

Vermittler im smarten Zuhause

Teil 2

FHEM – Freundliche Hausautomation und Energie-Messung

Energiesparen ist angesagter denn je. Dafür ist es notwendig, den eigenen Energieverbrauch zu kennen und die Werte im Blick zu behalten. Der Aufbau von Smart Homes (oder auch smarten Wohnungen) boomt und damit auch der Einsatz einer kleinen Smart Home Zentrale. Auf dem schnell wachsenden Markt der Haussteuerungen tummeln sich immer mehr Softwareprodukte, und es ist nicht leicht, sich zu entscheiden. Eine Hilfestellung will diese mehrteilige Artikelserie bieten: Nach dem Start im ELVjournal 3/2022 mit dem ioBroker ist nun die Software FHEM an der Reihe.



FHEM
Grundriss

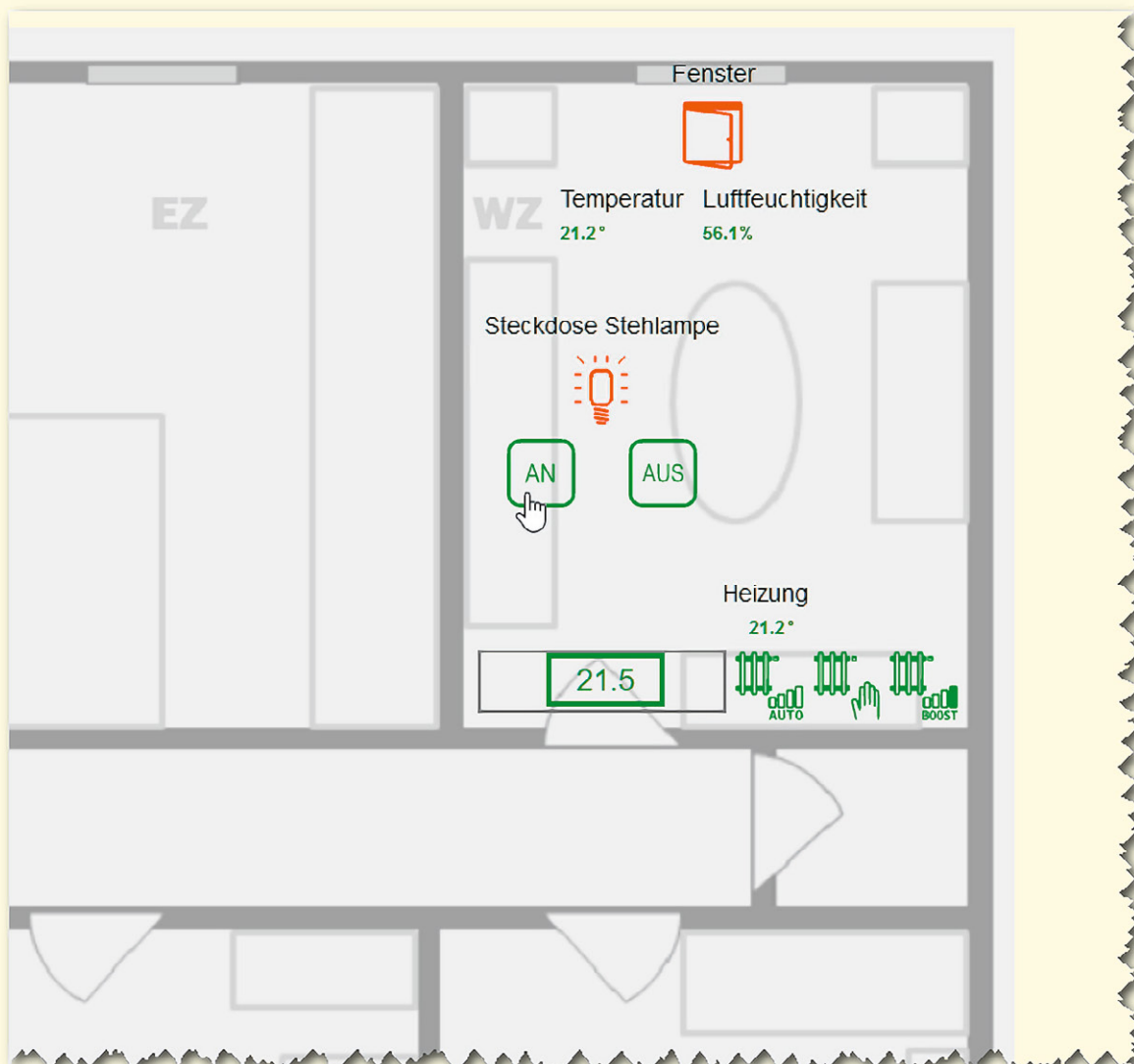


Bild 1: Zahlreiche Dokumentationen findet man auf der Internetseite von FHEM.

Freundliches Haus

FHEM ist ein Akronym und steht für „Freundliche Hausautomation und Energie-Messung“. Es wird ohne h wie bei feminin ausgesprochen und war eine der ersten Smart-Home-Vermittlersoftware bzw. -Broker am Markt. FHEM ist eine Open-Source-Lösung und ein unter GPL lizenzierter Server für die Heimautomatisierung. In der Programmiersprache Perl geschrieben, wurde die Software von Rudolf König programmiert und wird bis heute weiterentwickelt. FHEM unterstützt Hunderte verschiedene Gerätetypen in einer gemeinsamen Anwendung und Visualisierung. Durch die Vielzahl von Modulen und einer Nutzergemeinschaft von über 20.000 Anwendern kann man mit nahezu allen Smart-Home-Systemen interagieren. Ein Ende 2015 gegründeter, eingetragener gemeinnütziger Verein sichert nachhaltig den Fortbestand der Software und stellt die Webseiten, das Forum und das Wiki zur Verfügung. Auf den Internetseiten von FHEM (Bild 1), [1] finden Sie zahlreiche Informationen, Dokumentationen und Links zum Verein sowie ein umfangreiches Wiki. Zusätzliche interessante Beispiele und viele Anwendungen und Erläuterungen gibt es in dem 2019 erschienen Buch von Peter A. Henning „Smart Home mit FHEM“ [2].

