

## Kurzanleitung ScriptDeveloper SDV V3.09.04 (26. Oktober 2019)

Der Scriptdeveloper (SDV) soll ein Hilfsmittel im Alltag bei der Erstellung von Homematic Skripten 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.

Da mittlerweile aber schon einige tiefgreifende Operationen möglich sind, sind 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 SDV V3.09.04 (26. Oktober 2019) .....	1
1. Installation.....	5
1.1 Lizensierung.....	7
1.2 Donate .....	12
1.3 Systemvoraussetzungen.....	13
1.4 Was tut es bis jetzt .....	14
1.5 Bekannte Einschränkungen / Bugs.....	15
1.6 Geänderte Anforderung an Auflösung .....	15
1.6.1 Mehrfacher Start .....	15
1.7 Changelog .....	16
1.7.1 Changelog 03.09.03 LCL.....	16
1.7.2 Changelog 03.09.02 LCL.....	16
1.7.3 Changelog 03.09.01 LCL.....	16
1.7.4 Changelog 03.08.14 LCL.....	16
1.7.5 Changelog 03.08.13 LCL.....	16
1.7.6 Changelog 03.08.12 (A-D) LCL .....	16
1.7.7 Changelog 03.08.11 LCL.....	16
1.7.8 Changelog 03.08.10 LCL.....	16
1.7.9 Changelog 03.08.09 LCL.....	17
1.7.10 Changelog 03.08.07 LCL.....	17
1.7.11 Changelog 03.08.05 LCL.....	17
1.7.12 Changelog 03.08.04 LCL.....	17
1.7.13 Changelog 03.08.01 LCL.....	17
1.7.14 Changelog 03.07.13 LZL.....	17
1.7.15 Changelog 03.07.12 LZL.....	17
1.7.16 Changelog 03.07.11 LZL.....	18
1.7.17 Changelog 03.07.10 LZL.....	18
1.7.18 Changelog 03.07.07 LZL.....	18
1.7.19 Changelog 03.07.05 LZL.....	18
1.7.20 Changelog 03.07.04 LZL.....	18
1.8 Neue Authentifizierungsmechanismen Raspberrymatik .....	19
1.8.1 SSL.....	19
1.8.2 Authentifizierung über Nutzername / Passwort .....	19
1.9 Ein Wort des Authors zum Thema „gewerbliche Nutzung“ .....	21
2. Oberfläche .....	22

2.1	Word Markup (seit 03.07.02) .....	23
2.1.1	SyncroEdit (ab 3.08.10) .....	24
2.1.2	Schriftgrößen und Schriftart (ab 3.08.07) .....	25
2.2	Blockweise auskommentieren. ....	26
3	Scripteditor.....	27
3.1	Voreinstellungen Editor.....	29
3.2	Vervollständigen Funktion .....	30
3.3	Methodenhilfe.....	32
3.4	Symbolischer Zugriff auf selektierte Datenpunkte des Inspektors .....	34
4	Inspektor.....	35
4.1	Selektionswahl: DomScan .....	36
4.2	Selektionskriterium Types .....	39
4.3	Zusätzliche Selektionsbedingungen .....	40
4.4	Daten aus Inspektor in Editor übernehmen.....	47
4.4.1	Mehrfachauswahl als Enum String .....	49
4.4.2	Übernahme von einem Script aus einem Programm direkt in den Editor .....	51
4.5	Selektion von Selektion .....	54
4.6	Objekte löschen.....	55
4.7	Anwenderdefinierte Sichten .....	57
4.8	Browsing durch Rekursionsebenen.....	58
4.9	Auflösen komplexer Objekte .....	61
4.9.1	Auflösen von Devices .....	61
4.9.2	Auflösen von Programmen.....	63
4.10	Zeitmodule .....	64
4.11	Suchen in Scripten nach Variablen, Devices etc.....	65
4.12	Volltextsuche in Scripten.....	68
4.13	SingleDestinations in ihrer Reihenfolge ändern.....	70
4.14	History Data Points und Systemprotokoll .....	73
5	Favoritenansicht .....	74
5.1	Hinzufügen von Objekten in die Favoriten.....	74
5.2	Verwendete Objekte eines Programmes in die Favoriten laden .....	75
5.3	Favoritenliste löschen.....	75
5.4	Einschränkungen .....	75
6	Kleine Helfer im Alltag .....	76
6.1	Backups.....	76
6.1.1	Räume.....	76

6.1.2 Gewerke .....	77
6.1.3 Systemvariablen .....	78
6.1.4 Devices und Kanäle.....	78
6.1.5 Backup Programme .....	79
6.1.6 Backup Masterset.....	84
6.1.7 Backup Linkset.....	85
6.1.8 Backup Gerät von der WebUI.....	86
6.2 Umbenennen von Kanälen von Geräten .....	88
6.3 Paramset Master .....	89
6.4 Linkset eines Gerätes.....	91
6.5 Rega Push auf Datenpunkte via Rega event.....	92
6.6 Querverweise zu Datenpunkten in WebUI-Programmen .....	93
<b>7 Diagnosebild .....</b>	<b>94</b>
7.1 Allgemein.....	94
7.1.1 Liste Konstanten .....	94
7.1.2 Regadom.....	94
7.1.3 Systemübersicht .....	94
7.1.4 CacheInfo.....	95
7.1.5 Regadom Aktualisierung.....	95
7.1 SSH Funktionalität .....	96
7.1.1 SSH Realisierung (CUxD ab 2.3.1 oder plink.exe) .....	98
7.2 Programm Condition Konsistenz.....	99
7.3 Programmanalyse.....	101
7.4 Systemvariablen Metadatenkonsistenz .....	105
7.5 Systemvariablen Test auf NirvanaChannelBezüge .....	107
7.6 Test und Korrektur Zeitmodule .....	108
7.7 Test und Korrektur HistoryDPs.....	109
7.8 Check Internal Sysvar .....	111
<b>8 License Disclaimer .....</b>	<b>112</b>

## 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
; Zusätzliche Dateitypen zur Filterung
; *.BMP;*.JPG;*.GIF
; die Aufzählung muss mit Semikolon getrennt sein
FILETYPE=                                Beispiel *.htm;*.scr
; Scriptdir
; ist Scriptdir leer, so wird Standartmäßig das Verzeichnis
; genommen, in dem der SDV gestartet wird
SCRIPTDIR=                                Beispiel c:\Scriptdir\

[HOST]
NICKNAME=DerNickNameAusDerLizenzenanfrage
CUXD=CUXD.CUX2801001:5
SYSTEMEXEC=true                           Aufruf wenn möglich über systemExec statt CuxD
LICENCE1=DerLizenzschlüssel1
LICENCE2=DerLizenzschlüssel2

[CCU1]                                     Neuer Schlüssel
IP=192.168.2.19                            Ip Von CCU1
USERNAME=ExternAdmin                        Nutzernname auf der CCU1
PASSWORD=XXXXXXXXXXXXX                      Passwort der Nutzers auf der CCU1
USEHTTPS=true                               Zugriff über HTTPS
SSHUSERNAME=root                            root wenn SSH Zugriff erwünscht
SSHPW=xxxxx^                                Das SSH Passwort der CCU eintragen

[CCU2]                                     Dito wie bei CCU1
IP=192.168.2.6
USERNAME=
PASSWORD=
USEHTTPS=false                             root wenn SSH Zugriff erwünscht
SSHUSERNAME=root                            Das SSH Passwort der CCU eintragen
SSHPW=xxxxx^

[HOSTCCU]                                    Die Werte der aktuellen CCU bei Start
IP=192.168.2.19                            im Idealfall hier die Daten von CCU1 oder 2
                                              eintragen für 1. Start
USERNAME=ExternAdmin
PASSWORD=XXXXXXXXXXXXX
USEHTTPS=True
SSHUSERNAME=root                            root wenn SSH Zugriff erwünscht
SSHPW=xxxxx^                                Das SSH Passwort der CCU eintragen

