

Heim-Automatisierung

Am Anfang steht das Problem. Oder deren mehrere ;-)

- 1) Dachwohnung, Sommer, es wird sehr spät dunkel (Rolladen runter zum Einschlafen), sehr früh hell (Rolladen runter um nicht von der Sonne statt vom Wecker geweckt zu werden) und nachts soll das Schlafzimmerfenster und der Rolladen offen sein, damit die Hitze raus kann. Statt mehrmals aufzustehen und Zustände zu ändern: Automatisierung!
- 2) Die Heizungsthermostate sollen zeitlich regelbar werden, Energie sparen. Aber wenn man heimkommt solls schon warm sein... Ok, gibts auch "standalone", aber freie Tage einzeln an mehreren Heizkörpern programmieren?!?! Alles ruckzuck auf "ich-bin-nicht-da"-Sparprogramm stellen???
- 3) Die üblichen elektronischen Heizungsthermostate haben zwar eine (auf schnellen Temperaturabfall triggernde) "Fenster-offen"-Erkennung, aber die regelt nur eine zeitlang runter, dann wird wieder geheizt. Wenn das Fenster dann noch offen ist... :- (Ergo müssen die Fenster überwachbar werden. Nebenbei regelt das den Fall "Man ist unterwegs, es fängt an zu regnen und die bange Frage kommt: hab ich alle Fenster zugemacht???" Wenn man dann den Rolladen aus Punkt 1 von Ferne steuern kann, muss man keine ungeplante Heimfahrt auf Verdacht antreten.
- 4) Wenn man mal wieder vergessen hat, nach der Wäsche den Wasserhahn an der Waschmaschine zu schliessen und der Schlauch platzt, kanns teuer werden. Wenn erst derjenige in der Etage drunter merkt, was los ist, ist es längst zu spät. Da lässt sich doch was machen...?

Was geht, was geht nicht?

Erstes Kriterium: Eigenheim oder Mietobjekt? In einem Mietobjekt neue Strippen ziehen ist nicht ohne. Mindestens wenns gut aussehen soll. Stichwort WAF. In solchen Fällen helfen Funksysteme. Vorsicht: Sensoren mit Batteriebetrieb sind dann nicht beliebig abfragbar! Kabelgebundene Systeme haben da eindeutig einen Vorteil.

Es gibt fertige Systeme. Die haben oft die Einschränkung, dass man auf die Komponenten des Herstellers angewiesen ist. Die dann zu Mondpreisen zu haben sind. Oft haben sie den Vor/Nachteil (je nach Standpunkt), dass sie cloudbasiert arbeiten, mindestens aber eine Anbindung ans Internet haben, oftmals auch zwingend benötigen. Im Fall einer Insolvenz des Anbieters hat man dann einen Haufen Elektroschrott.

Achtung: Bzgl. der Sicherheit ist man hier auf Gedeih und Verderb den Herstellern ausgeliefert. Im Zweifelsfall sogar dann, wenn man am heimischen Router den Netzzugang per Firewall kontrolliert. Negativbeispiele finden sich alle paar Wochen in der Presse.

