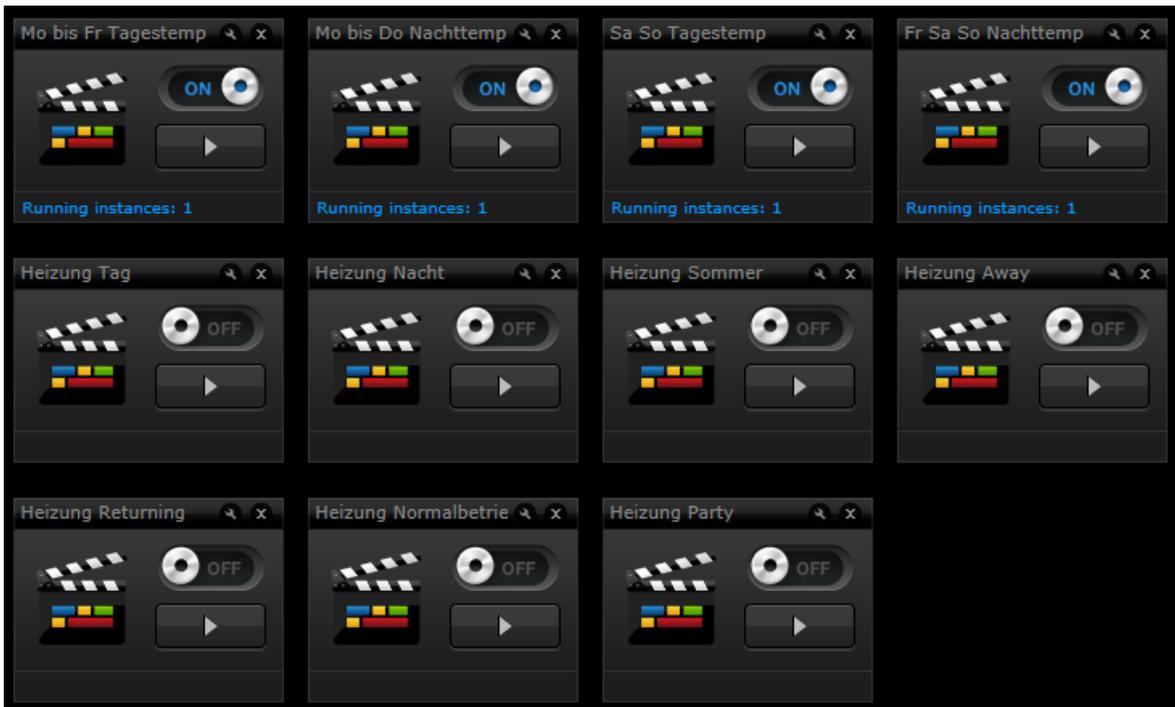
The 'Triggering device' section is empty. The 'Triggering variables' section shows 'Heizbetrieb' with a checked checkbox. The 'Weather' and 'GPS' sections are empty. The 'Days of Week' section shows 'Mon', 'Tue', 'Wed', and 'Thu' selected. The 'Time' is set to 'At exact time' with '21' and '15' selected. The 'And' button is active, and the 'Then' button is active. The 'Normal' button is active. The 'Heizung Nacht' scene is called with a duration of '0 s'.

9.2.2.4 Night Temperature from Friday to Sunday:



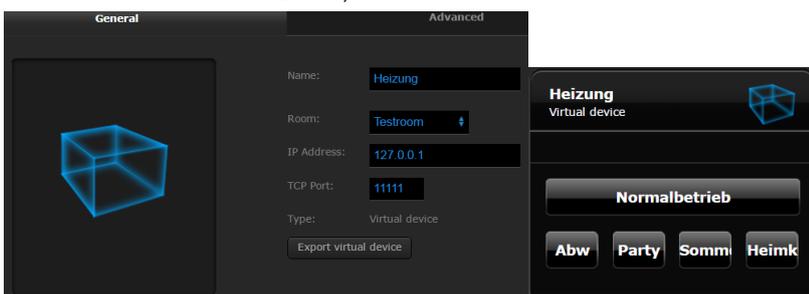
9.2.3 All the Heating Scenes

These are all the scenes defined for the heating system. All the scenes setting a temperature can be set to OFF, as they are called explicitly.