[SECURITY]                                    Die Ports die der SDV benutzt
HTTPREGASCRIPt=8181
HTTPREGAXMLRPC=1999
HTTPSREGASCRIPt=48181
HTTPSREGAXMLRPC=41999
```

[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

Hinweis für CCU2 Nutzer oder Nutzer älterer Firmware: Wenn sich kein Anfrageschlüssel generieren lässt (keine Hashs erkannt), in der INI Datei prüfen, ob dort der entsprechenden CCU der Eintrag USEHTTPS=True steht. Damit klappt es nicht. In diesem Fall muss der Eintrag in USEHTTPS=False geändert werden, dann klappt auch. Alte Firmwares mögen es auch nicht, wenn über Nutzernname und PW ein authentifizierter Zugriff über Name:PW@Host versucht wird. Bei älteren Firmware oder CCU keinen Nutzernamen oder Passwort eintragen bei CCU1 / CCU2 CCUHOST

Bei CCU2:

Auf jeden Fall USEHTTPS=false sowie USERNAME= und PASSWORD= (Username und Password leerlassen, sonst gibt's keine HashAnfrage

Bei CCU3:

Bei Authentifizierung auf CCU3 ein: USERNAME und PASSWORT des Admins müssen eingetragen sein

BestandsNutzer:

Die Ini Datei hat sich ziemlich vergrößert. Hilfreich ist, in der mitgelieferten INI alle Schlüssel ab [CFG\_ChanView1\_Methods]

Address=True.....

Und in die bestehende INI Datei anzuhängen. Erspart eine Menge manueller Hakensetzen.

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 exclusiven Zugriff hat. Auf Systemen ohne CUxD kann der SDV nicht eingesetzt werden.

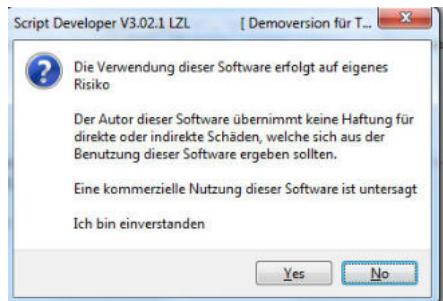
## 1.1 Lizenierung

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 einzuschränken. Dies geschieht durch Vergabe von bis zu 2 Lizenzschlüsseln. Der SDV ist dadurch an bis zu 2 CCU / Raspberrymatic gepaart.

Wie arbeitet das ?

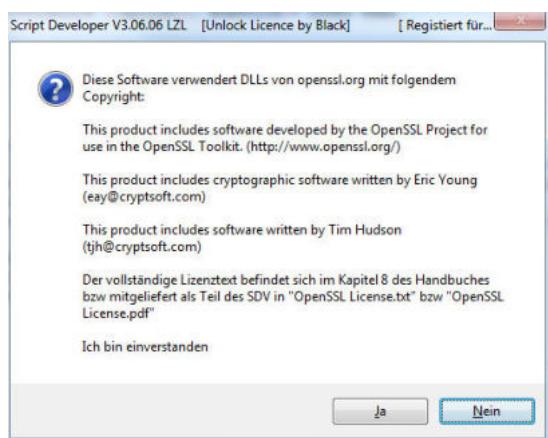
Der SDV telefoniert nicht nach Hause. Um eine Lizenz anzufragen ist folgender Weg einzuschlagen.

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



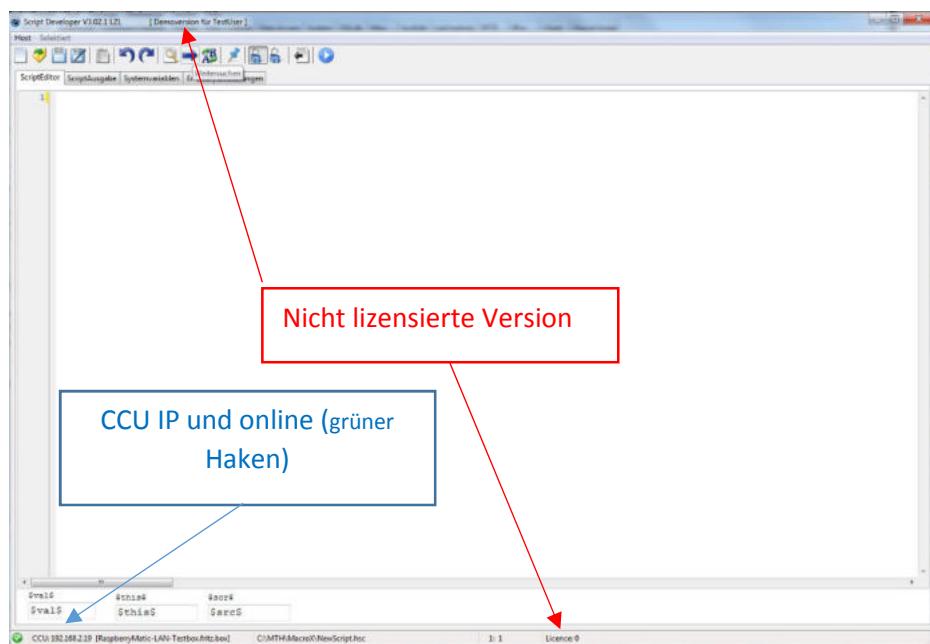
Bei allerersten Start muss dieses Fenster mit yes bestätigt werden. No führt so einem sofortigen Programmabbruch

Ab der Version 3.06.06 befinden sich in dem Package des SDV 2 DLL's von OpenSSL.org, welche für den HTTPS Zugang zur CCU benötigt werden. Hierbei ist einmalig ebenfalls eine Zustimmung über die Verwendung der Software nötig. Der komplette Lizenztext kann im Kapitel 8 nachgelesen werden bzw. ist als Datei mit in dem Package enthalten.

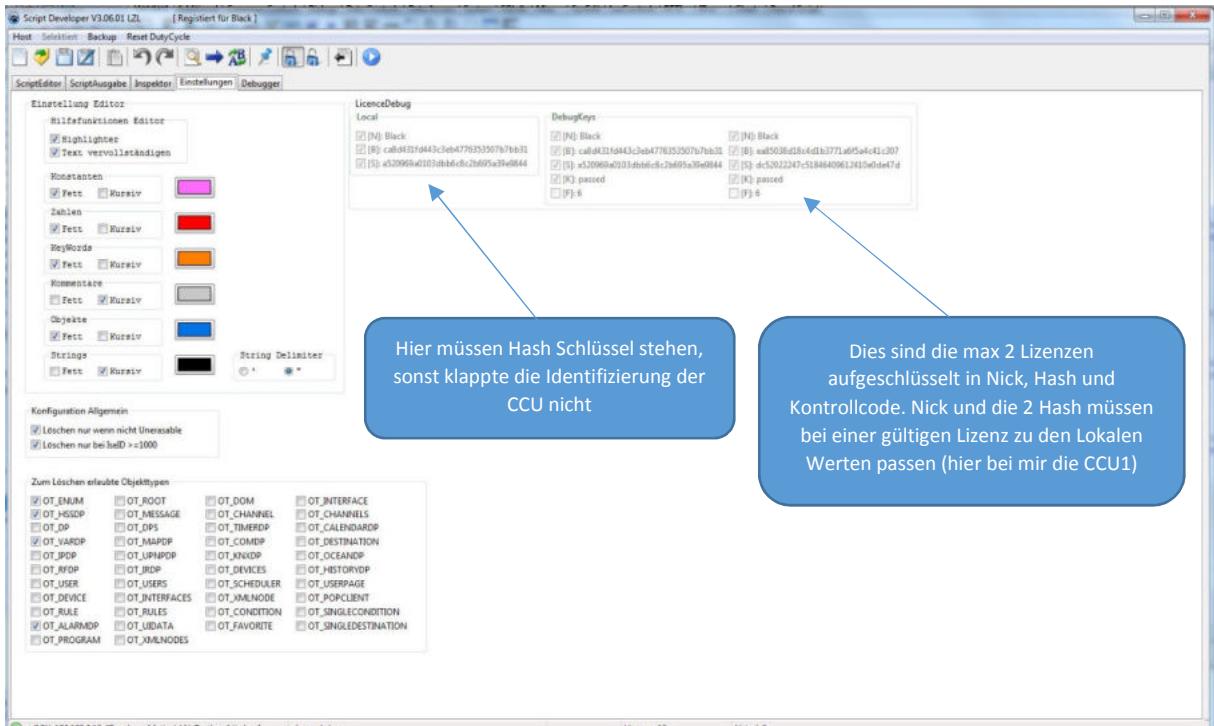


„Nein“ führt auch hier zu einem sofortigen Programmabbruch

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



Wenn die CCU, für die der Schlüssel angefragt werden soll, als grün angezeigt wird, bitte vorher einmal unter dem Reiter Einstellungen kontrollieren

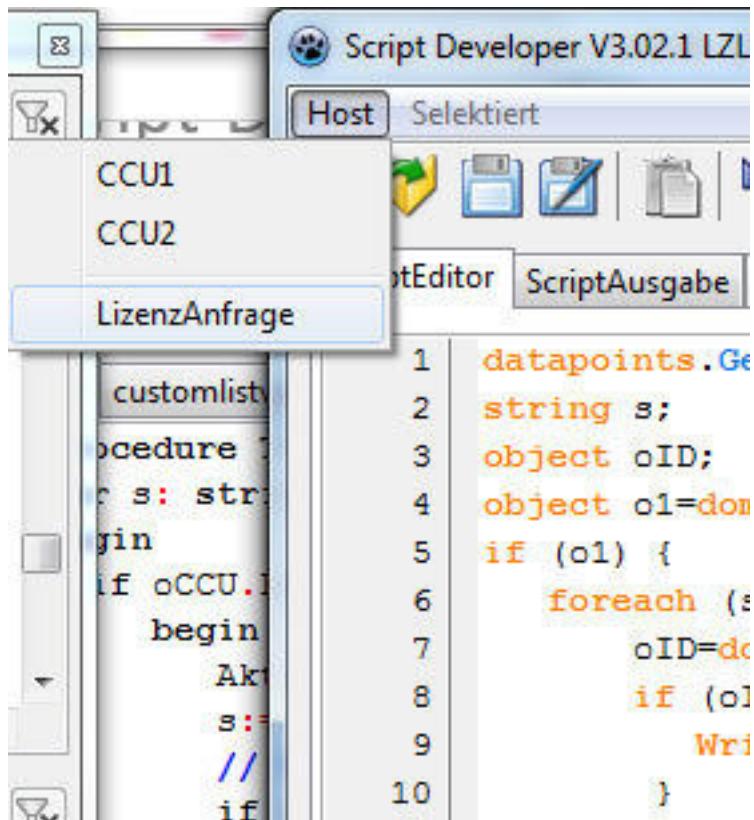


Aus einer Anfrage ohne unter local sinnige Einträge zu sehen lässt sich kein gültiger key generieren.

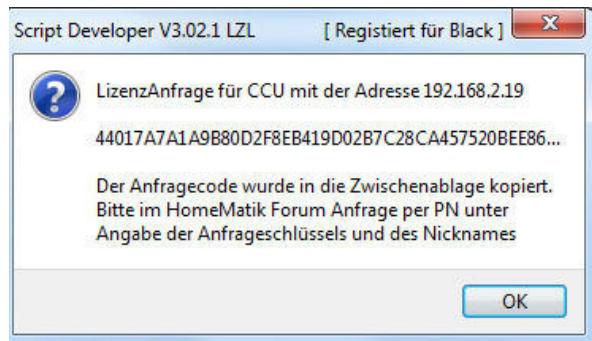
Aufgrund einer Programmungsgenauigkeit bei der Ausformulierung einer RegEx ist es leider sehr wahrscheinlich, dass alte Lizenzschlüssel vor 3.5.2 als ungültig erkannt werden. Sorry dafür, in dem Fall bitte die Anfrageschlüssel neu erstellen (wenn die Hashwerte gültig sind) und mir schicken. Der SDV 3.5.0 ist davon nicht betroffen, also am besten den 3.5.2 in ein neues Verzeichnis entpacken, die SDV.INI von dem 3.5.0 kopieren ins 3.5.2 Verzeichnis und neue Anfrage machen unter dem 3.5.2, da die alte INI nicht verändert wurde im alten Verzeichnis kann solange dann noch mit der 3.5.0 Version gearbeitet werden.

Im Auslieferungszustand ist unter NICKNAME= noch TestUser eingetragen. Mit TestUser lässt sich kein Key generieren. Hierbei dann bitte die INI nochmal anpassen und den SDV neu starten-

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.

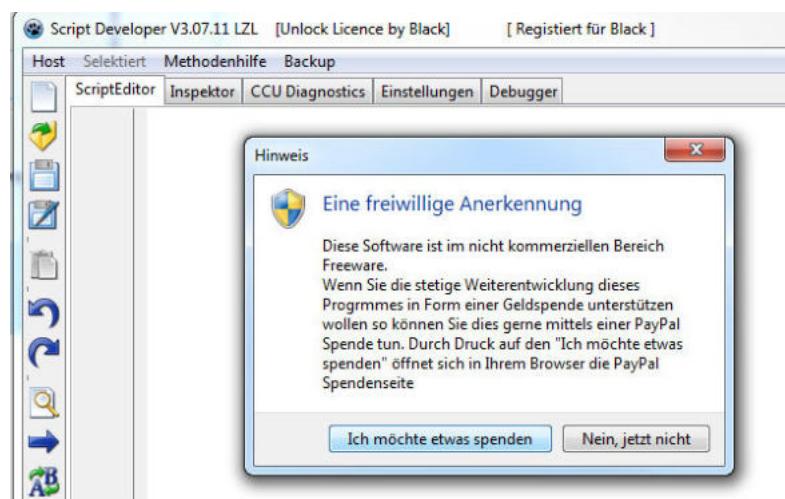
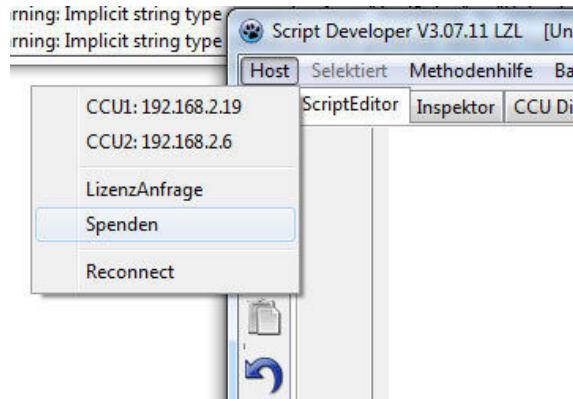
Wer mit diesem Verfahren nicht einverstanden ist, möge bitte an dieser Stelle die PDF Datei schließen und kann die Dateien beruhigt löschen.

Geplant hab ich folgende Lizenzabstufungen

Level	Editor	Script ausführen	Highlighter Und Vervollständiger	Enums	SysVar	Programs	Backup Restore		Special Funktions (resetDC, restor950 etc)
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									
7	X	X	X	X	X	X	X		X

## 1.2 Donate

Es besteht die Möglichkeit, dem Autor dieses Programmes für seine Arbeit eine Spende über Paypal zukommen zu lassen. Diese Möglichkeit ist völlig losgelöst von der Vergabe von Lizenzschlüsseln und rein freiwillig.



### 1.3 Systemvoraussetzungen

Der SDV lief bisher in Testinstallionen 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). In einer AppleVM haben es auch schon welche stabil zum laufen bekommen.

Auf der Homematic-Seite wurde bei mir auf einer Raspberrymatic 3.47.18.20190918 getestet.

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

Eventuell werde ich im Laufe des Jahres 2019 mal schauen, ob ich es nativ unter Debian auch kompiliert und zum Laufen bekomme. Wahrscheinlich wird dieses Projekt aber zurückgestellt, da leider das Threathandling unter Linux sich völlig anders wie Windows verhält.

Ab der Version 3.09.xx wird der SDV unter Lazarus 2.0.4 als 64Bit Programm kompiliert.

In dem Package sind nun auch die SSL DLLs für die 64bit Version enthalten. In einem Bestandssystem müssen dann die DLLs ausgetauscht werden. Ein 64bit SDV mit 32bit DLLs führt zu einem SIGTERM.

## 1.4 Was tut es 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

Devices

Aufschlüsseln der MetaDaten

Datenpunkte

Kanäle

Programme

Favoriten

User

Paramset Master

Paramset Links

Script Bearbeitung aus Programmen heraus

Verändert von Verzögert um von Scripten

Verändern von Retriggern

Komplettes Aufschlüsseln von Programmen in Rules, Subrules, Conditions, SingleConditions,

Destination und SingleDestination

MarkupUp im Editor

Suchen in Scripten nach Namen von Systemvariablen, Devices, Channels, Rooms und Functions

Suchen in Scripten nach Seriennummer von Devices und Channels

VollTextsuche in Scripten auch mit Regex Ausdrücken

Backup von Räumen, Gewerken, Favoriten, Systemvariablen, Devicenamen und Programmen

Backup von Mastersets

Backup von LinkSets

Backup von WebUI Einstellungen eines Gerätes (Zuordnung zu Räumen, Gewerken Favoriten und Programmen)

Diverse Konsistenstests

Auflösen von Direktverbindungen

## 1.5 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.

## 1.6 Geänderte Anforderung an Auflösung

Nach Rückmeldung aus dem Testerkreis bezüglich dem Herabsetzen der nötigen Bildschirmauflösung habe ich dieses etwas neu aufgebaut.

Der SDV startet mit Höhe 769 und Breite 1300 und kann angepasst werden. Die Einstellungen werden beim Verlassen gespeichert.

Mit dem Kommandozeilenparameter SDV\_xxxx formreset kann eine Bildschirmauflösung wieder auf den Grundzustand zurückgesetzt werden. Alternativ die Einträge unter dem Schlüssel FORMVIEW in der INI Datei löschen.

### 1.6.1 Mehrfacher Start

Der SDV kann mittlerweile mit mehreren Instanzen Gleichzeitig gestartet werden. Dabei hat allerdings nur die zuerst gestartete Instanz Schreibrechte auf die INI Datei. So lassen sich mehrere Scripte gleichzeitig bearbeiten und testen oder auch gleichzeitig mehrere Inspektoren benutzen.

Zu empfehlen ist auch, z.b. die Dateiendung .hsc mit dem SDV zu verknüpfen, so öffnet ein Doppelklick auf eine Scriptdatei dann automatisch den SDV mit dem geladenen Script

## 1.7 Changelog

### 1.7.1 Changelog 03.09.03 LCL

Mehrere kleinere Bugfixes

Restore Möglichkeit, wenn ein Device gelöscht wurde, um sämtliche Punkte der WEBUI wieder zu rekonstruieren

Linkset Auflösen von Level6 auf Level 5 herabgesetzt

Master und Linkset Level von 6 auf 5 heruntergesetzt

Logmöglichkeit für Backup programme eingeführt

### 1.7.2 Changelog 03.09.02 LCL

Kleinere Bugfixes

Parameter im Linkset können markiert werden und automatischer Editorcodeerzeugt werden

Editor kann mit Human friendly JSON umgehen

Auflösen von Direktverbindungen selektierte Geräte

Sortieren der Direktverbindungen nach Sender, Empfänger

Finden Gleichtartiger Geräte

Echtzeithilfe eingefügt

Automatische Codegenerierung für Datenpunkte (Vollsymbolischer Zugriff)

Konfigurierbarer Threatkill Timeout

Für die jeweiligen CCUs separates BackupDir eingefügt

### 1.7.3 Changelog 03.09.01 LCL

Update der RAD auf 2.0.4

Umbau im Untergrund, um die Classen für die geplanten Master und Linkset Manipulationen vorzubereiten

Auflösen von Linksets möglich

Ändern von Namen und Beschreibung von Linksets

Backup eines Linksets

### 1.7.4 Changelog 03.08.14 LCL

Letzte Version unter 32 bit unter Lazarus 1.8.4

Backup Devices eingeführt

Auflösen von Linksets möglich

### 1.7.5 Changelog 03.08.13 LCL

Inspektor: Device mit allen Kanälen aus Raumlisten löschen

Inspektor: Device mit allen Kanälen aus Gewerkelisten löschen

Konsistenzcheck History DPS (Ungültige und verwaiste History DPs aufspüren und beseitigen)

### 1.7.6 Changelog 03.08.12 (A-D) LCL

Proof of Concept Version mit Umgebauter interner Verarbeitung des Highlighters

B-Hash Berechnung entfernt, EQ3 kriegt unter bestimmten Voraussetzungen keine valide B-Ausgabe hin)

Filter für HUE und Lightify Geräte dem Package hinzugefügt

Möglichkeit 10 Eigener Methodenhilfen mit Tastenkürzanwahl

### 1.7.7 Changelog 03.08.11 LCL

Verbesserung im Editor (Stringdelimiter angabe nicht mehr nötig)

Highlighter verbessert in Hinsicht auf String und Kommentarverarbeitung

Ansatz von Echtzeit Hilfe zu Methoden

Möglichkeit ungültiger Dateinamen durch ein Backup-Programme beseitigt

Diverse Kleinere Bugfixes

### 1.7.8 Changelog 03.08.10 LCL

SynchoEdit eingefügt

Parallele Completions eingefügt

Echtzeit Completion für definierte Variablen eingefügt

## 1.7.9 Changelog 03.08.09 LCL

Interne Modifikationen, so das für Scriptausführen nun nicht mehr ein interner Aufruf con CUXD gebraucht wird  
Wahlmöglichkeit über INI Datei: SSH Funktionalität über pling oder über CUXD  
Ein paar kleinere Fixes zur Verbereitung auf den Syntax checker  
Menügesteuerter RTU Upate  
Für Raspberrymatik muss die CUXD version Ungleicht 2.3.0 sein !!!! aktuell heute 2.3.1

## 1.7.10 Changelog 03.08.07 LCL

Sortieralgorithmus der Listendarstellung optimiert  
Erkennen und Beseitigen von Conditionchannel und Destinationchannel Inkonsistenzen hinzugefügt  
ein kleiner Bugfix bei einem Ausgabetext

## 1.7.11 Changelog 03.08.05 LCL

- Bugfix : Backup von Räumen / Gewerken und Sysvars speicherte nicht als Datei. (behoben)
- Neu: - SDV kann SingleConditons auch so bearbeiten, das indirekte Vergleiche möglich sind. (Patch für Konsistenzcheck fehlt aber noch !)
- Editor und Listendarstellung Schrift und Grösse änderbar
- History Datapoints hinz

## 1.7.12 Changelog 03.08.04 LCL

- ein paar weitere Methoden hinzugefügt (auch die neuen der aktuellen RMMatik)
- Backup Restore Funktionalität nun auch für Programme (Einzelne oder mehrere)
- Möglichkeit nun mehrere Instanzen des SDV zu öffnen, ohne dass sich die INI Dateien in die Quere kommen (Nur die erste geöffnete Instanz, (Main) hat Schreibrechte auf die INI)
- Möglichkeit, dem SDV als Kommandozeilenparameter einen Dateinamen mitzugeben, damit ist bei Doppelklicken auf eine Datei automatisches öffnen des Scripts im SDV möglich

## 1.7.13 Changelog 03.08.01 LCL

- ein paar Änderungen in der internen Abarbeitung als Vorbereitung für die nächsten Steps
- Selektionshandling verbessert (ist nun schneller und Bildaufbau ist ruhiger, sor allem bei Multiselektionen)
- Condition Inkonsistenz beseitigen verbessert (hat nun auch mein Produktivsystem glattgezogen)
- SSH kommt nun mit CUxd Aus und braucht plink nun doch nicht mehr (auch im Hinblick auf irgendwann mal Linux)
- Ein paar Methoden und Konstanten noch hinzugefügt
- Editor hat Funktion bekommen für Block auskommentieren und Auskommentieren entfernen (fand ich nützlich für Programmeentwickeln zum Testen)
- Masterparameter können direkt in den Editor als programmCode übernommen werden (Level 6) (unter Berücksichtung vom Gerät, dem Kanal und dem ValueType des Parameters)
- aus einer Selektierten Singledestination kann der Editor unter Selektiert mit dem Neuen Menüpunkt "Singledestination Zugriff" direkt den programmcode zum selektieren der SDest erzeugen, um verzögert um in einem eigenen Script zu bearbeiten
- unter Methodenhilfe kann der Code zum Berechnen der "verzögert um" zeit direkt abgerufen werden.
- Sichten für Systemvariablen nun endlich fertig ausprogrammiert
- Favoritenansicht eingefügt Favoriten können aus der Listenübersicht übernommen werden, ein Doppelklick in der Favoritenansicht auf ein Object führt zum Laden der Favoritenansicht in die Listenübersicht mit detaildarstellung des Selektierten Objectes (undo redo Browsing ist dabei berücksichtigt)

## 1.7.14 Changelog 03.07.13 LZL

- eine rudimentäre SSH Funktionalität.
- Condition Inkonsistenzen sind nun auch behebbar (war behebbar ist)

## 1.7.15 Changelog 03.07.12 LZL

- kleiner Fix bei Gleitkommawerteingabe von Sysvars, DP (Hier wurde der Nachkommateil unterdrückt)
- Bugfix bei der Fensterdarstellung im Detail- Edit Feld
- Firmware der verbundenen CCU wird nun in der Statusbar angezeigt
- Fix für Level 7 Anwendung aufgrund Firmware 3.45.5 bei RMatik

## 1.7.16 Changelog 03.07.11 LZL

### Kleine Bugfixes

Tab Scriptausgabe entfernt, dafür sitzt die Scriptausgabe nun in einem Fenster unter dem Editor (Grösse anpassbar)

Neue Keys eingeführt, um Script Run und Script Test auch über FKeys zu starten

Mit dieser Version kennt der SDV 620 Rega-Methoden und Konstanten

Methodenhilfe eingeführt

## 1.7.17 Changelog 03.07.10 LZL

### Kleine Bugfixes

Properties von Zeitmodulen und Singledestination / Singleconditions über Dropboxes änderbar

Systemvariablen Metadaten Check und Reparatur eingeführt

Systemvariablen NirvanaChannel Check und Reparatur eingeführt

## 1.7.18 Changelog 03.07.07 LZL

### Kleine Bugfixes

Bugfix: Alt-S aus dem Editor brachte nicht die Special Spalte: geändert in Alt-p

Ein Paar Methoden hinzugefügt

Reiter Scriptsuche fertiggestellt (ab level 6)

Damit nun möglich: Volltextsuche in allen Scripten

Selektierbar, auch suchen in GeisterDestinations möglich

Suchen auch als Regular Expression möglich

Beschleunigte Suche in Ram möglich

ProgrammObjektanalyse

## 1.7.19 Changelog 03.07.05 LZL

### Kleine Bugfixes

Bugfix bei Programme Testen („\n“ in einem Kommentar führte zu einem Fehler) Bugfix tut nur auf RM, auf der die Rega bezüglich Doublequote gefixt wurde.

Im Editor folgende Tastenkürzel:

Ctrl F – Suchen Dialog, Ctrl R Replace Dialog, F3 – Weitersuchen

Regaports werden nun Automatisch ermittelt

Alternative Endungen für Scripte in INI hinzugefügt

Alternatives Scriptverzeichnis in INI hinzugefügt

## 1.7.20 Changelog 03.07.04 LZL

Auflösung geändert und ein wenig das Design

Programm Testen hinzugefügt

Suchen in Scripten nach Seriennummern von Devices / Kanälen

Ein paar weiterer Methoden eingepflegt

## 1.8 Neue Authentifizierungsmechanismen Raspberrymatik

Die neuen Versionen der Raspberrymatik unterstützen SSL und Authentifizierung über Nutzername / Password. Nach Wunsch und Tips aus dem Testerkreis habe ich auch dies versucht umzusetzen.

### 1.8.1 SSL

SSL wird nun realisiert über die OpenSSL Library, welche auch von der Lazarus Foundation empfohlen wird:

Quelle: [https://sourceforge.net/p/lazarus-ccr/s...en\\_ssl.pas](https://sourceforge.net/p/lazarus-ccr/s...en_ssl.pas).

Der SDV liegt als 32bit Compilat vor, wenn man sich die dazu benötigten DLL's selber herunterladen möchte, die Quellen sind folgende;

[http://packages.lazarus-ide.org/openssl-1.0.2j-x64\\_86-win64.zip](http://packages.lazarus-ide.org/openssl-1.0.2j-x64_86-win64.zip) für die 64 Bit Version  
und  
<http://packages.lazarus-ide.org/openssl-1.0.2j-i386-win32.zip> für die 32 Bit Version

Das Zip File auspacken und die 2 DLL's in das Verzeichnis kopieren, im dem sich auch die SDV.Exe befindet. In dem SDV rar File befindet sich auch die 32 Bit Version der DLL's inkl. dem Lizenztext als TXT und PDF. Damit sollte der Zugriff über HTTPS schon mal funktionieren.

### 1.8.2 Authentifizierung über Nutzername / Passwort.

Die Struktur der INI Datei wurde geändert, damit sich pro CCU nun auswählen lässt zwischen HTTP und HTTPS Zugriff und die Authentifizierung Nutzername / Passwort wahlweise genutzt wird.

Wird bei Nutzername oder Passwort nix eingetragen, so generiert der SDV einen Zugriff ohne die Kennung NN:PW@HOST. Sind beide Werte eingetragen, so wird der Zugriff über NN:PW@HOST generiert.

Beispielhafter Aufbau der neuen INI

