


Kurzanleitung Scriptdeveloper V3.04.xx (29.November 2018)

Der Scriptdeveloper (SDV) soll ein Hilfsmittel im Alltag bei der Erstellung von Homematic Scripten und deren Tests darstellen. Ein gewisses Wissen über Scripterstellung sowie den Aufbau einer CCU wird vorausgesetzt.

Die Software läuft auf Windows PC, ist bei nicht kommerzieller Nutzung Freeware und ist nicht an die Nichtverwendung einer Raspberrymatic oder sonstiger Einschränkungen gebunden.

Da mittlerweile aber schon einige tiefgreifende Operationen möglich sind, sind Mehrfachselektionen und Löschfunktionen erst nach Drücken von Unlock  zugänglich.

Trotzdem an der Stelle der Hinweis, welcher auch beim ersten Start des Programmes bestätigt werden muss:

Dies ist eine BetaTestversion.

Die Verwendung dieser Software erfolgt auf eigenes Risiko
Der Autor dieser Software übernimmt keine Haftung für direkte oder indirekte Schäden, welche sich aus der Benutzung dieser Software ergeben sollten.
Eine kommerzielle Nutzung dieser Software ist untersagt

Ich bin einverstanden (Ja, Nein, wobei nein zum Programmende führt)

Hinweise über undokumentierte Methoden, die im Alltag nützlich sind aber ich bis jetzt auch noch nicht kannte, nehme ich gerne an und baue die auch gerne hier in das Programm mit ein.

Inhalt

Kurzanleitung Scriptdeveloper V3.04.xx (29.November 2018)	1
1. Installation	3
1.1 Lizenzierung	4
1.2 Systemvoraussetzungen	7
1.3 Was tuts bis jetzt	7
1.4 Bekannte Einschränkungen / Bugs	7
2 Oberfläche	8
3 Scripteditor	9
3.1 Voreinstellungen Editor	10
3.2 Vervollständigen Funktion	11
4 Inspektor	12
4.1 Selektionswahl: DomScan	13
4.2 Selektionskriterium Types	16
4.3 Zusätzliche Selektionsbedingungen	17
4.4 Daten aus Inspektor in Editor übernehmen	23
4.4.1 Mehrfachauswahl als Enum String	25
4.5 Selektion von Selektion	27
4.6 Objekte löschen	28

1. Installation

Das *.rar File in ein beliebiges Verzeichnis entpacken. Ein Installer ist nicht notwendig. In diesem Verzeichnis befindet sich auch das Konfigurationsfile SDV.INI. Bei der erstmaligen Verwendung muss dieses angepasst werden

[LAST]

DATEI=c:\MTH\Homematic\NewScript.hsc

[HOST]

CCU=192.168.2.19

NICKNAME=Benutzername

← anpassen Benutzername / Nick

CCU1=192.168.2.XX

← IP Adresse der 1. CCU

CCU2=192.168.2.XX

← Wenn vorhanden, IP der 2. CCU

CUXD=CUXD.CUX2801001:5

← CUXd KANAL (ist nötig, dafür braucht es kein pscp mehr)

LICENCE1=

← Lizenzschlüssel für 1. CCU

LICENCE2=

← Lizenzschlüssel für 2. CCU

[ENUM_NORM]

← Ab hier kommen dann interne Werte, Finger Weg

C1=65

C2=200

C3=293

C4=65

[ENUM_MAX]

C1=65

C2=200

C3=293

C4=65

Warum CUXD ? Der SDV Version 2.x nutzte noch pscp für den Zugriff auf die Logdatei und auf das System. Dies war immer ein Schwachpunkt (zusätzliches Programm, Bestätigung Serverzertifikat. Dies wird jetzt mit CUXD realisiert. Es muss ein Kanal angegeben werden auf einem CUXD exec Gerät, auf das der SDV exklusiven Zugriff hat. Auf Systemen ohne CUXd kann der SDV nicht eingesetzt werden.

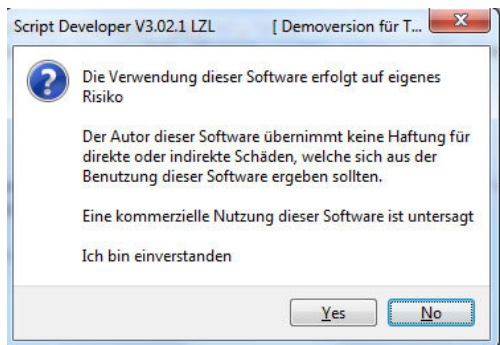
1.1 Lizenzierung

Der SDV ist bei nicht kommerzieller Nutzung Freeware. Trotzdem habe ich mich entschlossen, aufgrund von Erfahrungen der Vergangenheit den Nutzerkreis oder die möglichen Features bestimmter Nutzer einzugrenzen. Dies geschieht durch Vergabe von bis zu 2 Lizenzschlüsseln. Der SDV ist dadurch an bis zu 2 CCU / Raspberrymatic gepairt.

Wie arbeitet das ?

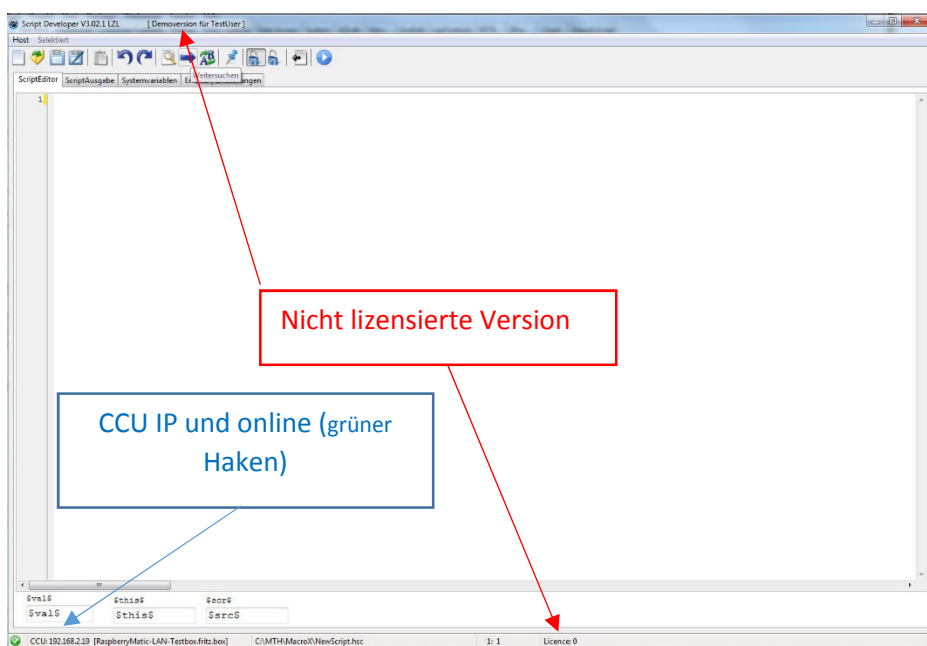
Der SDV telefoniert nicht nach Hause. Um eine Lizenz anzufragen ist folgender Weg einzuschlagen.

1. Die Konfigurationsdatei SDV.INI mit einem Editor öffnen.
2. Nickname Anpassen
3. IP der CCU 1 eintragen
4. IP der CCU 2 eintragen
5. CUXD Kanal eintragen
6. Konfiguration abspeichern
7. Script Developer starten

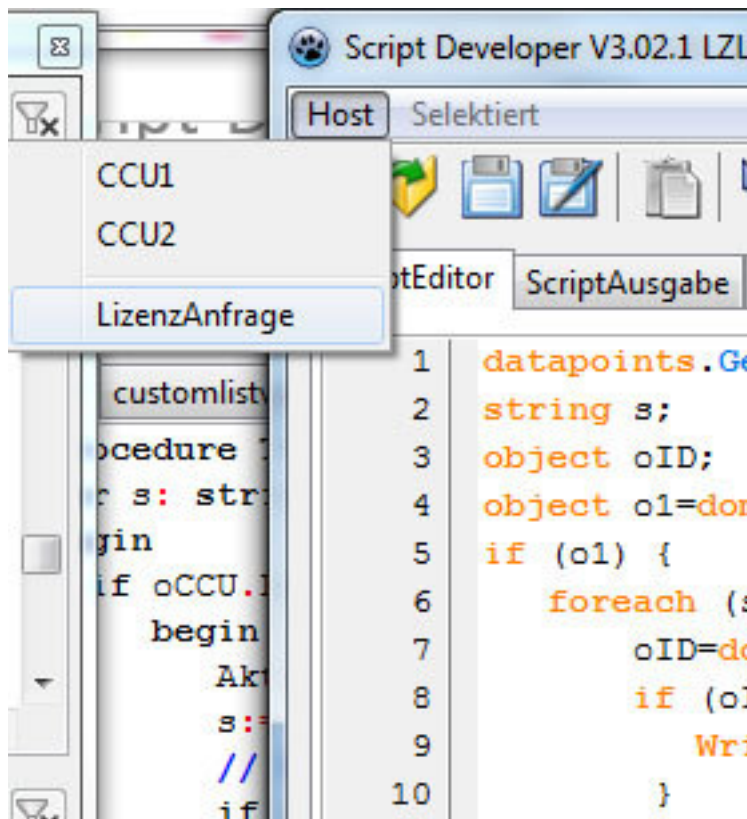


Bei allerersten Start muss dieses Fenster mit yes bestätigt werden. No führt so einem sofortigen Programmabbruch

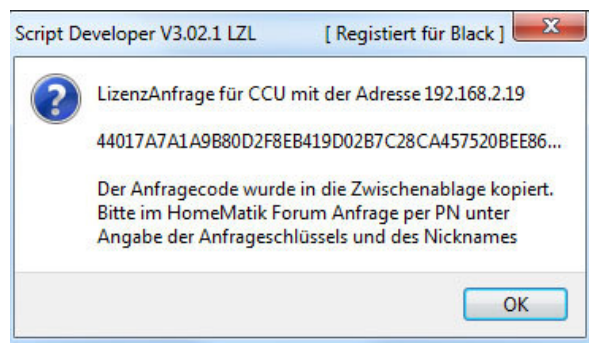
Bei Bestätigung mit Yes startet nun zum erstenmal der SDV als Demoversion



Für die weiteren Schritte muss der SDV mit der CCU verbunden sein und die CCU auch als online erkannt worden sein.



unter Host auf Lizenzanfrage drücken. Als nächstes öffnet sich ein Fenster mit einem Anfrage Hexstring.



