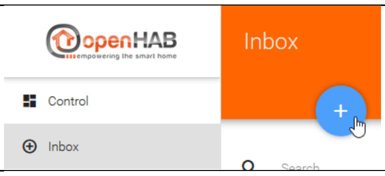
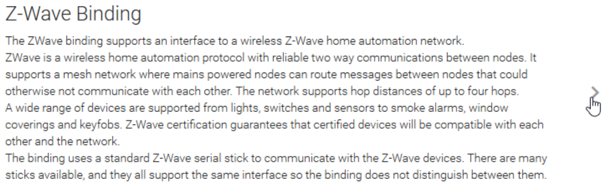
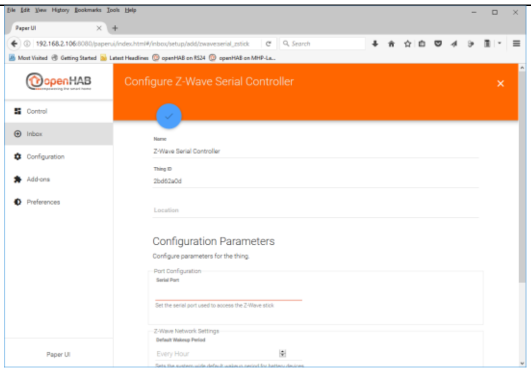
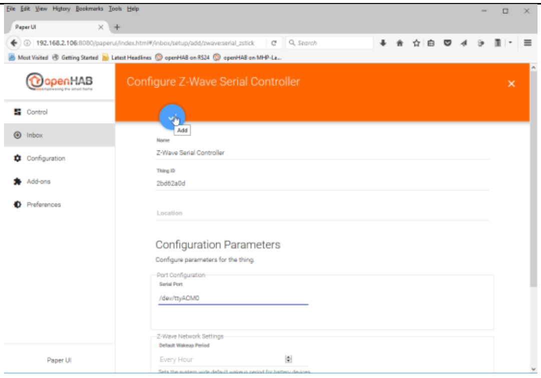
openHAB2 Raspberry beginner’s walkthrough – Using Raspberry Pi 3 and openHAB2 to create a home automation controller for Z-Wave, WiFi LED, 433MHz plugs, Yahoo Weather and meteoblue.com weather widget

Contents

Adding things connected via Z-Wave controller

NOTE: first you have to connect the Z-Wave controller as a thing. After this you will use HABmin to further include things into the Z-Wave network. These things should show up automatically in the inbox of PAPER UI.

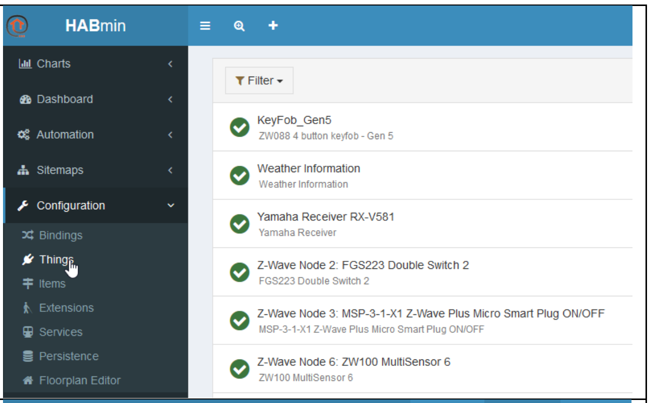
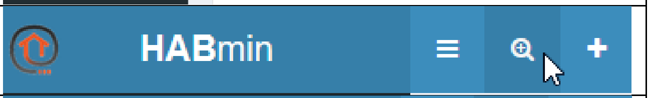
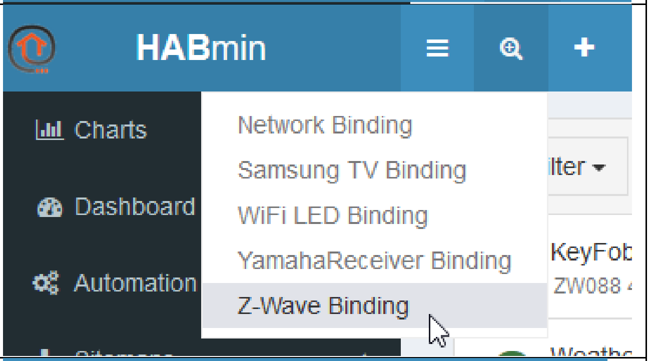
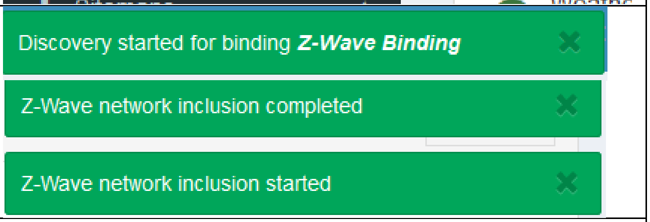
Adding Z-Wave controller

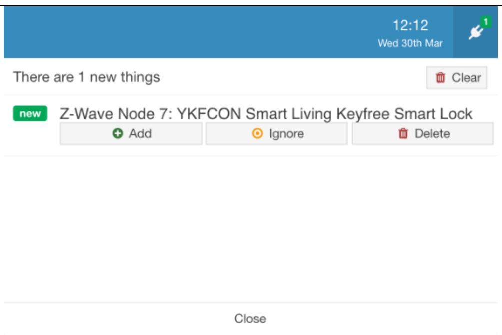
Select in your PAPER UI inbox the add blue add icon (+)	 openHAB empowering the smart home Inbox + Control Inbox Search
Select the Binding <Z-Wave Binding>	 Z-Wave Binding The ZWave binding supports an interface to a wireless Z-Wave home automation network. ZWave is a wireless home automation protocol with reliable two way communications between nodes. It supports a mesh network where mains powered nodes can route messages between nodes that could otherwise not communicate with each other. The network supports hop distances of up to four hops. A wide range of devices are supported from lights, switches and sensors to smoke alarms, window coverings and keyfobs. Z-Wave certification guarantees that certified devices will be compatible with each other and the network. The binding uses a standard Z-Wave serial stick to communicate with the Z-Wave devices. There are many sticks available, and they all support the same interface so the binding does not distinguish between them.
Select the thing <Z-Wave Serial Controller>	 Z-Wave Serial Controller Z-Wave USB Stick with Serial Interface
Now you have to configure the thing In this case you have to enter the serial Port of the Z-Wave controller The standard port where the UBS-Z-Wave controller should come up it: /dev/ttyACM0	 192.168.2.106:8080/paperui/index.html#/inbox/setup/add/zwaveserial_zstick Configure Z-Wave Serial Controller Name: Z-Wave Serial Controller Thing ID: 2bd62a0d Location Configuration Parameters Configure parameters for the thing. Port Configuration Serial Port Set the serial port used to access the Z-Wave stick Z-Wave Network Settings Default Wakeup Period: Every Hour /dev/ttyACM0
Then just add the thing by clicking on the blue check icon	 192.168.2.106:8080/paperui/index.html#/inbox/setup/add/zwaveserial_zstick Configure Z-Wave Serial Controller Add Name: Z-Wave Serial Controller Thing ID: 2bd62a0d Location Configuration Parameters Configure parameters for the thing. Port Configuration Serial Port: /dev/ttyACM0 Z-Wave Network Settings Default Wakeup Period: Every Hour <(+)>

Adding Z-Wave things using HABmin UI

To include further devices into your Z-Wave network and make the available as openHAB2 things you have to use the HABmin inclusion functionality.

Please also consult the online documentation for general information about Z-Wave :
https://github.com/openhab/org.openhab.ui.habmin/wiki/ZWave-Device-Installation

Start HABmin Direct URL: http://xxx.xxx.xxx.xxx:8080/habmin/index.html#/home The select the Things tab -Configuration --Things HABmin will show you all the things which are available in PAPER UI as well	 HABmin Charts Dashboard Automation Sitemaps Configuration Bindings Things Items Extensions Services Persistence Floorplan Editor Filter KeyFob_Gen5 ZW088 4 button keyfob - Gen 5 Weather Information Weather Information Yamaha Receiver RX-V581 Yamaha Receiver Z-Wave Node 2: FGS223 Double Switch 2 FGS223 Double Switch 2 Z-Wave Node 3: MSP-3-1-X1 Z-Wave Plus Micro Smart Plug ON/OFF MSP-3-1-X1 Z-Wave Plus Micro Smart Plug ON/OFF Z-Wave Node 6: ZW100 MultiSensor 6 ZW100 MultiSensor 6
To start the inclusion in HABmin you have to select the add thing icon of HABmin (magnifying glass)	
Then select the Z-Wave binding	 HABmin Charts Dashboard Automation Network Binding Samsung TV Binding WiFi LED Binding YamahaReceiver Binding Z-Wave Binding lter KeyFob ZW088
This now is triggering the inclusion mode of your Z-Wave controller	 Discovery started for binding ***Z-Wave Binding*** Z-Wave network inclusion completed Z-Wave network inclusion started
Now you have to put your Z-Wave devices in inclusion mode to be detected for online detection (applicable when using UZB Z-Wave PLUS USB stick by Z-Wave.Me as a Z-Wave controller	Please refer to the Z-Wave devices manual how to put them in inclusion mode
Optional: Using the Aeotec by Aeon Labs Z-Stick Gen5 which is allowing for offline inclusion Since this Z-Wave controller stick is allowing for offline inclusion, you have to unplug the stick before you start the inclusion on HABmin and use the inclusion button on the stick to start the inclusion mode inclusion mode of your Z-Wave controller The plug in the stick again do the HABmin inclusion. You do not have to put your Z-Wave devices again in inclusion mode.	Please refer to the Aeotec by Aeon Labs Z-Stick Gen5 documentation for further details