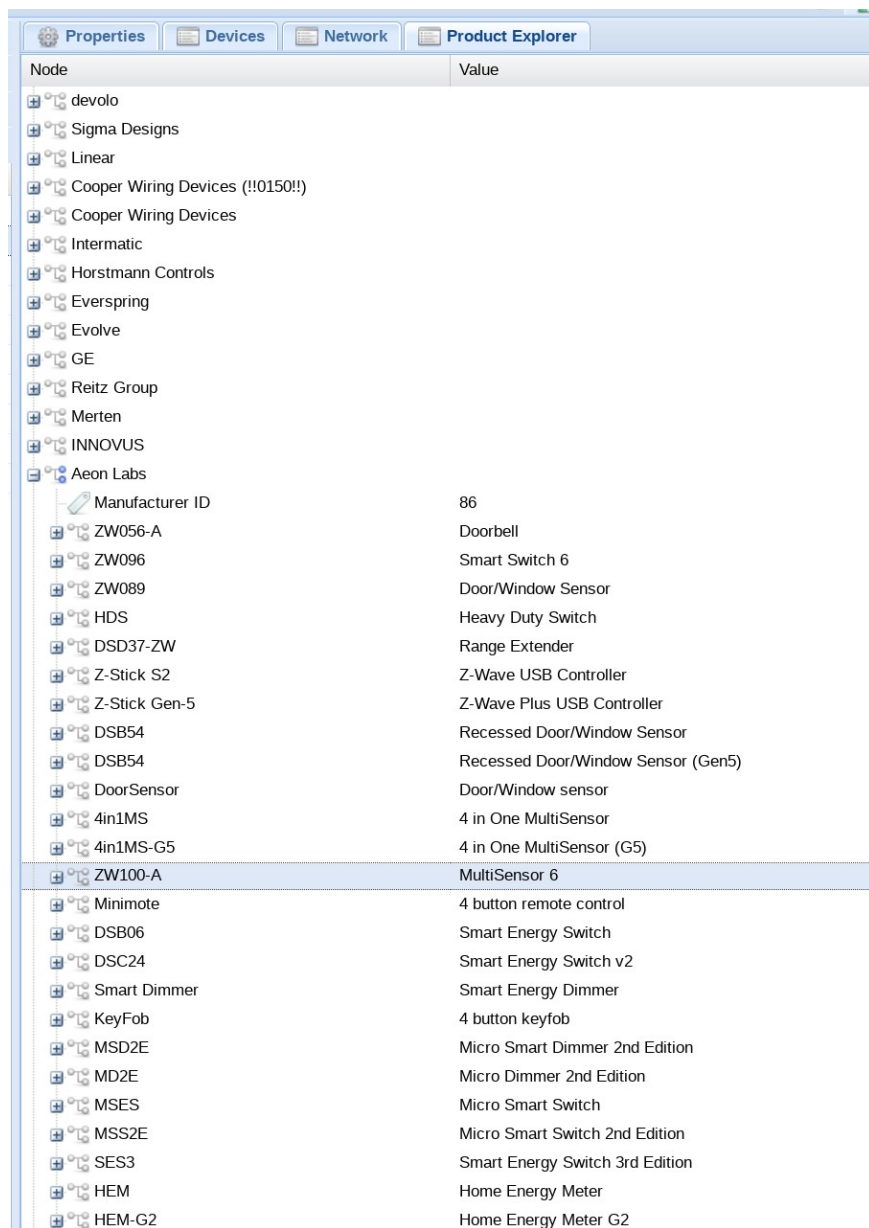
Eigenbau aus Komponenten: Man ist flexibel, kann Komponenten mehrerer verschiedener Hersteller einsetzen, so wie es benötigt wird. Nachteil: zeitintensiv!!! Und mit Forschungsarbeit verbunden, wenn man nicht beim Einkauf schon aufpasst.

Entscheidungen

Zur Steuerung habe ich mir OpenHAB ausgesucht. Da kann man an allen Ecken und Enden drehen und schrauben, hat damit auch mehr Möglichkeiten aber auch mehr Chancen, in einer Sackgasse zu landen. Es gibt alternative OpenSource-Software, die einfacher zu handhaben sein soll, aber auch einschränkt.

OpenHAB hat den Riesen-Vorteil, dass es Anbindungen an quasi alle möglichen und unmöglichen Geräte und Systeme gibt. Ob man kabelbasiert arbeiten will oder per Funk, ob man die FritzBox ansprechen will oder die Stereoanlage, ob man Webseiten auslesen will (Wetter, Sonnenstand...) oder eine Webcam auf dem eigenen Dashboard einbinden will, alles geht.