Der Hexstring ist in die Zwischenanlage kopiert und kann in beliebige Text Dokumente eingefügt werden. Als nächste dann im Homematik.de Forum eine PN an mich schreiben mit dem String und Angabe des Nicknames, welcher zum Zeitpunkt der Lizenzanfrage in der INI Datei eingetragen war.

Was enthält dieser Hexstring ?

Kodiert und verschlüsselt: 1. den Nicknamen, 2. die Seriennummer des Funkmodules der verbundenen CCU , einen VerifizierCode von mir.

Die Seriennummer des Funkmodules ist nötig zur Verifizierung des Pairings. Diese wird bei mir nirgends gespeichert, mit diesem Hexschlüssel wird nach der Anfrage der LizenzLevel definiert und ebenfalls in einen Hexstring kopiert. Dieser dann zurückgesandte Hexstring wird unter Licence1 oder Licence2 in der INI Datei eingetragen. Es kann mit bis zu 2 CCU bearbeitet werden, sollte ein Lizenzlevel höherwertiger sein so gilt dieser höherwertige Level für beide CCUs.

Wer mit diesem Verfahren nicht einverstanden ist, möge bitte an dieser Stelle die PDF Datei schließen und kann die Dateien beruhigt löschen.

Geplant hab ich folgende Lizenzabstufungen

Level	Editor	Script ausführen	Highlighter Und Vervollständiger	Enums	SysVar	Programs	Backup Restore		
0	X								
1	X	X							
2	X	X	X						
3	X	X	X	X	X				
4	X	X	X	X	X	X			
5	X	X	X	X	X	X	X		
6									

1.2 Systemvoraussetzungen

Der SDV lief bisher in Testinstallationen unter WIN 7 64/32 bit, und unter Win 10 64bit. Da unter recht konservativen Compilereinstellungen übersetzt wurde, sollte er eigentlich unter allen Windows Version laufen (ab Win 7)

Auf der Homematic-Seite wurde bei mir auf einer Raspberrymatic 3.37.8.20181026 getestet.

Auf einer CCU sind die erzeugten internen Progs auch lauffähig, wenn Rega-Community eingestellt wird. Unter Legacy läuft es NICHT !

1.3 Was tuts bis jetzt

Der Editor funktioniert inkl. Suchen und Suchen / ersetzen. Der Highlighter und der Code vervollständiger arbeiten auch.

Undo / Redo arbeiten

Script ausführen arbeitet und liefert wie in der alten Version die Antworten der CCU.

Enums und Sysvars arbeiten auch schon inkl. Detaildaten und Editiermöglichkeiten.

Darstellbarkeit zumindest der Grundmethoden aller Objekte

DomScan

1.4 Bekannte Einschränkungen / Bugs

Auswahldialoge sind auf Englisch. Weiß ich, zurzeit benutze ich die in der Laufzeitumgebung integrierten Dialoge, und die sind leider trotz Landeseinstellung englisch.

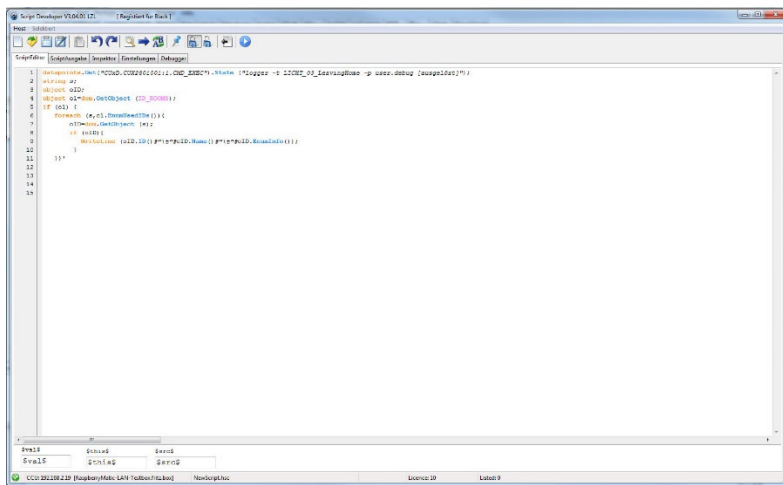
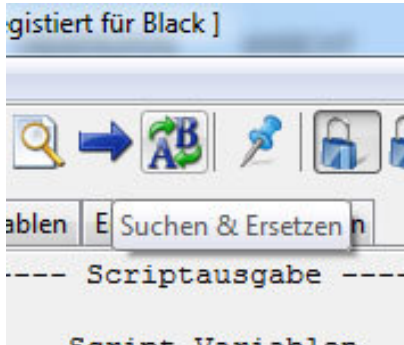
Folding im Editor arbeitet noch nicht. Wenn der Rest läuft gucke ich da mal nach.




Kommentare im Script müssen als `!-` geschrieben werden. Kann man sich dran gewöhnen, das anzupassen wäre ein Haufen Aufwand, da EQ3 ja klugerweise Negation und Kommentar mit demselben Zeichen bedacht hat. Hurra. Ich kann jedenfalls mit dem `!-` gut leben, folglich ist die Chance, das ich das ändere, recht gering: xD

Aufgrund dessen, dass als Middleware bei mir IOBroker läuft und ich die Diagramm und die History Funktion der CCU nicht nutze, werde ich diese im SDV auch nicht ausprogrammieren.

2 Oberfläche

Zu fast allen Funktionen sind die Hint parametrierbar, so dass es da Hilfestellung gibt.




Im Menüreiter Skripte finden sich die Einstellungen zum Anlegen eines neuen Skriptes , zum Laden eines bestehenden Skriptes  und zum Speichern eines Skriptes im Scripteditor  sowie speichern unter.

In der Statuszeile finden sich Information über:

1. IP der Host CCU
2. Dateinames des Skriptes im Scripteditor
3. Anzahl der Elemente in der Listendarstellung

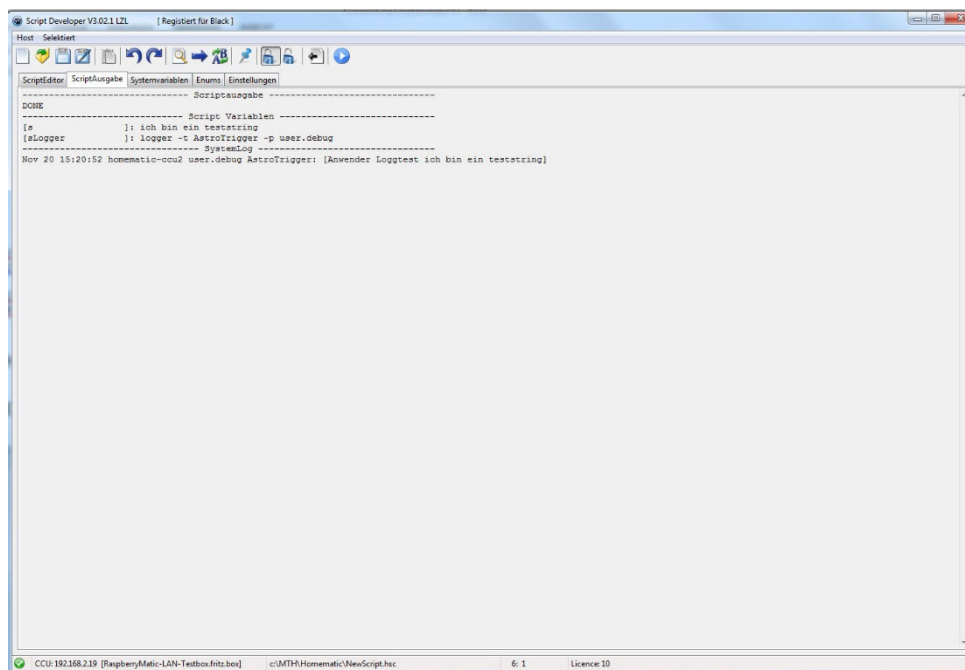
3 Scripteditor

Im Scripteditor werden die Scripte geschrieben oder geladen, die mittels Run Script oder  an die CCU zum Ausführen gesendet werden. Das Scriptergebnis wird dann im Reiter Ausgabe angezeigt. Dieses kleine TestScript zum Beispiel:

```
string s= "ich bin ein teststring";  
string sLogger = "logger -t AstroTrigger -p user.debug ";  
  
datapoints.Get("CUXD.CUX2801001:1.CMD_EXEC").State (sLogger # "[Anwender Loggtest " # s # "]");  
WriteLine ("DONE");
```

Hier testweise aus State

Erzeugt folgende Ausgabe:



Script Ausgabe stellt alles dar, was in dem Script mit Write, WriteLine oder Derivaten zur Ausgabe gebracht wurde,

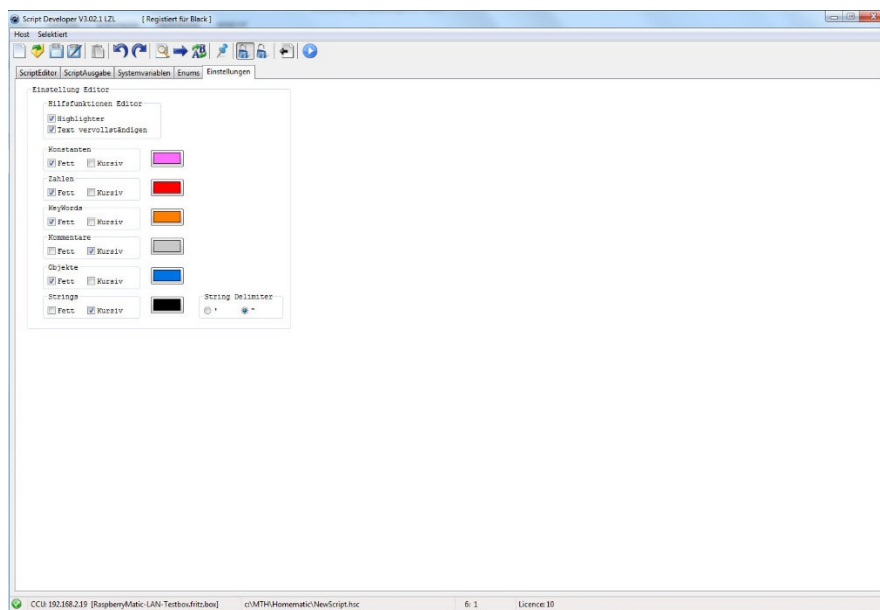
Unter lokale Script variablen stehen die Variablen welche im Script definiert wurden mit ihren Namen. In dem Fall hier sind das die Beiden String Variablen s und sLogger.