And when a device is found it will be listed Just select the <Add> button to include this device as a thing for openHAB2 NOTE: some devices may come up as **“unknown device”**. You can still add the device and “cure” the information later. The normal reason for this is that the device was not able to finish the communication with HABmin during the inclusion process This might be very likely for battery powered devices since the will go to sleep mode after a certain amount of time. To cure this you just have to manually wake up the device (see device manual) multiple times (up to 10 times) before the communication is completed and the device is recognized as a proper thing in HABmin ATTENTION: Using the **Aeotec by Aeon Labs Z-Stick Gen5** almost every battery powered item will come up as **“unknown device”** since in the timespan you include the battery device to the stick and you do the HABmin inclusion after you plugged the stick back in will be to long an most of the devices will go back to sleep. You might prevent this by manually waking up the device again just seconds before you start the inclusion on HABmin. The other reason is that the device is not supported by the openHAB2 Z-Wave binding. You can check the supported devices at: http://www.cd-jackson.com/index.php/zwave/zwave-device-database/zwave-device-list	 12:12 Wed 30th Mar There are 1 new things Z-Wave Node 7: YKFCON Smart Living Keyfree Smart Lock
The newly added thing should now be visible in the HABmin GUI things tab Now you can switch to the PAPER UI and find the new things in the inbox or check the new things in the things tab	

For details documentation on the Z-Wave Binding usage in HABmin consult:

http://www.cd-jackson.com/index.php/openhab/habmin/10-habmin-zwave-binding-initialisation

Creating items form things

The concept of Things, Channels, Items and Links

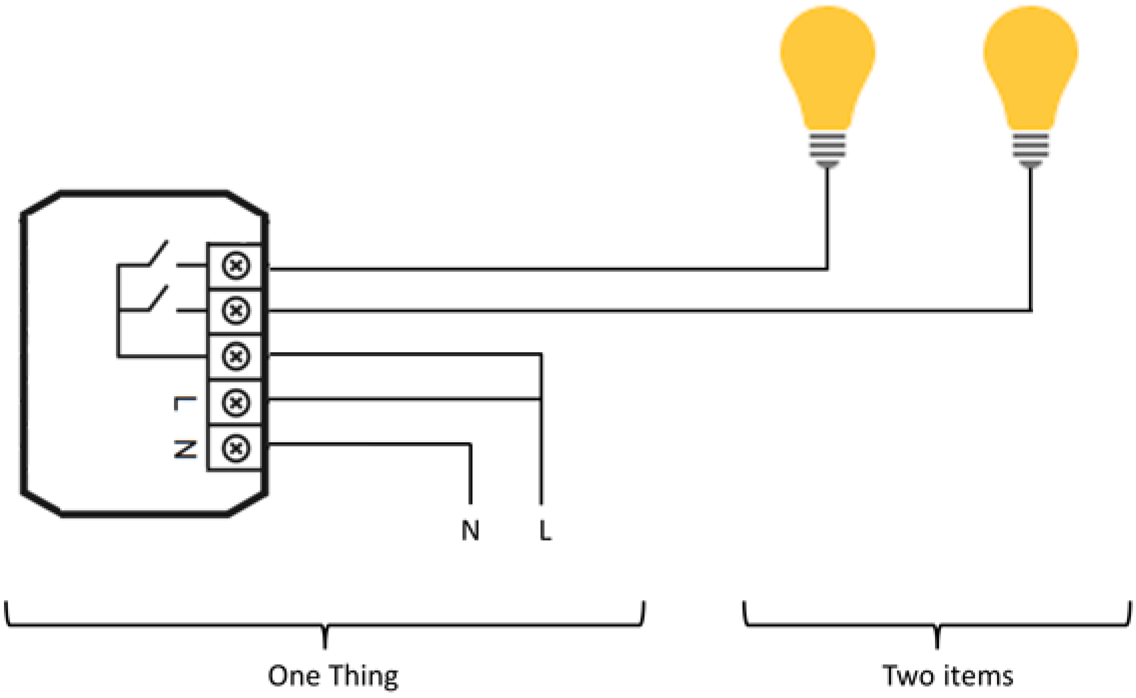
Before we start please read first the concept of Things, Channels, Items and Links which is directly tanken from the openHAB2 User Manual page *http://docs.openhab.org/concepts/index.html#things-channels-items-and-links*

Things are the entities that can be physically added to a system and which can potentially provide many functionalities at once. It is important to note that things do not have to be devices, but they can also represent a web service or any other manageable source of information and functionality. Things provide their functionality through a set of Channels. Channels are "passive" and can be regarded as a declaration of a Thing, what it can offer. It is up to the individual setup, which of the Channels are actively used through Items (see below).

Items represent (fine-grained) functionality that is used by applications - as user interfaces or automation logic. Items have a state and they can receive commands.

The glue between Things and Items are Links. Links are associations between exactly one Thing Channel and one Item. If a Channel is linked to an Item, it is "enabled", which means that the functionality that the Item represents is handled through the given Channel. Channels can be linked to multiple Items and Items can be linked to multiple Channels.

To illustrate these concepts, take a two-channel actuator that controls two lights:

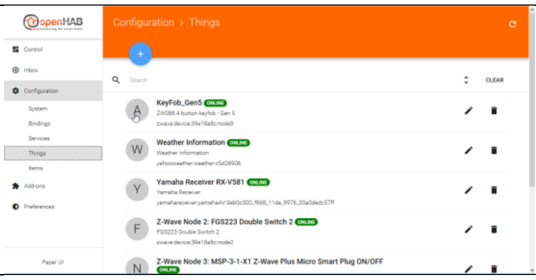
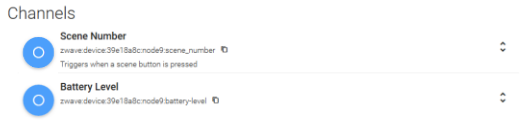
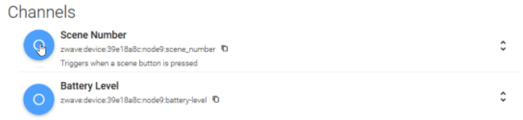
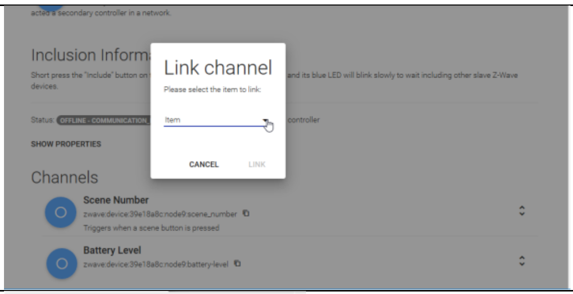
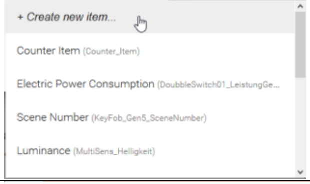
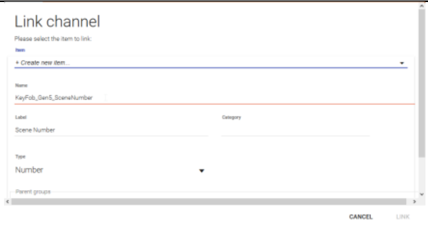
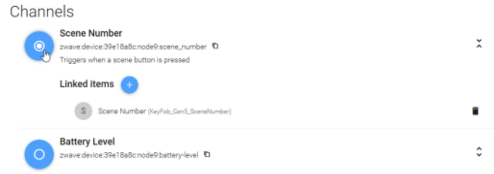
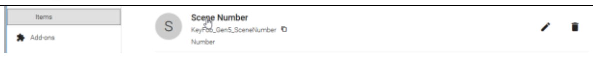


The actuator is the Thing. This might be installed in the electrical cabinet, it has a physical address and needs to be setup and configured in order to be used. The user is instead interested in the two lights, which are located at different locations in his home. These lights are the desired functionality, thus the Items and they are linked to the Channels of the actuator. A Link can be regarded like a physical wire in this example.