Hier mal ein Blick auf die nutzbaren Geräte eines einzigen Herstellers, und zwar nur die ZWave-basierten Geräte:



Node	Value
devolo	
Sigma Designs	
Linear	
Cooper Wiring Devices (!!0150!!)	
Cooper Wiring Devices	
Intermatic	
Horstmann Controls	
Everspring	
Evolve	
GE	
Reitz Group	
Merten	
INNOVUS	
Aeon Labs	
Manufacturer ID	86
ZW056-A	Doorbell
ZW096	Smart Switch 6
ZW089	Door/Window Sensor
HDS	Heavy Duty Switch
DSD37-ZW	Range Extender
Z-Stick S2	Z-Wave USB Controller
Z-Stick Gen-5	Z-Wave Plus USB Controller
DSB54	Recessed Door/Window Sensor
DSB54	Recessed Door/Window Sensor (Gen5)
DoorSensor	Door/Window sensor
4in1MS	4 in One MultiSensor
4in1MS-G5	4 in One MultiSensor (G5)
ZW100-A	MultiSensor 6
Minimote	4 button remote control
DSB06	Smart Energy Switch
DSC24	Smart Energy Switch v2
Smart Dimmer	Smart Energy Dimmer
KeyFob	4 button keyfob
MSD2E	Micro Smart Dimmer 2nd Edition
MD2E	Micro Dimmer 2nd Edition
MSES	Micro Smart Switch
MSS2E	Micro Smart Switch 2nd Edition
SES3	Smart Energy Switch 3rd Edition
HEM	Home Energy Meter
HEM-G2	Home Energy Meter G2

Die Liste ist ungefähr doppelt so lang, aber irgendwo hört der Bildschirm halt auf :-)

Es sind, über alle Hersteller hinweg, hunderte Sensoren und Aktoren für so ziemlich alles, was man braucht oder sich vorstellen kann.

Man nehme...

Einen Raspberry Pi (2er oder 3er-Generation)

Speicherkarte dazu

Netzteil dazu

Gehäuse dazu. Beim 3er-RasPi KEIN Metallgehäuse wenn man WLAN/Bluetooth nutzen will

Einen USB-Stick, der das Funknetz (ZWave) aufspannt, darüber die Daten sammelt und die Aktoren steuert.

Verschiedene Sensoren und Aktoren:

Ein Rollo-Funkrelais (und zur Elektrifizierung des Rolladens einen Rolladenmotor und für die manuelle Steuerung einen Wippenschalter dazu)

Ein Heizungsthermostat

Eine Funksteckdose

Eine Funk-Lampe, monochromatisch mit Dimmer (Projekt: RGB-Variante zur Simulation eines Sonnenaufgangs ;-)

Einen Flutsensor, den man im Bad auf den Boden stellt

Einen Multisensor für jedes Fenster und einen für die Wohnungstür

etc.

Erfahrungen

Vieles kann man im Netz nachlesen, die ein oder andere Erfahrung muss man selbst machen. Und manchmal mit Lehrgeld bezahlen :-/

Heizungsthermostat

Deutscher Hersteller (Comet), nutzt leider nicht die aktuelle Funktechnik (ZWave+), obwohl das Teil erst im Herbst 2015 auf den Markt kam und die Chips schon deutlich länger verfügbar sind. Ist nicht dokumentiert gewesen zum Zeitpunkt des Einkaufs. Ausserdem hat der Hersteller die Konfigurationsmöglichkeiten nicht dokumentiert. Inzwischen weiss ich, dass es eine Datenbank im Netz gibt, in der für alle bekannten Geräte in Bezug auf OpenHAB die Daten zur Verfügung stehen bzw. deren Fehlen ersichtlich ist.

Bei Thermostaten sieht es ohnehin recht mau aus. Es gibt wohl nur 3 Hersteller, die sich historisch auch eher auf den Bereich öffentliche Gebäude konzentriert haben und den Endverbrauchermarkt erst langsam entdecken. Habe jetzt von einem OEM (Devolvo) eine Version des Danfoss LC13, aber noch nicht montiert/eingebunden. Die Devolo-Variante funkt auch die Temperatur am Regler, anders als das Original.