```
[LAST]
DATEI=c:\MTH\Homematic\NewScript.hsc

[HOST]
NICKNAME=DerNickNameAusDerLizenzenanfrage
CUxD=CUxD.CUX2801001:5
SYSTEMEXEC=true
LICENCE1=DerLizenzschlüssel1
LICENCE2=DerLizenzschlüssel2

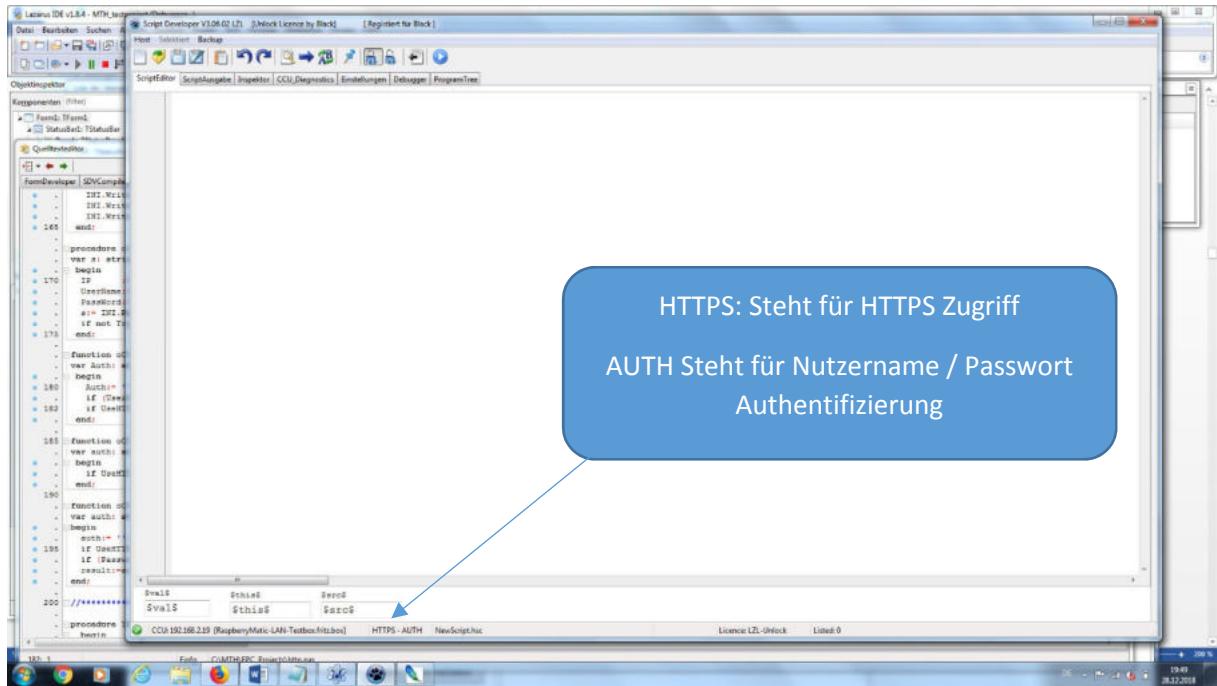
[CCU1]                                     Neuer Schlüssel
IP=192.168.2.19                           Ip Von CCU1
USERNAME=Admin                             Nutzernname auf der CCU1
PASSWORD=XXXXXXXXXXXX                      Passwort der Nutzers auf der CCU1
USEHTTPS=true                                Zugriff über HTTPS

[CCU2]                                     Dito wie bei CCU1
IP=192.168.2.6
USERNAME=
PASSWORD=
USEHTTPS=false

[HOSTCCU]                                    Die Werte der aktuellen CCU bei Start
IP=192.168.2.19                           im Idealfall hier die Daten von CCU1 oder 2
                                             eintragen für 1. Start
USERNAME=Admin
PASSWORD=XXXXXXXXXXXX
USEHTTPS=True

[SECURITY]                                    Die Ports die der SDV benutzt
HTTPREGASCRPT=8181
HTTPREGAXMLRPC=1999
HTTPSREGASCRPT=48181
HTTPSREGAXMLRPC=41999
```

Die gelben Texte müssen auf jeden in einer Bestehenden INI geändert werden. Ist das alles richtig gemacht, so zeigt der SDV im Statusfeld auch die Art des Zugriffs an:



Ein Debugger Breakpoint im SDV zeigt hier den zusammengesetzten HostString, der bei HTTPS und Authentifizierung benutzt wird

```
function oCCUNetwork.CCUHostString: string;
var Auth: string;
begin
  Auth:= '';
  if (UserName<>'') and (PassWord<>'') then auth:=UserName+':'+PassWord+'@';
  if UseHTTPS then result:= 'https://'+auth+IP else result:= 'http://'+IP;
end;
Result = "ansstring(50AC2FF10) = https://Admin:123456@192.168.2.19:48181/black.exe"

```

## 1.9 Ein Wort des Authors zum Thema „gewerbliche Nutzung“

Aufgrund der Tatsache, dass ich von einem Supplier versehentlich hörte, man freue sich schon auf die Backup Funktionalitäten der Direktverknüpfungen, Blöd nur, dass ich die kommerzielle Nutzung explizit untersagt habe. Immerhin zwingt mich diese kommerzielle Unverfrorenheit ein wenig an dem Ausgabeformat zu feilen. Das derartige Vervielfältigungsmechanismen einen immensen zeitlichen Vorteil bei der Implementierung von baugleichen Anlagen bieten leuchtet allen ein. Möglichkeit wäre diese Funktion dann ganz zu sperren oder per CompilerDirective erst gar nicht einzocompilieren. Hilft aber allen anderen privaten Anwendern nicht.

Also bleibt mir nur um das zu unterbinden bzw. möglichst zu erschweren:

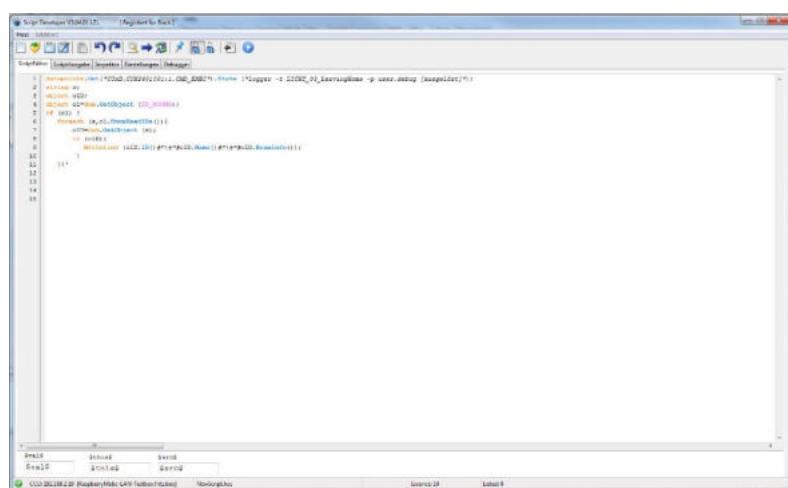
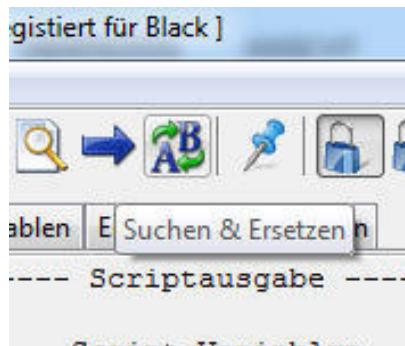
**Als Ausgabeformat für das Backup von Geräten wird es nur noch JSON geben.**

Damit ist direktes 1 zu 1 Kopien ziehen aus einer Masteranlage zwar möglich, das JSON kann auch manuell angepasst werden, allerdings lassen sich diese JSON Files aber aufgrund des Pairings nicht mehr in die Child Anlagen einspielen. Als privater Anwender sollte man davon nichts merken, der SDV verbindet sich ja mit der CCU und kann das JSON in ein Programm umrechnen und damit das Restore ausführen.

Ich hoffe der Schritt von mir ist nachvollziehbar

## 2. Oberfläche

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



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

In der Statuszeile finden sich Information über:

1. IP der Host CCU
2. DateiNames des Scriptes im Scripteditor
3. Anzahl der Elemente in der Listendarstellung

Folgende Shortcuts wurden implementiert:

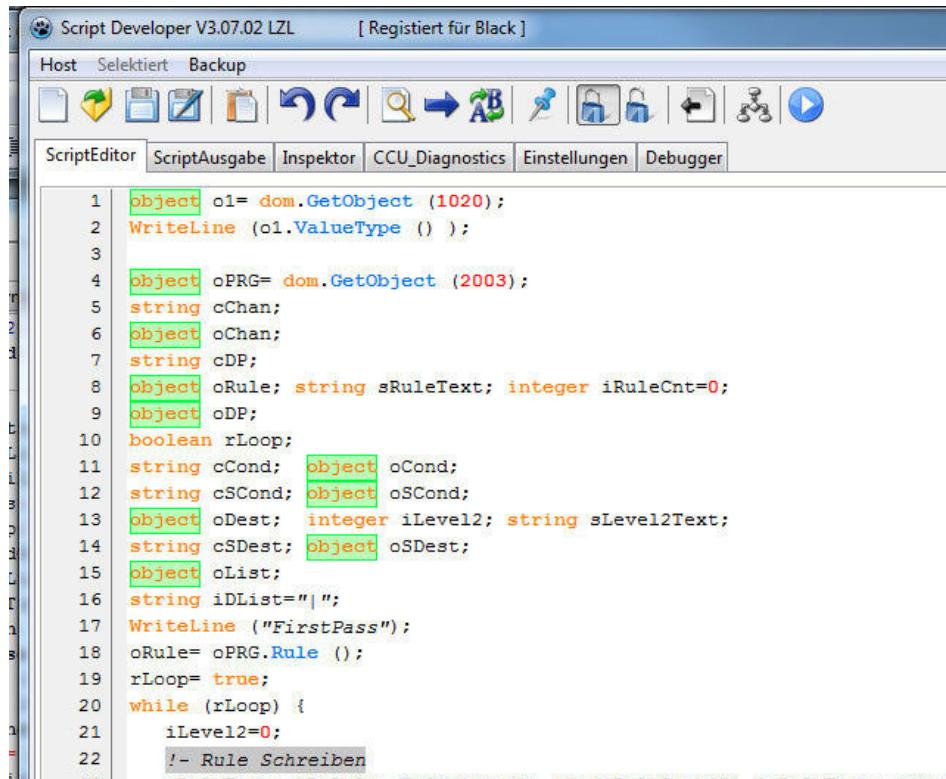
Ctrl-F: Suchen Dialog

Ctrl-R: Ersetzen Dialog

F3: Weitersuchen

## 2.1 Word Markup (seit 03.07.02)

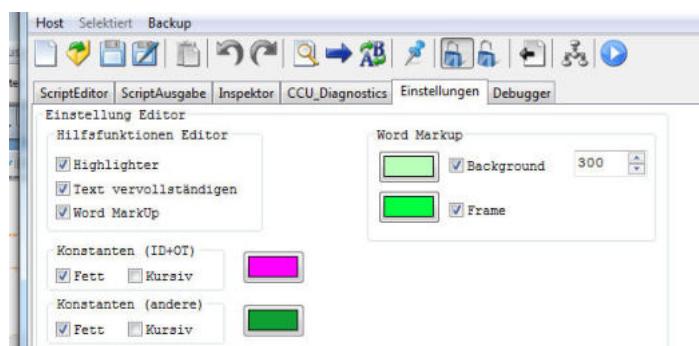
Der Editor beherrscht nun auch Wordmarkup. Bei einem Word, welches angeklickt wurde, oder geschrieben wurde (auf dem sich der Focus befindet ^^), werden sämtliche Vorkommen im Text farblich hervorgehoben



The screenshot shows the Script Developer interface with the title bar "Script Developer V3.07.02 L2L [Registriert für Black]". The menu bar includes "Host", "Selektiert", and "Backup". The toolbar contains various icons for file operations and script management. The main window has tabs for "ScriptEditor", "ScriptAusgabe", "Inspektor", "CCU\_Diagnostics", "Einstellungen", and "Debugger". The "ScriptEditor" tab is active, displaying the following script code:

```
1 object o1= dom.GetObject (1020);
2 WriteLine (o1.ValueType () );
3
4 object oPRG= dom.GetObject (2003);
5 string cChan;
6 object oChan;
7 string cDP;
8 object oRule; string sRuleText; integer iRuleCnt=0;
9 object oDP;
10 boolean rLoop;
11 string cCond; object oCond;
12 string cSCond; object oSCond;
13 object oDest; integer iLevel2; string sLevel2Text;
14 string cSDest; object oSDest;
15 object oList;
16 string iDList="| ";
17 WriteLine ("FirstPass");
18 oRule= oPRG.Rule ();
19 rLoop= true;
20 while (rLoop) {
21     iLevel2=0;
22     !- Rule Schreiben
```

Die Funktion ist konfigurierbar unter Einstellungen



Die Einstellungen sollten selbsterklärend sein, die Zahlenangabe bezieht sich auf die Zeit in ms, ab wann seit dem letzten Tastenanschlag die Hervorhebung beginnt.

### 2.1.1 SyncroEdit (ab 3.08.10)

Mit SyncroEdit lässt sich in einem Bereich synchron an mehreren gleichen Stellen ändern.  
Vorgehensweise:

```

17 integer CMT_HUB=0000
18 integer SMB=0000
19 integer i
20 integer index=0
21 integer index2=0
22 integer index3=0
23 integer index4=0
24 integer index5=0
25 integer index6=0
26 integer index7=0
27 integer index8=0
28 integer index9=0

```

Den Textbereich in dem geändert werden soll, markieren, anschliessend mit Strg-J zum Synchroneditieren vorbereiten. Der Bereich wird dann mintgrün, die Worte werden hervorgehoben und ich klick nun dahin wo ich ändern will. Hier Hinter das old (ist durch den markup auch hellgrün hervorgehoben