The basic process of in PAPER UI is explained on:

http://docs.openhab.org/tutorials/beginner/configuration.html

Creating the required items for my project

Switch to -Configuration --Things and select the KeyFob_Gen5 thing	 openHAB Control Inbox Configuration System Bindings Services Things Items Add-ons Preferences Paper UI Configuration › Things Search CLEAR KeyFob_Gen5 ONLINE ZW088 4 button keyfob - Gen 5 zwave:device:39e18a8c:node9 Weather Information ONLINE Weather Information yahooweather:weather:c5d26906 Yamaha Receiver RX-V581 ONLINE Yamaha Receiver yamahareceiver:yamahaAV:9ab0c000_f668_11de_9976_00a0dedc57ff Z-Wave Node 2: FGS223 Double Switch 2 ONLINE FGS223 Double Switch 2 zwave:device:39e18a8c:node2 Z-Wave Node 3: MSP-3-1-X1 Z-Wave Plus Micro Smart Plug ON/OFF ONLINE
A list of all the available channels of the thing will be shown	 Channels Scene Number zwave:device:39e18a8c:node9:scene_number Triggers when a scene button is pressed Battery Level zwave:device:39e18a8c:node9:battery-level
Now click on the blue icon in front of the channel number to link this channel to an item	 Channels Scene Number zwave:device:39e18a8c:node9:scene_number Triggers when a scene button is pressed Battery Level zwave:device:39e18a8c:node9:battery-level
A link channel window will come up Select the pull down option for the item	 acted a secondary controller in a network. Inclusion Inform Short press the "Include" button on and its blue LED will blink slowly to wait including other slave Z-Wave devices. Status: OFFLINE - COMMUNICATION controller SHOW PROPERTIES Link channel Please select the item to link: Item CANCEL LINK Channels Scene Number zwave:device:39e18a8c:node9:scene_number Triggers when a scene button is pressed Battery Level zwave:device:39e18a8c:node9:battery-level
The next window coming up will allow you to select already existing items or: In our case <+Create new item…>	 + Create new item... Counter Item (Counter_Item) Electric Power Consumption (DoubbleSwitch01_LeistungGe... Scene Number (KeyFob_Gen5_SceneNumber) Luminance (MultiSens_Helligkeit)
The next window will allow you to configure the item you want to link to the thing channel You can also change the name of the item if you are planning to run with your own naming convention Then select <LINK> to create your new item	 Link channel Please select the item to link: Item + Create new item... Name KeyFob_Gen5_SceneNumber Label Scene Number Category Type Number Parent groups CANCEL LINK
Now the blue icon to the left of the name will change (white dot in the center) By clicking on this icon the channel will expand and show you the linked items to this cannel	 Channels Scene Number zwave:device:39e18a8c:node9:scene_number Triggers when a scene button is pressed Linked items Scene Number (KeyFob_Gen5_SceneNumber) Battery Level zwave:device:39e18a8c:node9:battery-level
You can now find the new item in -Configuration --Items	 Items Add-ons Scene Number KeyFob_Gen5_SceneNumber Number

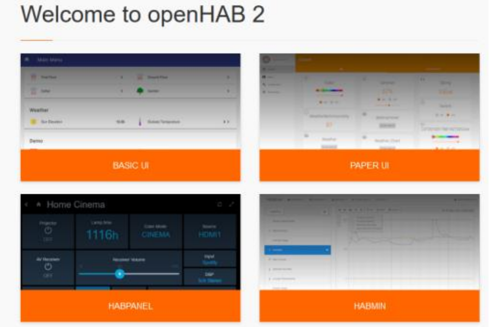
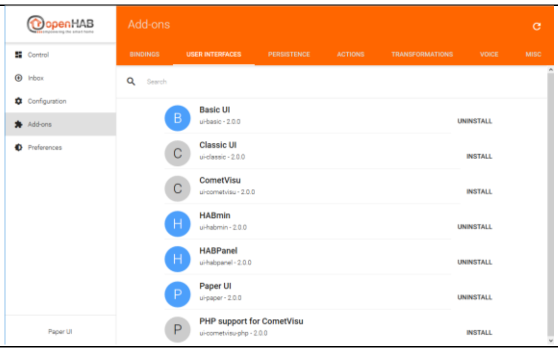
Now go on creating the items according to the list below

Thing	Channel	Item
KeyFob_Gen5	zwave:device:39e18a8c:node9:scene_number	KeyFob_Gen5_SceneNumber
Weather Information	yahooweather:weather:c5d26906:temperature	Yahoo_Temperature
Weather Information	yahooweather:weather:c5d26906:humidity	Yahoo_Humidity
Yamaha Receiver RX-V581	yamahareceiver:yamahaAV:9ab0c000_f668_11de_9976_00a0dedc57ff:power	YamahaReceiverRXV581_Power
Yamaha Receiver RX-V581	yamahareceiver:yamahaAV:9ab0c000_f668_11de_9976_00a0dedc57ff:volume	YamahaReceiverRXV581_Volume
Yamaha Receiver RX-V581	yamahareceiver:yamahaAV:9ab0c000_f668_11de_9976_00a0dedc57ff:mute	YamahaReceiverRXV581_Mute
Z-Wave Node 2: FGS223 Double Switch 2	zwave:device:39e18a8c:node2:meter_watts	DoubbleSwitch01_Leistung Gesamt
Z-Wave Node 2: FGS223 Double Switch 2	zwave:device:39e18a8c:node2:switch_binary1	DoubbleSwitch01_Relais1
Z-Wave Node 3: MSP-3-1-X1 Z-Wave Plus Micro Smart Plug ON/OFF	zwave:device:39e18a8c:node3:switch_binary	SchuKo01
Z-Wave Node 3: MSP-3-1-X1 Z-Wave Plus Micro Smart Plug ON/OFF	zwave:device:39e18a8c:node3:meter_watts	SchuKo01_Leistung
Z-Wave Node 6: ZW100 MultiSensor 6	zwave:device:39e18a8c:node6:sensor_relhumidity	MultiSens_Luftfeuchte
Z-Wave Node 6: ZW100 MultiSensor 6	zwave:device:39e18a8c:node6:sensor_temperature	MultiSens_Temperatur
Z-Wave Node 6: ZW100 MultiSensor 6	zwave:device:39e18a8c:node6:sensor_luminance	MultiSens_Helligkeit
HF-LPB100-ZJ200	wifiled:wifiled:F0FE6B314910:power	HFLPB100ZJ200_Power
HF-LPB100-ZJ200	wifiled:wifiled:F0FE6B314910:color	HFLPB100ZJ200_Color
HF-LPB100-ZJ200	wifiled:wifiled:F0FE6B314910:white	HFLPB100ZJ200_White
DUMMYTHING_SamsungTV	Data missing	DUMMYITEM_SamsungTV_Power

Chapter 12: Creating a dashboard for your home automation project

Every User Interface on openHAB2 is providing its own style of dashboards to control your home automation project, display current item states and attribute values or even include online information like web pages.

In our configuration of openHAB2 you will have the choice of 4 different user interfaces which can be selected

Start openHAB2 start screen Direct URL: http://xxx.xxx.xxx.xxx:8080/ start/index Here you will now find the GUIs: <BASIC UI> (requires setup using textual *.sitemap files) <PAPER UI> (which were already using to configure openHAB2) <HABPANEL> (this UI is all about creating a dashboard) <HABMIN> (which we already used for inclusion of Z-Wave devices)	
You can also install more UIs in PAPER UI Add-ons section: `http://xxx.xxx.xxx.xxx:8080/paperui/index.html#/extensions` on the tab `<USER INTERFACES>`	

Basic UI ui-basic - 2.0.0 UNINSTALL
Classic UI ui-classic - 2.0.0 INSTALL
CometVisu ui-cometvisu - 2.0.0 INSTALL
HABmin ui-habmin - 2.0.0 UNINSTALL
HABPanel ui-habpanel - 2.0.0 UNINSTALL
Paper UI ui-paper - 2.0.0 UNINSTALL
PHP support for CometVisu ui-cometvisu-php - 2.0.0 INSTALL

BASIC UI dashboard

Since this is a beginner's tutorial where I am trying to get things done using as less textual coding as possible, I will not go into the details of creating a dashboard for the BASIC UI since it requires the file based approach.