Fenster-Offen-Kontakte/Multisensoren

Habe von DLink Sensoren im Einsatz. Neben "Kontakt offen/zu" melden sie auch die Helligkeit in Lux und die Temperatur. Dachte, das spart weitere Temperatursensoren. Problem: es wird die Temperatur am Befestigungspunkt gemessen, nicht die davor im Raum. Da die Dinger zwangsläufig wegen ihrer Primärfunktion an der Temperaturgrenze nach aussen hängen, messen sie im Sommer eine zu hohe und jetzt bei tieferen Aussentemperaturen eine zu tiefe Temperatur. Also für *Temperaturmessungen* nur im reinen Innenbereich (mit-)nutzbar. Wie z.B. bei mir an der Wohnungstür.

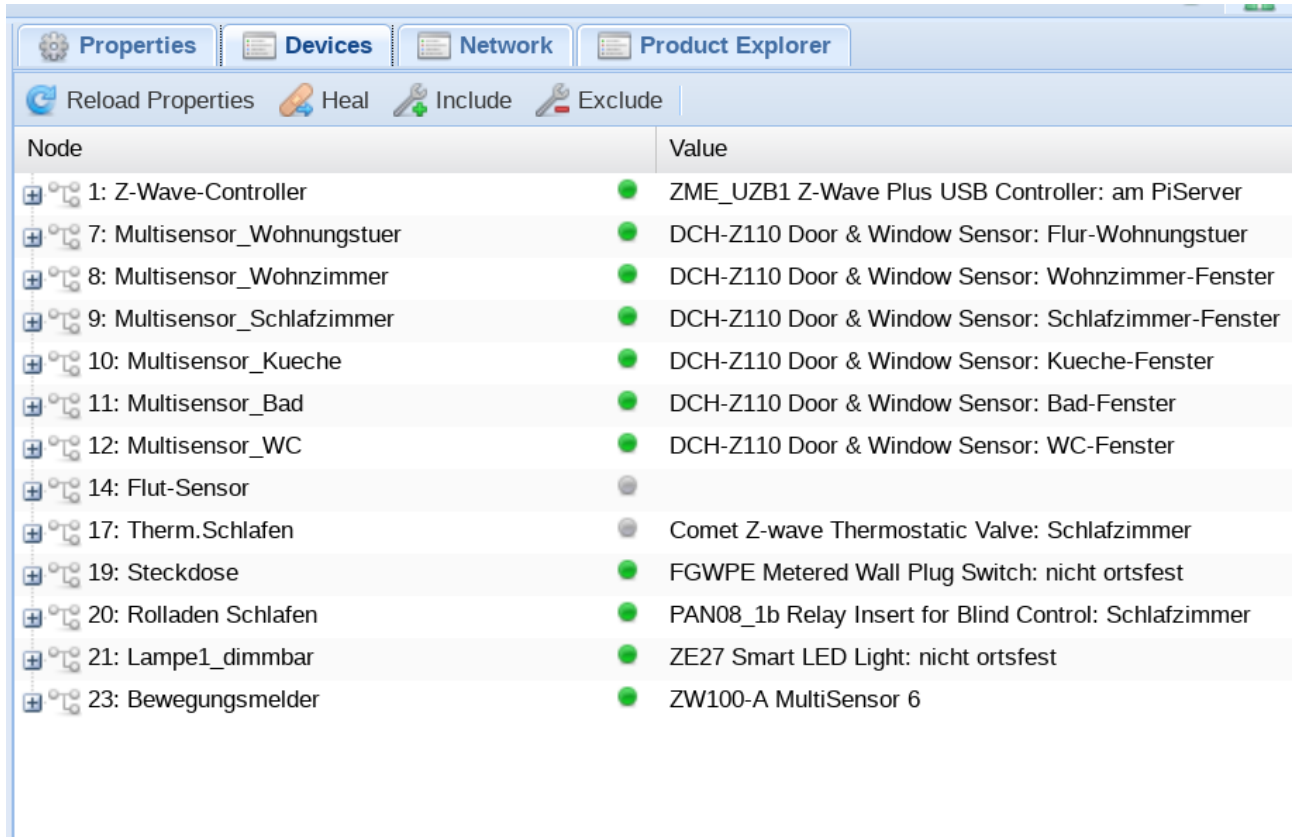
Es gibt die Teile auch mit Bewegungsmelder statt "Tür/Fenster"-Kontakt. Die wären dann für den Innenbereich als Temperaturfühler verwendbar. Da gibts aber Alternativen, die mehr können. Habe jetzt einen Aeon Multisensor 6 testweise in Betrieb genommen aber noch nicht fertig eingebunden. Mein Urlaub kommt ja erst noch ;-)