```

17 integer CMT_HUB=0000
18 integer SMB=0000
19 integer i
20 integer index=0
21 integer index2=0
22 integer index3=0
23 integer index4=0
24 integer index5=0
25 integer index6=0
26 integer index7=0
27 integer index8=0
28 integer index9=0

```

Jetzt kann man schreiben und an allen hellgrün hervorgehobenen Stellen im Mindgrünen Bereich wird Synchron mitgeändert.

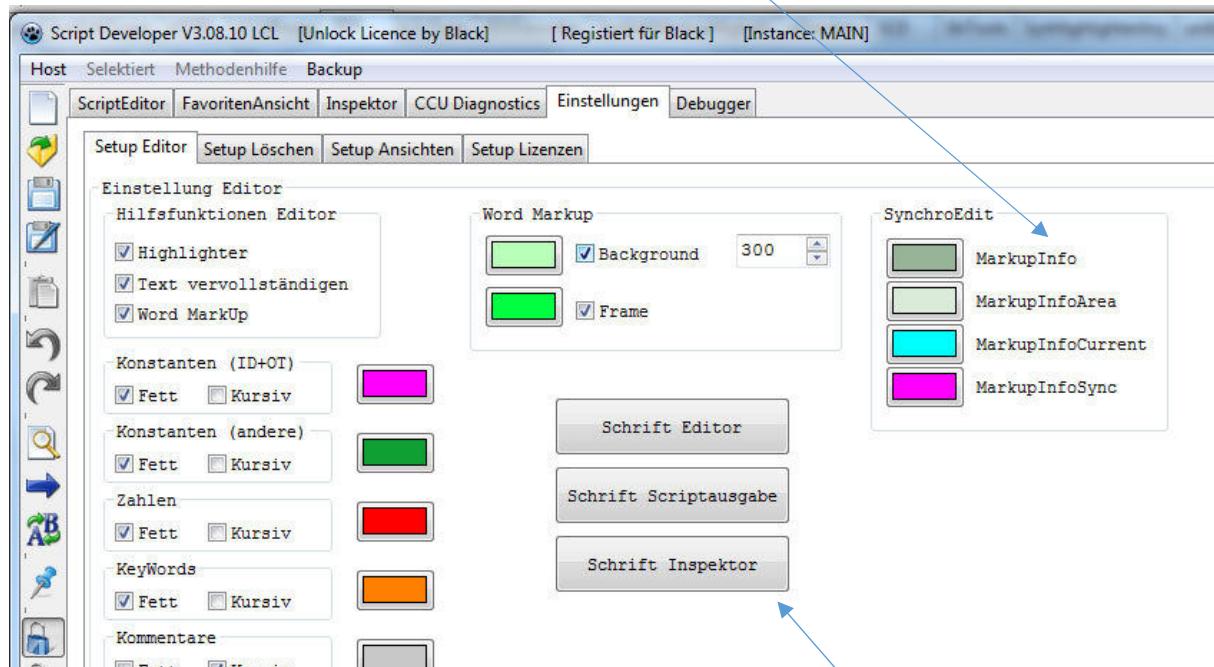
```

17 integer CMT_HUB=0000
18 integer SMB=0000
19 integer i
20 integer index=0
21 integer index2=0
22 integer index3=0
23 integer index4=0
24 integer index5=0
25 integer index6=0
26 integer index7=0
27 integer index8=0
28 integer index9=0

```

Die Markierung aufheben geht dann mit Escape.

Die Farben lassen sich natürlich in Setup Menü einstellen

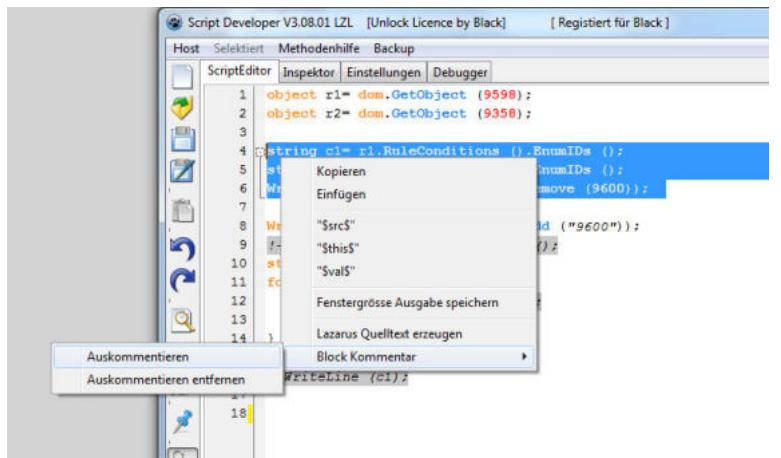


## 2.1.2 Schriftgrößen und Schriftart (ab 3.08.07)

Einstellungen für Schriftwart Editor, dem Editor Ausgabefenster und im Inspektor

## 2.2 Blockweise auskommentieren.

Ab der Version V3.08.01 kann der SDV Editor Blöcke zum Testen auskommentieren oder wieder die Auskommentierung entfernen.



The screenshot shows the same interface after the 'Auskommentieren' (Comment Out) operation was performed. The previously selected block of code is now displayed with leading semi-colons and a double slash at the start of each line, indicating it is now a comment. The code block is:

```
1 object r1= dom.GetObject (9598);
2 object r2= dom.GetObject (9358);
3
4 //string c1= r1.RuleConditions ().EnumIDs ();
5 //string c2= r2.RuleConditions ().EnumIDs ();
6 //r1.RuleConditions().Remove (9600);
7
8 //WriteLine (r1.RuleConditions().Add ("9600"));
9 //r2.RuleConditions().RemoveAll ();
10 string sCND;
11 foreach (sCND,c1) {
12     //r1.RuleConditions.Add (sCND);
13     //WriteLine ("---#sCND#---");
14 }
15
16 //WriteLine (c1);
```

## 3 Scripteditor

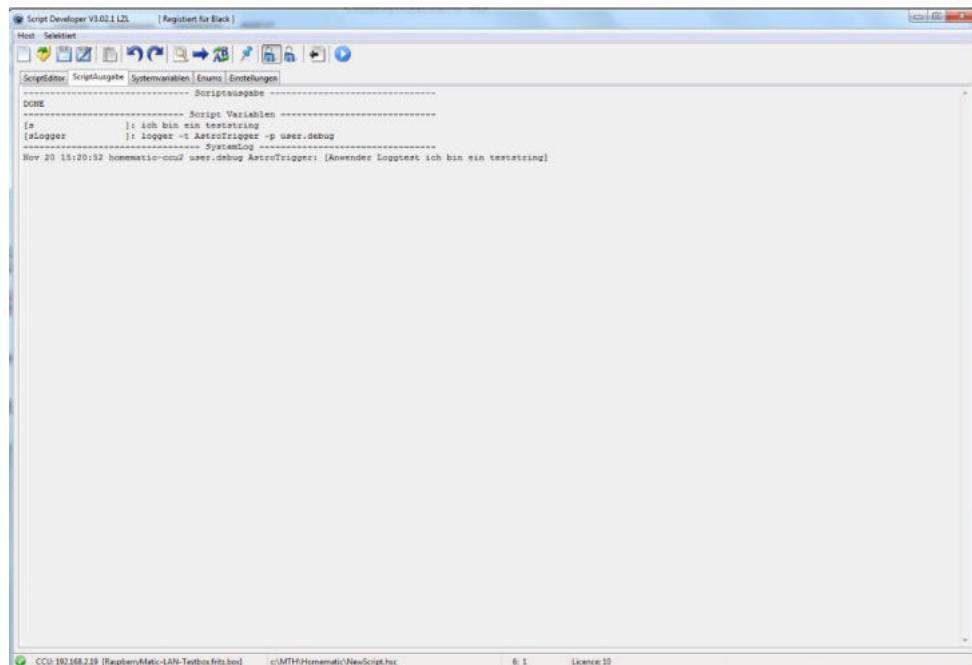
Im Scripteditor werden die Scripte geschrieben oder geladen, die mittels Run Script oder  an die CCU zum Ausführen gesendet werden. Das Scriptergebnis wird dann im Reiter Ausgabe angezeigt. Dieses kleine TestScript zum Beispiel:

```
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 testweile aus State

Erzeugt folgende Ausgabe:



The screenshot shows the 'Scriptausgabe' tab of the Script Developer software. The output window displays the following text:  
DONE  
[sLogger ]: ich bin ein teststring  
[sLogger ] logger -t AstroTrigger -p user.debug  
Systemlog  
Nov 20 15:20:32 homematic-cu2 user.debug AstroTrigger: [Anwender Loggtest ich bin ein teststring]

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 Logtest " # s # "]");^M WriteLine ("DONE");^M  
[iseESP.cpp:1121]  
Jun 15 12:41:49 homematic-raspi local0.err ReGaHss: Error: ParseProgram: SyntaxError=  
(sLogger # "[Anwender Logtest " # s # "]");^M WriteLine ("DONE"); [iseESP.cpp:374]
```

Ab 3.07.04 : Script testen eingefügt, hierbei wird das Skript nur nach Fehlern geparst aber nicht ausgeführt (Syntax Check). Funktionsweise äquivalent zu der KlickiBunti Funktion: Script Testen.

Dazu im Scripteditor auf  drücken, es wird dann der Syntaxcheck der CCU aufgerufen. Bei erfolgreichem Syntaxcheck erfolgt für 2 Sekunden die Meldung

Test Script erfolgreich

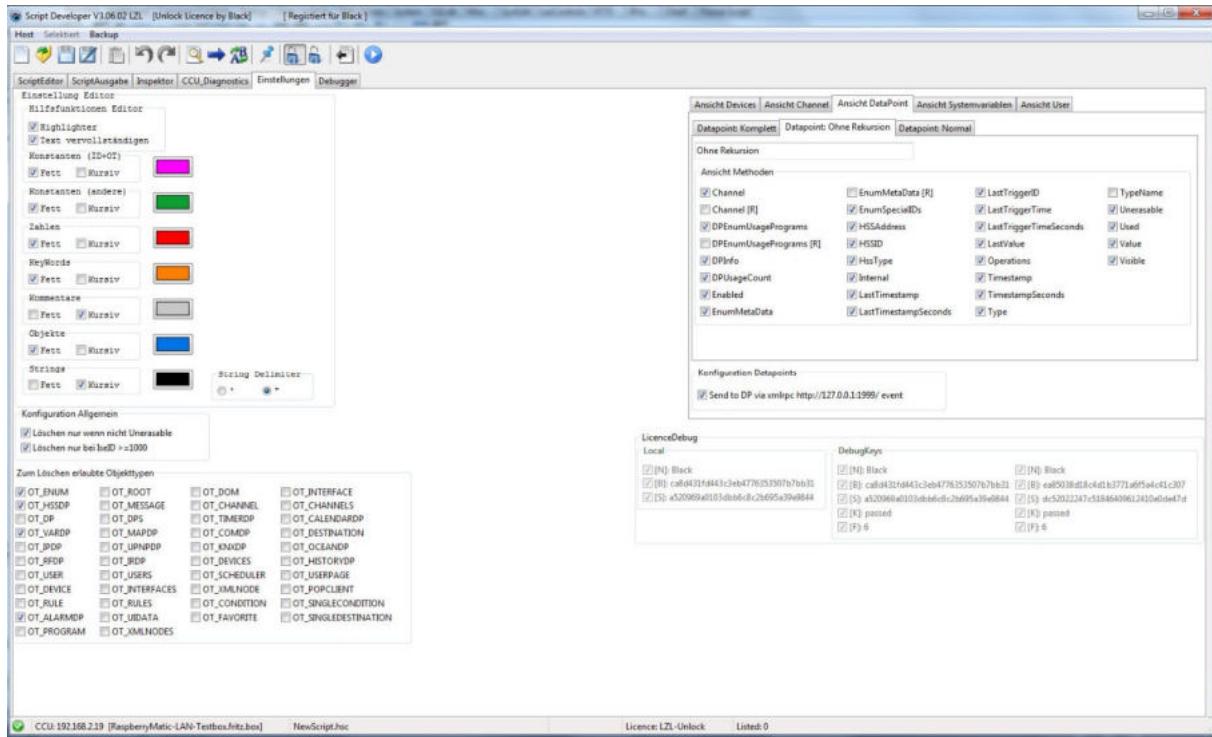
Trat ein Fehler auf, so erfolgt Sprung auf den Reiter Scriptausgabe mit Anzeige der Fehlermeldung

Bestandsnutzer können den folgenden Schlüssel in die SDV.INI schreiben:

```
[SCRIPTRUN]  
RUN=F1  
TEST=F12
```

Die legt Script RUN auf F1 und Script testen auf F12. Kann selber angepasst werden. F3 ist reserviert !!! für suche erneut

### 3.1 Voreinstellungen Editor



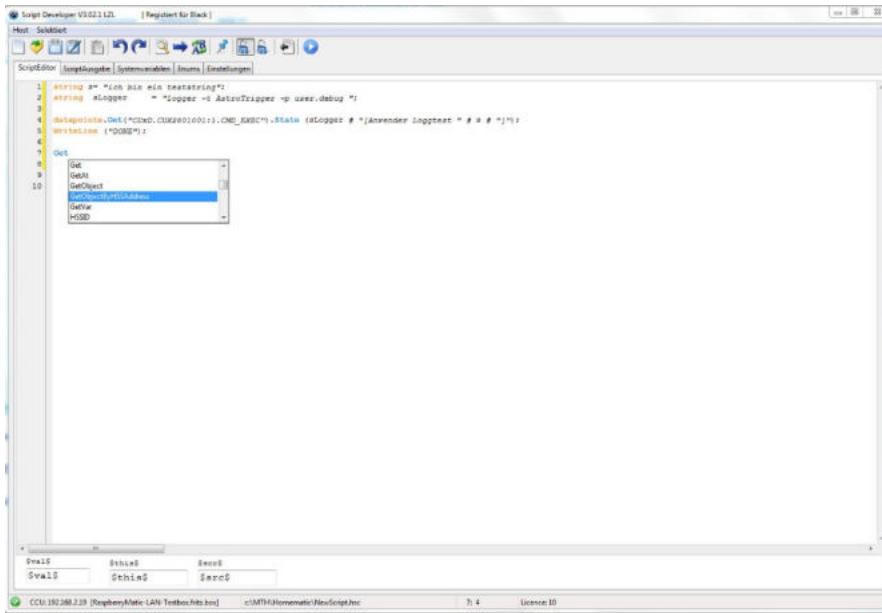
Hier kann nach Vorliebe der Highlighter konfiguriert oder auch ausgeschaltet werden

Ebenso lässt sich die Vervollständigen Funktion an oder abwählen.

Vorbedingung natürlich: Lizenzlevel muss vorhanden sein.

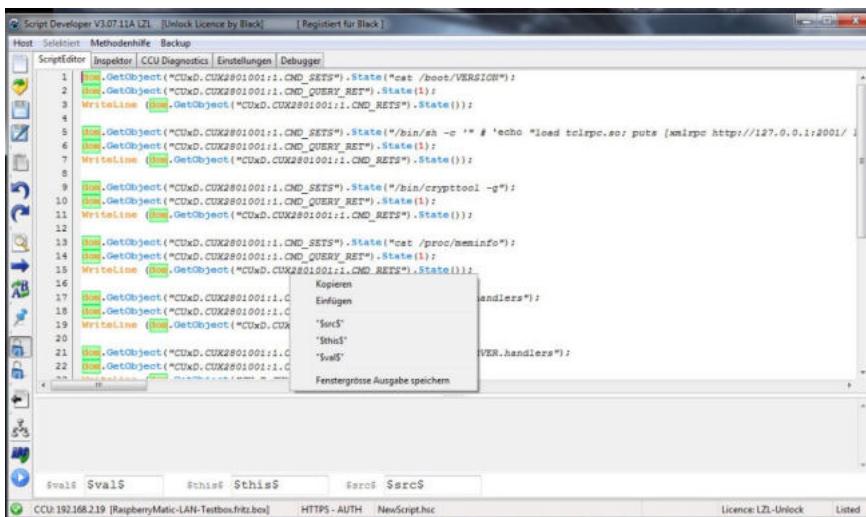
## 3.2 Vervollständigen Funktion

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.

Die Grösse des Ausgabefensters kann abgespeichert werden.  
Rechte Maustaste in den Editor und Fenstergrösse Ausgabe speichern



Ab V3.08.10

## Erweiterung der Vervollständigen Funktion

Folgende Vervollständiger sind nun parallel implementiert:

Ctrl-Space : Die bekannte komplette Version

Ctrl-Shift-M: Nur Regascript Methoden

Ctrl-Shift-I: Regascript Konstanten ID\_xxx

Ctrl-Shift-O: Regascript Konstanten OT\_xx

Ctrl-Shift-S: Untermenge Regascript Methoden (StringMethoden)

Ctrl-Shift-C: Untermenge Regascript Methoden (Mathematische Methoden)

Ctrl-Shift-D: Deklarationen: hier wird in Echtzeit das Script analysiert und die gefundenen Variablen-deklarationen vorgeschlagen

The screenshot shows the Script Developer V3.08.10 LCL interface. The main window displays a script editor with the following code:

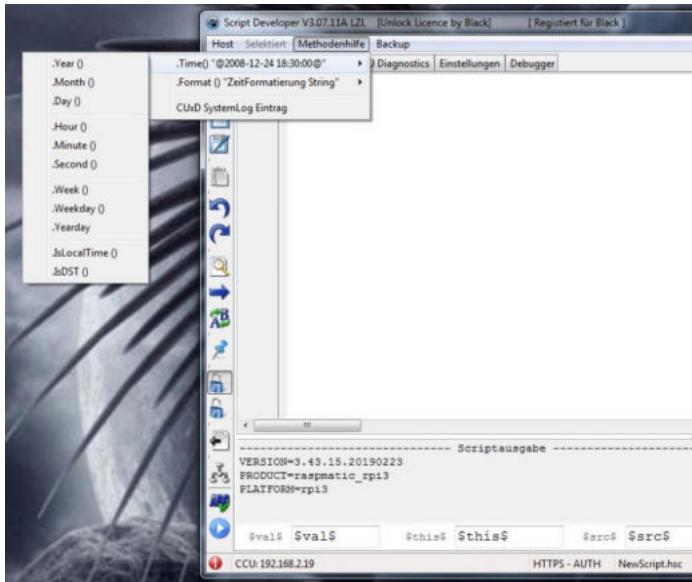
```
25 boolean falszDP;
26 WriteLine ("ProgrammAnalyse Konsistenz ConditionChannel und DestinationChannel");
27 WriteLine ("Black in Juni 2019");
28 WriteLine (sline);
29 foreach (sPRG, dom.GetObject (ID_PROGRAMS).EnumIDs () ) {
30     oPRG= dom.GetObject (sPRG);
31     if (oPRG) {
32         infoDown= false;
33         sProgInfo= "Analyse Program: " # oPRG.Name () # " [ID: "# oPRG.ID () #"]";
34         if (infoAll) {WriteLine (sProgInfo);}
35         oRule= oPRG.Rule ();
36         iRule=1;
37         while (oRule) {
38             foreach (sCond, system.GenerateEnum (0,oRule.RuleConditions().Count()-1)) {
39                 oCond= oRule.RuleCondition (sCond.ToInteger ());
40                 foreach (sSCond, system.GenerateEnum (0,oCond.CndSingleCount ()-1)) {
41                     oSCond= oCond.CndSingleCondition (sSCond);
42                     /* Hier sind wir in der Singlecondition
43                     - Testen auf eine Systemvariable/HSS
44                     if ((oSCond.LeftValType () == ivtSystemId) || (oSCond.LeftValType () == ivtObjectId)) {
45                         oOB= dom.GetObject (oSCond.LeftVal () );
46                         os=;
47                         if (nRn
48                             nPRG
49                             oCond
50                             oDet
51                             oOB
52                             oPRG
53                             oRule
54                             oSCond
55                             WriteLine ("#tInkonsistenz [SCND-ID: " # oSCond.ID () # "] Rule-Nr: " # iRule # ", CND-Nr: " # (sCond.ToInteger()+1) # ", SCND-Nr: " # (s5
56                             if (coRr) {
57                                 nCorr=nCorr+1;
58                                 x= oSCond.LeftVal ();
59                                 oSCond.LeftValType (ivtSystemId);
```

A tooltip is displayed over the variable 'oSCond' at line 54, listing several variable names: 'nRn', 'nPRG', 'oCond', 'oDet', 'oOB', 'oPRG', 'oRule', and 'oSCond'. The status bar at the bottom shows the license information: CCU:192.168.2.19 [] and CCU Diagnostics.

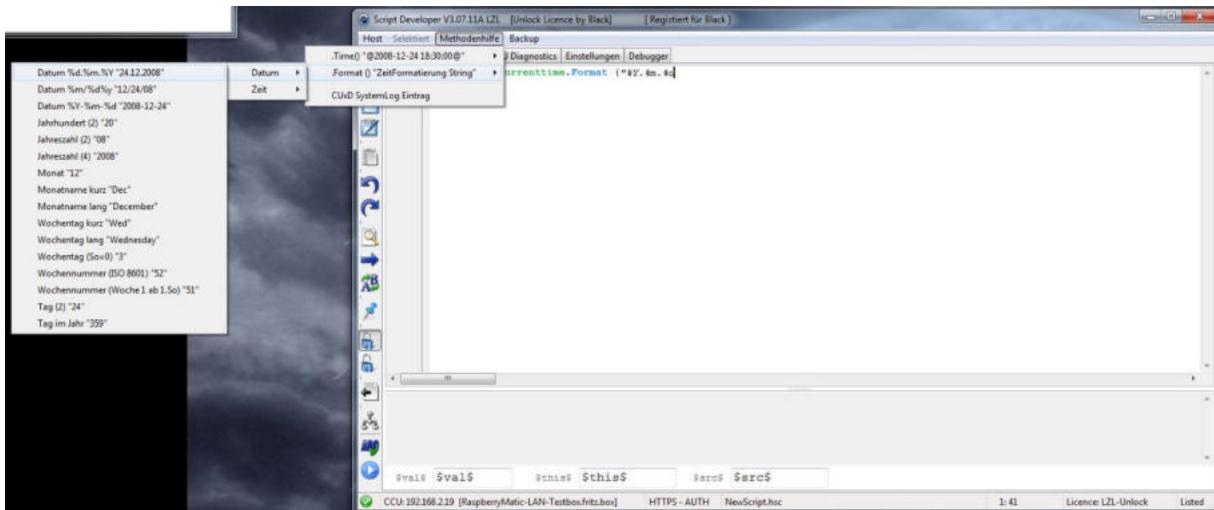
### 3.3 Methodenhilfe

Da es mich immer nervte, bei bestimmten Funktionen doch in Handbuch gucken zu müssen, gibt es nun den Menüpunkt Methodenhilfe (Aktiv im Editor)

Damit lassen sich zu Themen Time, Zeit, und Zeitformatierung Funktionen und die Formatstring schnell finden



Durch Druck wird die Passende Funktion ausgewählt. Dito bei Zeitformatierung



Ab dem Version 3.08.12D verfügt der SDV über die Möglichkeit, bis zu zehn eigene Methodenhilfen zu definieren. Diese werden in der INI angelegt.