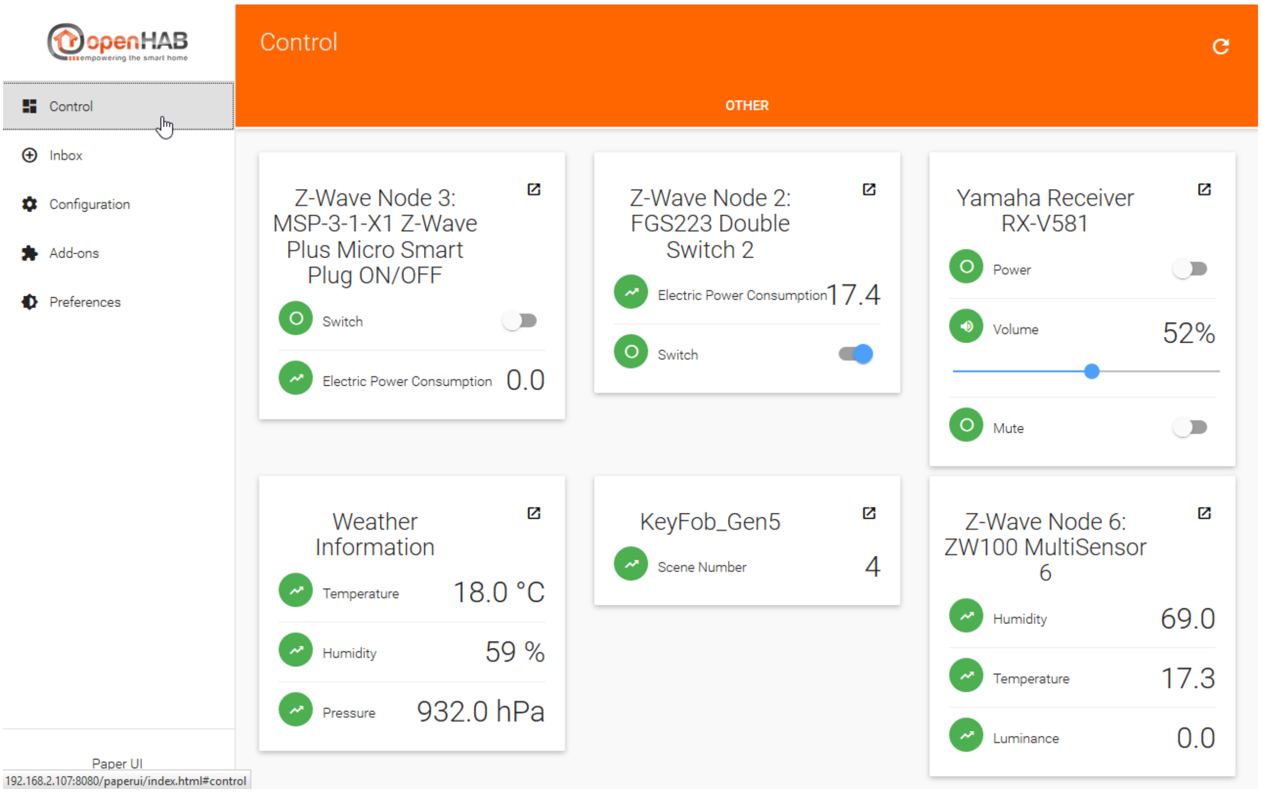
Tough you might find that some specific requirements to a dashboard can only be solved creating this textual sitemap configuration and using BASIC UI for displaying your dashboard.

You can find more information in how to setup and use BASIC UI on:

http://docs.openhab.org/configuration/sitemaps.html

PAPER UI dashboard

The simplest way of creating an interactive user interface is in just clicking on the <CONTROL> tab to PAPTER UI. Now you should see all the items grouped by the things they belong to.


openHAB
empowering the smart home
Control
Inbox
Configuration
Add-ons
Preferences
Paper UI
192.168.2.107:8080/paperui/index.html#control
Control
OTHER

Z-Wave Node 3: MSP-3-1-X1 Z-Wave Plus Micro Smart Plug ON/OFF
Switch
Electric Power Consumption 0.0

Z-Wave Node 2: FGS223 Double Switch 2
Electric Power Consumption 17.4
Switch

Yamaha Receiver RX-V581
Power
Volume 52%
Mute

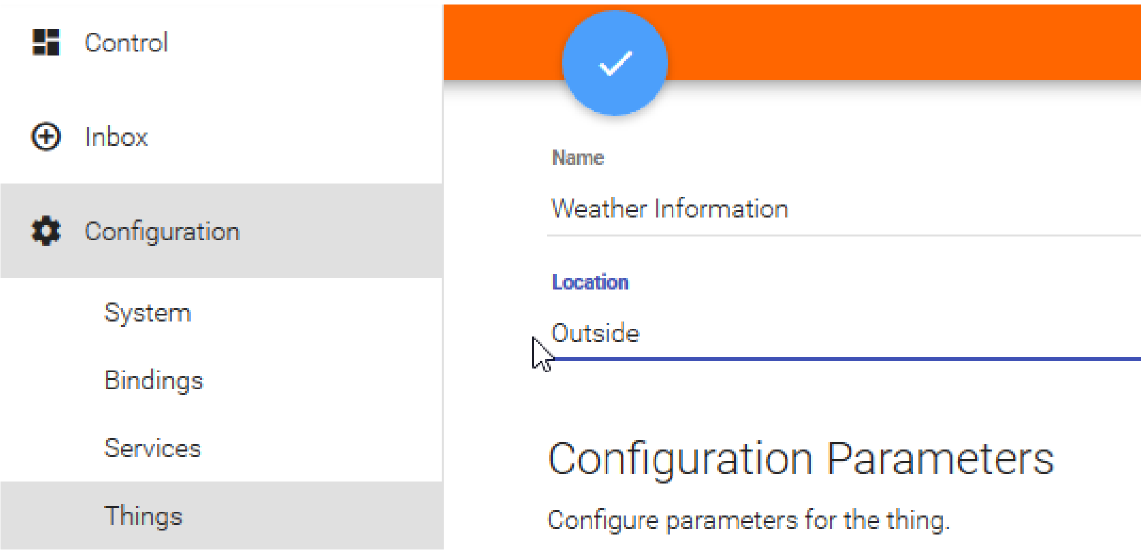
Weather Information
Temperature 18.0 °C
Humidity 59 %
Pressure 932.0 hPa

KeyFob_Gen5
Scene Number 4

Z-Wave Node 6: ZW100 MultiSensor 6
Humidity 69.0
Temperature 17.3
Luminance 0.0

You can also easily create different tabs of things by adding <Location> information to the thing in the thing tab:

NOTE: I had issues in updating a few things, so it will be good to it while creating the things



Name
Weather Information

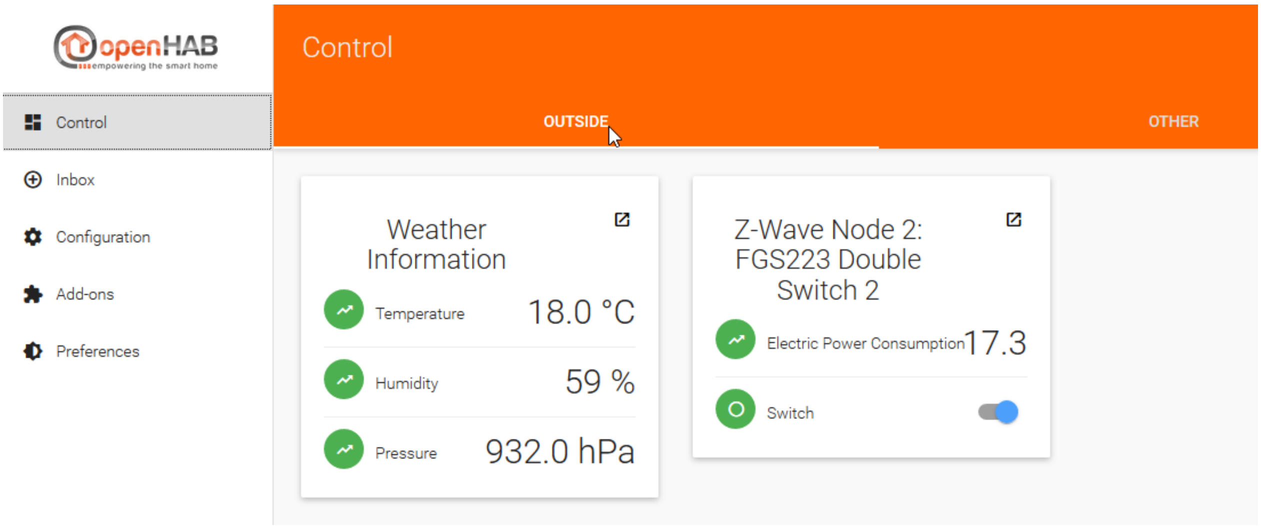
Location
Outside

Configuration Parameters

Configure parameters for the thing.