Batteriebetriebene Geräte

Bei batteriebetriebenen Geräten beim Einbinden unbedingt SOFORT die Konfiguration prüfen und korrigieren. Beim Flutsensor z.B. waren die initialen Parameter so unglücklich gesetzt, dass ich nach 4h Betrieb einen Batterielevel von 80% zu Gesicht bekam. WTF?!?! Das Gerät war schlichtweg nicht auf "leg dich schlafen" parametrisiert gewesen. Seitdem (4 Monate) sind nur weitere 5% Kapazität verbraucht worden. Schon besser :)

Was schon funktioniert

Eingebunden sind der Controller und ein Dutzend Sensoren und Aktoren. Bis auf den neu dazu gekommenen Bewegungsmelder ist das der Stand Mai, der ist im Wesentlichen seit damals im Einsatz:



Node	Value
1: Z-Wave-Controller	ZME_UZB1 Z-Wave Plus USB Controller: am PiServer
7: Multisensor_Wohnungstuer	DCH-Z110 Door & Window Sensor: Flur-Wohnungstuer
8: Multisensor_Wohnzimmer	DCH-Z110 Door & Window Sensor: Wohnzimmer-Fenster
9: Multisensor_Schlafzimmer	DCH-Z110 Door & Window Sensor: Schlafzimmer-Fenster
10: Multisensor_Kueche	DCH-Z110 Door & Window Sensor: Kueche-Fenster
11: Multisensor_Bad	DCH-Z110 Door & Window Sensor: Bad-Fenster
12: Multisensor_WC	DCH-Z110 Door & Window Sensor: WC-Fenster
14: Flut-Sensor	
17: Therm.Schlafen	Comet Z-wave Thermostatic Valve: Schlafzimmer
19: Steckdose	FGWPE Metered Wall Plug Switch: nicht ortsfest
20: Rolladen Schlafen	PAN08_1b Relay Insert for Blind Control: Schlafzimmer
21: Lampe1_dimmbar	ZE27 Smart LED Light: nicht ortsfest
23: Bewegungsmelder	ZW100-A MultiSensor 6

Was man hier sieht ist die Konfigurationsoberfläche für die Geräte selbst. Hier kann man z.B. einstellen, wie oft die Geräte "aufwachen" sollen (soweit sie batteriebetrieben sind) oder welche Farbe der Farbring an der Steckdose haben soll, wenn sie an ist etc.

Diese Teile sind dann logisch in einer Datei "items" definiert, die die Kopplung zwischen dem technischen Gerät und der Programmlogik herstellt.