9.3 VD

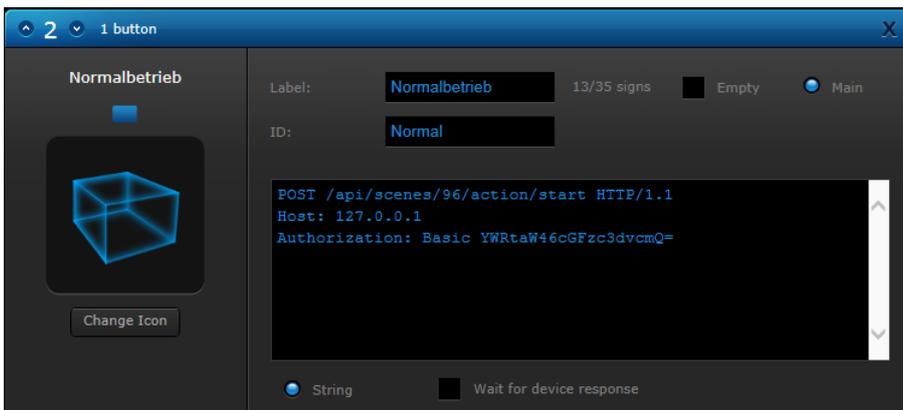
Create a device with a Label, one button and four buttons



The label is only used to show the status of the heating system, so the label does not need a name, the text shown there should be self-explaining.

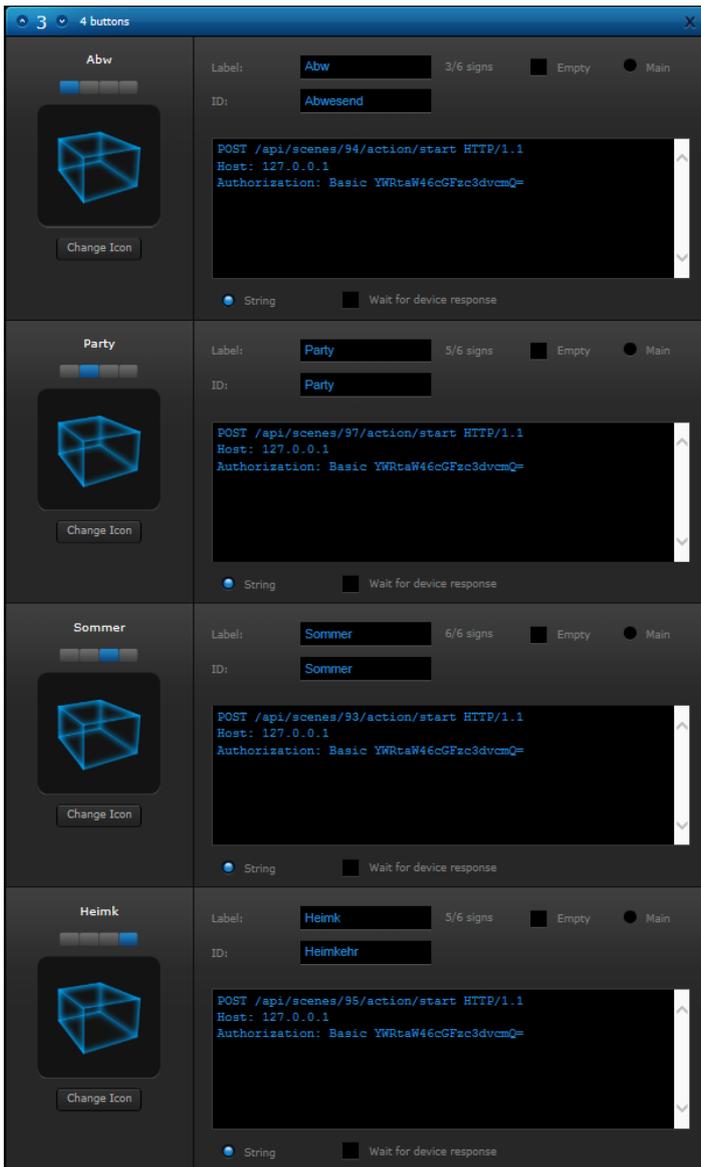


The button for Normalbetrieb is also set as the main button, so this will force the heating system into normal operation mode.