Wurde via Userlog ein Eintrag im Logfile erzeugt, so wird dieser nach Scriptende auch hier angezeigt.

Sollte in dem Script ein Fehler sein (hier testweise State zu Stat geändert) erhält man die gleiche Ausgabe wie im Syslog:

```
[----- Fehler im Script -----]
Jun 15 12:41:49 homematic-raspi local0.err ReGaHss: Error: IseESP::SyntaxError= Error 1 at
row 4 col 88 near ^ (sLogger # "[Anwender Loggtest " # s # "]);^M WriteLine ("DONE");^M
[iseESP.cpp:1121]
Jun 15 12:41:49 homematic-raspi local0.err ReGaHss: Error: ParseProgram: SyntaxError=
(sLogger # "[Anwender Loggtest " # s # "]);^M WriteLine ("DONE"); [iseESP.cpp:374]
```

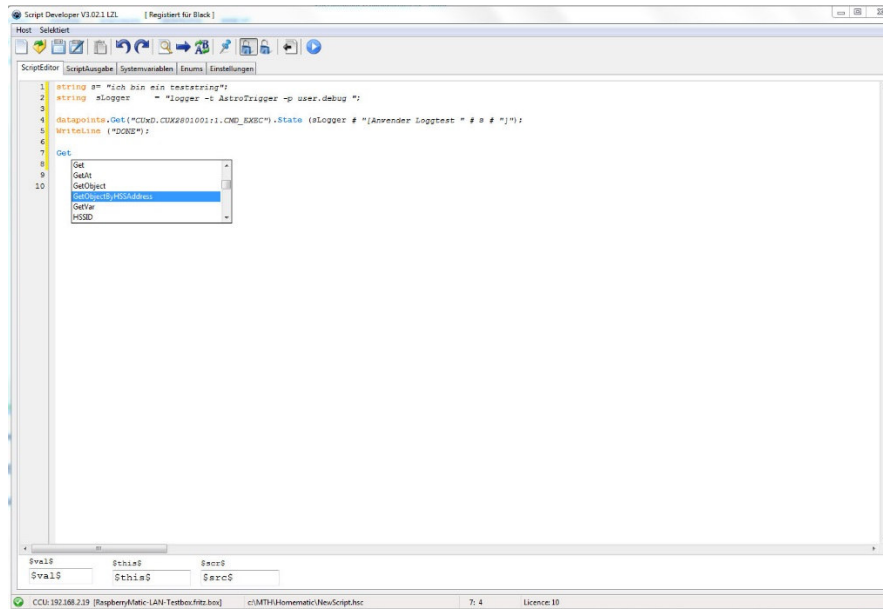
3.1 Voreinstellungen Editor



Hier kann nach Vorliebe der Highlighter konfiguriert oder auch ausgeschaltet werden.
Ebenso lässt sich die Vervollständigen Funktion an oder abwählen.
Vorbereitung natürlich: Lizenzlevel muss vorhanden sein.

3.2 Vervollständigen Funktion

Methoden und Konstanten Namen muss man sich nicht auswendig merken. Der Editor verfügt über einen Auto Vervollständiger. Man schreibt den Wortanfang, hier z.B Get , drückt Strg+Space und wählt in dem sich öffnenden Menü die passende Funktion aus.

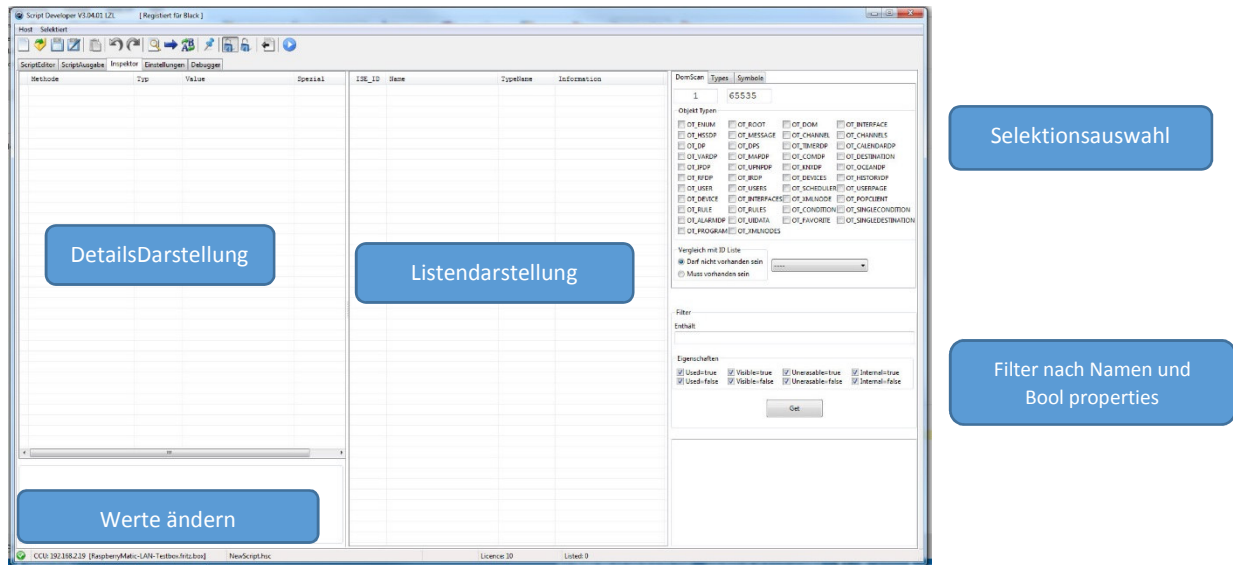


Nach Druck von Enter erscheint das Wort im Editor.

4 Inspektor

Der Inspektor dient zum Suchen, Anzeigen und Ändern von Objekten auf der CCU/Raspberrymatik.

Es existieren verschiedene Selektionskriterien.



Filteroptionen:

Auswahl der Aufzählungen (Räume, Gewerke, Favoriten, Interfaces , Systemvariablen sind bisher implementiert)

Enthält: leerer Eintrag = es wird nicht nach enthaltener Buchstabensequenz selektiert
Eingegebener Text. Die Systemvariable muss im Namen die Buchstabensequenz enthalten.

Eigenschaften: Es wird nach den Eigenschaften Used, Visible, Unerasable und Internal selektiert.
Am Beispiel used:

1. Kein Haken bei Used= false und kein Haken bei Used= true
Die Eigenschaft Used wird bei der Auswahl nicht beachtet
2. Haken bei Used= false und Haken bei Used= true
Die Eigenschaft Used wird bei der Auswahl nicht beachtet
3. Kein Haken bei Used= false und Haken bei Used= true
Um gelistet zu werden muss das Objekt die Eigenschaft Used=true haben
4. Haken bei Used= false und kein Haken bei Used= true
Um gelistet zu werden muss das Objekt die Eigenschaft Used=false haben

4.1 Selektionswahl: DomScan

DomScan Types Symbole

1 65535

Objekt Typen

<input type="checkbox"/> OT_ENUM	<input type="checkbox"/> OT_ROOT	<input type="checkbox"/> OT_DOM	<input type="checkbox"/> OT_INTERFACE
<input type="checkbox"/> OT_HSSDP	<input type="checkbox"/> OT_MESSAGE	<input type="checkbox"/> OT_CHANNEL	<input type="checkbox"/> OT_CHANNELS
<input type="checkbox"/> OT_DP	<input type="checkbox"/> OT_DPS	<input type="checkbox"/> OT_TIMERDP	<input type="checkbox"/> OT_CALENDARDP
<input type="checkbox"/> OT_VARDP	<input type="checkbox"/> OT_MAPDP	<input type="checkbox"/> OT_COMDP	<input type="checkbox"/> OT_DESTINATION
<input type="checkbox"/> OT_IPDP	<input type="checkbox"/> OT_UPNPDP	<input type="checkbox"/> OT_KNXDP	<input type="checkbox"/> OT_OCEANDP
<input type="checkbox"/> OT_RFDP	<input type="checkbox"/> OT_IRDP	<input type="checkbox"/> OT_DEVICES	<input type="checkbox"/> OT_HISTORYDP
<input type="checkbox"/> OT_USER	<input type="checkbox"/> OT_USERS	<input type="checkbox"/> OT_SCHEDULER	<input type="checkbox"/> OT_USERPAGE
<input type="checkbox"/> OT_DEVICE	<input type="checkbox"/> OT_INTERFACES	<input type="checkbox"/> OT_XMLNODE	<input type="checkbox"/> OT_POPCLIENT
<input type="checkbox"/> OT_RULE	<input type="checkbox"/> OT_RULES	<input type="checkbox"/> OT_CONDITION	<input type="checkbox"/> OT_SINGLECONDITION
<input type="checkbox"/> OT_ALARMDP	<input type="checkbox"/> OT_UIDATA	<input type="checkbox"/> OT_FAVORITE	<input type="checkbox"/> OT_SINGLEDESTINATION
<input type="checkbox"/> OT_PROGRAM	<input type="checkbox"/> OT_XMLNODES		

Vergleich mit ID Liste

☒ Darf nicht vorhanden sein

☐ Muss vorhanden sein

Eingabe des Scan Bereiches der IseID's (hier von 1-65535)

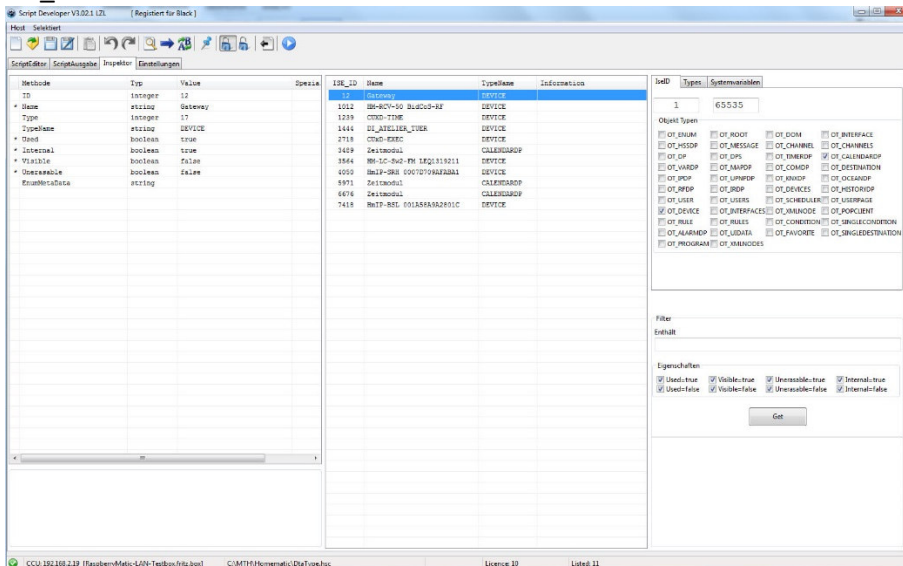
Achtung

Schrott Eingabe von Millionenwerten werden die CCU lahmlegen. Der SDV ist schließlich kein Spielzeug sondern ein Werkzeug, man sollte schon wissen, was man tut.