Das sieht dann z.B. so aus (im Original eine Zeile):

```
Number Temp_Flur_Eingangstuer    "Flur Eingangstür [%0.1f °C]"          <temperature>
(gFlur)  { zwave="7:COMMAND=SENSOR_MULTILEVEL,SENSOR_TYPE=1" }
```

Das Item "Temp_Flur_Eingangstuer" ist numerisch, hat ein Textlabel (=Anzeige in der Oberfläche), bekommt in der GUI ein Icon namens "temperature", ist der Gruppe "gFlur" zugeordnet und hängt technisch am Gerät 7 (siehe Grafik oben). Genauer gesagt einem Sensor vom Typ 1 (=Temperatur), der viele verschiedene Werte liefern kann.

Die Oberfläche selbst ist in einer weiteren Datei "sitemap" definiert. Deren ganzer Inhalt sieht so aus:

```
sitemap default label="Unterm Dach"
{
    Frame {
        Text item=AktDatumZeit
        Text item=Anzeige_Sonnenzeiten
        Text item=AktDatumZeitString
    }
    Frame label="Status" {
        Group item=gDG {
            Group item=gFlur
            Group item=gWC
            Group item=gBad
            Group item=gKueche
            Group item=gSchlZi
            Group item=gWohnZi
        }
        Group item=gAlarm
        Group item=gTemp
        Group item=gFT
        Group item=gAkku
        Group item=gManip
    }
    Frame label="Steuerung" {
        Group item=gSteckdose1
        Switch item=Roll_Schlafen label="Rollladen Schlafen [(%d)]"
    }
    Frame label="noch im Testbetrieb" {
        Slider item=Lampe1
        Setpoint item=Solltemp_Schlafen_Thermostat minValue=5 maxValue=25 step=1
    }
    Frame label="Statistiken" {
        //period: the time span for the x-axis. Value can be h,4h,8h,12h,D,3D,W,2W,M,2M,4M,Y
        Chart item=gTemp period=2W refresh=30000
    }
}
```

Was dann zu diesem Dashboard wird:



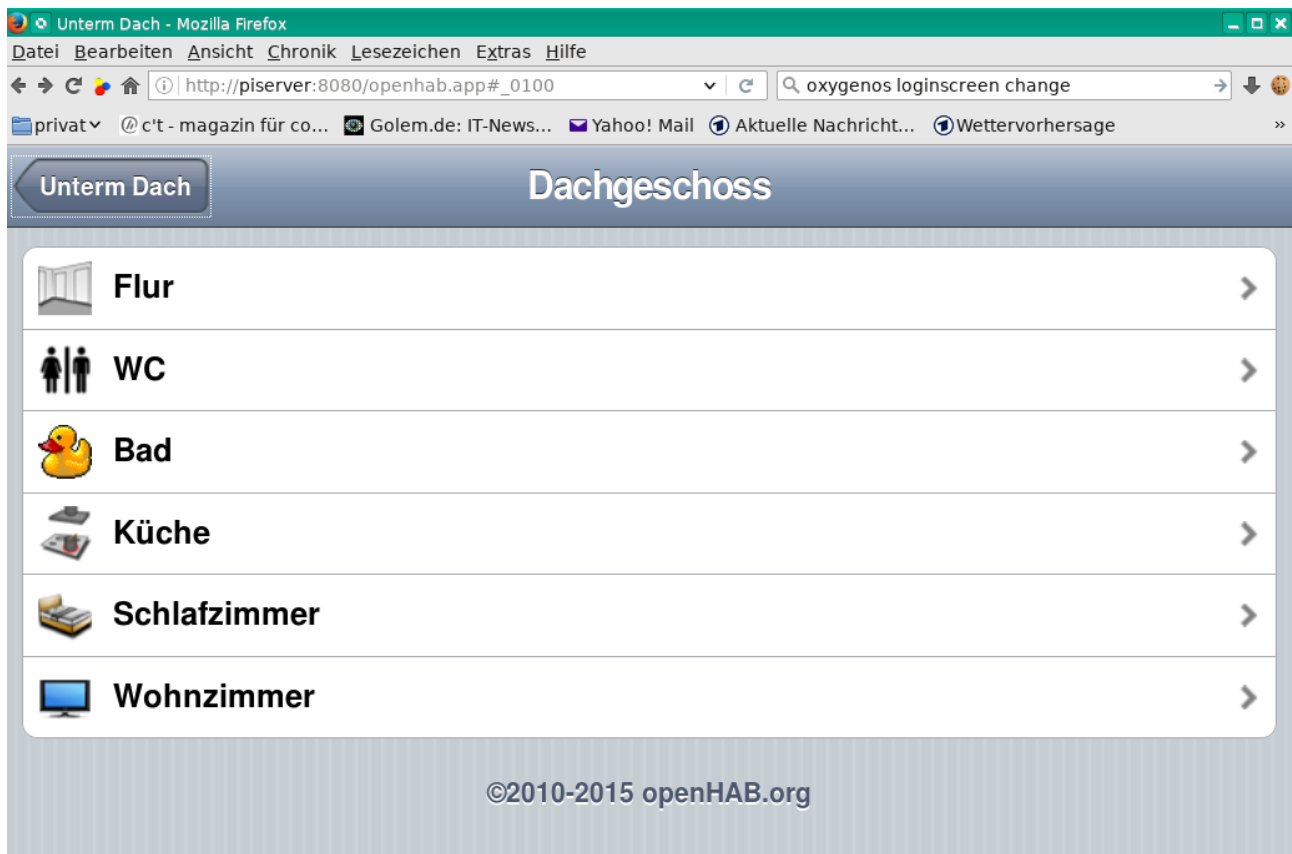
Oben sieht man das aktuelle Datum / die Zeit. Das wird alle paar Minuten aktualisiert, so dass man sehen kann, ob das System "lebt".