```
; Common escape sequences
;\\      \ (a single backslash, escaping the escape character)
;\'     Apostrophe
;\"    Double quotes
;\t     Tab character
;\r    Carriage return
;\n    Line feed
;\;    Semicolon
;\#    Number sign
;\=    Equals sign
;\:    Colon

[METHODHELP1]
NAME=dom.GetObject
TEXT=dom.GetObject (

[METHODHELP2]
NAME=
TEXT=

[METHODHELP3]
NAME=
TEXT=

[METHODHELP4]
NAME=
TEXT=

[METHODHELP5]
NAME=
TEXT=

[METHODHELP6]
NAME=
TEXT=

[METHODHELP7]
NAME=
TEXT=

[METHODHELP8]
NAME=
TEXT=

[METHODHELP9]
NAME=
TEXT=

[METHODHELP10]
NAME=
TEXT=
```

Sollte fast selbsterklärend sein, mit Name wird definiert, was in der Menüzeile dargestellt wird, mit text wird das definiert, was dann im Editor eingefügt wird. Es muss mindestens ein Eigener MenüHelp definiert sein, damit der menüpunkt unter Methodenhilfe dargestellt wird.

Ebenso ist direkte Verwendung con Kurztasten möglich Shift+Ctrl + 1,2,3...9,0

#### **dabei bitte auf die Falle aufpassen !**

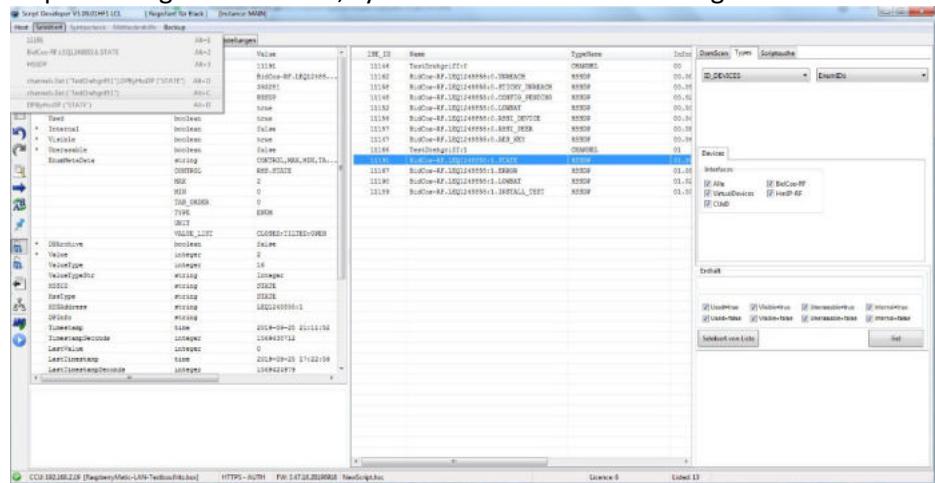
ich hab extra die Kommentare da mit reingesetzt, sie dort aufgeführten Sonderzeichen müssen gequotet sein rechts des Gleichheitszeichens.

heißt will ich dom.GetObject (" "); als Text schreiben so muss ich eintragen  
TEXT=dom.GetObject (\\"\\");

Zweizeilig geht auch, dann muss \n benutzt werden.

### 3.4 Symbolischer Zugriff auf selektierte Datenpunkte des Inspektors

Zur Vereinfachung beim Programmieren besteht die Möglichkeit, einen Datenpunkt, welcher im Inspektor angewählt wurde, symbolisch in den Editor einfügen zu können.



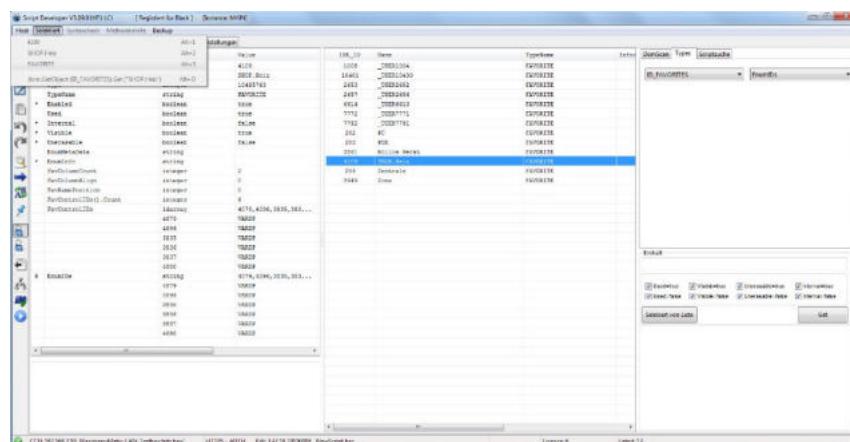
Bei einem HSSDP werden 3 Möglichkeiten angeboten (Menüpunkt selektiert)

Vollzugriff über Kanal.HSSDP

Nur Kanal

Nur HSSDP

Fast alle typen (Favoriten, User, VarDP, AlarmDP) werden automatisch richtig erkannt und dafür der richtige Symbolische Zugriff vorgeschlagen.

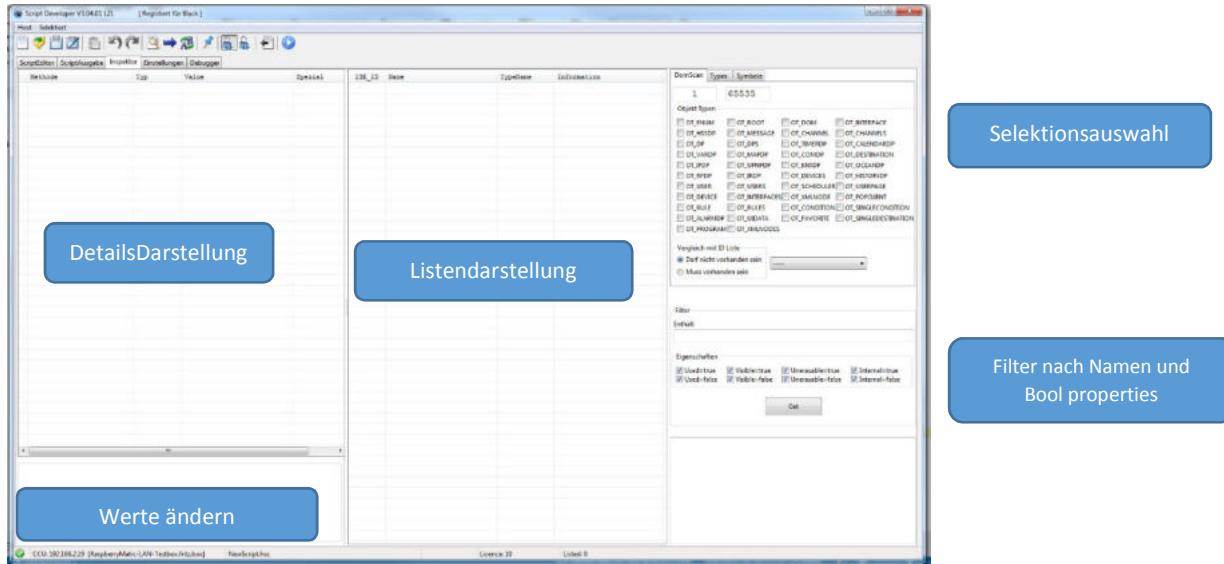


Eine einfache, schnelle und effektive Methode um unnötige Schreibfehler zu vermeiden.

## 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)

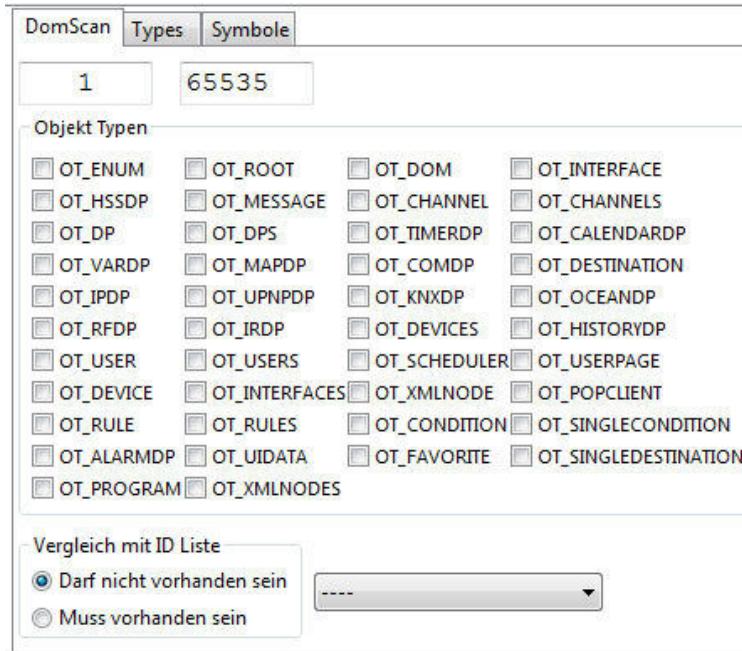
Enthält: leerer Eintrag = es wird nicht nach enthaltener Buchstabensequenz selektiert  
Eingegebener Text. Die Systemvariable muss im Namen die Buchstabensequenz enthalten.

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



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

The screenshot shows the Script-Developer interface with a search results table on the left and the DomScan dialog on the right. The search results table has columns: Wertebereich, Typ, Value, Spezies, ISE\_ID, Name, Typename, undressValue. The DomScan dialog on the right shows the same search parameters and the 'Object Types' list with the 'OT\_DEVICE' checkbox selected.

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 Beispieldeweise: Scanlauf über alle Objekte aus DOM mit der Eigenschaft VARDP, die aber nicht unter ID\_SYSTEM\_VARIABLES gelistet sind:

The screenshot shows the Script-Developer interface with the 'DomScan' tab selected. The search results table has the following columns: Method, Type, Value, Special, ID\_ID, Name, Typepath, and Information. A filter row above the table specifies 'VARDP' in the 'Typepath' column. The results list several entries, including 'System' (Typepath: 'System'), 'Windows-IP' (Typepath: 'Windows-IP'), and various ActiveState objects (Typepath: 'ActiveState'). A detailed view of one entry, 'Windows-IP', is shown in the bottom pane, displaying its properties like 'Name' (Windows-IP), 'Typepath' (Windows-IP), 'Information' (Windows-IP), and a 'Details' section with a tree view of object types.

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

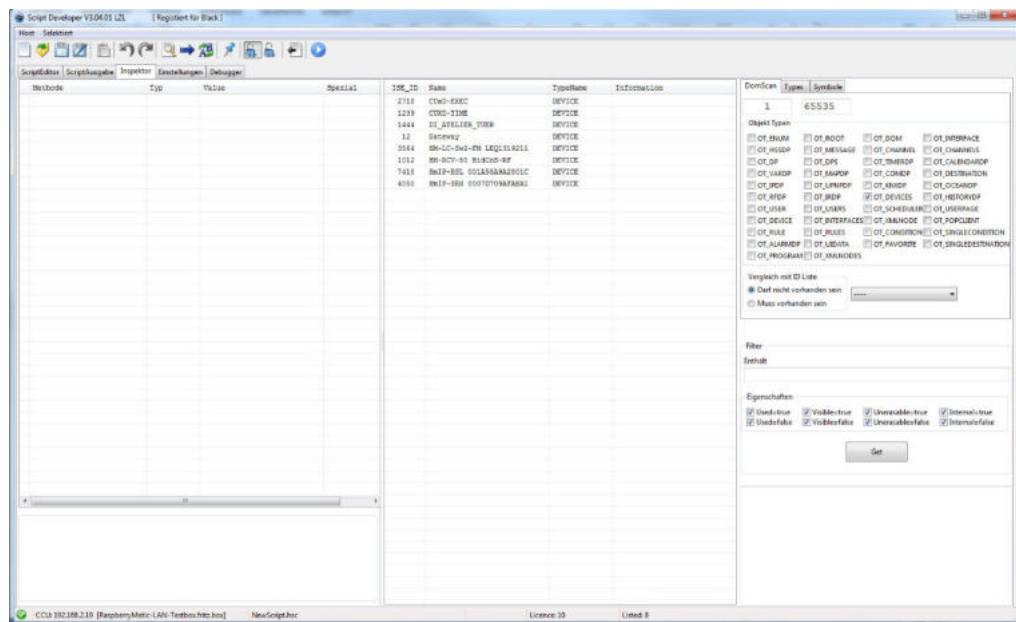
Weiterhin ist auch rekursives Arbeiten nun möglich

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

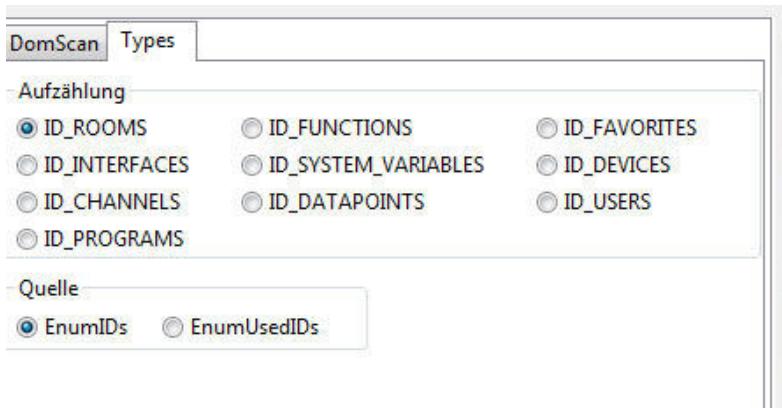
The screenshot shows the Script-Developer interface with the 'DomScan' tab selected. The search results table has the same columns as the previous screenshot. A filter row above the table specifies 'Root Device' in the 'Name' column. The results list several entries, including 'Root Device' (Name: Root Device) and various device entries (e.g., '2718 DEVICE', '3229 DEVICE', '1444 DEVICE', etc.). A detailed view of one entry, 'Root Device', is shown in the bottom pane, displaying its properties like 'Name' (Root Device), 'Typepath' (Root Device), and a 'Details' section with a tree view of object types.

Das „R“ in der ersten Spalte zeigt an, dass ein Rekursiver Aufruf möglich ist. Doppelklicken löst und die Liste auf und lädt diese in die Listenauswahl. Ist es nur ein Wert, so wird dieser Wert auch direkt in der Detailansicht geöffnet.

Die neue Auswahllist steht nun zur weiteren Bearbeitung bereit.



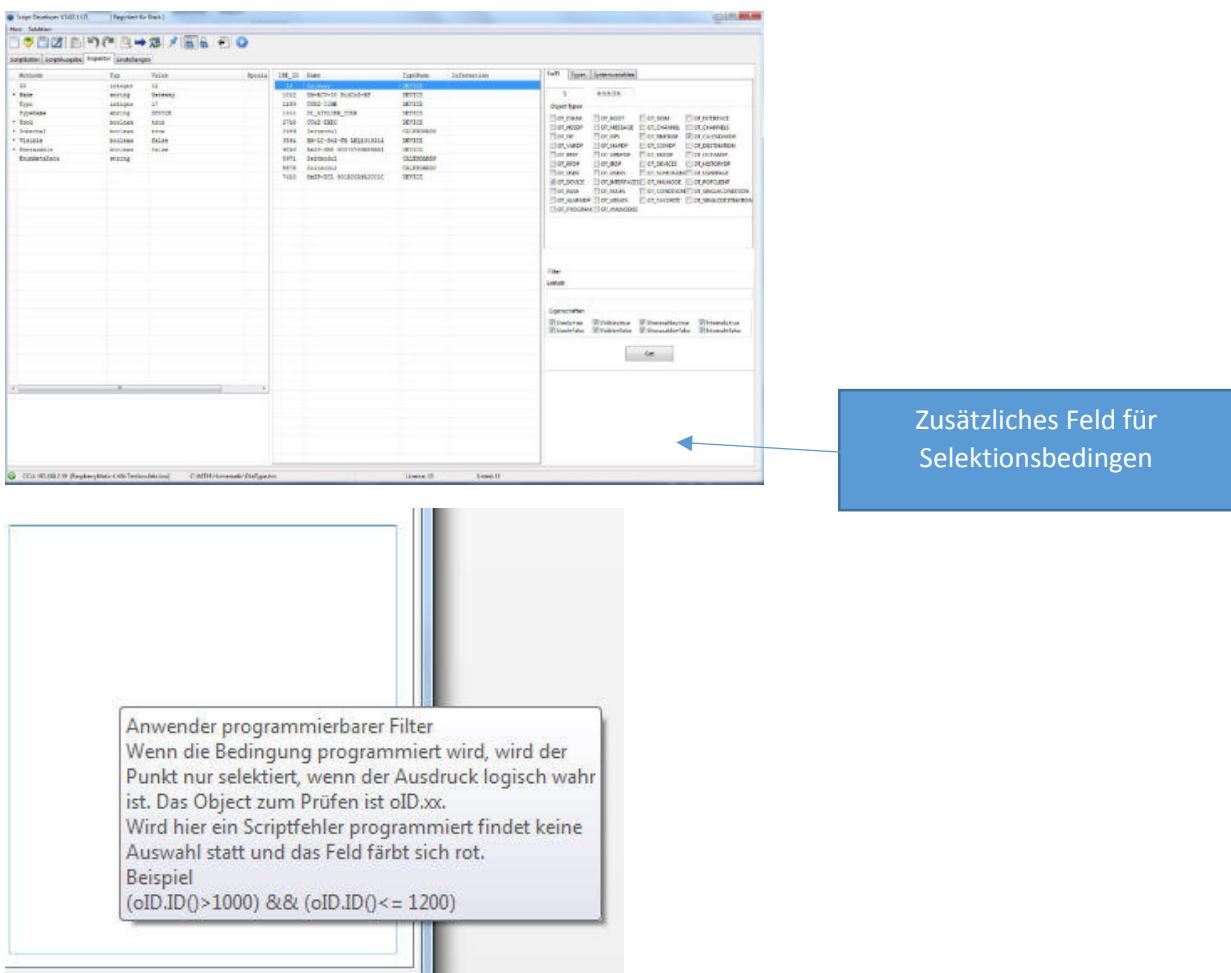
## 4.2 Selektionskriterium Types



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)



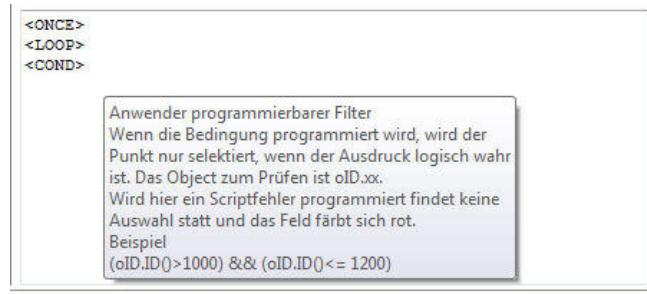
Filter sind ein mächtiges Werkzeug zum komplexen Eingrenzen und für Komplexe Abfragen.

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 () );
}
}
```