Damit ein Objekt Selektiert wird, muss es die angeklickte Objekteigenschaft haben.

Mehrfachangaben sind möglich

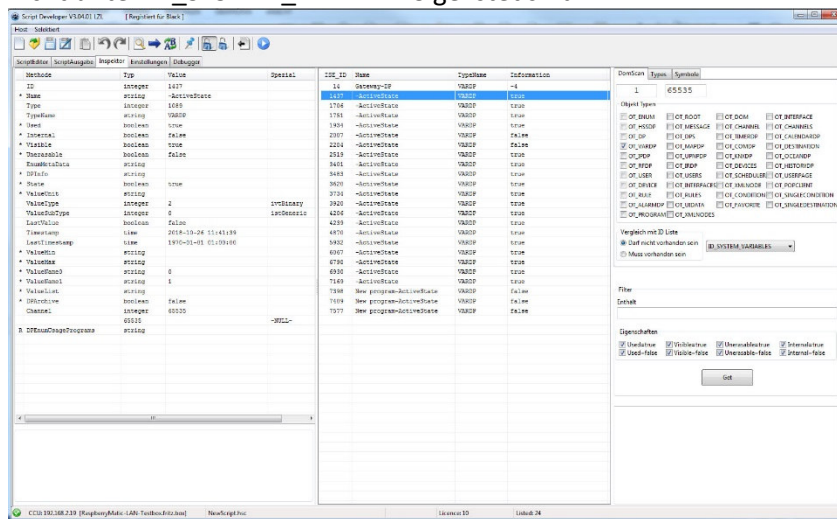
Beispiel für Suchen aller Objekte mit der Eigenschaft OT_DEVICE im Bereich der ISE_Nummern 1-65535



Anklicken eines Wertes in der Listendarstellung öffnet die Detaildarstellung des Objektes.

Ebenso ist es möglich, im DomScan Bereich Einträge zu suchen, welche beispielsweise nicht in den Aufzählungen gelistet sind.

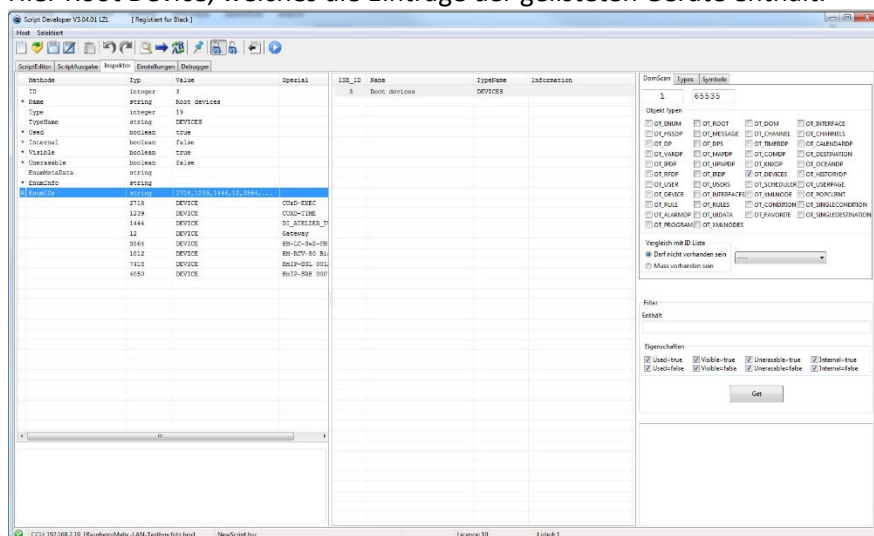
Hier Beispielsweise: Scanlauf über alle Objekte aus DOM mit der Eigenschaft VARDP, die aber nicht unter ID_SYSTEM_VARIABLES gelistet sind:



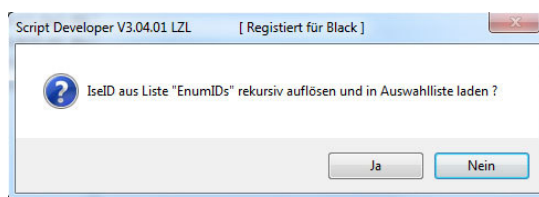
Hier tauchen dann einige interne Datenpunkte auf, im dem Falle sind die –ActiveState keine Leichen, sondern der Anwahr Punkt Programm aktiv unter Programme.

Weiterhin ist auch rekursives Arbeiten nun möglich

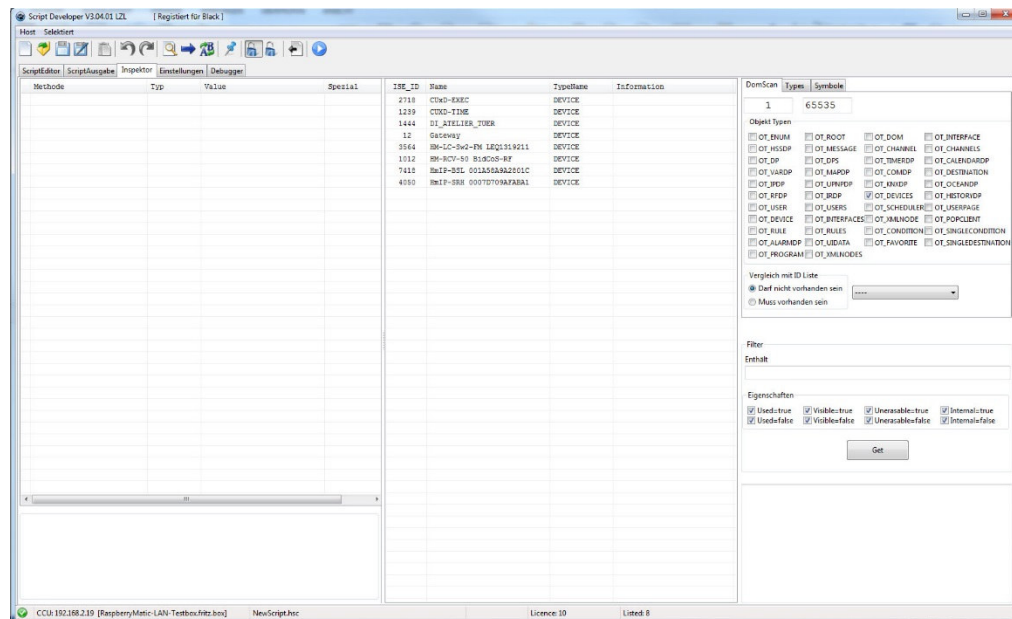
Hier Root Device, welches die Einträge der gelisteten Geräte enthält.



Das „R“ in der ersten Spalte zeigt an, dass ein Rekursiver Aufruf möglich ist. Doppelklicken auf die Zeile öffnet eine Sicherheitsabfrage



Wurde mit JA bestätigt, so wird die EnumList nun aufgelöst und als neue Auswahlliste zur weiteren Bearbeitung zur Verfügung gestellt.



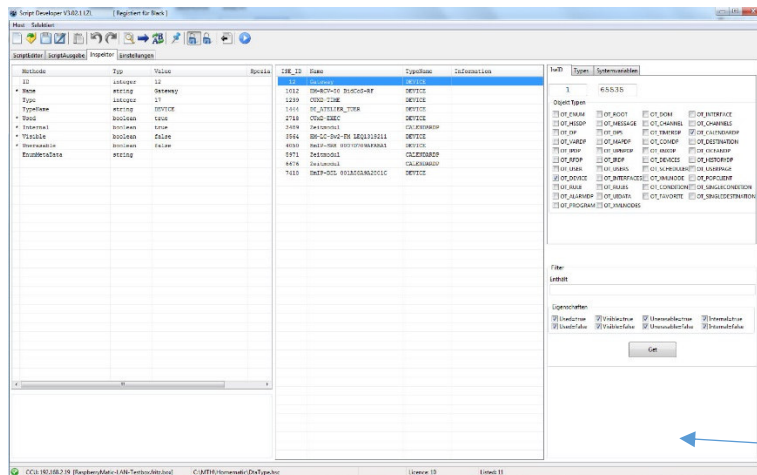
4.2 Selektionskriterium Types

The screenshot shows a software interface with three tabs: 'IselD', 'Types', and 'Systemvariablen'. The 'Types' tab is active. It contains two sections: 'Aufzählung' and 'Quelle'. In the 'Aufzählung' section, there are three radio buttons: 'ID_ROOMS' (selected), 'ID_FUNCTIONS', and 'ID_FAVORITES'. Below it, there is a radio button for 'ID_INTERFACES'. In the 'Quelle' section, there are two radio buttons: 'EnumIDs' (selected) and 'EnumUsedIDs'.