This button is calling the scene with the ID 96, which is the normal operation mode of the heating system

The buttons for the different special modes of the heating system



These buttons are calling the scenes with the appropriate ID

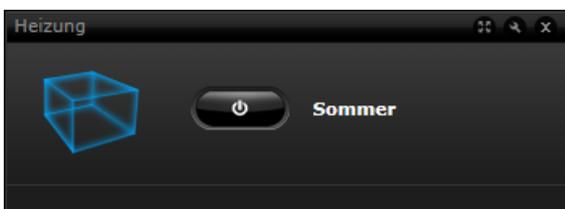
Scene ID 94: Abwesend (being away)

Scene ID 97: Party

Scene ID 93: Sommer (Summer)

Scene ID 95: Heimkehr (Returning)

When all is set up and hopefully not Scene ID mismatch, the VD looks like this when the variable is updated for the first time



Opening the VD gives this view with not so nice text on the buttons:

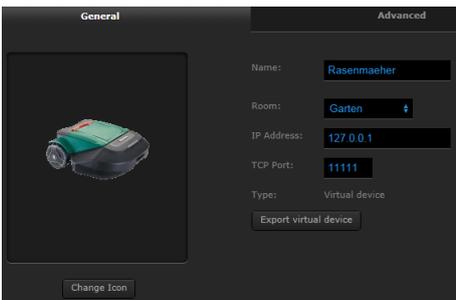


10 Robomow® RM615 Telling Operational State

Unfortunately Robomow® only has a Bluetooth interface and not web interface nor a Z-Wave interface. To get at least some info from the mower, the following approach can be used.

The VD Scenes are based on the power measurement of a AEON Labs Smart Switch Gen5. With other devices it might look different

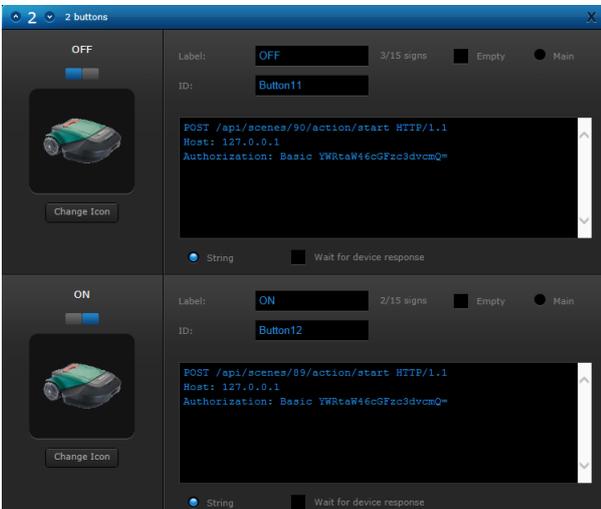
10.1 The VD representing the mower



With one label:



And two buttons:



Attention: the buttons do not make really sense, as the mower will drive out of the charging station if the power is switched off. If the power off remains for longer than 1 or 2 hours, the mower will not return to the charging station. So for general purpose the label would be good enough to provide the feedback from the mower.

10.2 Global Variable

Even it would also work without a global variable if only the state of the mower should be shown in the VD the mower state is stored in a global variable, so it can be used as a condition as well.

The image shows a configuration interface for a global variable named 'MaeherZustand'. It consists of several rows, each representing a possible state of the mower. Each row has a 'Value' label, a text input field, and a minus sign button. The states shown are: Balancing, Charging, Mowing, OFF, ON, and Waiting.