Die Hardware-Komponenten

In der Artikelserie kommt für die verschiedenen Broker-Programme immer ein kostengünstiger Raspberry Pi 3B+ (Bild 2 1) als Zentrale zum Einsatz. Auf der Endgeräteseite nutzen wir Geräte aus dem Programm von Homematic/Homematic IP. Wir haben uns für folgende Homematic Komponenten entschieden:

- Schalt-Mess-Steckdose HmIP-PSM 2
- Zählersensor-Sendeeinheit Strom/Gas HM-ES-TX-WM 3
- Energiesensor für Smart Meter ES-IEC 4
- Temperatur- und Luftfeuchtigkeits-Sensor HmIP-STH 5
- Fenster- und Türkontakt HmIP-SWDO-I 6
- Heizkörperthermostat HmIP-eTRV 7

Eine ausführliche Beschreibung der einzelnen Geräte finden Sie in Teil 1 dieser Beitragsreihe. Den zugehörigen Fachbeitrag finden Sie als Download auf den Seiten des ELVjournals im Internet [3]. Alle Geräte sind an einer CCU3 angelernt, und wir bleiben bei unserem sogenannten Schreibtisch-Aufbau für die Vorstellung dieses und der weiteren Broker-Systeme. Bevor wir uns nun mit der Installation von FHEM befassen, setzen wir noch voraus, dass sowohl die CCU3 als auch alle Geräte mit der aktuellen Firmware ausgestattet sind.

Für unsere CCU3 nutzen wir die Version 3.61.7, und bei den Geräten sind es jeweils die im März 2022 aktuellen Versionen. Beim Raspberry Pi ist die OS-Version Raspberry Pi OS Lite (Release-Datum 28.01.2022 [4]) auf einer 16-GB-SD-Karte vorinstalliert. Über einen SSH-Client (hier empfiehlt sich das Programm PuTTY [5]) greifen wir auf den Minirechner per Kommandozeile zu. Über die Befehle "sudo apt update" und

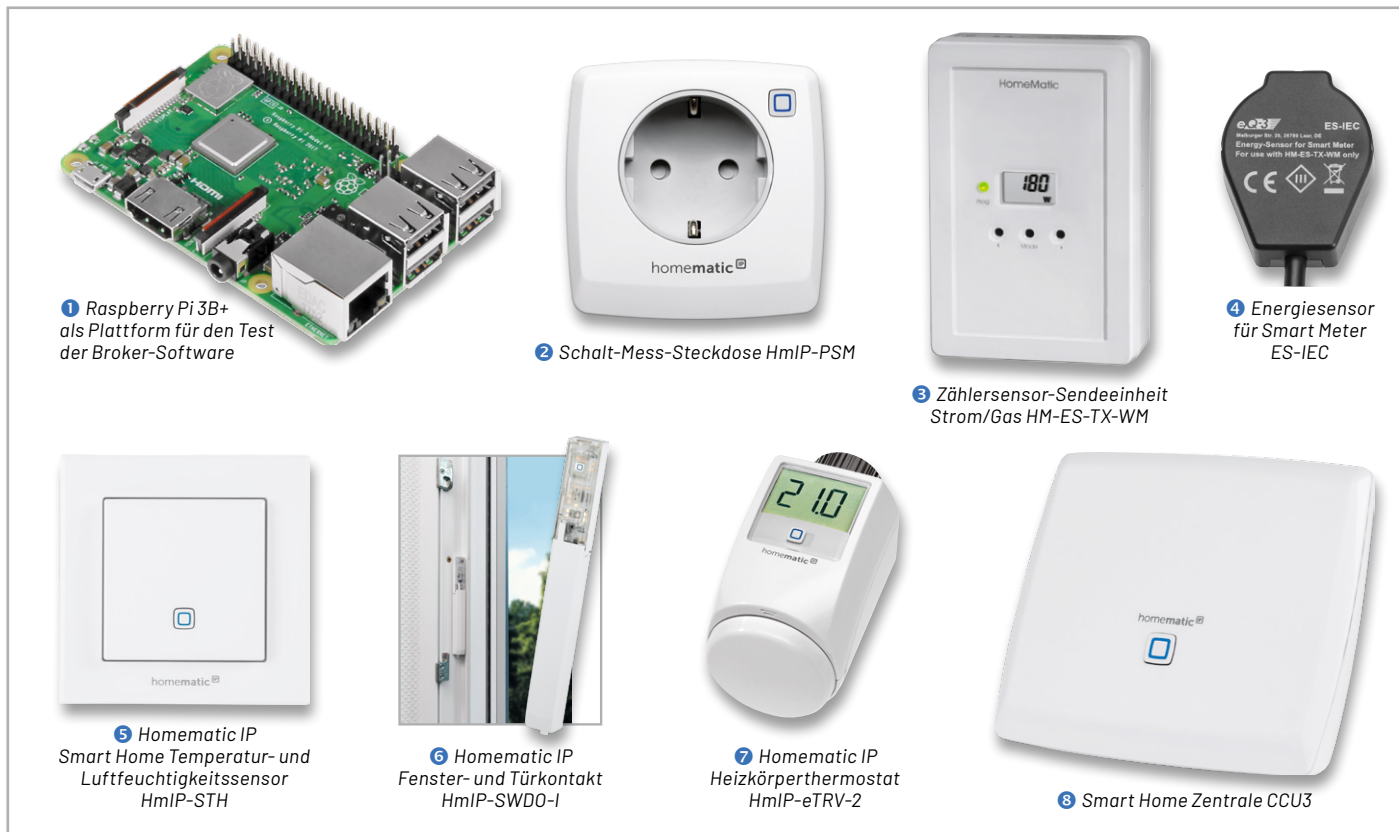


Bild 2: Überblick über alle Hardware-Komponenten für die Haussteuerungslösung

"sudo apt full-upgrade" (Bild 3) werden nach der Grundinstallation vorab alle vorhandenen Pakete aktualisiert, und nach einem Neustart über "sudo reboot" ist der Raspberry Pi auf dem aktuellen Stand.

Jetzt geht es los – FHEM wird installiert

Die Installation auf einem Raspberry Pi ist etwas knifflig, und es bietet sich folgende Vorgehensweise an: Zunächst erfolgt die Grundinstallation des Raspberry Pi wie in Teil 1 [3] beschrieben. Da die Hausautomatisierungssoftware FHEM bislang in den Paketquellen von Raspbian fehlt, muss sie manuell auf einem bestehenden System eingespielt werden (Bild 4).

```
pi@raspberrypi:~$ sudo apt update
pi@raspberrypi:~$ sudo apt full-upgrade
pi@raspberrypi:~$ sudo shutdown -r now
oder
pi@raspberrypi:~$ sudo reboot
```

Bild 3: Vorbereiten des Raspberry Pi für die Installation von FHEM

FHEM

The easy way: Use apt to install FHEM and all dependencies

- All steps have to be run as root or with appropriate sudo privileges.
- Import repository gpg key:


```
wget -O- https://debian.fhem.de/archive.key | gpg --dearmor | sudo tee /usr/share/keyrings/debianfhemde-archive-keyring.gpg
```
- Add repository to /etc/apt/sources.list:


```
deb [signed-by=/usr/share/keyrings/debianfhemde-archive-keyring.gpg] https://debian.fhem.de/nightly/ /
```
- Update your package administration:


```
apt update
```
- Install fhem:


```
apt install fhem
```
- You will find a special version number like


```
Inst fhem (6.1.26091 debian.fhem.de [all])
```

 with 6.1 representing the major version
with 26091 representing the latest SVN commit included.