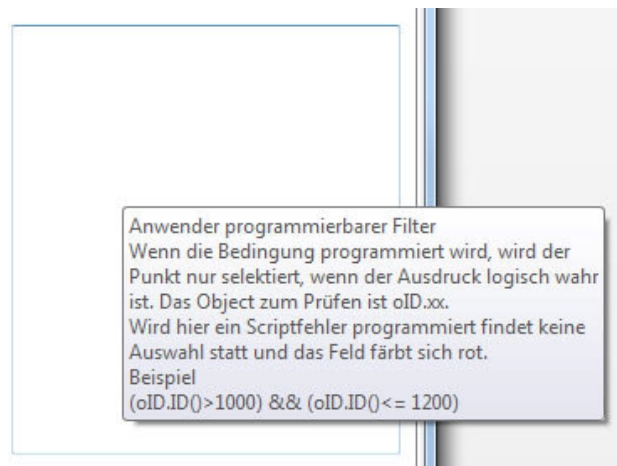
Hierbei wird wie schon in der Version 3.2 in festen Bereichen Gesucht und selektiert. Schneller und einfacher zu handeln als die Objekt Selektion, dafür nicht so umfangreich.

4.3 Zusätzliche Selektionsbedingungen

Durch Druck auf Get wird die Liste gemäß Selektion von der CCU angefordert, aufbereitet und dargestellt. (Lizenzlevel vorausgesetzt)

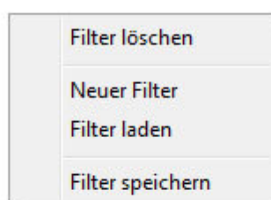


Zusätzliches Feld für Selektionsbedingen



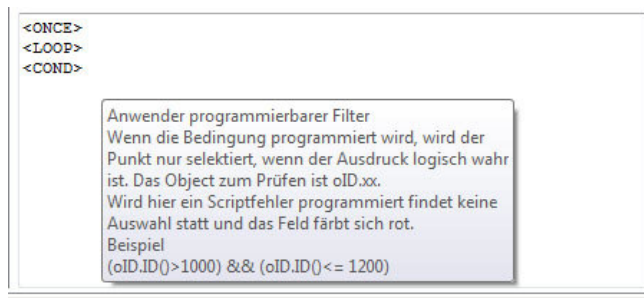
Filter sind ein mächtiges Werkzeug zum komplexen Eingrenzen und für Komplexe Abfragen.

Für die Filter existiert mittlerweile ein Kontext Menü mit rechter Maustaste:



Filter löschen entfernt sämtliche Filterbedingungen

Neuer Filter legt von der Syntax einen neuen, leeren Filter an



Mit Filter laden und speichern lassen sich nun Anwenderfilter als *.flt Datei im Verzeichnis des SDV abspeichern.

Ein Filter besteht aus den 3 Abschnitten:

<ONCE> Der Text dahinter wird am Anfang des internen Abfragescriptes quasi im einmaligen Durchlauf eingefügt. Normalerweise stehen hier Definitionen, welche nicht bei jedem Durchlauf aktualisiert werden müssen

<LOOP> Der Text dahinter wird im Zyklischen Durchlauf des Programmes innerhalb der Programmschleife eingefügt.

<COND> der Text hinter COND wird in die IF Abfrage eingefügt, welche letztlich das Objekt zur Darstellung in der Liste selektiert.

Vereinfachter Ablauf: so sieht vereinfacht das Listenselektionsprogramm aus:

```
object oID;
string s;
foreach (s,Schleifenbedingung) {
oID= dom.GetObject (s);
if (oID) {
    if (ElementBedingung) {WriteLine („Element in Liste: „ # oID.ID () );}
}
```

Ein Anwenderdefinierter Filter wird dann in diese Grundschleife so eingebaut:

```
object oID;
string s;
ONCETEXT;

foreach (s,Schleifenbedingung) {
oID= dom.GetObject (s);
if (oID) {
    LOOPTEXT;
    if (ElementBedingung && (CONDTEXT)) {WriteLine („Element in Liste: „ # oID.ID () );}
}
```

```
<CONCE>boolean c;  
<LOOP>c=(oID.EnumMetaData().Contains("VALUE_LIST"));  
<COND>c
```

```
object oID;
string s;
boolean c;

foreach (s,Schleifenbedingung) {
oID= dom.GetObject (s);
if (oID) {
    c=(oID.EnumMetaData().Contains ("VALUE_LIST"));
    if (ElementBedingung && (c)) {WriteLine („Element in Liste: „ # oID.ID () );
}
}
```

[illegible]

```
object oID: darf benutzt werden, ist der Bezug auf das Objekt, welches im Filter überprüft werden soll
var v: intern benutzt zur Typerkennung: Fingers weg
string sInfo: intern benutzt zur Listengenerierung: Fingers weg
boolean b: interner Filder, auch Finger weg
string done: auch interne Benutzung, auch Finger weg
```

Die Filterbedingung wird in HM Script ausformuliert. Das gefundene Object kommt nur in die Liste, wenn die ausformulierte Bedingung True ist. Das Teil ist mächtig, aber auch nicht ungefährlich, man kann auch Müll als Bedingung schreiben. Dabei kommt dann aber eine Warnung:

```
<ONCE>boolean c;  
<LOOP>c=(oID.EnumMetaData().Contains ("VALUE_LIST";  
<COND>c
```

Anwender programmiert
Wenn die Bedingung
Punkt nur selektiert
ist. Das Object zu

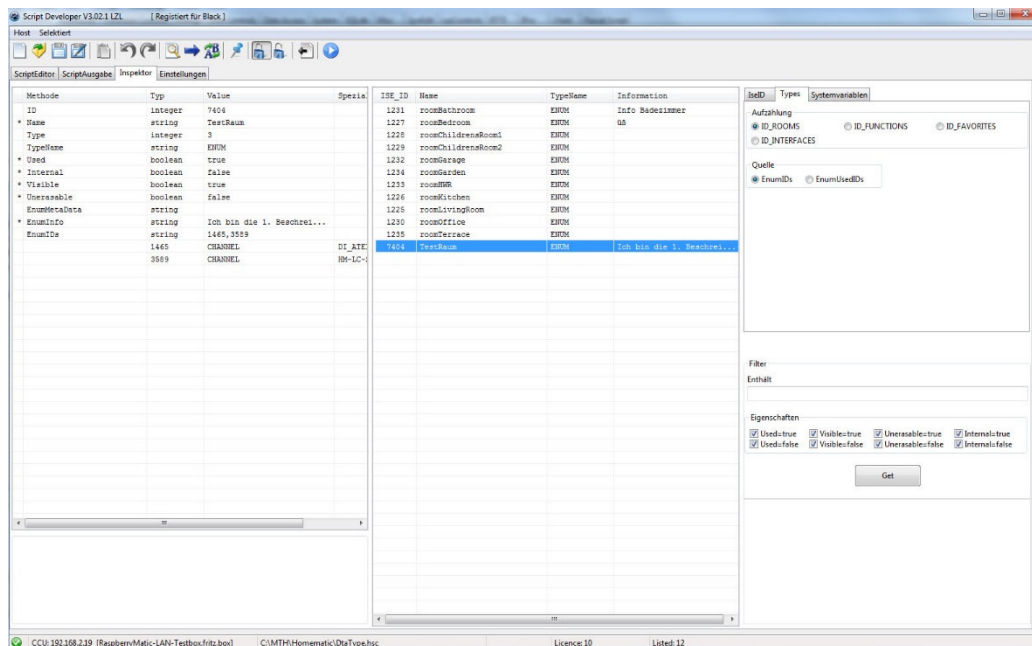
Bedingung ist falsch: erzeugt Scriptfehler
(Klammer zu fehlt). In dem fall färbt sich nach
Druck auf Get das Feld rot

Durch Click auf die Beschreibungszeile IseID bzw Name können die Felder entsprechend sortiert werden.

Click auf eine selektierte Aufzählung öffnet im Detailfenster die Methodenansicht des Objektes

Changelog V3.03xx

Da die internen Sortialgorithmen suboptimal arbeiteten, hat das ListView Object neue selektive Sortialgorithmen bekommen. IseID sortiert nun wie man erwartet nach Integer aufsteigend, Name sortiert alphabetisch aufsteigend, TypeName sortiert alphabetisch, sind die Typenames gleich, wird innerhalb gleicher Typenames nach IseID numerisch sortiert.



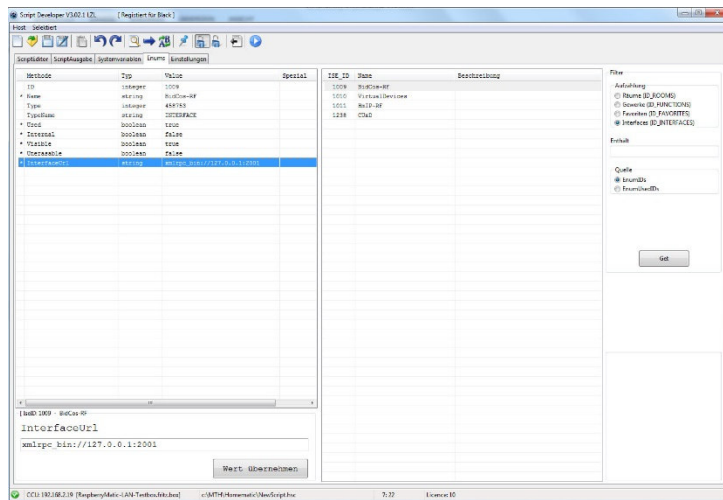
Die Spaltenbreite kann sowohl in Normalansicht als auch in Maximize separat eingestellt werden (Das Programm sollte sich die Breiten merken und je nach Darstellungsart automatisch wieder einstellen, sollte...)

Dargestellt werden die Methode, der Vartype und die Property.