Darunter die vom geografischen Standort abhängigen Sonnenaufgangs- und Untergangszeiten. Mit denen wird der Rolladen im Schlafzimmer zeitabhängig gesteuert: 1h vor Sonnenuntergang runter, 1h danach wieder hoch und 1h vor Sonnenaufgang wieder runter. Und da bleibt er dann auch. Manchmal ist ja auch Wochenende und dann will ich ausschlafen. Noch werden die Wochentage nicht berücksichtigt. Die Steuerung hat diesen Sommer aber vielfaches Aufstehen und Rolladen runterlassen/hochziehen erspart, ich konnte durchschlafen. ROI erreicht ;-)

Unter "Dachgeschoss" versteckt sich die Hierarchie von Anzeigen der Sensoren nach Unterbringungsort. Ergo pro Zimmer. So hat man den Zustand einzelner Sensoren bzw. den Gesamtzustand eines Zimmers im Überblick.

Die Punkte "Temperaturen" bis "Manipulationen" sind Messwerte aller Sensoren, nach Funktion/Messobjekt zusammengestellt.

Beide Varianten sind rein durch Gruppenzugehörigkeiten in der "Items"-Datei definiert. Bei "Dachgeschoss" kommt noch eine Ebene (Übersicht der Räume) aus der "sitemap" dazu:



Beispiel: Im Schlafzimmer gibts einen Sensor und 2 Aktoren (die auch Sensorelemente mitbringen):

Dachgeschoss		Schlafzimmer		Home	
	Schlafen Fenster		20,0 °C		
	Schlafen Fenster		geschlossen		
	Schlafen		14 Lux		
	Schlafen Fenster		100 %		
	Schlafen Thermostat		24,5 °C		
	Schlafen Thermostat Status		0 %		
	Schlafen Thermostat Modus		1		
	Schlafen Soll-Temp.		17		
	Schlafen Thermostat		100 %		
	Schlafen, geschlossen zu:	0 %			

©2010-2015 openHAB.org

Bei der Anordnung der Elemente auf dem Dashboard bzw. in darunter liegenden Ebenen gibt es leider Einschränkungen. Nicht jedes Element lässt sich überall nutzen/verwenden. Daher fehlt hier z.B. die Temperatureinstellung des Heizungsthermostats. Das geht nur im Dashboard. Oder ich hab's noch nicht ganz durchschaut, kann in diesem speziellen Zusammenhang gut der Fall sein weil ich mich bislang eher oberflächlich mit der Thematik befasst habe. Das mit dem Rolladen war mir für den Sommer viiiiiiieel wichtiger ;-)

Ein nettes Feature ist die Möglichkeit, die Icons vom Zustand der Sensoren abhängig zu machen. Das macht OpenHAB automatisch, wenn es entsprechend benannte Grafiken gibt.