An diesem Kleinen Filter mal verdeutlicht:

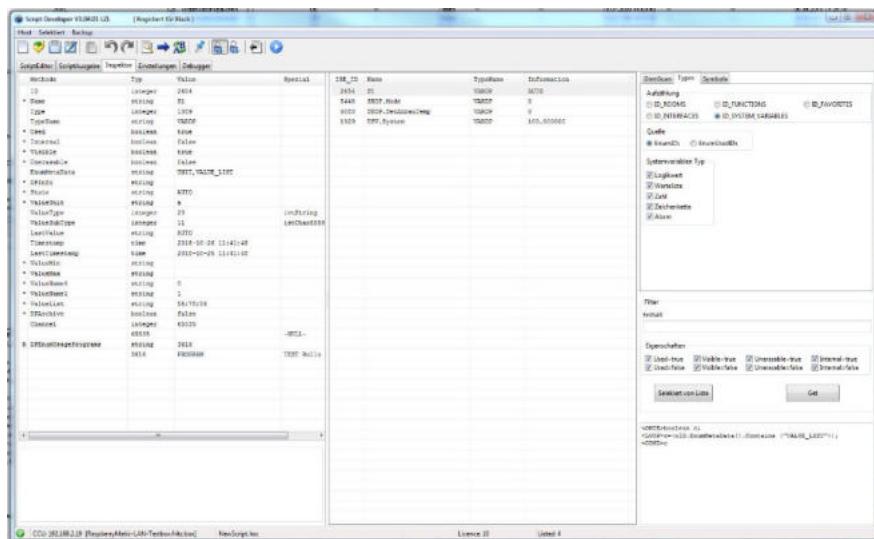
```
<ONCE>boolean c;
<LOOP>c=(oID.EnumMetaData().Contains ("VALUE_LIST"));
<COND>c
```

Daraus generiert der Scriptdeveloper folgende interne Filterabfrage:

```
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 () );
}
}
```

Filtert aus der Gruppe der Systemvariablen alle, in deren Eigenschaft EnumMetadata das Wort VALUE\_LIST vorkommt.



So lassen sich dann Filter in epischer Komplexität basteln, die man über die RegaDom stülpen kann. Zu beachten, die folgenden Variablennamen sind schon Intern vorbelegt:

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 Filter, 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 progr  
Wenn die Bedin  
Punkt nur selekt  
ist. Das Object zu

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

Der rar Datei liegen Standardmäßig nun schon mal 2 Filter bei:

PROGRAM\_GeisterProg\_CopyID - Filter um Geisterprogramme mit gesetzter CopyID zu finden

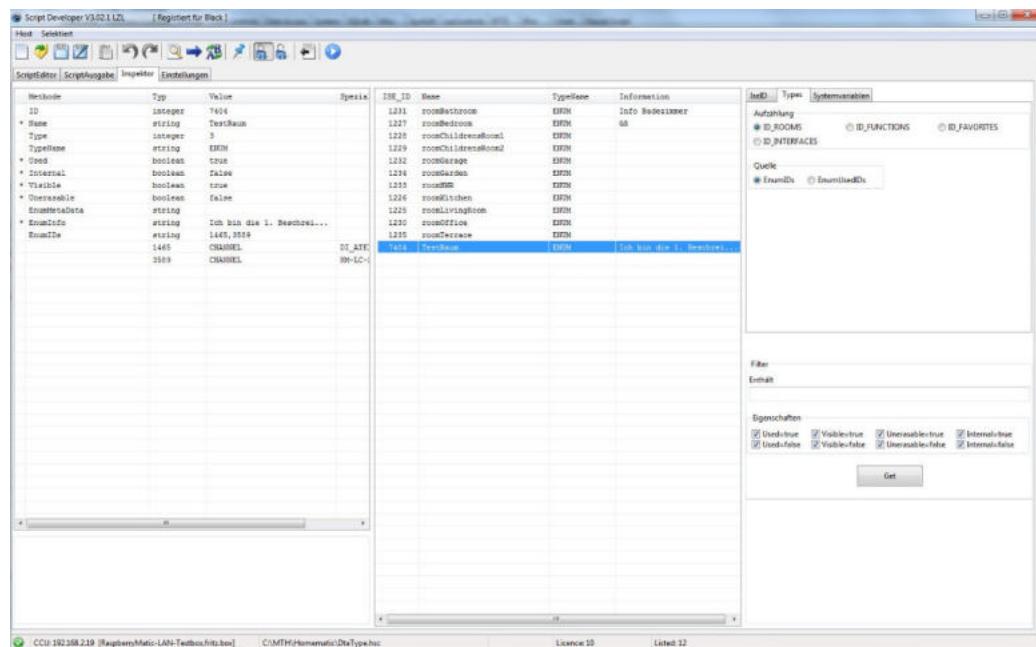
SYSVAR\_VerwaisterChannel – Filter um Systemvariablen zu finden, deren Channel verweis in Nirvana zeigt

Durch Click auf die Beschreibungszeile IsID 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 Sortieralgorithmen suboptimal arbeiteten, hat das ListView Object neue selektive Sortieralgorithmen bekommen. IsID sortiert nun wie man erwartet nach Integer aufsteigend, Name sortiert alphabetisch aufsteigend, TypeName sortiert alphabetisch, sind die Typenames gleich, wird innerhalb gleicher Typenames nach IsID numerisch sortiert. (Ab Version 3.08.08 verbesserter Sortieralgorithmus, hier wird durch Klick auf die Spalte zwischen Sortieren aufsteigend und Sortieren absteigend getoggelt.)



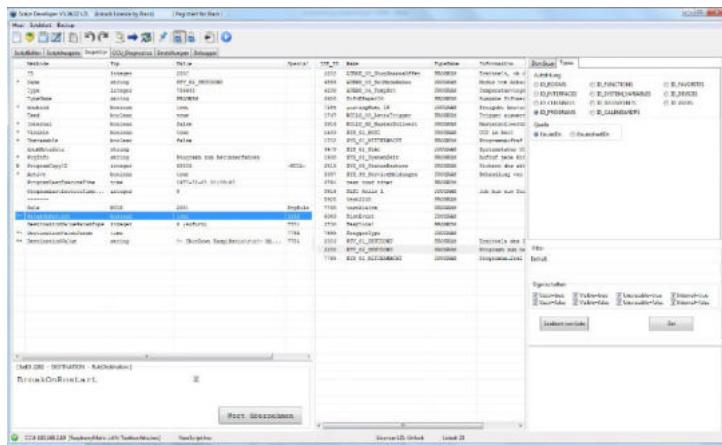
Die Spaltenbreite kann sowohl in Normalansicht als auch in Maximize separat eingestellt werden (Das Programm sollte sich die Breiten merken und je nach Darstellungsart automatisch wieder einstellen, sollte...)

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)

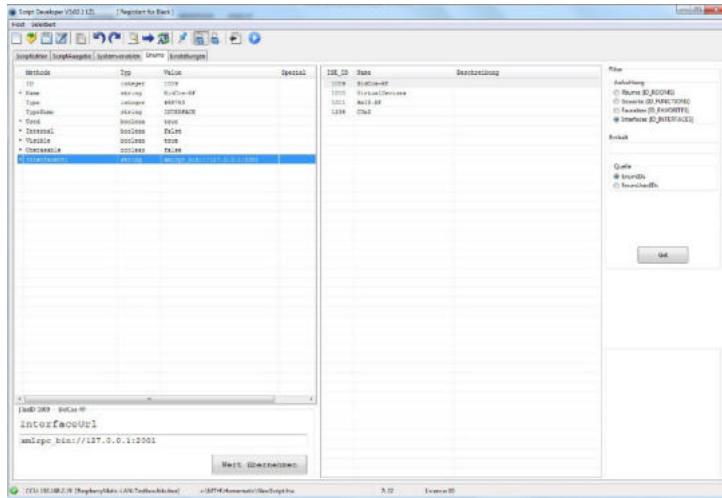
Properties, die in der ersten Zeile mit einem Stern (\*) gekennzeichnet sind, können in ihrem Wert geändert werden.

Ab version 3.06.04: Properties, die in der ersten Zeile mit einem Doppelstern (\*\*\*) gekennzeichnet sind, können ebenfalls verändert werden. Der Doppelstern bedeutet dabei, die geänderte Property nicht element der Haupt ID ist, sondern sich rekursiv in einer untergeordneten Rekursionsebene befand.

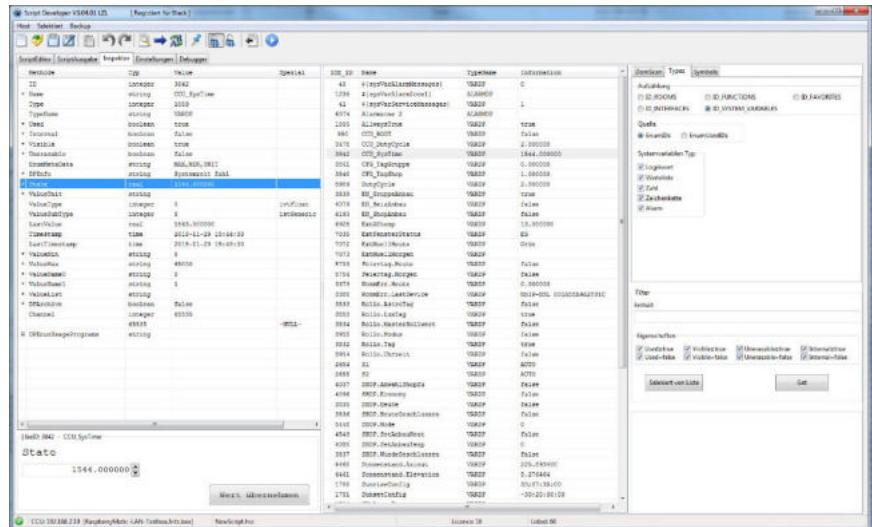


BreakOnRestart ist hier nicht Member von OT\_PROGRAM sondern rekursiv von der entsprechenden Rule bzw SubRule.

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.