Now you will have Control panel giving you various tabs with items grouped by their things

Tab <OUTSIDE>



Weather Information

Temperature 18.0 °C

Humidity 59 %

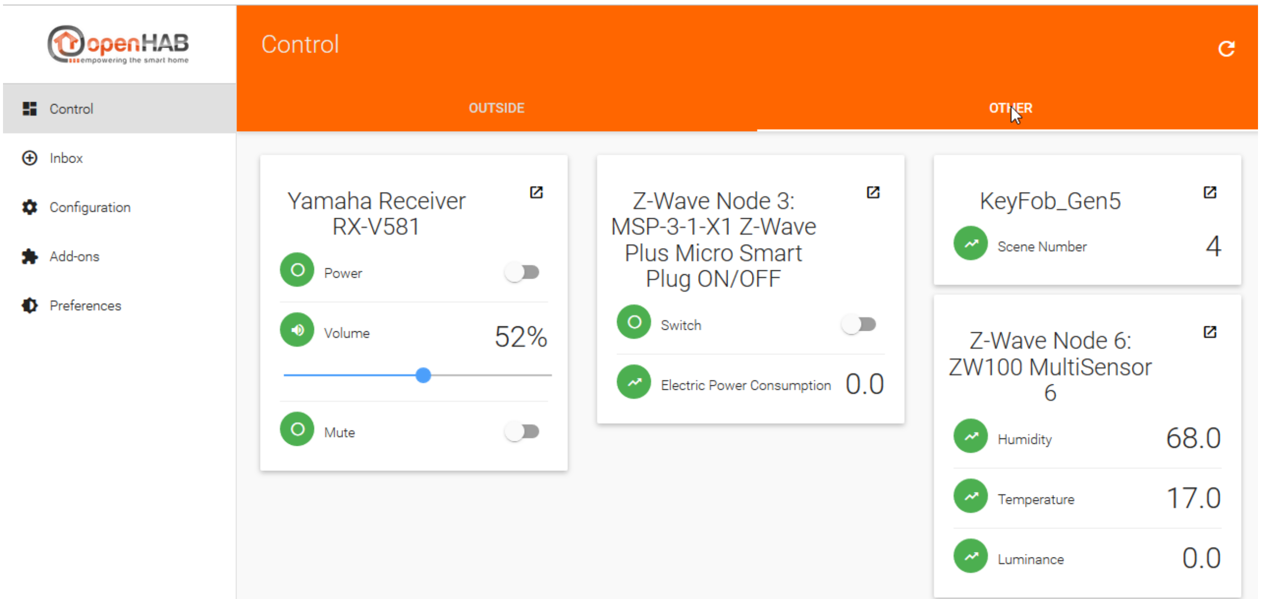
Pressure 932.0 hPa

Z-Wave Node 2: FGS223 Double Switch 2

Electric Power Consumption 17.3

Switch

Standard tab <OTHER>



Yamaha Receiver RX-V581

Power

Volume 52%

Mute

Z-Wave Node 3: MSP-3-1-X1 Z-Wave Plus Micro Smart Plug ON/OFF

Switch

Electric Power Consumption 0.0

KeyFob_Gen5

Scene Number 4

Z-Wave Node 6: ZW100 MultiSensor 6

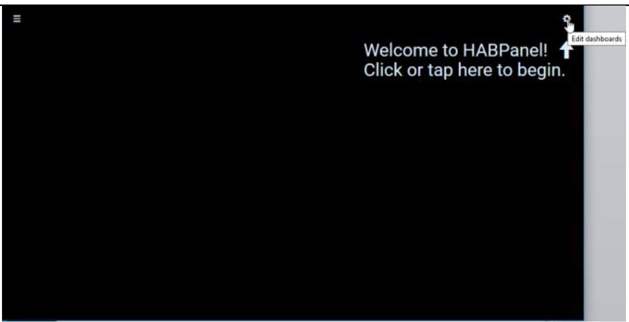
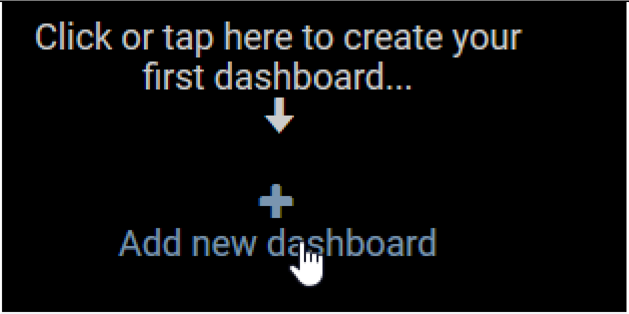
Humidity 68.0

Temperature 17.0

Luminance 0.0

HABPANEL UI dashboard

The HABPANEL UI is all about creation a dashboard for your home automation project. Therefore you can not to do all the configuration work for openHAB2 in this UI. The configuration hast to be done in the other UIs

Start HABPANEL http://xxx.xxx.xxx.xxx:8080/HABPANEL/index.html#/ It will come up a complete blank panel asking you to start configuration	 Welcome to HABPanel! Click or tap here to begin.
You now can choose to <Add new dashboard>	 Click or tap here to create your first dashboard...
Create switches to control the Z-Wave switches Details MISSING	Picture MISSING

You can find the standard documentation on:

http://docs.openhab.org/addons/uis/HABPANEL/readme.html

HABMIN dashboard

And once again, since this is a beginner's tutorial I can't go into all the possibilities of how to create a dashboard in openHAB2. So I will not go into creating a dashboard in HABMIN. But be aware the HABMIN is providing a powerful graphical way of creating dashboards you might want to have a look at.

You can find the standard documentation on:

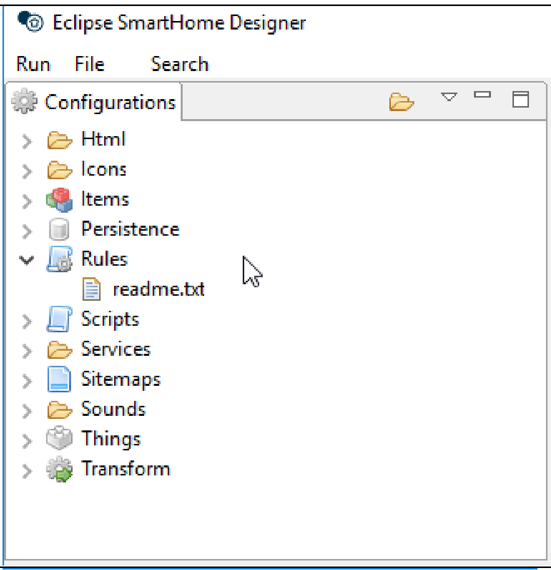
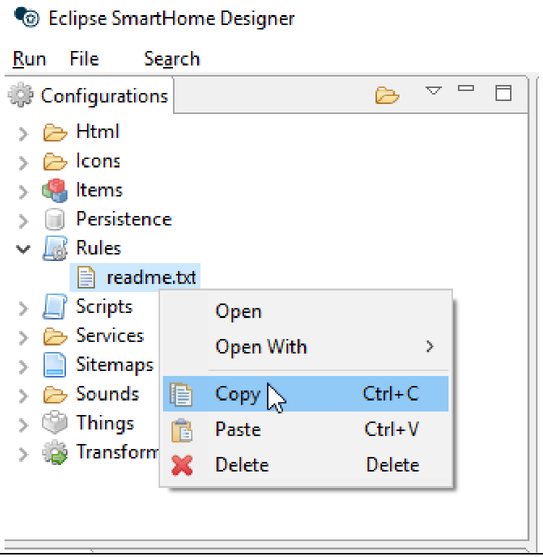
http://docs.openhab.org/addons/uis/habmin/readme.html

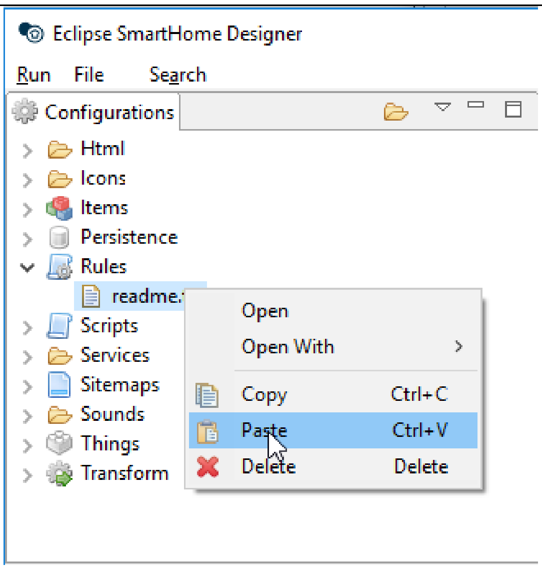
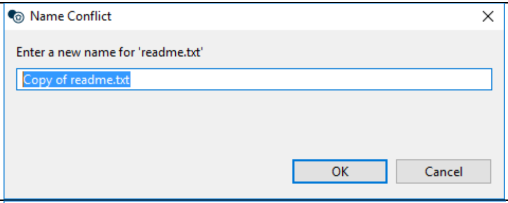
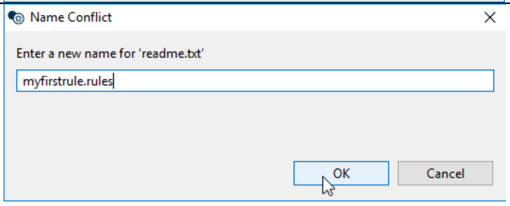
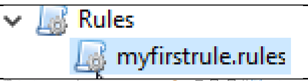
Chapter 13: Creating rules

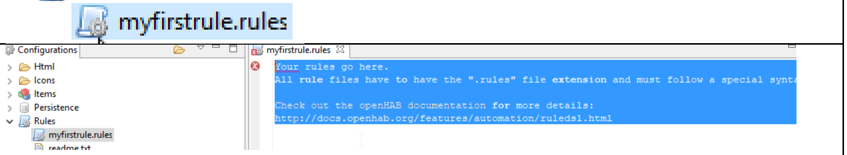
Since rules can't be configured using the PAPER UI (stable version) you now have to go to the text files for now. For this part we will now use the Eclipse Smart Home Designer since it creates at least some syntax highlighting.