Content:
Start
The easy way: use apt
Manual installation
[https Trouble?](https://debian.fhem.de)
[gpg key Trouble?](#)

Nightly build:
26.05.22 ==TIME==
Rev 6.1.26091

Links:
[FHEM Home](#)
[FHEM Forum](#)
[FHEM Wiki](#)
[FHEM Verein](#)

Bild 4: FHEM-Installationsseite für Debian/Raspbian

Zunächst verbinden Sie sich über PuTTY mit dem Raspberry Pi – das Standard-Passwort (falls Sie es noch nicht geändert haben) lautet „raspberrypi“. Dann wird zunächst ein gpg-Key für das Repository importiert:

```
wget -O- https://debian.fhem.de/archive.key | gpg --dearmor | sudo tee /usr/share/keyrings/ ↵
debianfhemde-archive-keyring.gpg
```

Danach muss das Repository durch einen Eintrag in der Datei /etc/apt/sources.list hinzugefügt werden. Dies geschieht durch die Erweiterung der Datei um folgenden Eintrag:

```
deb[signed-by=/usr/share/keyrings/debianfhemde-archive-keyring.gpg] https://debian.fhem.de/nightly/ /
```

```
pi@raspberrypi:~ $ sudo nano sources.list

pi@raspberrypi:~ $ sudo apt update
Hit:1 http://raspbian.raspberrypi.org/raspbian bullseye InRelease
Hit:2 http://archive.raspberrypi.org/debian bullseye InRelease
Get:3 https://debian.fhem.de/nightly InRelease [1,912 B]
Get:4 https://debian.fhem.de/nightly Packages [973 B]
Fetched 2,885 B in 1s (2,116 B/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
All packages are up to date.

pi@raspberrypi:~ $ sudo apt install fhem
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Suggested packages:
  libarchive-extract-perl libarchive-zip-perl
The following NEW packages will be installed:
  fhem
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 25.6 MB of archives.
After this operation, 0 B of additional disk space will be used.
Get:1 https://debian.fhem.de/nightly fhem 6.1.26091 [25.6 MB]
Fetched 25.6 MB in 4s (7,061 kB/s)
Selecting previously unselected package fhem.
(Reading database ... 43662 files and directories currently installed.)
Preparing to unpack .../fhem_6.1.26091_all.deb ...
Unpacking fhem (6.1.26091) ...
Setting up fhem (6.1.26091) ...
creating user fhem
Created symlink /etc/systemd/system/multi-user.target.wants/fhem.service → /etc/systemd/system/fhem.service.

pi@raspberrypi:~ $ sudo service fhem start
```

Bild 5: Befehle zur Installation von FHEM

Dieser Eintrag muss mit Root-Rechten erfolgen – für die Ergänzung der Datei bietet sich der Editor nano an. Bei Problemen mit der Installation gibt es auf den FHEM-Seiten einige Troubleshooting-Tipps [6]. Danach wird zunächst erneut ein

```
sudo apt update
```

durchgeführt und anschließend mit

```
sudo apt install fhem
```

die Installation gestartet. Den Abschluss (Bild 5) bildet der Start des FHEM-Servers über

```
sudo service fhem start
```

Nach durchgeführter Installation und erfolgreichem Start des Dienstes ist die Laufzeitumgebung von Perl mit der IP-Adresse des Raspberry Pi und dem Port 8083 über einen Webbrowser erreichbar

(Bild 6). Auf der gelb-grünen Weboberfläche sehen Sie ganz oben das zentrale Element in FHEM: die Kommandozeile für die Definitionen und Befehle. Das grüne Pluszeichen links davon öffnet ein größeres Eingabefenster für mehrzeilige Anweisungen. Über das kleine FHEM-Logo (ein lächelndes Haus) links oben kommen Sie aus allen Unterseiten immer wieder zurück zur Startseite des Servers.

Die Meldung „FHEM:SecurityCheck:“ bedeutet, dass für FHEM noch kein Passwort festgelegt wurde. Mit dem vorgeschlagenen Befehl „attr global motd none“ werden Sie diese Meldung los. Das Menü in der linken Spalte weist danach mit einem kleinen roten Fragezeichen neben dem Link „Save config“ auf noch nicht gespeicherte Konfigurationsänderungen hin. Nach dem nächsten Systemstart oder über den Befehl „rereadcfg“ wird die fhem.cfg eingelesen (Bild 7).

Grundsätzlich lässt sich jedoch recht einfach ein Passwort für FHEM über diese Definition vergeben:

```
attr WEB basicAuth <benutzername:passwort>
```

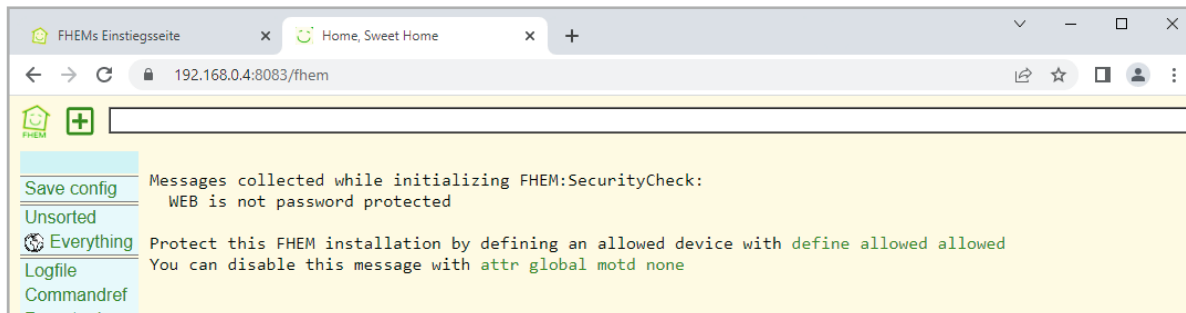


Bild 6: FHEM-Startseite nach erfolgreicher Installation

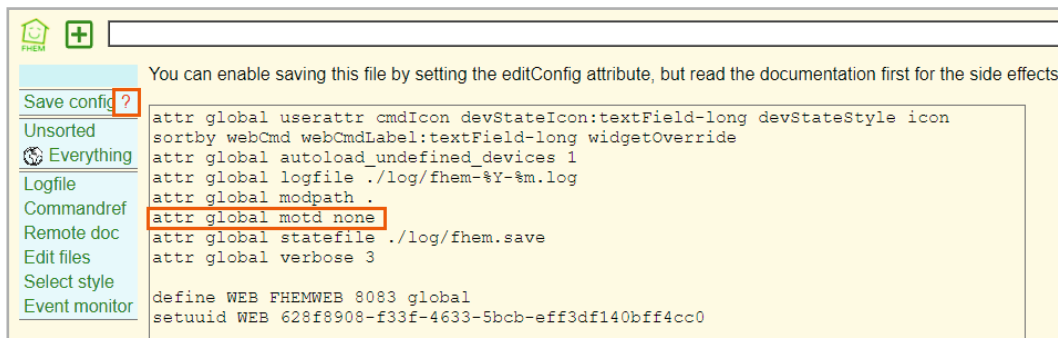


Bild 7: Die FHEM-Config-Datei

Die Kombination <benutzername:passwort> muss base64-kodiert sein und kann über PuTTY in der RaspberryPi-Konsole mit folgender Anweisung erzeugt werden:

```
echo -n MeinBenutzername:MeinPasswort | base64
```

Der Rückgabewert (z. B. TWVpbkJIbnV0emVybmFtZTpNZWluUGFzc3dvcnQ=) wird dann in der obigen FHEM-Anweisung als Benutzername-Passwort-Kombination verwendet.