Value	Button
Balancing	-
Charging	-
Mowing	-
OFF	-
ON	-
Waiting	-

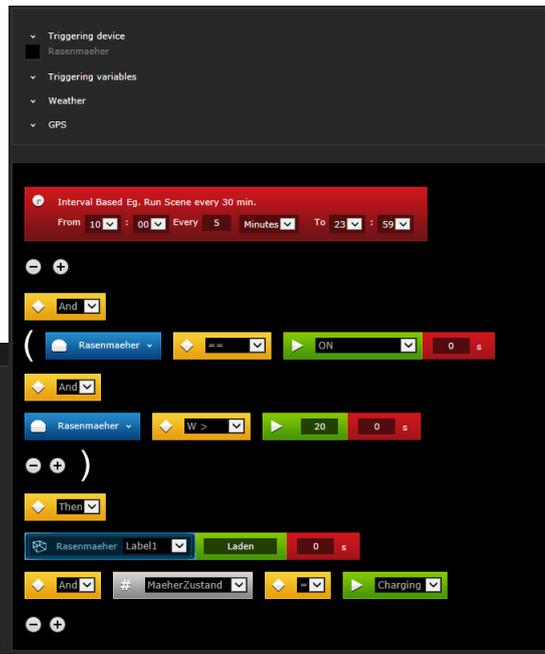
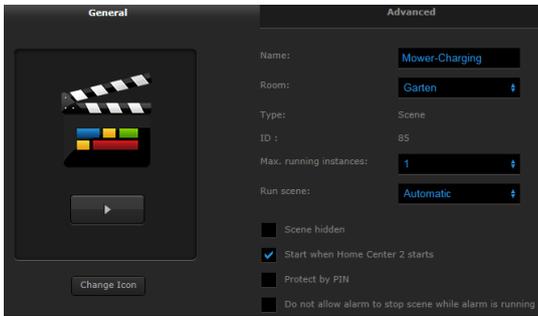
10.3 Scenes to evaluate the mower state

The scenes evaluate the power consumption of the mower and set the global variable based on this and the label of the VD.

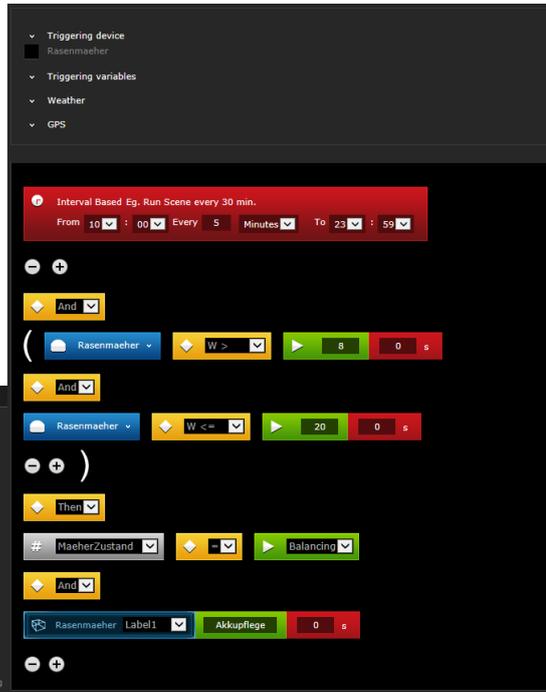
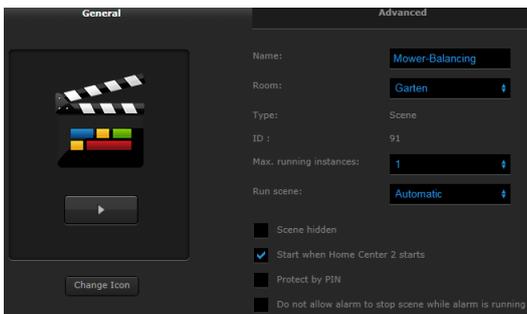
Took some time to find out the correct levels to be set in the scenes so there is no false information shown. Could also be much simpler with e.g. mowing, charging and waiting only.

Charging = Laden, Mowing = Mähen, Balancing = Akkupflege, Waiting = Warten, ON = EIN, OFF = AUS