Das wirkt sich z.B. bei der "Fenster/Türen"-Gruppe bis aufs Dashboard aus: wenn nur ein einziges Fenster offen ist, wird dort ein geöffnetes Fenster angezeigt. Sind alle zu, ein geschlossenes.

Analog ist das bei der "Alarme"-Gruppe: ein einziger aktiver Alarm färbt das Icon orange ein. Dann kann man reinklicken und sich die Ursache anzeigen lassen. Ansonsten (Icon ist grau) weiss man schon, dass alles ok ist.

Im Bereich "Steuerung" sind die nutzbaren Aktoren ansprechbar. Hier kann die Steckdose geschaltet und deren Stromverbrauch überwacht sowie der Rolladen "manuell" gesteuert werden (vom echten Handbetrieb per Schalter im Schlafzimmer abgesehen).

Darunter Sachen, die noch nicht ganz so funktionieren, wie ich das will.

Die Zeile "Chart item=gTemp period=2W refresh=30000" in der sitemap produziert auf einfachste Weise die Grafik ganz unten. Gezeichnet werden alle Mitglieder der Gruppe "gTemp", und zwar für einen Zeitraum von 2 Wochen zurück. Die Skalierung, Farbgebung, Beschriftung etc. passiert automatisch.

Alle Werte werden übrigens gespeichert. Beim Hochfahren von OpenHAB bekommen die Items den Wert, den sie vor dem Shutdown zuletzt hatten. Der wird dann aktualisiert, wenn ein Sensor den nächsten neuen Wert liefert. Wenn das, z.B. bei batteriebetriebenen Sensoren, ein paar Stunden dauert, stimmt die Anzeige ggf. nicht.

Fernzugriff

Ein wesentlicher Aspekt war ja auch die Möglichkeit, nicht nur vor Ort kontrollieren und steuern zu können. Um von aussen zugreifen zu können, gibt es folgende Wege:

1) Die Webseite der eigenen OpenHAB-Installation von aussen zugänglich zu machen. Also ein Port-Forwarding auf dem eigenen Router, unter der Voraussetzung, dass der eigene Anschluss über einen DynDNS unter einer URL aufrufbar ist.

Achtung: Damit in einem solchen Fall kein Fremder <http://blinkenlights.net/> spielt, muss für einen effektiven Passwortschutz der Installation gesorgt werden!!!

Die Konfiguration auf einen "schrägen" Port anstelle von 80 oder 8080 ist KEIN Schutz!

In dieser Variante muss man der OpenHAB-Community vertrauen, dass niemand am Passwortschutz vorbeikommt.

2) Analog 1), aber statt eines (beliebigen) Browsers kann eine App "HAB-Droid" verwendet werden.

Es gelten die selben Sicherheitshinweise wie bei 1)

3) Die Webseite der eigenen OpenHAB-Installation NICHT nach aussen freigeben, aber das interne Netz quasi nach aussen verlängern: via VPN-Zugriff.

Sinnvollerweise nutzt man aber auch in diesem Konstrukt die Userverwaltung von OpenHAB.

Sicherheitstechnisch ist man in diesem Fall von der Vertrauenswürdigkeit der VPN-

Implementierung abhängig. Die Userverwaltung zum Schutz des Zugriffs auf OpenHAB ist im Fall eines Eindringens von Fremden ins Netz dann eh das kleinere Problem.