Der mittlere Menüblock (Bild 6/7, links) zeigt die Strukturierung der Hausautomatisierungszentrale. FHEM bezeichnet sie als Räume, dies können tatsächliche Räume im Smart Home sein oder auch Funktionsgruppen innerhalb der Zentrale. Die Links „Unsorted“ und „Everything“ sind voreingestellte Räume für noch nicht zugewiesene oder für globale Elemente. Über „attr <Gerätename> room <Raumname>“ wird beispielsweise ein neuer Raum eingefügt. Dazu braucht man zuerst jedoch die Bezeichnung des Geräts, das man in diesem Raum hinzufügen möchte. Das ist der Name, mit dem das Gerät über einen „define“-Befehl hinzugefügt wurde.

Der untere Menüblock (Bild 7) zeigt links die häufig benötigten Funktionen oder Systemwerkzeuge:

- das Systemprotokoll (Logfile)
- eine Befehlsreferenz (Commandref)
- einen Link zur FHEM-Dokumentationsseite [9] (Remote doc)
- eine Liste mit editierbaren Dateien (Edit files)
- eine Auswahl der Darstellung (Select style)
- den Ereignismonitor (Event monitor)

CCU anbinden und Geräte hinzufügen

Um unsere Smart Home Zentrale CCU3 mit FHEM zu verbinden, muss das Modul HMCCU installiert werden. Es ermöglicht zusammen mit weiteren Modulen (HMCCUDEV, HMCCUCHN und HMCCUPROCRPC) eine Integration der CCU3 sowie der dort angelegten Geräte in FHEM. Die einzelnen Module haben folgende Aufgaben:

- HMCCU: Kommunikation zwischen FHEM und CCU3 (I/O-Device)
- HMCCUDEV: Definition von FHEM-Devices für Homematic Geräte
- HMCCUCHN: Definition von FHEM-Devices für einzelne Kanäle von Homematic Geräten
- HMCCURPCPROC: RPC-Server zum Empfang von Events der CCU3
- HMCCUConf: Templates mit Default-Attributen für bestimmte Homematic Gerätetypen

Die an der CCU3 angelegten Geräte haben einen oder mehrere Kanäle. Ein Kanal wiederum hat einen oder mehrere Datenpunkte. Über diese Datenpunkte kann ein Gerät gesteuert oder es können Statusinformationen ausgelesen werden. Sofern ein Datenpunkt lesbar ist, kann er als Reading in einem HMCCUCHN- oder HMCCUDEV-Device dargestellt werden. Ausgehend von dieser Abbildung in der CCU3 kann man nun entscheiden, ob man ein CCU3-Gerät über HMCCUCHN oder HMCCUDEV definiert.

Es gibt jedoch einiges zu beachten: In den CCU3-Namen dürfen keine Umlaute verwendet werden. Leerzeichen sind zulässig, können aber bei einigen Funktionen zu Problemen führen. Die in der CCU3 vergebenen Namen müssen über alle Objekttypen hinweg eindeutig sein. Ein Gerät und ein Raum dürfen also nicht den gleichen Namen haben. Nach einem Neustart der CCU3 muss auch der RPC-Server neu gestartet werden, da die CCU3 bei einem Neustart die registrierten RPC-Server „vergisst“.

Auch bei Änderungen an Devices oder von Attributen empfiehlt sich ein Neustart. Vor der Ausführung eines FHEM-Updates oder vor der Installation eines CCU3-Firmware-Updates muss der RPC-Server gestoppt werden. Seit der CCU3-Firmware 2.27.7 sind die Zugriffsmöglichkeiten auf die CCU3 durch eine Firewall abgesichert, sodass der Zugriff durch FHEM explizit freigeschaltet werden muss. Dazu im CCU3-Menü „Einstellungen – Systemsteuerung“ den Button „Firewall konfigurieren“ anklicken und bei „Firewall-Richtlinie“ die Auswahl „Ports offen“ wählen (oder die notwendigen Ports 2001 und 2010 im Feld „Port-Freigabe“ explizit angeben). Die Rechte für „HomeMatic XML-RPC API“ und „Remote HomeMatic-Script API“ sind auf „Vollzugriff“ zu setzen. Oder Sie können die Rechte auf „Eingeschränkter Zugriff“ belassen und dann im Feld „IP-Adressen für eingeschränkten Zugriff“ die IP des FHEM-Servers oder das komplette Subnetz in Subnet-Notation angeben.

```

pi@raspberrypi:~ $ sudo apt-get update && sudo apt-get install -y librpc-xml-perl
pi@raspberrypi:~ $ sudo apt update
pi@raspberrypi:~ $ sudo apt full-upgrade
pi@raspberrypi:~ $ sudo reboot
    
```

Bild 8: Nachinstallationen für das Modul HMCCU

Wenn der Zugriff von FHEM auf die CCU3 ohne Anmeldung erfolgen soll, muss zusätzlich in der Systemsteuerung unter „Sicherheit“ die Option „Authentifizierung“ deaktiviert werden.

Alle HMCCU-Module sind ein Teil von FHEM, und HMCCU benötigt die Perl Module RPC::XML::Server und RPC::XML::Client. Über PuTTY muss auf dem Raspberry Pi dafür das Paket librpc-xml-perl nachinstalliert werden:

```
sudo apt-get update && sudo apt-get install -y librpc-xml-perl
```

Danach wie gewohnt über die Befehle "sudo apt update" und "sudo apt full-upgrade" alle vorhandenen Pakete aktualisieren und den Raspberry Pi über "sudo reboot" neu starten (Bild 8).