REMARK: You might find already some functionality about configuration of rules in the PAPER UI snapshot versions so there will be some changes in the way of doing rules more easily in the future. Also HABmin is providing some kind of graphical rule engine.

Creating the myfirstrule.rules file

Start Eclipse Smart Home Designer on your PC Make sure you have mapped the Raspbian samba drive to your PC and Start Eclipse Smart Home Designer is set to the right folder in this samba drive (see Chapter 7: Installation of Eclipse Smart Home Designer -: part launching first time) You should now the augmented icons for the different folders If you check the Rules folder you will only find a readme.txt file	 Eclipse SmartHome Designer Run File Search Configurations Html Icons Items Persistence Rules readme.txt Scripts Services Sitemaps Sounds Things Transform
You need to create a file with the ending .rules to store your rules there. The fastest way of doing it is using the basic file management functionality of Eclipse Smart Home Designer Right click on the readme.txt file and select copy	 Eclipse SmartHome Designer Run File Search Configurations Html Icons Items Persistence Rules readme.txt Scripts Services Sitemaps Sounds Things Transform Open Open With Copy Ctrl+C Paste Ctrl+V Delete Delete

Then right click again and select Paste	 Eclipse SmartHome Designer Run File Search Configurations Html Icons Items Persistence Rules readme. Scripts Services Sitemaps Sounds Things Transform Open Open With Copy Ctrl+C Paste Ctrl+V Delete Delete
A new window will ask you to enter a new file name.	 Enter a new name for 'readme.txt' Copy of readme.txt
Change the file name to myfirstrule.rules an press <OK>	 Enter a new name for 'readme.txt' myfirstrule.rules myfirstrule.rules <OK>
A new file will appear in the Rules folder showing the rule icon	

Now double click on the file to open it and delete the old content to have a plain rule rile
 	
And save the rule file again
	 Configurations
	Html
	Icons
	Items
	Persistence
	Rules
	myfirstrule.rules
	readme.txt
	myfirstrule.rules
	`Your rules go here.`
	`All rule files have to have the ".rules" file extension and must follow a special synta`
	
	`Check out the openHAB documentation for more details:`
	`http://docs.openhab.org/features/automation/ruledsl.html`
	<ctrl+a>
	<del>
	<ctrlss>

Creating a basic rules

I will try to give some basic rule examples to start with but you have to be aware:

DISCLAIMER:

***I am no coding expert at all and writing rules is still the most difficult part in my home automation project (finding the right commands, the right syntax and so on), so you might be better off using to other tutorials*.**

For the standard rules documentation incl. the rule syntax please refer to the online documentation:

http://docs.openhab.org/configuration/rules-dsl.html

You can also find some rules samples on:

https://github.com/openhab/openhab1-addons/wiki/Samples-Rules

And maybe some coding experts in the community will find the time to create a kind of an openHAB2 compendium to make it easier the non-experts to do rules.

REMAKR: Yes, I know there are plenty of online documentation sites available, but the problem for me was the "plenty" part of it since I always had to go through plenty different websites to finally get the syntax right and make the rule do, what I wanted it to do.

Basic rule example:
We will use the:

KeyFob_Gen5

to control some items linked to

Z-Wave Node 2: FGS223 Double Switch 2

Z-Wave Node 3: MSP-3-1-X1 Z-Wave Plus Micro Smart Plug ON/OFF

Yamaha Receiver RX-V581

DUMMYTHING_SamsungTV

HF-LPB100-ZJ200 (WiFi LED)

Just copy the code to the right directly into your myfirstrule.rules file

```
rule "KeyFob"

when
      Item ZWaveNode9ZW0884ButtonKeyfobGen5_SceneNumber
received update
then
//Scene number 1 - Button 1 (up left) pressed short
      if
(ZWaveNode9ZW0884ButtonKeyfobGen5_SceneNumber.state == 1) {
//Swithing ON the items DoubbleSwitch01_Relais1 and
SchuKo01
                  sendCommand (DoubbleSwitch01_Relais1, ON)
                  sendCommand (SchuKo01, ON)
      }

//Scene number 2 - Button 1 (up left) pressed long
      if
(ZWaveNode9ZW0884ButtonKeyfobGen5_SceneNumber.state == 2) {
//Swithing OFF the items DoubbleSwitch01_Relais1 and
SchuKo01
                  sendCommand (DoubbleSwitch01_Relais1,
OFF)
                  sendCommand (SchuKo01, OFF)
      }

//Scene number 3 - Button 2 (up left) pressed short
      if
(ZWaveNode9ZW0884ButtonKeyfobGen5_SceneNumber.state == 3) {
//Switching ON the items YamahaReceiverRXV581_Power and
DUMMYITEM_SamsungTV_Power
                  sendCommand (YamahaReceiverRXV581_Power,
ON)
                  sendCommand (DUMMYITEM_SamsungTV_Power,
ON)
      }

//Scene number 4 - Button 2 (up left) pressed long
      if
(ZWaveNode9ZW0884ButtonKeyfobGen5_SceneNumber.state == 4) {
//Switching OFF the items YamahaReceiverRXV581_Power and
DUMMYITEM_SamsungTV_Power
                  sendCommand (YamahaReceiverRXV581_Power,
OFF)
                  sendCommand (DUMMYITEM_SamsungTV_Power,
OFF)
      }

//Scene number 5 - Button 3 (up left) pressed short
      if
(ZWaveNode9ZW0884ButtonKeyfobGen5_SceneNumber.state == 5) {
//Switching ON the item HFLPB100ZJ200_Power
                  sendCommand (HFLPB100ZJ200_Power, ON)
      }

//Scene number 6 - Button 3 (up left) pressed long
      if
(ZWaveNode9ZW0884ButtonKeyfobGen5_SceneNumber.state == 6) {
//Switching OFF the item HFLPB100ZJ200_Power
                  sendCommand (HFLPB100ZJ200_Power, OFF)
      }


//Scene number 7 - Button 4 (up left) pressed short
      if
(ZWaveNode9ZW0884ButtonKeyfobGen5_SceneNumber.state == 7) {
//NOTE: since setting the color will influence the dimming
of the WHITE
//          so if you want to set both values at the same
time, always
//          set color first and than
//      set the dimming of WHITE

//Changing the color of the item HFLPB100ZJ200_Color to RED
//openHAB2 is requesting the color to be set in HSL code
(Hue, Saturation, Lightness" or in openHAB terms "HSBType"
(Hue, Saturation, Brightness)
//To convert RGB code to HSL code just go to e.g.
http://www.rapidtables.com/convert/color/rgb-to-hsl.htm
                  sendCommand (HFLPB100ZJ200_Color,(new
HSBType("0,100,50")))
//Dimming the item HFLPB100ZJ200_White to 50%
                  sendCommand (HFLPB100ZJ200_White, 50)
      }

//Scene number 8 - Button 4 (up left) pressed long
      if
(ZWaveNode9ZW0884ButtonKeyfobGen5_SceneNumber.state == 8) {
//NOTE: since setting the color will influence the dimming
of the WHITE
//          so if you want to set both values at the same
time, always
//          set color first and than
//      set the dimming of WHITE

//Changing the color of the item HFLPB100ZJ200_Color to
GREEN
//openHAB2 is requesting the color to be set in HSL code
(Hue, Saturation, Lightness" or in openHAB terms "HSBType"
(Hue, Saturation, Brightness)
//To convert RGB code to HSL code just go to e.g.
http://www.rapidtables.com/convert/color/rgb-to-hsl.htm
                  sendCommand (HFLPB100ZJ200_Color,(new
HSBType("120,100,50")))
                  //Dimming the item HFLPB100ZJ200_White to
100%
                  sendCommand (HFLPB100ZJ200_White, 100)
      }
end
```

Save the rule file. The rule should now be available	<crtl+s>

Test it by clicking through the buttons (pressed long and pressed short)

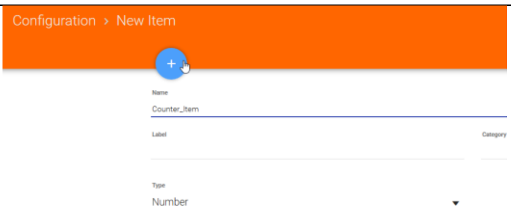
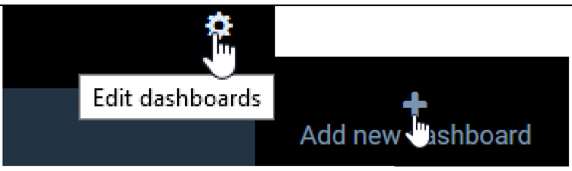
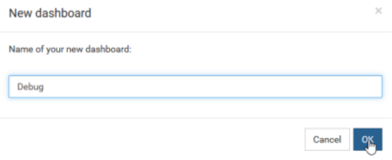
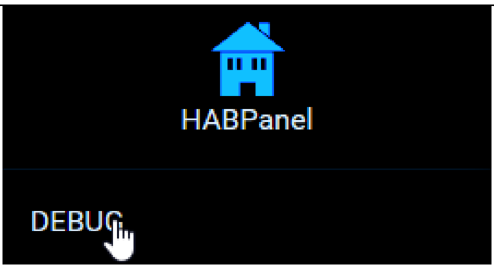
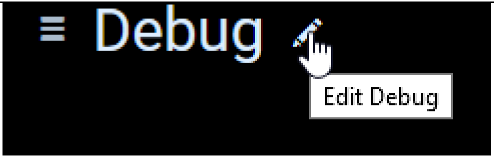
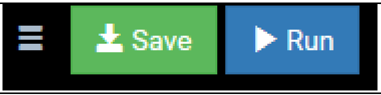
NOTE: When working with rules for the WiFi LED be aware, that the controller is storing the setting when you send the OFF command. So if you want you switch on the LED with a different color you might be better off in setting the color to black before switching off you if you switch on again the LED starts black and not in the old color.