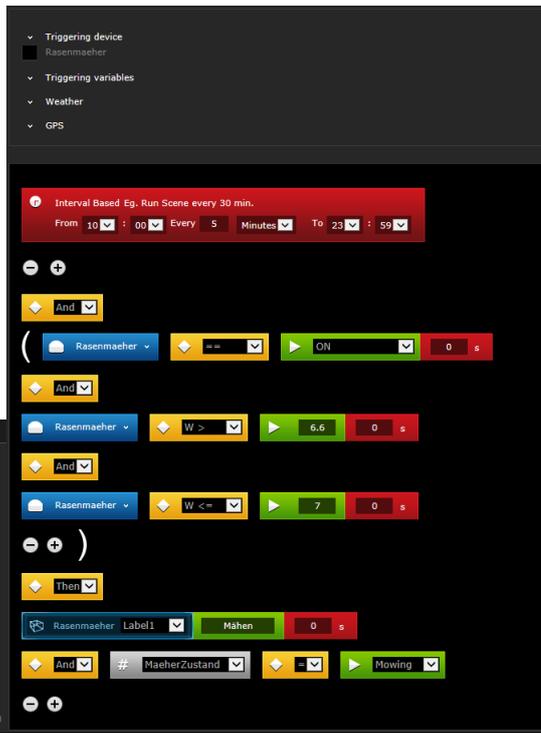
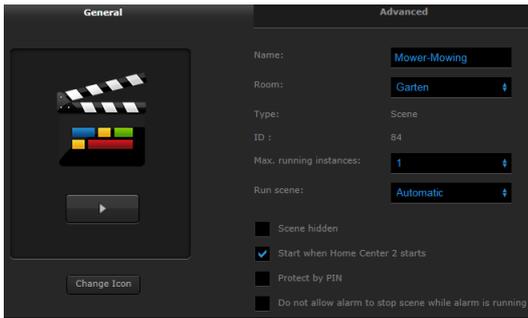
Charging:



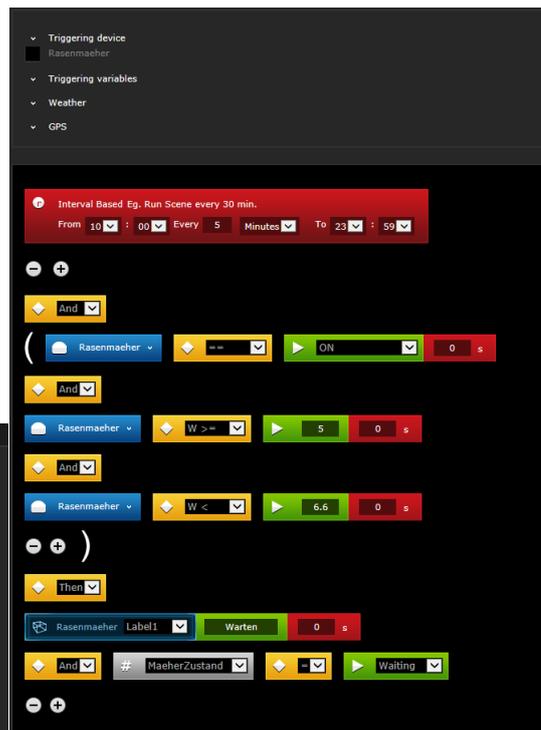
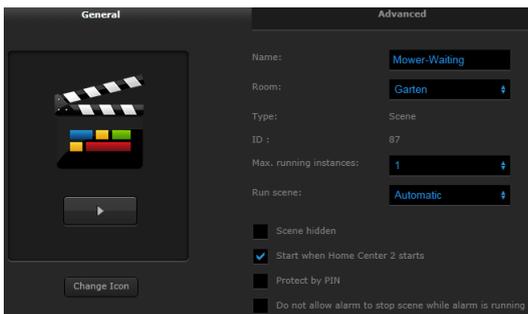
Balancing:



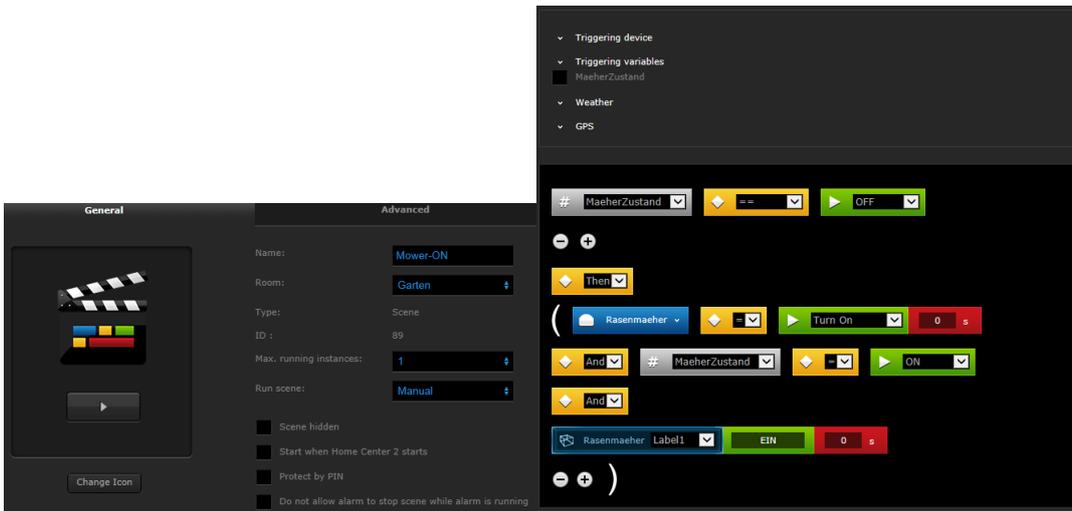
Mowing:



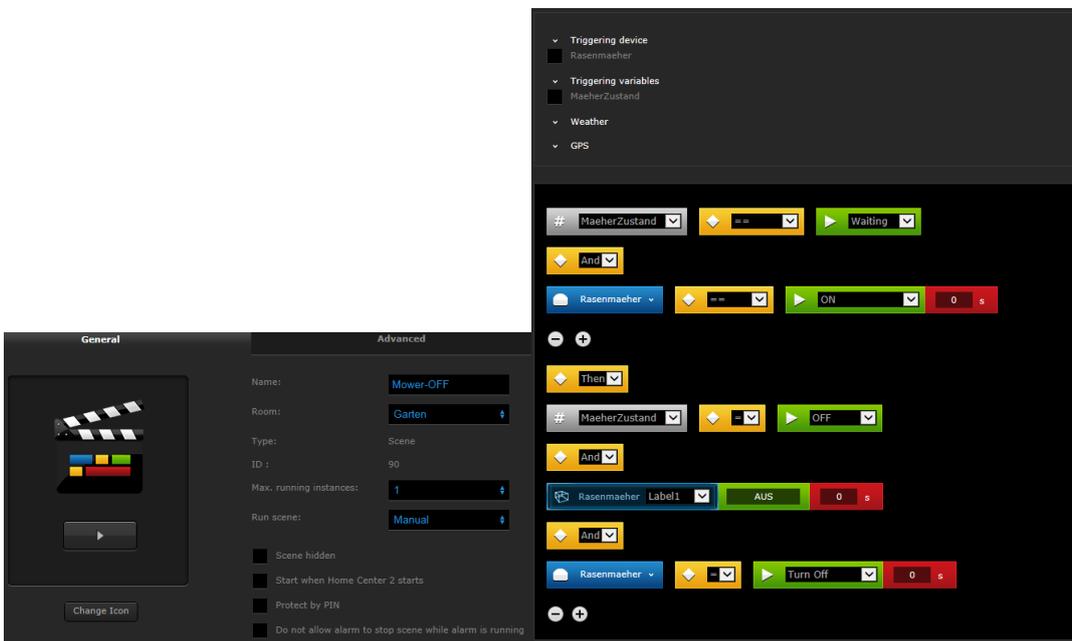
Waiting:



ON:



OFF



How it looks in the overview:



And when the VD is opened:

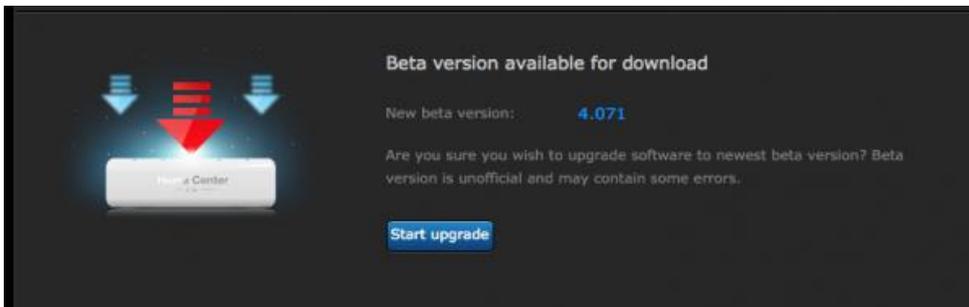


11 Downgrading HCL/HC2

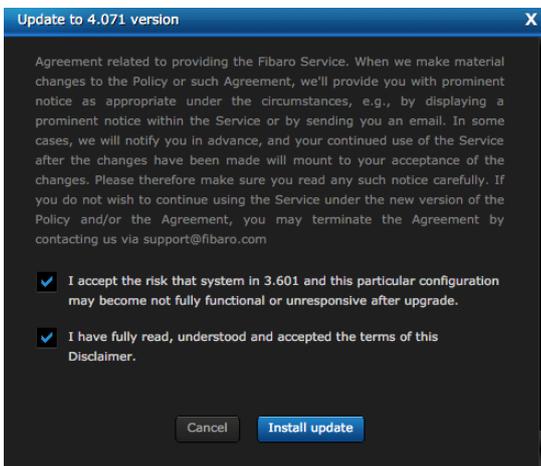
The following guideline is taken from:

<http://domotica4all.com/book-review/tutorial-how-to-downgrade-firmware-version-of-z-wave-controller-fibaro-home-center-2-or-home-center-lite/>

Users of home automation controllers Fibaro Home Center 2 and Fibaro Home Center Lite are often prompted to install a new firmware version. Sometimes the new version is a Beta. Despite the warnings, many users install it to enjoy or test the new features. On the other hand, there are some times, some former features that do not work. Firmware version it is not so stable (actually, that is a beta).



Fibaro firmware update beta available



We are often asked how it is the procedure put back the Z-Wave controller HCLite Fibaro HC2 or into a stable previous firmware version.

The first thing to do is a backup of the system, just in case, as usual.

The procedure to downgrade the firmware version of the controller requires putting the controller in recovery mode. To do this, you have to turn it off (by removing the power supply). Then you have to push the “+” button and keep it pressed while turning it on. You will see that the boot sequence of blue lights is different from the usual. If we enter the IP address of the controller, instead of the usual screen we find this one:



Fibaro Recovery mode

Here there are three options. Repair system, repair from a file, or fully repair (warning! This third option will delete all your data).

In this case, we chose the second option, but of course, we need the file with the firmware you want to load to your controller.

To download the file, manufacturer Fibaro have two pages available. One for and one for HC2 HCLite:

<http://updatehc2.fibaro.com/>

<http://updatehcl.fibaro.com/>

Fibaro do not publish all the firmware versions, but the most significant. For example, a stable firmware version today would be v.054. To download should go to:

<http://updatehc2.fibaro.com/4.054/>

<http://updatehcl.fibaro.com/4.054/>

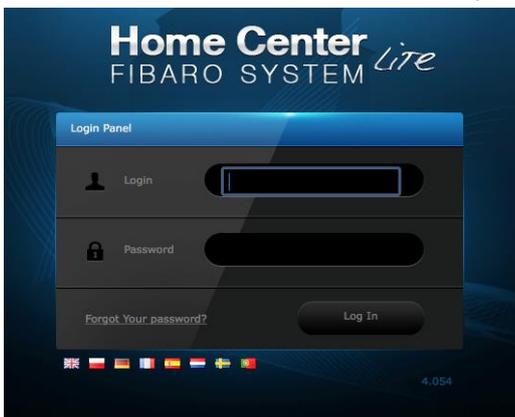
Lista plików w katalogu /4.054/

Nazwa pliku	Data modyfikacji	Rozmiar	Opis
..	03-Aug-2015 17:29	-	
checksum	03-Aug-2015 17:29	1k	
patch.tar.gz	03-Aug-2015 17:29	47610k	

LiteSpeed Web Server updatehc2.fibaro.com:80

Fibaro Firmware Download

After downloading the file, we can go to the recovery screen. Then choose the option "Recovery from File", select the downloaded file and click "Repair".



Fibaro HC Lite Restaurado

Finally we would get our controller with the desired firmware version, and it should have all our devices and scenes as when using the beta.