Im nächsten Schritt wird in FHEM ein I/O-Device angelegt, das für die Kommunikation zwischen FHEM und der CCU3 verantwortlich ist. Das FHEM-Device bekommt den Namen „d_ccu“ und die IP-Adresse der CCU3:

```
define d_ccu HMCCU 192.168.0.100
```

Falls die CCU3 als Software-Service auf demselben Rechner läuft wie FHEM, sollte bei der Definition der Parameter "ccudelay" angegeben werden. Die CCU-Services brauchen beim Starten länger als FHEM (bis zu 120 Sekunden). Durch den Parameter ccudelay wird das Device um die angegebene Anzahl Sekunden ver-

The screenshot shows the FHEM web interface for a CCU3 device. The 'Homematic' tab is selected. The configuration includes:

- set d_ccu: ackmessages
- get d_ccu: ccuConfig

The 'Internals' section shows the following parameters:

CCUNum	1
Clients	:HMCCUDEV:HMCCUCHN:HMCCURPCPROC:
DEF	192.168.0.100
FUUID	62936353-f33f-4633-eafa-e5d8e36043d1f3af
NAME	d_ccu
NOTIFYDEV	global
NR	14
NTFY_ORDER	50-d_ccu
RPCState	inactive
STATE	Initialized
TYPE	HMCCU
ccuip	192.168.0.100
ccustate	unreachable
ccutype	CCU2/3
config	5.0
host	192.168.0.100
prot	http
version	5.0 220431743

The 'Readings' section shows the following data:

PLATFORM	rpi3	2022-05-29 13:50:06
PRODUCT	ccu3	2022-05-29 13:50:06
VERSION	3.63.8	2022-05-29 13:50:06
count_channels	133	2022-05-29 13:50:06
count_devices	7	2022-05-29 13:50:06
count_groups	0	2022-05-29 13:50:06
count_interfaces	3	2022-05-29 13:50:06
count_programs	2	2022-05-29 13:50:06
rpcstate	inactive	2022-05-29 13:50:19
state	Initialized	2022-05-29 13:50:19

The 'Attributes' section shows:

room	Homematic	deleteattr
stateFormat	rpcstate/state	deleteattr

At the bottom, there are navigation options: Select icon, Extend devStateIcon, Raw definition, Delete this device (d_ccu), and Device specific help.

Bild 9: CCU-Device in FHEM

zögert initialisiert. Je nach Installationsvariante kann auch ein höherer Wert als 180 Sekunden notwendig sein:

```
define d_ccu HMCCU 192.168.0.100 ccudelay=180
```

Im nächsten Schritt muss der RPC-Server konfiguriert und gestartet werden. Zunächst werden mit dem Attribut "rpcinterfaces" die Schnittstellen bzw. Ports festgelegt, für die sich der RPC-Server bei der CCU registrieren soll. Das Attribut zeigt in einer Auswahlliste die möglichen Schnittstellen an. Nur für die hier ausgewählten Schnittstellen werden automatisch Readings aktualisiert. Beim ersten Start dieses RPC-Servers über das I/O-Device wird automatisch je Schnittstelle ein Device vom Typ HMCCURPCPROC angelegt. Die Attribute „room“ und „group“ werden vom I/O-Device übernommen (Bild 9).

Nun kann der RPC-Server gestartet werden. Dabei wird je RPC-Schnittstelle ein separater fhem.pl-Prozess mit eigenem Listening-Port gestartet. Beim ersten Start legt HMCCU für jede unter rpcinterfaces angegebene Schnittstelle ein Device vom Typ HMCCURPCPROC an. Alle RPC-Server werden über das I/O-Device mit einem Befehl gestartet:

```
set d_ccu on
```

Während des Starts der RPC-Server und der Registrierung bei der CCU kann das I/O-Device nur eingeschränkt verwendet werden. Dies wird durch den Status „starting/busy“ angezeigt. Nachdem die RPC-Server gestartet wurden, wechselt der Status zu „running/OK“. Anschließend sollte man noch das Attribut rpcserver auf „on“ setzen, damit die RPC-Server beim Start von FHEM automatisch gestartet werden:

```
attr d_ccu rpcserver on
```

Danach finden Sie im Raum „Unsorted“ das Modul HMCCURPCPROC und die RPC-Instanzen (Bild 10).

Nun geht es darum, die Homematic Geräte in FHEM anzulegen. Das Erzeugen der Geräte kann über Einzelbefehle:

```
get d_ccu createDev „HmIP-PSM Wohnzimmer“
get d_ccu createDev „HmIP-STH Wohnzimmer“
get d_ccu createDev „HmIP-SWDO-I Wohnzimmer“
get d_ccu createDev „HmIP-eTRV Wohnzimmer“
```

oder in unserem Fall über einen Sammelbefehl erfolgen:

```
get d_ccu create HmIP-* room=Wohnzimmer
```

Grundsätzlich ist es vorab wichtig, in der CCU3 sinnvolle, eindeutige Gerätenamen (z. B. HmIP-PSM Wohnzimmer) zu vergeben und vor allem keine Umlaute im Namen zu verwenden, da die auf dem Weg zu

FHEMWEB	
Save config	WEB Initialized
Homematic	
Unsorted	FileLog
Everything	Logfile fhem-2022-05.log text
Logfile	Global
Commandref	global no definition
Remote doc	
Edit files	HMCCURPCPROC
Select style	CCU 001123 RPC BidCos-RF running/OK
Event monitor	CCU 001123 RPC HmIP-RF running/OK
	CCU 001123 RPC VirtualDevices running/OK
autocreate	
autocreate	active
eventTypes	
eventTypes	active

Bild 10: Modul HMCCURPCPROC in FHEM

FHEM verloren gehen. Danach werden die Schaltmess-Steckdose HmIP-PSM, der Temperatur- und Luftfeuchtigkeits-Sensor HmIP-STH, der Fenster- und Türkontakt HmIP-SWDO-I und der Heizkörperthermostat HmIP-eTRV-2 im Raum Wohnzimmer angelegt (Bild 11).

Die Verbindung zu weiteren Smart-Home-Steuern erfolgt über zusätzliche Module – die Auswahl ist hier riesig und bietet die Anbindung nahezu aller gängigen Systeme.

Events und der Manager

Die Ereignisse sind neben den Geräten (Devices) das Grundkonzept in FHEM. Es sind Systemnachrichten, die Informationen über Änderungen von Zuständen enthalten und auch von allen anderen Devices gelesen werden. Über den Ereignismonitor – im Menüblock links ist das der letzte Link – sehen Sie, welche Ereignisse gerade durchlaufen werden (Bild 12 zeigt einen Ausschnitt).