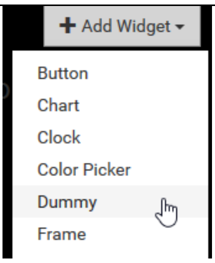
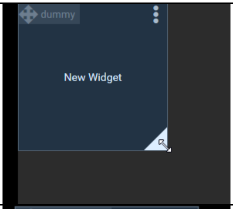
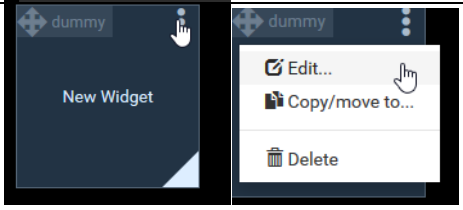
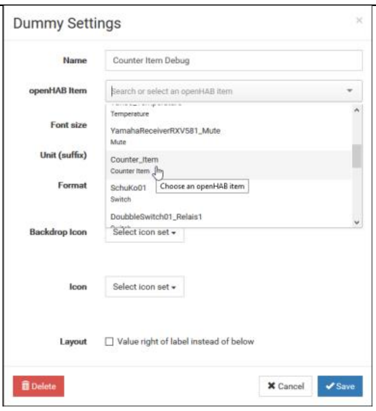
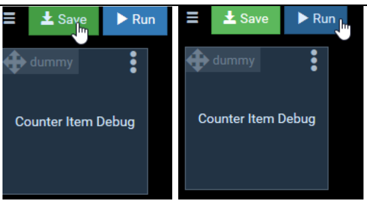
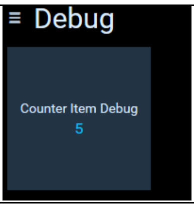
Basic tips for debugging rules

Creating an item the see the value of a variable of a rule online

The standard way of debugging a rule would be to use a logfile.

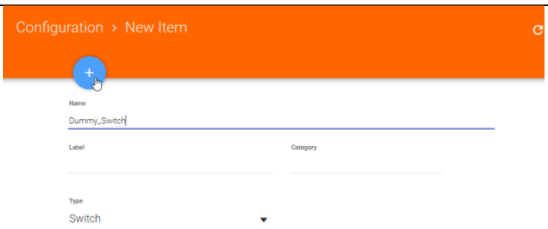
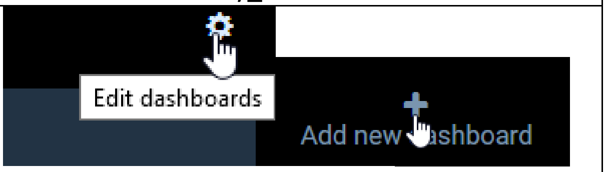
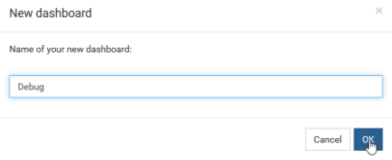
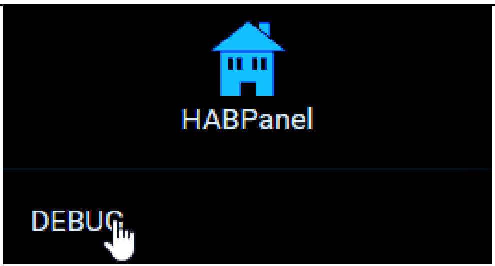
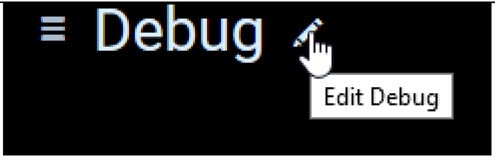
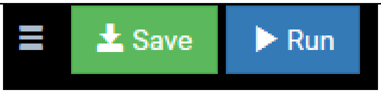
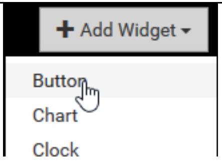
For simple problems you might also be able to visualize the variables online by creation item with the same type and posting the value of the variable inside the rule

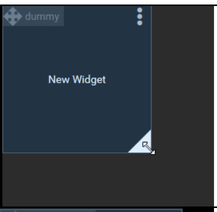
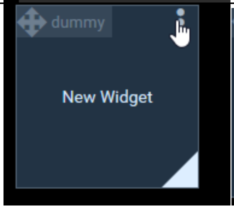
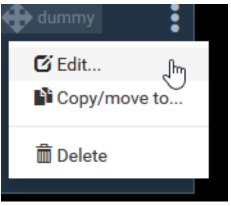
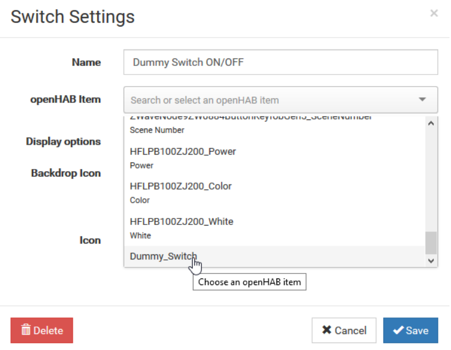
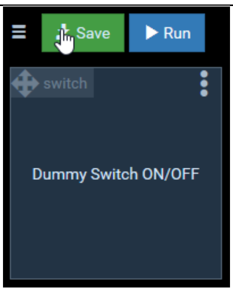
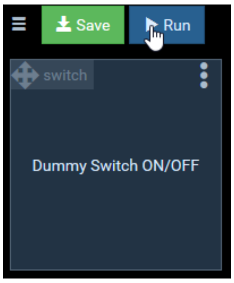
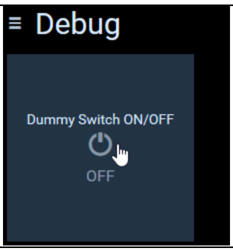
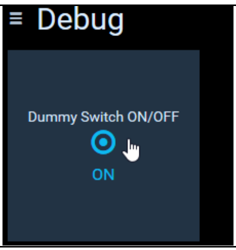
If you are using a variable in your rule initially set to	var Number loop_counter = 0
Create the item Counter_Item with the same type: In PAPER UI switch to -Configuration --Items And press the blue icon (+) Enter Name Counter_Item Select Typ <Number> And confirm the creation by clicking again on the blue icon (+)	 Configuration › New Item Name Counter_Item Label Category Type Number Counter_Item <Number> <(+)>
Result: A new item called Counter_Item should be visible in the item tab NOTE: Since this Item is not linked to a thing it will not show up in the <Control> Tab to PAPER UI.	 Counter Item Counter_Item Number
Now enter a line into your rule to post the value of the var to the Counter_Item	postUpdate(Counter_Item, loop_counter)
To show the value on a dashboard you have to use e.g. HABPANEL Start HABPANEL http://xxx.xxx.xxx.xxx:8080/HABPANEL/index.html#/ and select <Edit dashboards> in the upper right corner then <Add new dashboard> enter the name Debug and <OK>	 Edit dashboards Add new dashboard  New dashboard Name of your new dashboard: Debug Cancel OK Debug <OK>
Now you should have the dashboard <Debug> available in your HABPANEL Select the dashboard	 HABPanel DEBUG
If you now hover with your mouse right of the dashboard name the <Edit *Debug*> icon will appear. Click on it to enter the edit mode.	 Debug Edit Debug <Edit *Debug*>
The edit mode is indicated by the two icons <Save> and <Run>	 Save Run