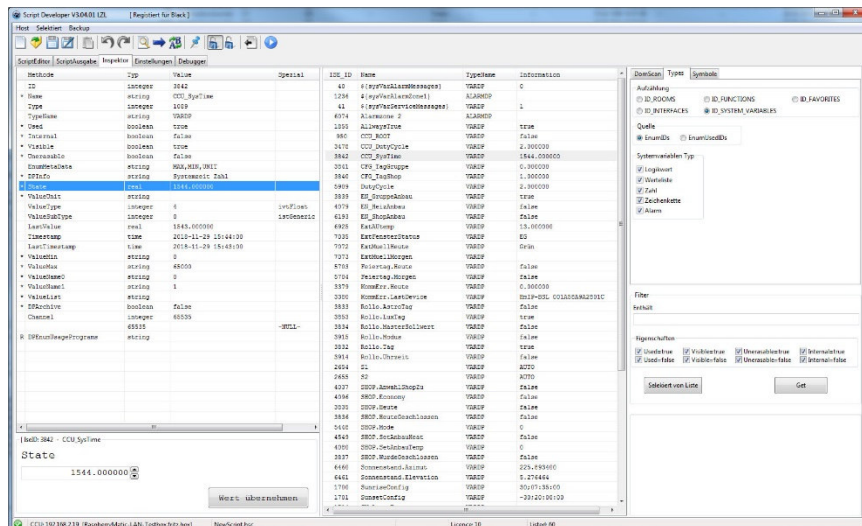
Bei den Aufzählungen wird jeweils eine Rekursionsstufe aufgelöst, um an die Detailinformationen zu kommen. Hier die Liste der Channels, die diesen Raum verwenden, aufgelöst in die ID, der Typ (hier Channels und der Name des Channels)

Properties, die in der ersten Zeile mit einem Stern (*) gekennzeichnet sind, können in ihrem Wert geändert werden.

Dazu auf die Zeile klicken



Nach Click auf Wert übernehmen wird der Wert in der CCU geändert. Also Vorsichtig mit dieser Funktion umgehen, hier gibt es kein redo.

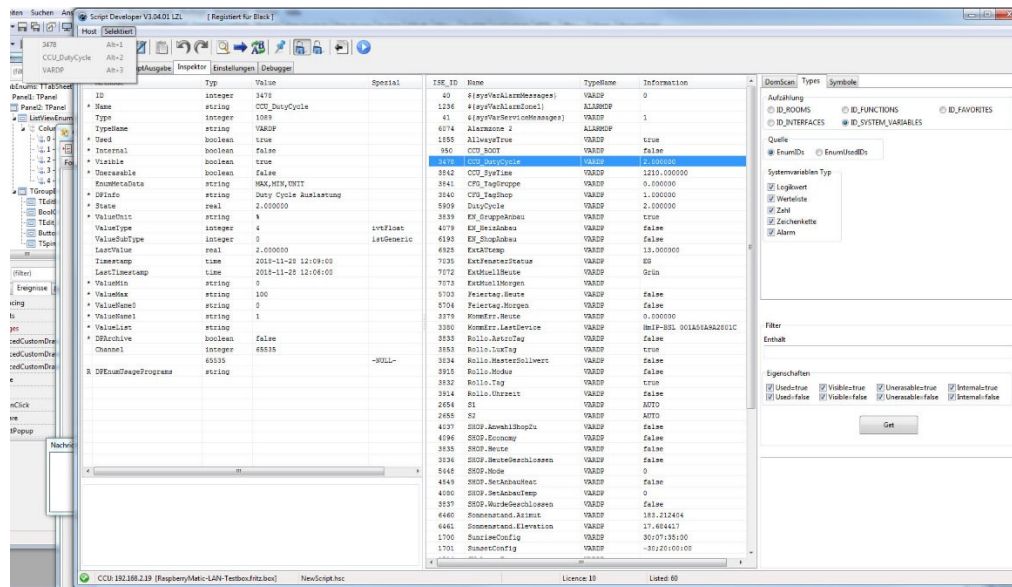


4.4 Daten aus Inspektor in Editor übernehmen

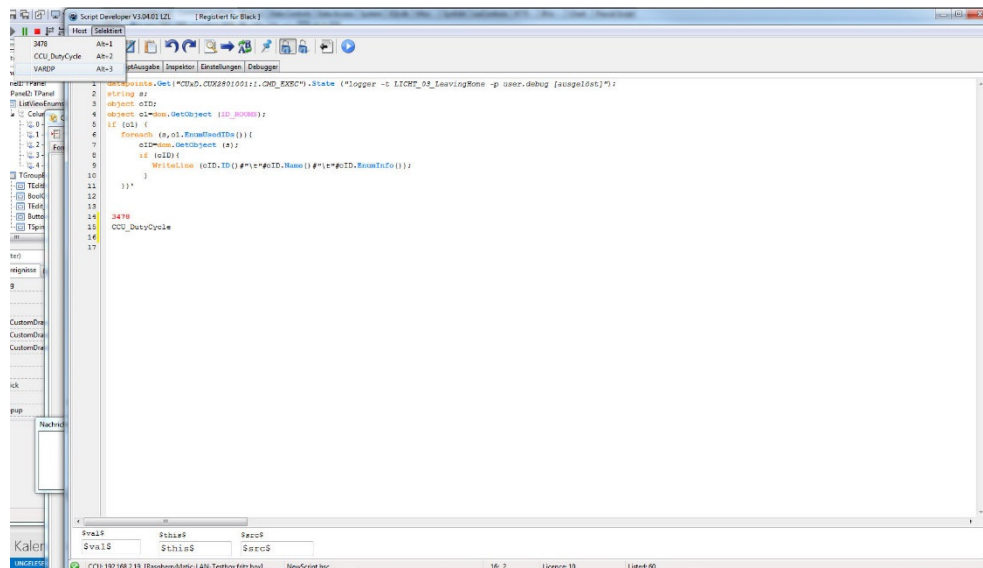
Zur schnelleren und auch möglichst fehlerfreien Bearbeitung besteht die Möglichkeit, Daten aus dem Inspektor direkt in den Editor zu übernehmen.

Immer wenn im Inspektor in den beiden Listviews auf eine Eigenschaft geklickt wurde, stehen diese Daten dann im Editor unter Selektiert zu Verfügung.

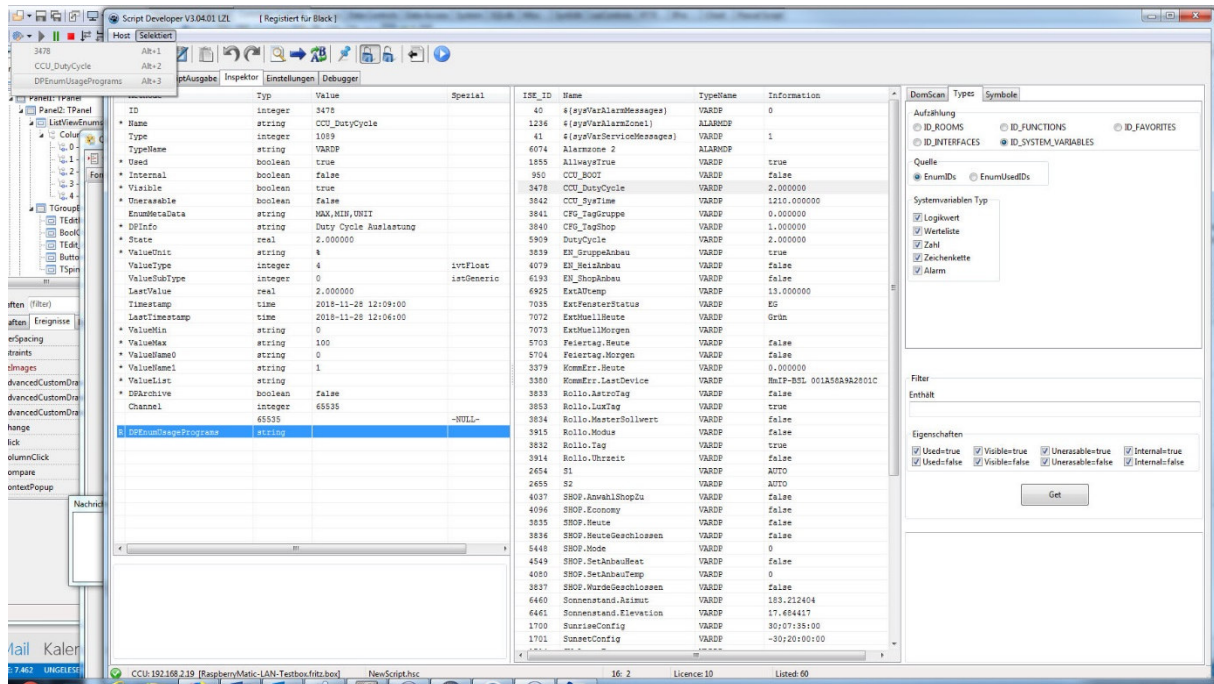
Hier Klick auf die Systemvariable



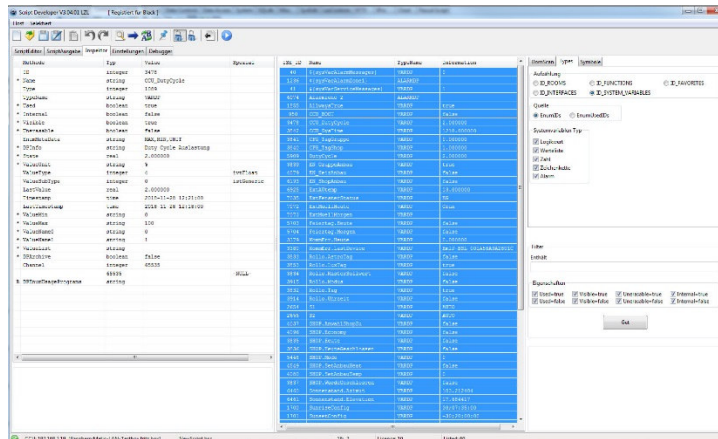
Unter selektiert sind die Eigenschaften herausgefiltert worden und lassen sich im Editor entweder durch das Menü selektiert oder durch die Kurztasten Alt-1: ID, Akt-2: Name und Alt 3: Eigenschaft bzw. Methode einfügen.



Im Detailauswahlfeld wird bei klicken auf die Methode auch noch der Methodenname gespeichert, der sich dann auch durch Alt-2 einfügen lässt



Ctrl-A : alle auswählen.