FHEM	
Save config	Fenster
Dashboard	HmIP_SWDO_I_Wohnzimmer
Floorplans	
Homematic	HMCCUCHN
Unsorted	HmIP_STH_Wohnzimmer 23.5 20.0
Wohnzimmer	
Everything	HMCCUDEV
Logfile	HmIP_PSM_Wohnzimmer
Commandref	
Remote doc	Heizung
Edit files	HmIP_eTRV_Wohnzimmer 23.5 21.5
Select style	
Event monitor	Stromzähler
	HM_HM_ES_TX_WM 10628.2074
	notify
	HmIP_PSM_Wohnzimmer_notify_1 2022-06-07 19:28:41

Bild 11: HmIP-Geräte in FHEM

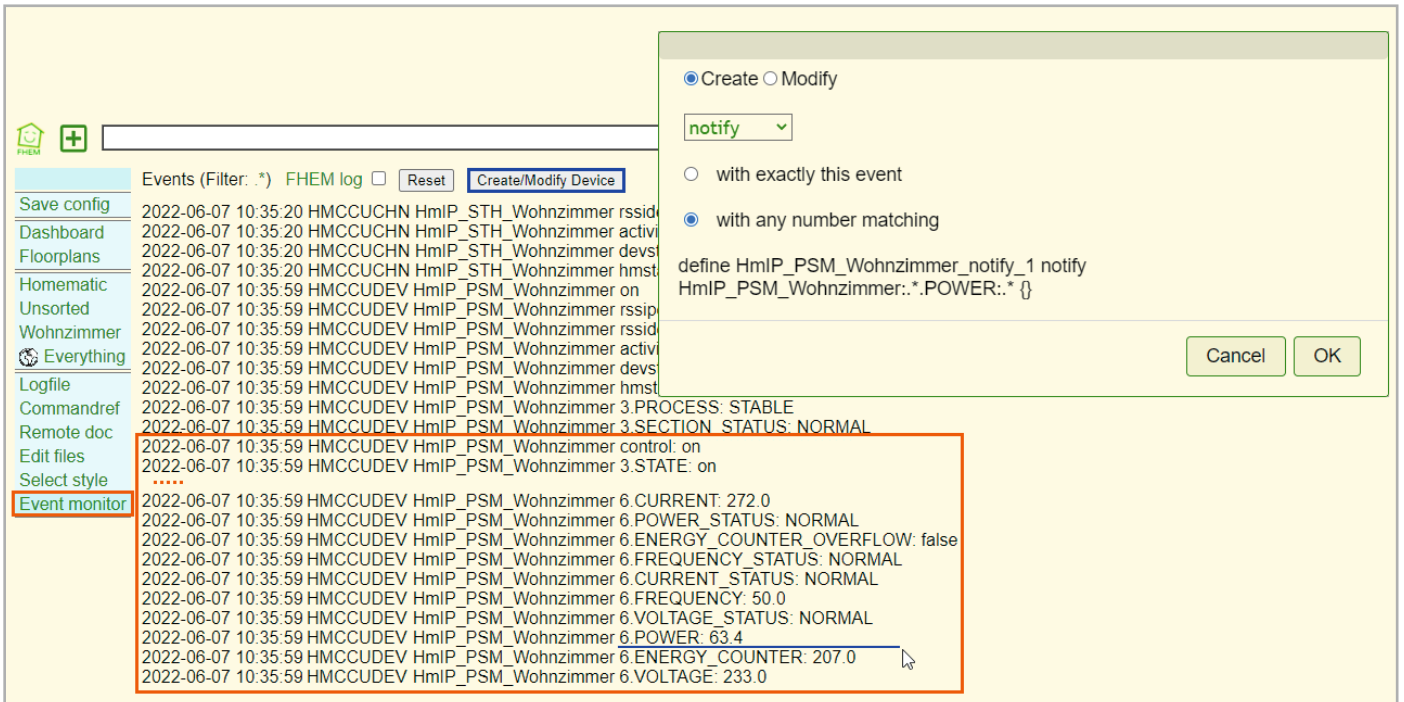


Bild 12: Schaltereignis im Event-Manager

Über den Button CREATE/MODIFY DEVICE oben rechts können Sie einen sogenannten Event-Handler anlegen (das erzeugt ein notify-Device) und damit auf bestimmte Ereignisse reagieren (Bild 13 zeigt den Stromverbrauch der PSM-Steckdose als Auslöser). Über diesen Mechanismus können Sie auch at-Devices definieren, die zu bestimmten Uhrzeiten weitere Aktionen auslösen und über IF- und DOIF-Befehle bedingte Schaltbefehle definieren.

Hierzu ist es jedoch absolut notwendig, sich intensiv mit regulären Ausdrücken (Regular Expressions), einem Begriff aus der Informatik und der Softwareentwicklung, zu befassen und damit die Bedingungen zu definieren. Hier werden der Anwender und seine exakte Syntax in der Kommandozeile sehr gefordert – ein Tribut an die historisch gewachsene Perl-Software FHEM.