Now you have to <+ Add Widget> You have to use the <Dummy> widget to display numbers	 + Add Widget Button Chart Clock Color Picker Dummy Frame <+ Add Widget> <Dummy>
The new Widget will appear on your dashboard NOTE: you can easily change the size of the widget by dragging the lower right corner. I will snap to a predefined grid	
To edit the widget you have to click on the 3 dots in the upper right corner of the widget and select <Edit>	 dummy New Widget dummy Edit... Copy/move to... Delete <.> <Edit>
In this window you can change the name to Counter Item Debug You have to select the openHAB item <Counter_Item> to link it to this widget. Optional you can change font size, add a unit and format or add some icons. Now <Save> the widget.	Dummy Settings Name: Counter Item Debug openHAB Item: Search or select an openHAB item Temperature YamahaReceiverRXV581_Mute Mute Counter_Item Counter Item SchuKo01 Switch DoubbleSwitch01_Relais1 Choose an openHAB item Font size Unit (suffix) Format Backdrop Icon: Select icon set Icon: Select icon set Layout: □ Value right of label instead of below Counter Item Debug <Counter_Item> <Save>
Now <Save> the dashboard and <Run> it.	 Save Run dummy Counter Item Debug Save Run dummy Counter Item Debug <Save> <Run>
You will now see the value of your rule variable displayed dynamically on your HABPANEL dashboard	

Creating a virtual switch on HABPANEL to use it in a rule

This switch might come handy if you are debugging your rule while physically having no access to the switch. Standing up and running to the switch might do you some good when it comes down to fitness but surely is disturbing while developing your rule

Create the item Dummy_Switch with the same type: In PAPER UI switch to -Configuration --Items And press the blue icon (+) Enter Name Dummy_Switch Select Typ <Switch> And confirm the creation by clicking again on the blue icon (+)	 Name Dummy_Switch Label Category Type Switch Dummy_Switch <Switch> <(+)>
Result: A new item called Dummy_Switch should be visible in the item tab NOTE: Since this Item is not linked to a thing it will not show up in the <Control> Tab to PAPER UI.	Dummy_Switch Switch
Now replace the item name of the *physical_switch* with the name of the Dummy_Switch	*physical_switch* will become Dummy_Switch
To use the Dummy_Switch on a dashboard you have to use e.g. HABPANEL Start HABPANEL http://xxx.xxx.xxx.xxx:8080/HABPANEL/index.html#/ and select <Edit dashboards> in the upper right corner then <Add new dashboard> enter the name Debug and <OK>	  New dashboard Name of your new dashboard: Debug Debug <OK>
Now you should have the dashboard <Debug> available in your HABPANEL Select the dashboard	
If you now hover with your mouse right of the dashboard name the <Edit *Debug*> icon will appear. Click on it to enter the edit mode.	 <Edit *Debug*>
The edit mode is indicated by the two icons <Save> and <Run>	
Now you have to <+ Add Widget>	

You have to use the <Button> widget to display numbers	<+ Add Widget> <Button>
The new Widget will appear on your dashboard NOTE: you can easily change the size of the widget by dragging the lower right corner. I will snap to a predefined grid	
To edit the widget you have to click on the 3 dots in the upper right corner of the widget and select <Edit>	  <.> <Edit>
In this window you can change the name to Dummy Switch ON/OFF You have to select the openHAB item <Dummy_Switch> to link it to this widget. Optional you can select display options and icons Now <Save> the widget.	 Switch Settings Name: Dummy Switch ON/OFF openHAB Item: Search or select an openHAB item Scene Number HFLPB100ZJ200_Power Power HFLPB100ZJ200_Color Color HFLPB100ZJ200_White White Dummy_Switch Choose an openHAB item Display options Backdrop Icon Icon Delete Cancel Save Dummy Switch ON/OFF <Dummy_Switch> <Save>
Now <Save> the dashboard and <Run> it.	  <Save> <Run>
You will now see the Dummy_Switch Item be visualized by the Dummy Switch ON/OFF widget. The initial state should be <Inactive> Just click on the widget to change the state to <Active>	 

Further Rule examples based on this home automation project:

Switching ON/OFF switches based on luminance reading of the multisensory

The rule is designed to switch on/off the Z-Wave Node 2: FGS223 Double Switch 2 based on the illumination measured by the Z-Wave Node 6: ZW100 MultiSensor 6

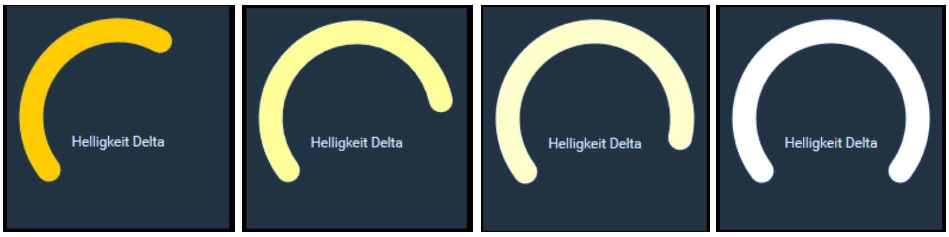
The trigger value is set to 10 LUX

To prevent von switching on/off if the illumination is around 10 lumen and e.g. just a cloud is casting a temporarily shadow, there is a counter included which is measuring multiple times the illumination and only allows to triggering the switch if there reading is consistently (10 times) above or below the trigger value.

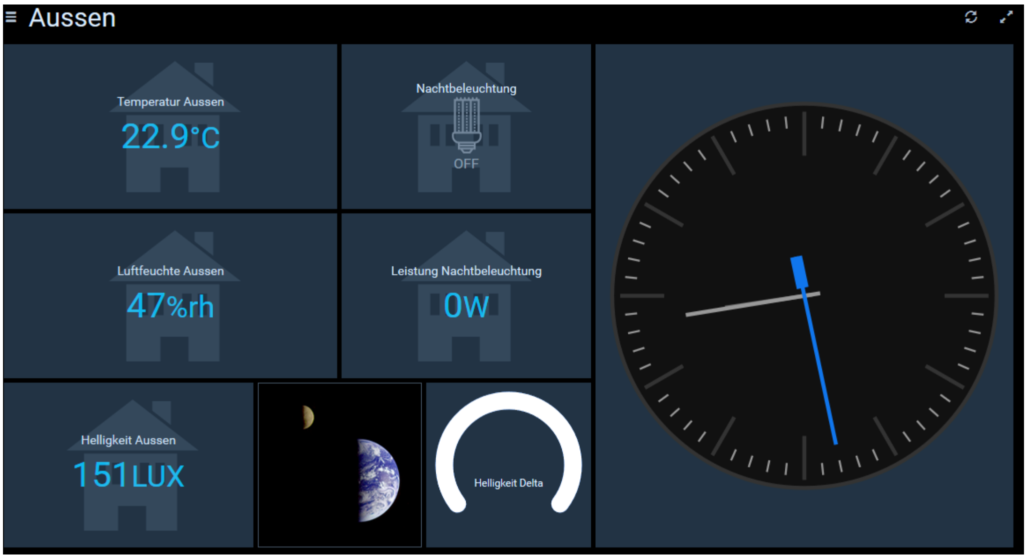
NOTE: Make sure you have create the item Counter_Item since it's used in the rule (see: part "Creating an item the see the value of a variable of a rule online" in this chapter).

REMARK: This was done as some kind of coding and configuring HABPANEL exercise. You might get the same result in just checking the reading in a less frequent period, forget about the counter and just trigger the switch when the illumination reading is above or below the trigger value.

You can also use this Counter_Item to display some tendency whether your light is about to switch ON/OFF depending on the value of the Counter_Item on your dashboard. So if you play a little with the "knob" widget in HABPANEL it might look like this:



Then you add some other readings of your multissor, the wattage reading and trigger of your switch, a clock widget and you have a complete dashboard of your real weather and let it trigger your outside ligth.



Aussen

Temperatur Aussen 22.9°C

Nachtbeleuchtung OFF

Luftfeuchte Aussen 47%rh

Leistung Nachtbeleuchtung 0W

Helligkeit Aussen 151LUX

Helligkeit Delta

You can just add these lines at the bottom of your existing rule file or create a new file in the same folder

```
var Number loop_counter = 0


rule "check_illumination"
// using the loop_counter to ensure that it is
// 10 times in a row darker/lighter before triggering switch

when
// every x seconds "0/x" the value is checked
// 0/30 means every 30 sec the value is checked
      Time cron "0/30 * * ? * * *"
then
// reset loop_counter if required (counter outside -5 +5 range)
if (loop_counter >= -5 && loop_counter <= 5){
// <= 10 is defining the LUX trigger value when ligth is swiched ON/OFF
      if (MultiSens_Helligkeit.state <= 10) {
            if (loop_counter > -5) {
                  loop_counter = loop_counter -1
                  postUpdate(Counter_Item, loop_counter)
            }
            else {
                  if (DoubbleSwitch01_Relais1.state == OFF)
                  sendCommand (DoubbleSwitch01_Relais1, ON)
            }
      }
      else
            if (loop_counter < 5) {
                  loop_counter =loop_counter +1
                  postUpdate(Counter_Item, loop_counter)
            }
            else {
                  if (DoubbleSwitch01_Relais1.state == ON)
                  sendCommand (DoubbleSwitch01_Relais1, OFF)
            }
            }
else {
            loop_counter = 0
            }
end
```

And save the file