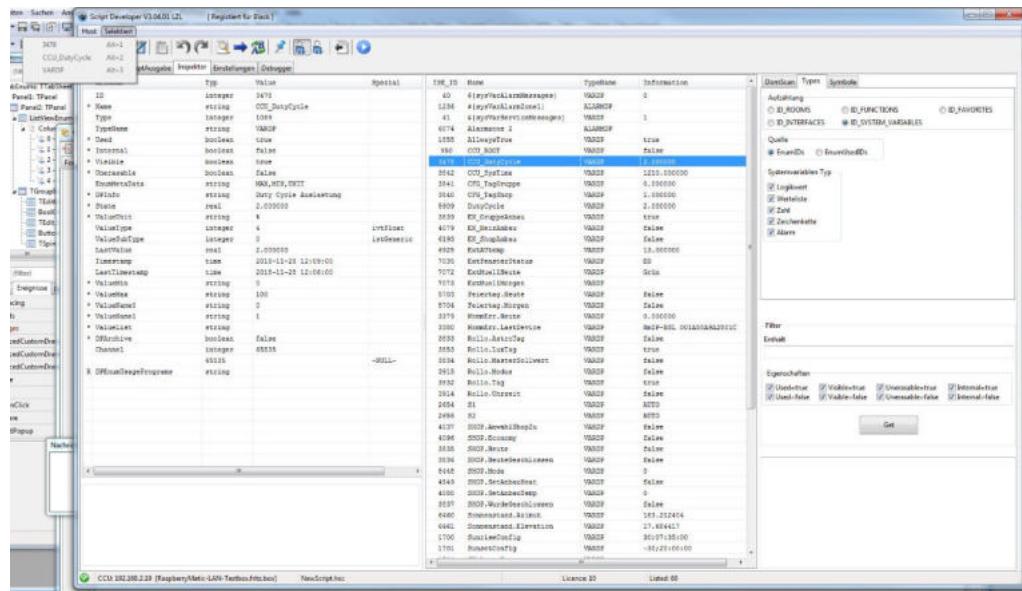


## 4.4 Daten aus Inspektor in Editor übernehmen

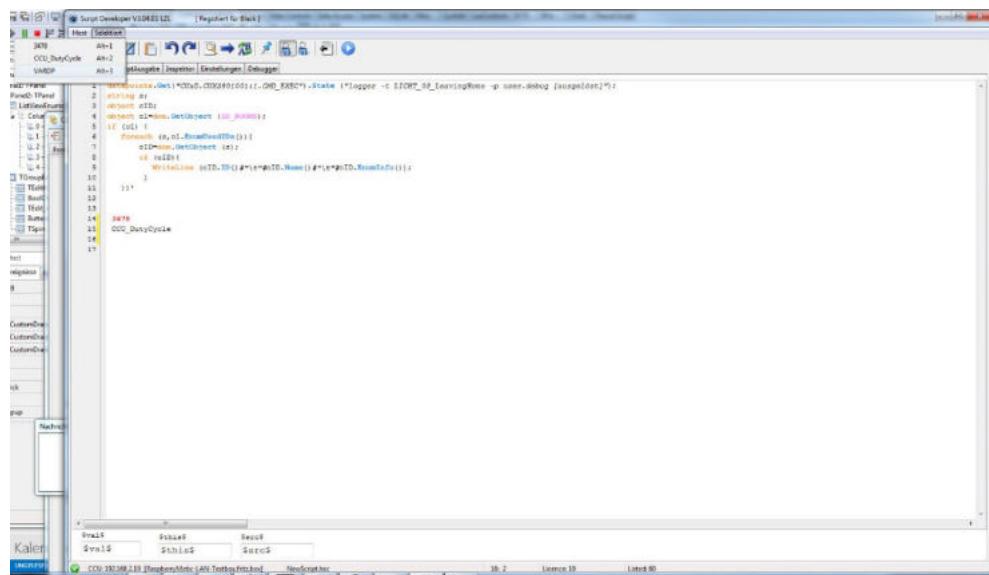
Zur schnelleren und auch möglichst fehlerfreien Bearbeitung besteht die Möglichkeit, Daten aus dem Inspektor direkt in den Editor zu übernehmen.

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, Alt-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

Ab 3.06.01

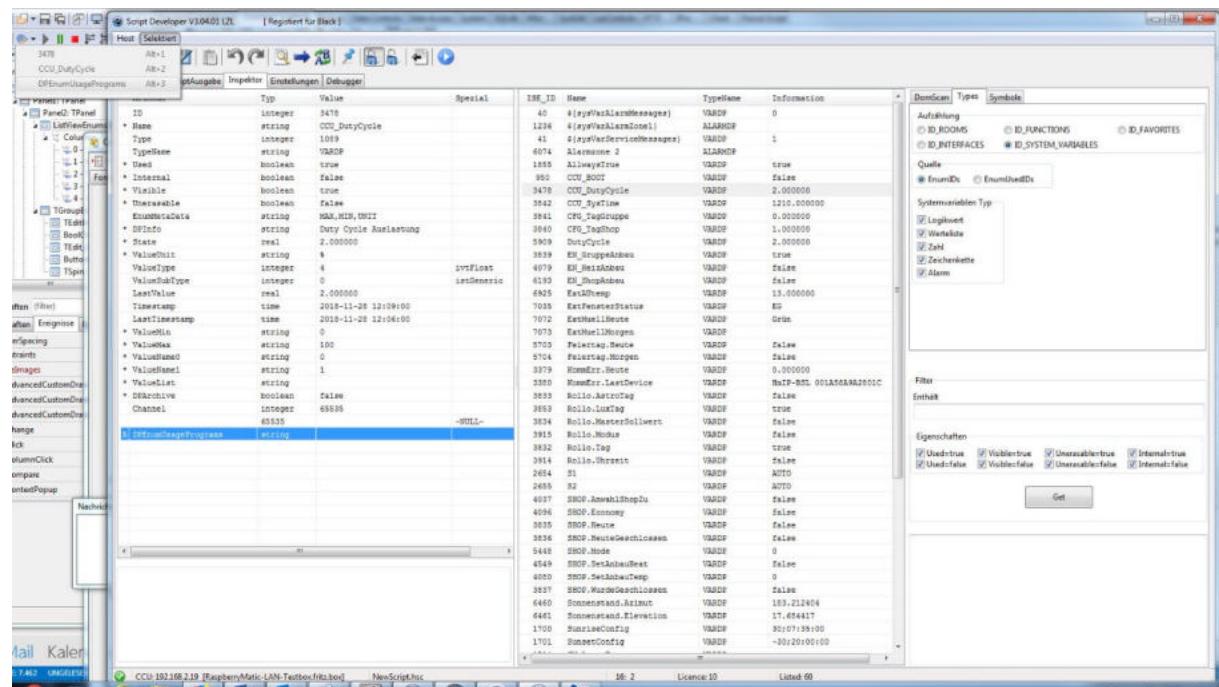
Mit den Tasten lassen sich folgende Spalteninhalte, welche vorher im Detailfeld selektiert worden sind, in den Editor übernehmen

Alt+m: Inhalt von Methode

Alt+t: Inhalt von typ

Alt+v: Inhalt von Value

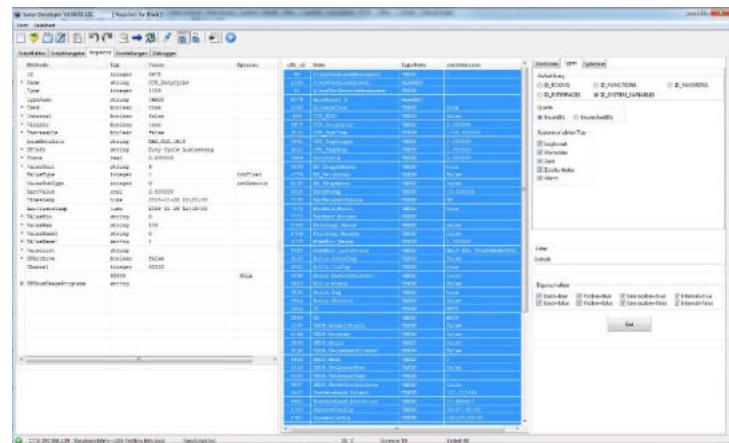
Alt+p: Inhalt von Spezial



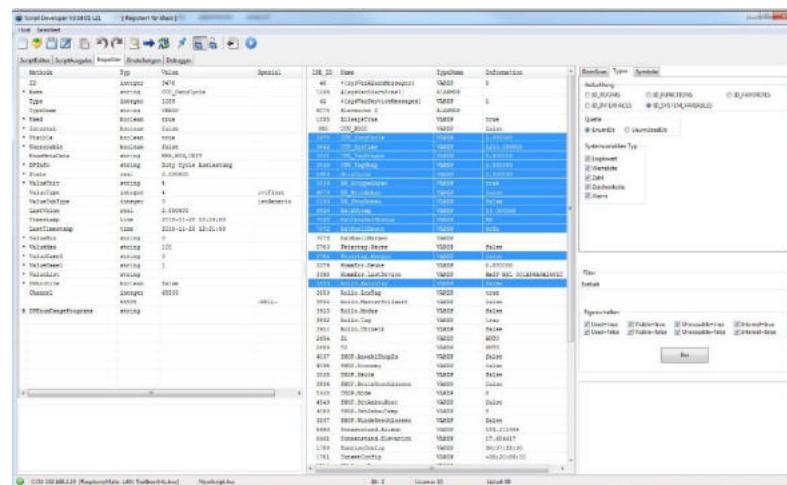
#### 4.4.1 Mehrfachauswahl als Enum String

Es lassen sich im Hauptauswahlfeld Mehrfachselektionen vornehmen.

Ctrl-A : alle auswählen.



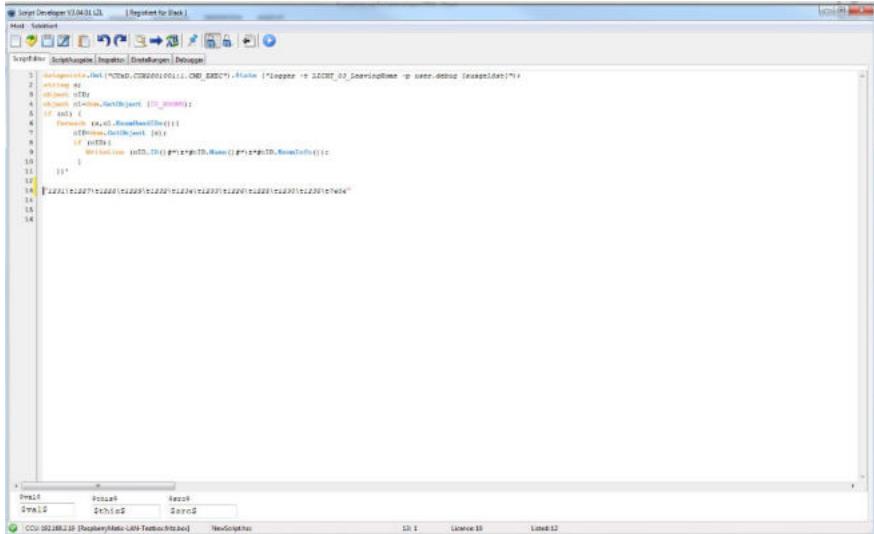
Oder die Übliche Mausbedienung:



Ab Version 3.07.07.

In der Listendarstellung kann nun mit den Cursortasten navigiert werden, ein Druck auf Return öffnet von dem selektierten Objekt die Detaildarstellung.

Mit der Taste markt sich der Inspektor die Auswahl, welche sich dann Script Konform im Editor Als ID-Enum durch die Einfügen Clipboard Taste, welche nach dem Pin Druck nicht mehr grau ist, lassen sie die selektierten ID,s im Editor einfügen (z.B. zur Verarbeitung in einem Script als foreach)



Die Pinliste funktioniert nicht nur mit dem Editor, auch im Inspektor lässt sich eine mit dem Pin gemerkte Selektionsliste wieder in die Auswahl laden:

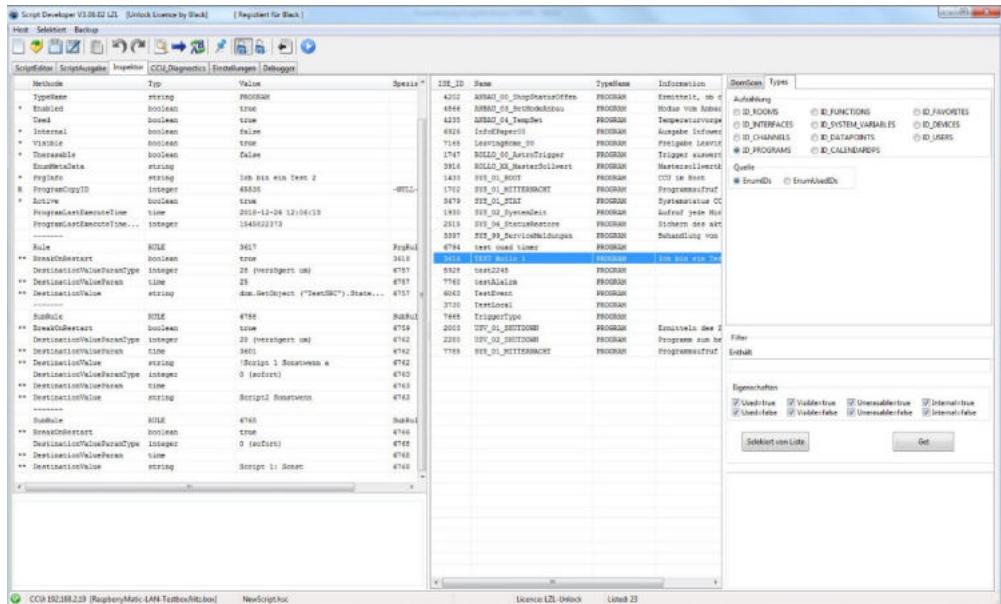
Mit rechter Maustaste im Mittleren Feld die Funktion „Einfügen aus Pinliste“ anwählen und die Sicherheitsabfrage bestätigen,



dann befinden sich die Selektierten Elemente wieder im mittleren Feld.

## 4.4.2 Übernahme von einem Script aus einem Programm direkt in den Editor

In der Darstellung eines Programmes werden in der Detaildarstellung auch die Rules und Subrules mit ihren Destinations/SingleDestinations aufgelöst, wenn diese ein Script enthalten



Dies entspricht der tabellarischen Darstellung dieses Programmes

Name	Beschreibung	Bedingung (Wenn...)	Aktion (Dann... Sonst...)	Aktion
TEST Rollo 1	Ich bin ein Test 2	Kanalzustand: DE_ATELIER_TUER:1 bei Fensterzustand: verriegelt bei Aktualisierung auslösen	Systemzustand: S1 sofort auf AUTO setzen	systemintern

Bedingung: Wenn... DE\_ATELIER\_TUER:1 bei Fensterzustand: verriegelt bei Aktualisierung auslösen

Aktivität: Dann... Vor dem Ausführen alle laufenden Verzögernungen für diese Aktivitäten beenden (z.B. Retriggers).  
Sollzustand: S1 sofort    Ablauf: 4010 Sekunden  
Zeige: dom.GetObject ("TestSRC").State ("sys") -> Weigert um: 5 Sekunden

Bedingung: Sonst, wenn... S1!sofort    Aktion: Bei Änderung auslösen

Aktivität: Dann... Vor dem Ausführen alle laufenden Verzögernungen für diese Aktivitäten beenden (z.B. Retriggers).  
Sollzustand: S0 sofort    Ablauf: 4010 Sekunden  
Zeige: dom.GetObject ("TestSRC").State ("sys") -> Weigert um: 5 Sekunden

Bedingung: Sonst, wenn... S0!sofort    Aktion: Bei Änderung auslösen

Aktivität: Dann... Vor dem Ausführen alle laufenden Verzögernungen für diese Aktivitäten beenden (z.B. Retriggers).  
Sollzustand: CF5\_TapGruppe sofort    Ablauf: 0.00 Sekunden

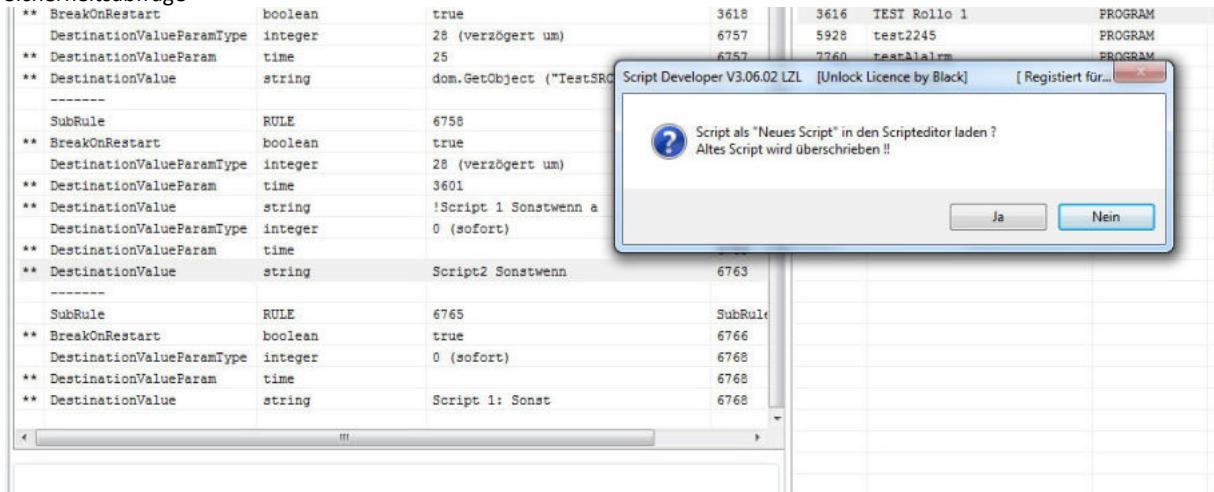
Bedingung: Sonst, wenn... HH-1C-Sw2-FM LEO131921110 sofort    Ablauf: 4010 Sekunden  
Zeige: Skript 1: Sonst... sofort

Rule entspricht hier der DANN Aktivität, dort wurde auch das Script gefunden, welches mit dom.GetObject („TESTSCR.....“) beginnt

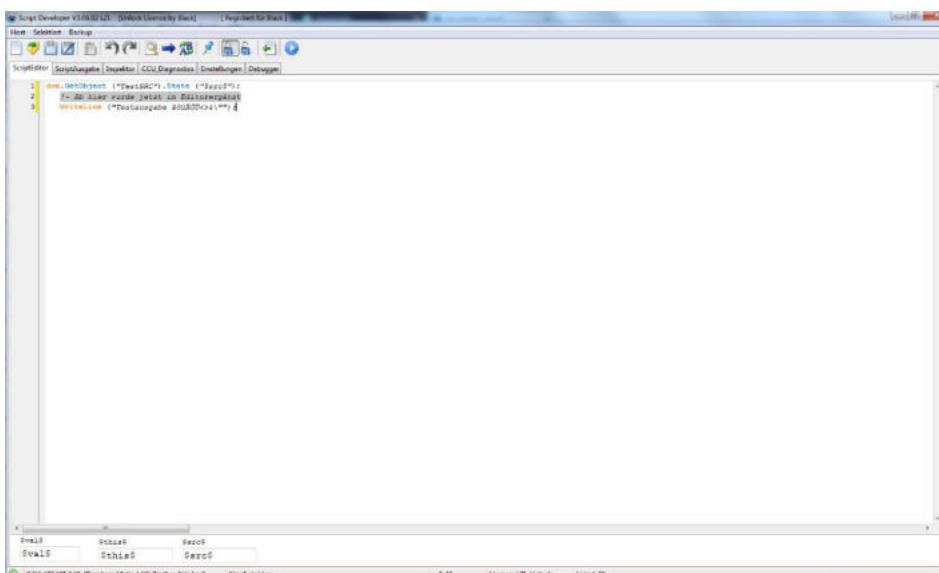
Die erste Subrule enthält dann die beiden Scripte aus der SonstWenn Bedingung.