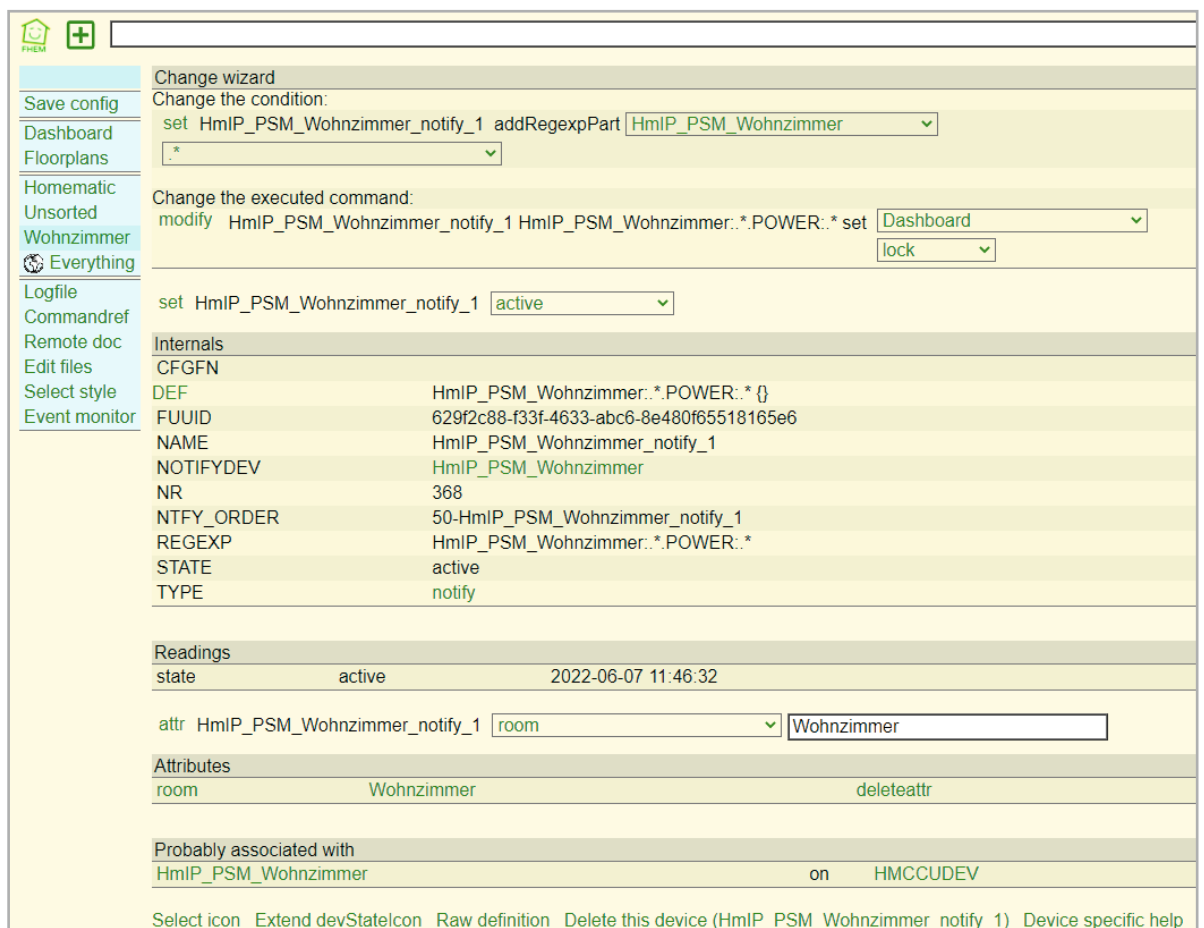


Bild 13: Notify-Device für ein Event

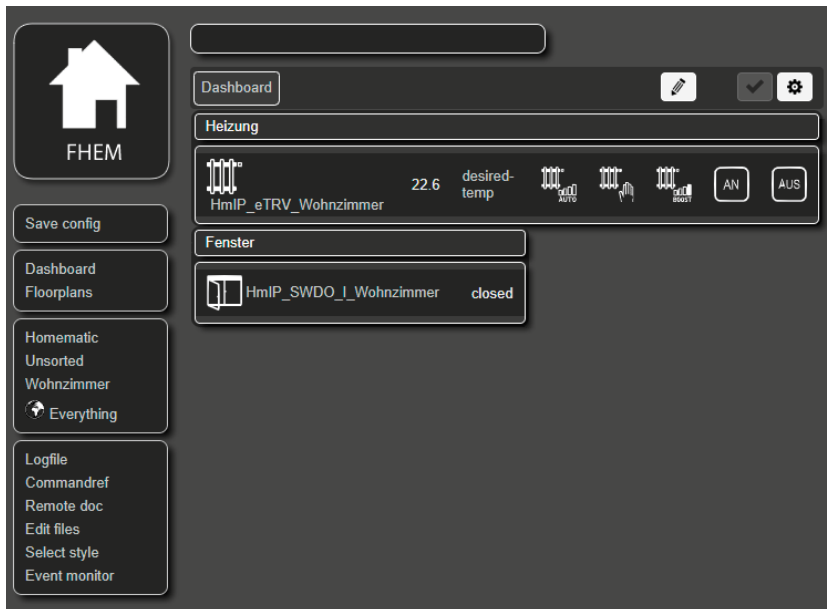


Bild 14: Dashboard-Beispiel im Style Dark

Visualisierung: Floorplan und Dashboard

Die graphische Standard-Benutzeroberfläche in FHEM ist ebenfalls ein Modul und nennt sich FLOORPLAN. Der FLOORPLAN ist eine Erweiterung, mit deren Hilfe Sie Geräte und deren mögliche Aktionen und Attribute auf einem selbst erstellten Hintergrundbild darstellen können. Zunächst ist wieder ein Definitionsbefehl notwendig:

```
define Grundriss FLOORPLAN
```

Nach dem Hochladen einer Hintergrundgrafik fp_Grundriss im Dateiformat .PNG in das Verzeichnis /www/images/default und einem set WEB rereadicons

erscheint der Grundriss bei Anklicken des Menüpunkts „Floorplans.“ Nun sind die einzelnen Gerätewerte über Koordinaten zu definieren. Bei vielen Geräten eine Sisyphusarbeit – bringen Sie hier Geduld und Zeit mit (s. [Titelbild](#)). Ein passendes Tutorial finden Sie im FHEM-Wiki unter dem Suchbegriff „Floorplan“ [7].

Ein Alternative zur Standard-Visualisierungs-Lösung ist das Modul DASHBOARD. Es ist ein Hilfsmodul, das umfangreiche Gestaltungsmöglichkeiten für das Web-Frontend bietet. So ist zum Beispiel die Anordnung von Gerätegruppen in Tabs und in mehreren Spalten möglich (Bild 14 im Style Dark).

Installiert wird das Dashboard über den Befehl "define <name> Dashboard", wobei Sie für <name> eine eindeutige Bezeichnung für das anzulegende Dashboard einsetzen müssen. Damit das Dashboard nicht leer bleibt, müssen Sie die einzelnen Geräte über das jeweils zur Komponente gehörende „groups“-Attribut einer Gruppe zuordnen. Auch hier gibt es ein schönes Tutorial im FHEM-Wiki [8].

Fazit und Ausblick

Wer sich weder durch die Kommandozeile noch durch die nicht gerade modern wirkende Oberfläche abschrecken lässt, bekommt mit FHEM eine zuverlässige, altgediente Lösung für seine Hausautomatisierung. Die Gründung eines Vereins birgt die Hoffnung, dass diese Software noch lange und fortdauernd unterstützt und weiterentwickelt wird. Auch die immense Anzahl an Modulen zur Unterstützung des großen Angebots an Steuerungssystemen ist ein weiterer Pluspunkt des Oldies FHEM.

Wie bereits beim ioBroker erwartet auch FHEM eine intensive Auseinandersetzung mit der eigentlichen Software und eine ausgiebige Einarbeitung in das System. Das gewohnungsbedürftige Definieren von Geräten erfordert viel Tipparbeit, etwas Ausdauer und das eine oder andere Mal mehrere Versuche, bis das gewünschte Ergebnis erreicht wird. Allerdings zeigt sich FHEM hier sehr gutmütig und nimmt auch das mehrmalige Löschen und Wiedereinfügen nicht übel. Vielfältige Hilfestellungen finden Sie in der FHEM-Dokumentation [9] sowie dem zugehörigen FHEM-Wiki [10] und dem aktiven FHEM-Forum [11].

Nach diesem zweiten Teil unserer Broker-Serie werden wir im nächsten ELVjournal auf openHAB, den „open Home Automation Bus“ eingehen. Die jeweiligen Vor- und Nachteile aller vorgestellten Broker finden Sie zum direkten Vergleich am Ende eines jeden Beitrags. **ELV**

i Weitere Infos

- [1] FHEM – freundliche Hausautomation und Energie-Messung: https://fhem.de//fhem_DE.html
- [2] Peter A. Henning: Smart Home mit FHEM, Carl Hanser Verlag GmbH & Company KG, ISBN 978-3-446-46247-2
- [3] Vermittler im smarten Zuhause, Teil 1 – ioBroker – automate your life, ELVjournal 3/2022: Artikel-Nr. 252720
- [4] Raspberry Pi – OS Download: <https://www.raspberrypi.com/software/>
- [5] PuTTY SSH Client – Download: <https://www.putty.org/>
- [6] Installation unter Debian mit Troubleshooting-Tipps: <https://debian.fhem.de/>
- [7] FHEM Floorplan Tutorial: <https://wiki.fhem.de/wiki/FLOORPLAN>
- [8] FHEM Dashboard Tutorial: <https://wiki.fhem.de/wiki/Dashboard>
- [9] FHEM Dokumentation: <https://fhem.de/#Documentation>
- [10] FHEM Wiki: <https://wiki.fhem.de/wiki/Hauptseite>
- [11] FHEM Forum: <https://forum.fhem.de/>

Alle Links finden Sie auch online unter: de.elv.com/elvjournal-links

Glossar

Attributes

Attribute sind die Eigenschaften der Geräte und Module. Es wird zwischen globalen Attributen für alle Devices und lokalen, gerätespezifischen Attributen unterschieden.