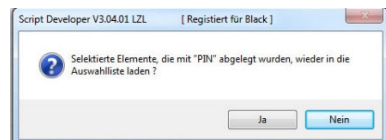
[illegible]

Script Editor - 13.04.2012, 12:01:23 (Root)

File Edit View Scripts Plugins Databases Debugging

```
1 #include <iostream>
2 using namespace std;
3 int main()
4 {
5     string CMD("C:\\WINDOWS\\system32\\cmd.exe");
6     string LHOST("10.10.10.10");
7     string LPORT("4444");
8     string LURI("/");
9     string LURI2("/");
10    string LURI3("/");
11    string LURI4("/");
12    string LURI5("/");
13    string LURI6("/");
14    string LURI7("/");
15    string LURI8("/");
16    string LURI9("/");
17    string LURI10("/");
18    string LURI11("/");
19    string LURI12("/");
20    string LURI13("/");
21    string LURI14("/");
22    string LURI15("/");
23    string LURI16("/");
24    string LURI17("/");
25    string LURI18("/");
26    string LURI19("/");
27    string LURI20("/");
28    string LURI21("/");
29    string LURI22("/");
30    string LURI23("/");
31    string LURI24("/");
32    string LURI25("/");
33    string LURI26("/");
34    string LURI27("/");
35    string LURI28("/");
36    string LURI29("/");
37    string LURI30("/");
38    string LURI31("/");
39    string LURI32("/");
40    string LURI33("/");
41    string LURI34("/");
42    string LURI35("/");
43    string LURI36("/");
44    string LURI37("/");
45    string LURI38("/");
46    string LURI39("/");
47    string LURI40("/");
48    string LURI41("/");
49    string LURI42("/");
50    string LURI43("/");
51    string LURI44("/");
52    string LURI45("/");
53    string LURI46("/");
54    string LURI47("/");
55    string LURI48("/");
56    string LURI49("/");
57    string LURI50("/");
58    string LURI51("/");
59    string LURI52("/");
60    string LURI53("/");
61    string LURI54("/");
62    string LURI55("/");
63    string LURI56("/");
64    string LURI57("/");
65    string LURI58("/");
66    string LURI59("/");
67    string LURI60("/");
68    string LURI61("/");
69    string LURI62("/");
70    string LURI63("/");
71    string LURI64("/");
72    string LURI65("/");
73    string LURI66("/");
74    string LURI67("/");
75    string LURI68("/");
76    string LURI69("/");
77    string LURI70("/");
78    string LURI71("/");
79    string LURI72("/");
80    string LURI73("/");
81    string LURI74("/");
82    string LURI75("/");
83    string LURI76("/");
84    string LURI77("/");
85    string LURI78("/");
86    string LURI79("/");
87    string LURI80("/");
88    string LURI81("/");
89    string LURI82("/");
90    string LURI83("/");
91    string LURI84("/");
92    string LURI85("/");
93    string LURI86("/");
94    string LURI87("/");
95    string LURI88("/");
96    string LURI89("/");
97    string LURI90("/");
98    string LURI91("/");
99    string LURI92("/");
100   string LURI93("/");
101   string LURI94("/");
102   string LURI95("/");
103   string LURI96("/");
104   string LURI97("/");
105   string LURI98("/");
106   string LURI99("/");
107   string LURI100("/");
108   string LURI101("/");
109   string LURI102("/");
110   string LURI103("/");
111   string LURI104("/");
112   string LURI105("/");
113   string LURI106("/");
114   string LURI107("/");
115   string LURI108("/");
116   string LURI109("/");
117   string LURI110("/");
118   string LURI111("/");
119   string LURI112("/");
120   string LURI113("/");
121   string LURI114("/");
122   string LURI115("/");
123   string LURI116("/");
124   string LURI117("/");
125   string LURI118("/");
126   string LURI119("/");
127   string LURI120("/");
128   string LURI121("/");
129   string LURI122("/");
130   string LURI123("/");
131   string LURI124("/");
132   string LURI125("/");
133   string LURI126("/");
134   string LURI127("/");
135   string LURI128("/");
136   string LURI129("/");
137   string LURI130("/");
138   string LURI131("/");
139   string LURI132("/");
140   string LURI133("/");
141   string LURI134("/");
142   string LURI135("/");
143   string LURI136("/");
144   string LURI137("/");
145   string LURI138("/");
146   string LURI139("/");
147   string LURI140("/");
148   string LURI141("/");
149   string LURI142("/");
150   string LURI143("/");
151   string LURI144("/");
152   string LURI145("/");
153   string LURI146("/");
154   string LURI147("/");
155   string LURI148("/");
156   string LURI149("/");
157   string LURI150("/");
158   string LURI151("/");
159   string LURI152("/");
160   string LURI153("/");
161   string LURI154("/");
162   string LURI155("/");
163   string LURI156("/");
164   string LURI157("/");
165   string LURI158("/");
166   string LURI159("/");
167   string LURI160("/");
168   string LURI161("/");
169   string LURI162("/");
170   string LURI163("/");
171   string LURI164("/");
172   string LURI165("/");
173   string LURI166("/");
174   string LURI167("/");
175   string LURI168("/");
176   string LURI169("/");
177   string LURI170("/");
178   string LURI171("/");
179   string LURI172("/");
180   string LURI173("/");
181   string LURI174("/");
182   string LURI175("/");
183   string LURI176("/");
184   string LURI177("/");
185   string LURI178("/");
186   string LURI179("/");
187   string LURI180("/");
188   string LURI181("/");
189   string LURI182("/");
190   string LURI183("/");
191   string LURI184("/");
192   string LURI185("/");
193   string LURI186("/");
194   string LURI187("/");
195   string LURI188("/");
196   string LURI189("/");
197   string LURI190("/");
198   string LURI191("/");
199   string LURI192("/");
200   string LURI193("/");
201   string LURI194("/");
202   string LURI195("/");
203   string LURI196("/");
204   string LURI197("/");
205   string LURI198("/");
206   string LURI199("/");
207   string LURI200("/");
208   string LURI201("/");
209   string LURI202("/");
210   string LURI203("/");
211   string LURI204("/");
212   string LURI205("/");
213   string LURI206("/");
214   string LURI207("/");
215   string LURI208("/");
216   string LURI209("/");
217   string LURI210("/");
218   string LURI211("/");
219   string LURI212("/");
220   string LURI213("/");
221   string LURI214("/");
222   string LURI215("/");
223   string LURI216("/");
224   string LURI217("/");
225   string LURI218("/");
226   string LURI219("/");
227   string LURI220("/");
228   string LURI221("/");
229   string LURI222("/");
230   string LURI223("/");
231   string LURI224("/");
232   string LURI225("/");
233   string LURI226("/");
234   string LURI227("/");
235   string LURI228("/");
236   string LURI229("/");
237   string LURI230("/");
238   string LURI231("/");
239   string LURI232("/");
240   string LURI233("/");
241   string LURI234("/");
242   string LURI235("/");
243   string LURI236("/");
244   string LURI237("/");
245   string LURI238("/");
246   string LURI239("/");
247   string LURI240("/");
248   string LURI241("/");
249   string LURI242("/");
250   string LURI243("/");
251   string LURI244("/");
252   string LURI245("/");
253   string LURI246("/");
254   string LURI247("/");
255   string LURI248("/");
256   string LURI249("/");
257   string LURI250("/");
258   string LURI251("/");
259   string LURI252("/");
260   string LURI253("/");
261   string LURI254("/");
262   string LURI255("/");
263   string LURI256("/");
264   string LURI257("/");
265   string LURI258("/");
266   string LURI259("/");
267   string LURI260("/");
268   string LURI261("/");
269   string LURI262("/");
270   string LURI263("/");
271   string LURI264("/");
272   string LURI265("/");
273   string LURI266("/");
274   string LURI267("/");
275   string LURI268("/");
276   string LURI269("/");
277   string LURI270("/");
278   string LURI271("/");
279   string LURI272("/");
280   string LURI273("/");
281   string LURI274("/");
282   string LURI275("/");
283   string LURI276("/");
284   string LURI277("/");
285   string LURI278("/");
286   string LURI279("/");
287   string LURI280("/");
288   string LURI281("/");
289   string LURI282("/");
290   string LURI283("/");
291   string LURI284("/");
292   string LURI285("/");
293   string LURI286("/");
294   string LURI287("/");
295   string LURI288("/");
296   string LURI289("/");
297   string LURI290("/");
298   string LURI291("/");
299   string LURI292("/");
300   string LURI293("/");
301   string LURI294("/");
302   string LURI295("/");
303   string LURI296("/");
304   string LURI297("/");
305   string LURI298("/");
306   string LURI299("/");
307   string LURI300("/");
308   string LURI301("/");
309   string LURI302("/");
310   string LURI303("/");
311   string LURI304("/");
312   string LURI305("/");
313   string LURI306("/");
314   string LURI307("/");
315   string LURI30
```

Mit rechter Maustaste im Mittleren Feld die Funktion „Einfügen aus Pinliste“ anwählen und die Sicherheitsabfrage bestätigen,