Dashboard

Das Modul DASHBOARD ist eine Erweiterung von FHEM und ermöglicht die Anordnung von Geräte-Gruppen mit Register-Tabs anstelle von Räumen.

Devices/Geräte

Als Devices werden in FHEM alle Geräte bezeichnet. Dies müssen nicht unbedingt echte Hardware-Komponenten sein, daher wird auch von logischen Geräten gesprochen.

Events/Ereignisse

Events sind neben den Devices das Grundkonzept in FHEM. Es sind Systemnachrichten, die alle Daten über Änderungen von Zuständen beinhalten und von allen Geräten gelesen werden können.

Floorplan

Das Modul FLOORPLAN erlaubt die Darstellung von Geräten mit einem Icon oder ihrem Status auf einer Hintergrundgrafik. Die Geräte sind durch Anklicken bedienbar und aktualisieren ihren Status. Das Smart Home wird als „Landkarte“ dargestellt.

Group/Gerätegruppe

Das FHEMWEB zeigt Geräte, die in die gleiche Gruppe gehören, auch in einer gemeinsamen Box an. Die Zugehörigkeit zu einer Gruppe wird über das Geräteattribut "group" definiert. Ein Gerät kann zu mehr als einer Gruppe gehören – in diesem Fall müssen die entsprechenden Gruppen durch Kommata getrennt eingetragen werden.

HMCCU

Durch das Modul HMCCU wird eine Verbindung zur Homematic CCU hergestellt. Weitere Module werden benötigt: HMCCUDEV, HMCCUCHN, HMCCURPCPROC und HMCCUConf

Module

Die Module sind die Grundlage der FHEM-Funktionalität, aus denen Geräte abgeleitet werden. Als Modul wird sowohl die funktionale Einheit (FHEM-Modul) als auch die Datei (Perl-Modul) bezeichnet, die den Programmcode enthält.

Regulärer Ausdruck

Das ist ein Begriff aus der Informatik und für die Arbeit mit FHEM sehr wichtig. Der reguläre Ausdruck beschreibt eine Menge von Zeichen mit syntaktischen Regeln und dient als Filter für eine andere Zeichenkette. Ein sehr einfaches Beispiel sind die sogenannten Wildcards, oder auch Platzhalter, bei der Dateisuche (z.B. *.doc).

Rooms/Räume

Mit Räumen bezeichnet FHEM die Gruppierung von Geräten. Dies können tatsächliche Räume im Smart Home sein oder auch Funktionsgruppen innerhalb der Zentrale. „Unsorted“ und „Everything“ sind voreingestellte Räume für nicht zugewiesene oder globale Elemente.

Scripting

FHEM bietet die Möglichkeit, Perl-Skripte auszuführen, welche dann beispielsweise bei der Änderung eines Sensorwerts eine bestimmte Aktion auslösen.

Visualisierung

Unter Visualisierung (Sichtbarmachen) versteht man im Allgemeinen, dass abstrakte Daten und Zusammenhänge (z. B. Sensordaten und Texte) in eine graphische bzw. visuell erfassbare Form gebracht werden. Die (Standard-)Visualisierung in FHEM läuft über das Modul FLOORPLAN.

Übersicht der Beiträge im ELVjournal zum Thema Broker-Software

ELVjournal Stand März 2022	Teil 1 ioBroker Automate your life	Teil 2 FHEM Freundliche Hausautomation und Energie-Messung	Teil 3 openHAB open Home Automation Bus
Ausgabe	3/2022	4/2022	5/2022
Download	https://www.iobroker.net/	https://fhem.de//fhem_DE.html	https://www.openhab.org/
Version(en)	v5.2.3 node.js V16.14.0 npm 8.3.1 js-controller 4.0.15	6.1	3.2.0 Java zulu 8.42.0.195
App	ioBroker (Android) ioBroker (iOS)	andFHEM (Android) FHEM-Remote (iOS)	openHAB (Android) openHAB (iOS)
Skripting	Rules, JavaScript, Typescript, Blockly	PERL in 99_myUtils	Rules, Groovy, JavaScript, JRuby, Jython
Homematic Anbindung	Adapter HomeMatic ReGaHSS 2x Adapter HomeMatic RPC	HMCCU	Homematic Binding
Server-Adresse	[IP-Adresse]:8080	[IP-Adresse]:8083	[IP-Adresse]:8080
Admin-Tool	[IP-Adresse]:8081	[IP-Adresse]:8083/fhem	[IP-Adresse]:8101
Visualisierung	Adapter VIS [IP-Adresse]:8082/vis/ Adapter Jarvis [IP-Adresse]:8082/jarvis/	Dashboard Floorplan	HABPANEL
Dokumentation/Wiki	https://www.iobroker.net/#de/documentation	https://www.fhemwiki.de/wiki https://wiki.fhem.de/wiki/Hauptseite	https://www.openhab.org/docs
Forum	https://forum.iobroker.net	https://forum.fhem.de	https://openhabforum.de
ELVjournal Stand März 2022	Teil 4 Node-RED a flow-based programming tool	Teil 5 IP-Symcon Smart Home ohne Kompromisse	
Ausgabe	6/2022	1/2023	
Download	https://nodered.org/	https://www.symcon.de/	
Version(en)	v2.2.0 node.js V14.19.0 npm 6.14.16	6.2	
App	Termux mit node-RED (Android) Remote-RED (iOS)	IP-Symcon Mobile (Android) IP-Symcon Mobile (iOS)	
Skripting	Flows, JavaScript	PHP 7.4	
Homematic Anbindung	node-red-contrib-ccu node-red-contrib-homematic	HomeMatic Socket	
Server-Adresse	[IP-Adresse]:1880	[IP-Adresse]	
Admin-Tool	[IP-Adresse]:1880	[IP-Adresse]:3777/console	
Visualisierung	Dashboard	Webfront	
Dokumentation/Wiki	https://nodered.org/docs	https://www.symcon.de/service/dokumentation	
Forum	https://discourse.nodered.org	https://community.symcon.de	

Tabelle 1