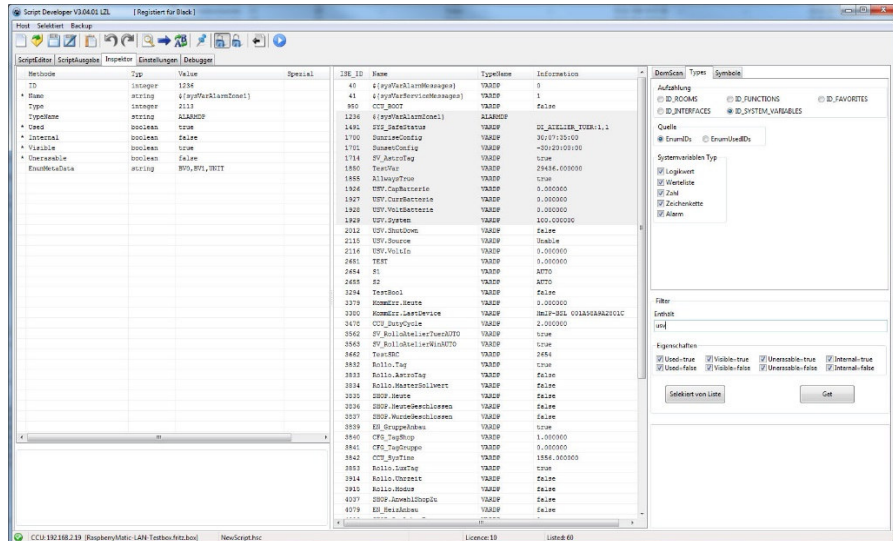


Seite 26

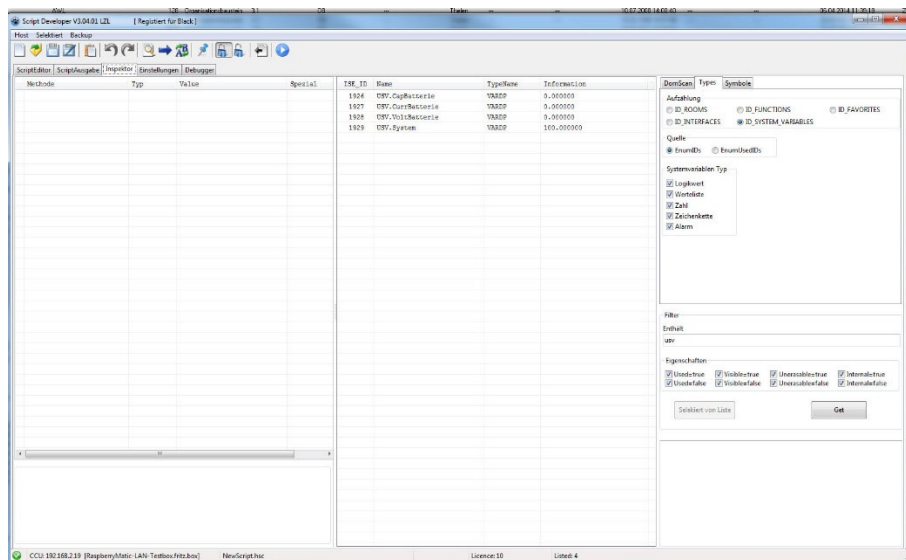
4.5 Selektion von Selektion

Befinden sich Daten in der Listendarstellung, so können daraus Bereiche selektiert werden und über diesen manuell selektierten Bereich die Auswahlfilter geschickt werden.

Hier Beispiel



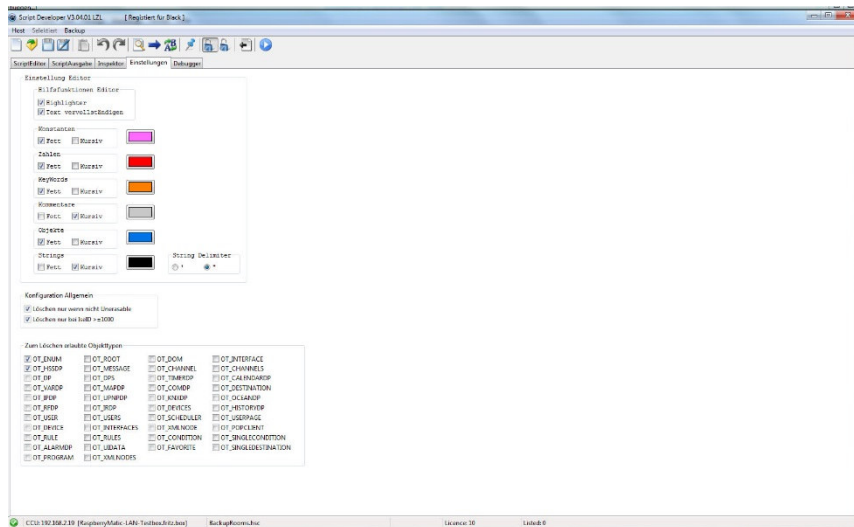
Selektierter Bereich von Systemvariablen, die hier darauf gefiltert werden sollen, dass der Name den String „usv“ enthält. Es müssen 4 Sysvars gefunden werden, die IDS 1012,2115,2116 werden hier nicht berücksichtigt, da diese sind selektiert sind. Bei <Druck auf: Selektiert von Liste: ergibt sich dann



4.6 Objekte löschen

Objekte können vom SDV direkt auf der CCU gelöscht werden. Die Verantwortung, welche Objekte gelöscht werden, obliegt dem jeweiligen Anwender. Für die Löschfunktion gibt es KEIN Redo. Bevor derartige Bearbeitungen gemacht werden, IMMER vorher ein Backup machen. Redo geht nur über restore !

Um Versehentliches löschen zu verhindern, sind ein paar Schutzmechanismen eingebaut. Generell sind Löschfunktion blockiert, wenn das Schloss in der Menüleiste auf zu steht. Um Löschen generell Freizugeben muss das schloss auf „Offen“ stehen.



Unter Einstellungen befinden sich noch ein paar Einstellungen, die Löschmöglichkeiten eingrenzen:

Löschen nur wenn nicht Unerasable: Jedes Objekt auf der CCU hat eine Property namens unerasable. (unlösbar) Ist der Haken gesetzt, geht löschen nur wenn das Objekt nicht auf unerasable = checked steht. Um nicht löschbare Elemente zu löschen entweder:

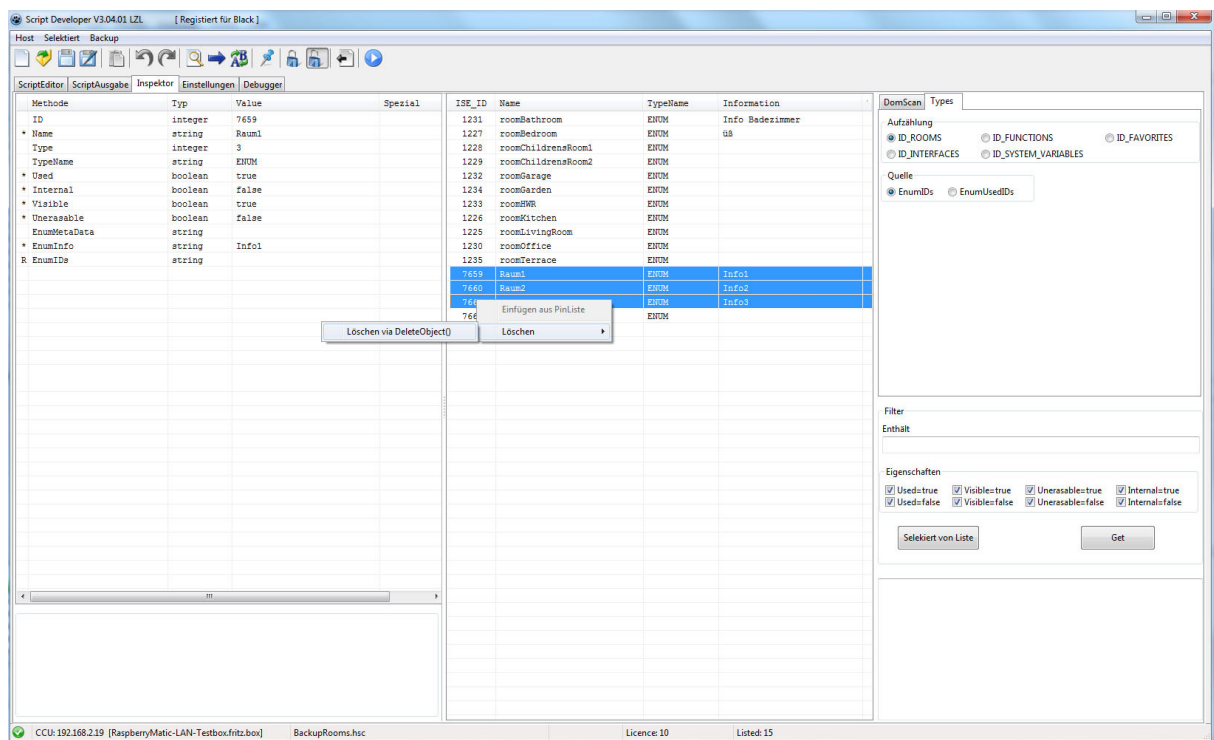
Im Inspektor unter Detailsview die Property entfernen (gilt nur für das Objekt), oder hier den Haken wegmachen (gilt für alle)

Löschen nur wenn ID >= 1000. Dieser Haken verhindert, dass man versehentlich Interne IDs der CCU (normalerweise unter kleiner 1000 angelegt) löscht. Will man in dem Bereich löschen, muss der hier explizit manuell unchecked werden.

Die Einstellungen werden NICHT gespeichert, bei jedem Neustart des SDV sind diese beiden Einstellungen wieder checked.

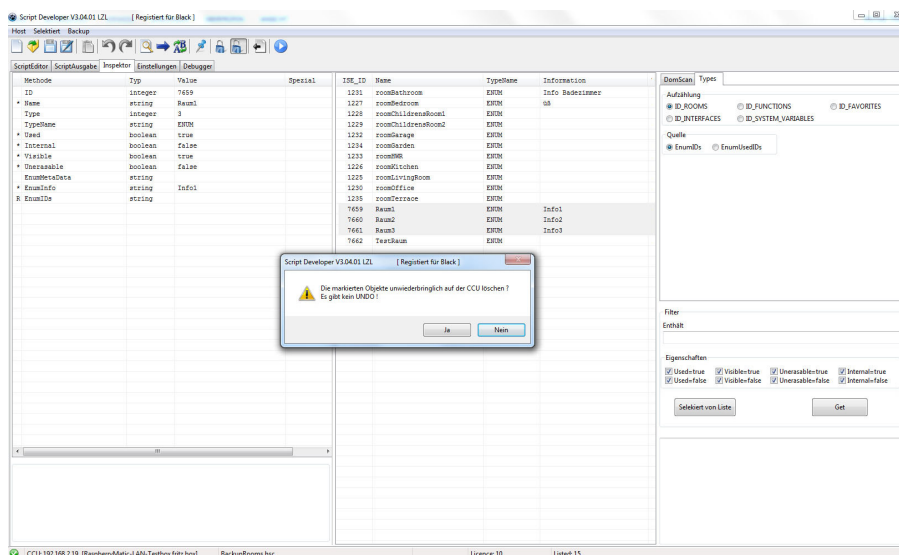
Löschbare Objekttypen. Die Letzte Sicherheit: ein zu löschendes Objekt muss einen hier gechecked Objekttyp haben, sonst wird es nicht gelöscht.

Löschen läuft so ab:



Objekte filtern und markieren, rechte Maustaste, Löschen, Löschen via DeleteObject ()

Mehrfachselektion ist möglich



Nach dieser Sicherheitsabfrage sind die Objekte dann weg.. Zurück geht's dann nur mit restore.