Die Letzte Subrule entspricht der Sonst Bedingung. Nach dieser Auflistung lassen sich die Scripte tabellatisch im SDV wiederfinden. Zur Hilfe wird in der Spalte Value die ersten 70 Zeichen des Scriptes dargestellt.

Klick auf den Doppelstern (Doppelstern= Element wurde rekursiv aus dem Hauptelement aufgelöst) führt nach einer Sicherheitsabfrage



Zum Laden des Scriptes in den Scripteditor. Dort können dann die Änderungen durchgeführt oder Tests gemacht werden.

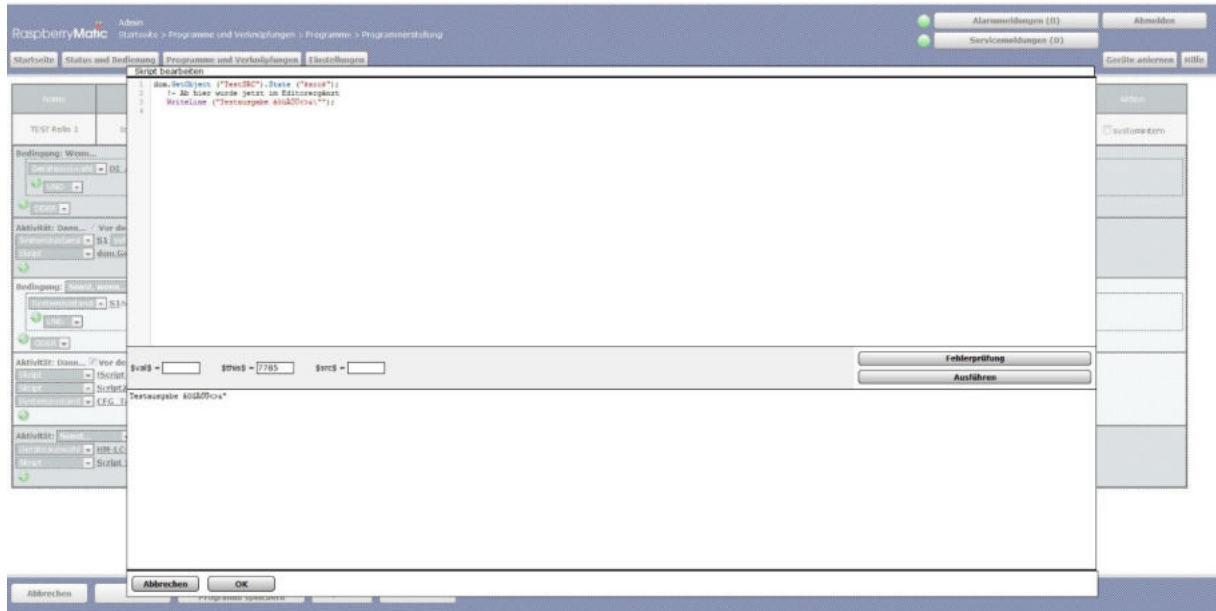


Danach kann durch Drücken der nun nicht mehr grauen Taste



Das Programm aus dem Editor direkt in das Programm der CCU geladen werden. Im Hint wird als Gedächtnissstütze der Name des Programmes mit angezeigt. Einschränkung: wird im Inspektor nun rumgeklickt und eine andere Property angewählt oder etwas anderes Selektiert. Wird laden in CCU Programm wieder grau (Grund: interner Bezug geht dann natürlich verloren) Dann ist kein direkter Upload mehr möglich (nur über im Editor CtrlA-CtrlC, im Inspektor neu Selektieren und laden des Scriptes erneut in den Editor, im Editor Ctrl-A, Del, Ctrl-V, Upload).

Auf der CCU lässt dich dann das geänderte und wieder uploadete Script öffnen, überprüfen und auch Ausführen.

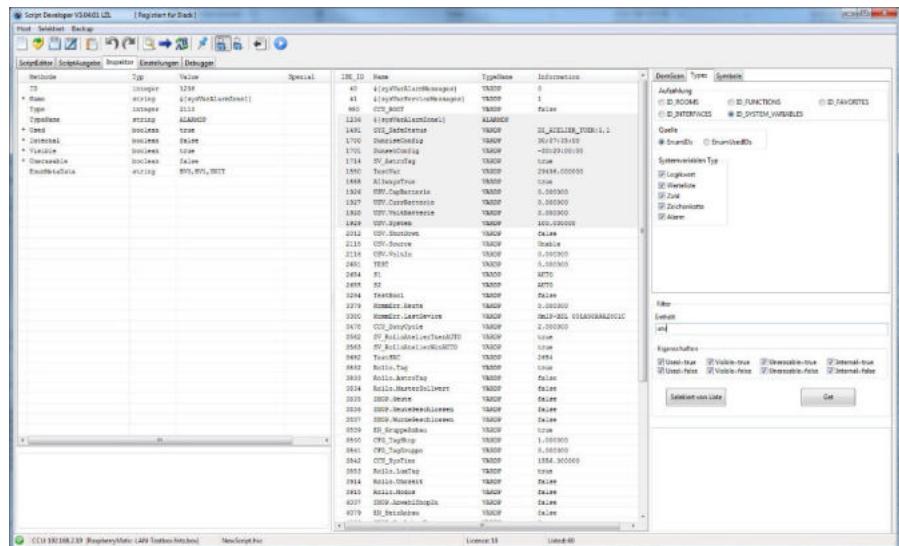


Das Programm würde nun das neue, geänderte Script bei Triggerung ausführen.

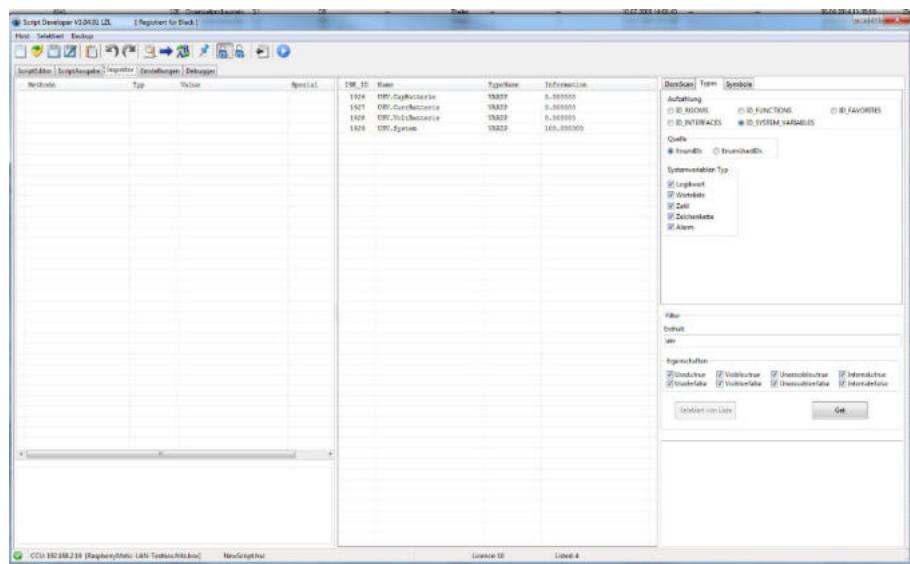
## 4.5 Selektion von Selektion

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 2012,2115,2116 werden hier nicht berücksichtigt, da diese sind selektiert sind. Bei <Druck auf: Selektiert von Liste: ergibt sich dann

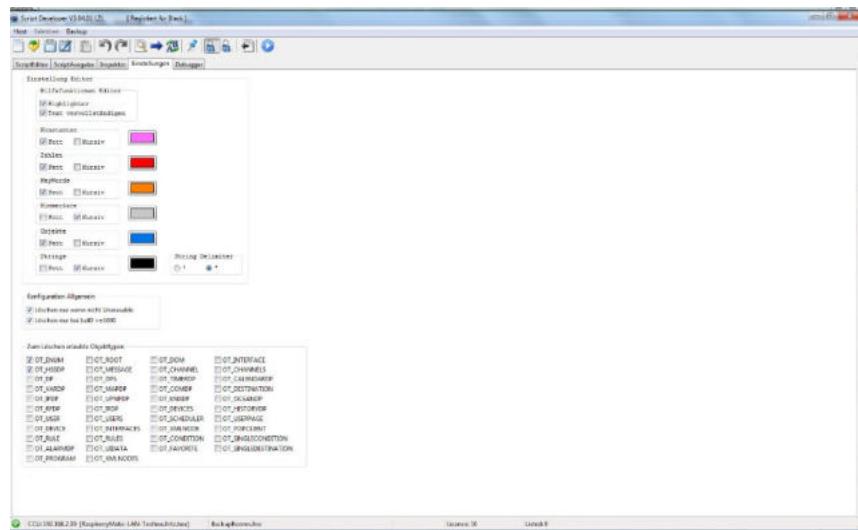


## 4.6 Objekte löschen

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 löscharbe Elemente zu löschen entweder:

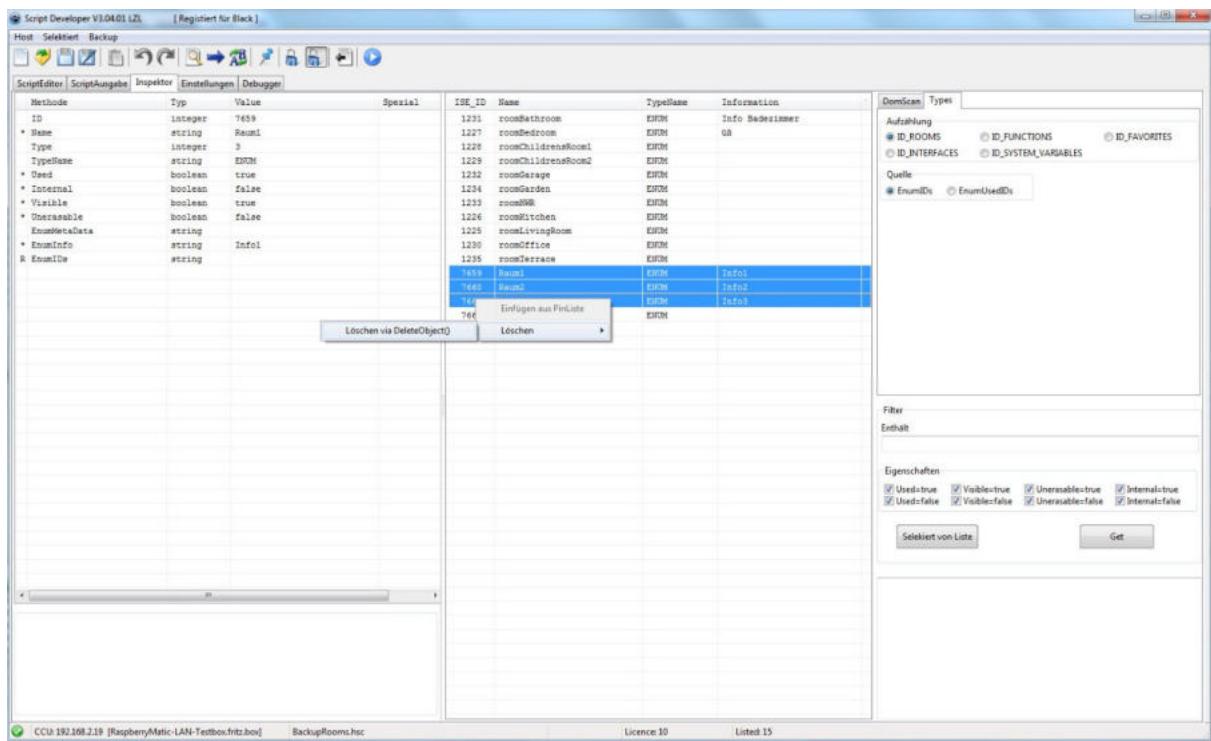
Im Inspektor unter Detailsview die Property entfernen (gilt nur für das Objekt), oder hier den Haken wegmachen (gilt für alle)

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.

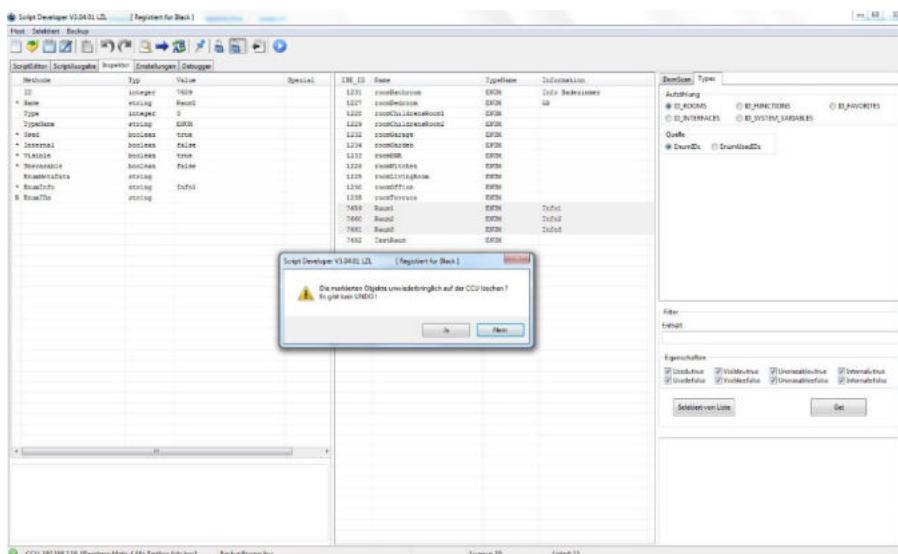
Löschbare Objekttypen. Die Letzte Sicherheit: ein zu lösches Objekt muss einen hier gechecked Objekttyp haben, sonst wird es nicht gelöscht.

Löschen läuft so ab:



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

Mehrfachselektion ist möglich

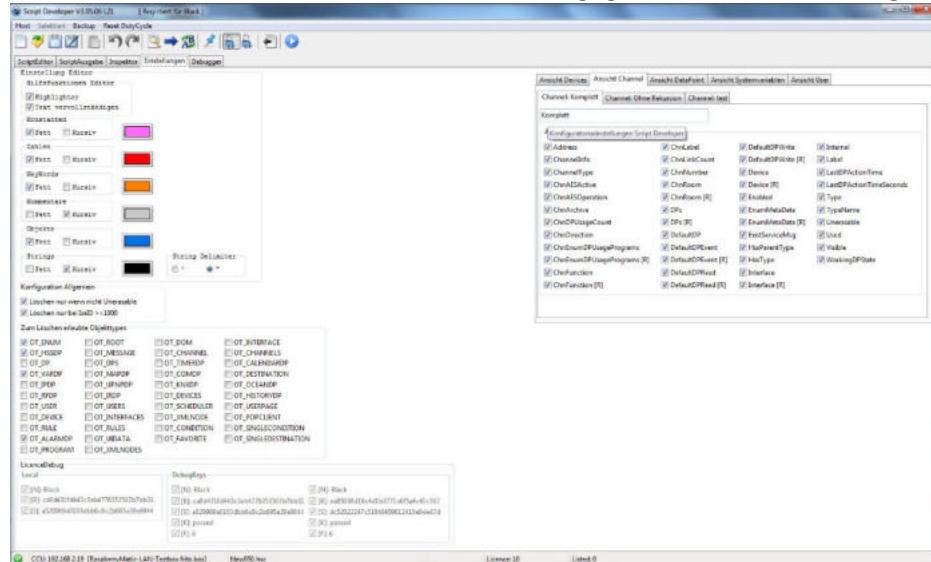


Nach dieser Sicherheitsabfrage sind die Objekte dann weg.. Zurück geht's dann nur mit Restore.

## 4.7 Anwenderdefinierte Sichten

Die Detailansichten können stellenweise sehr umfangreich sein und auf den ersten Blick mit Information zuwerfen. Deshalb ist es möglich, für manche Objekte drei Anwenderspezifische Sichten zu definieren.

Es werden dann in der Detailansicht nur die freigegebenen Methoden dargestellt.



Sichten können mit eigenen Namen versehen werden.

Die Entstellungen werden beim Verlassen gespeichert. Die jeweils geöffnete Sicht wird dann für das gefundene Objekt angewendet.

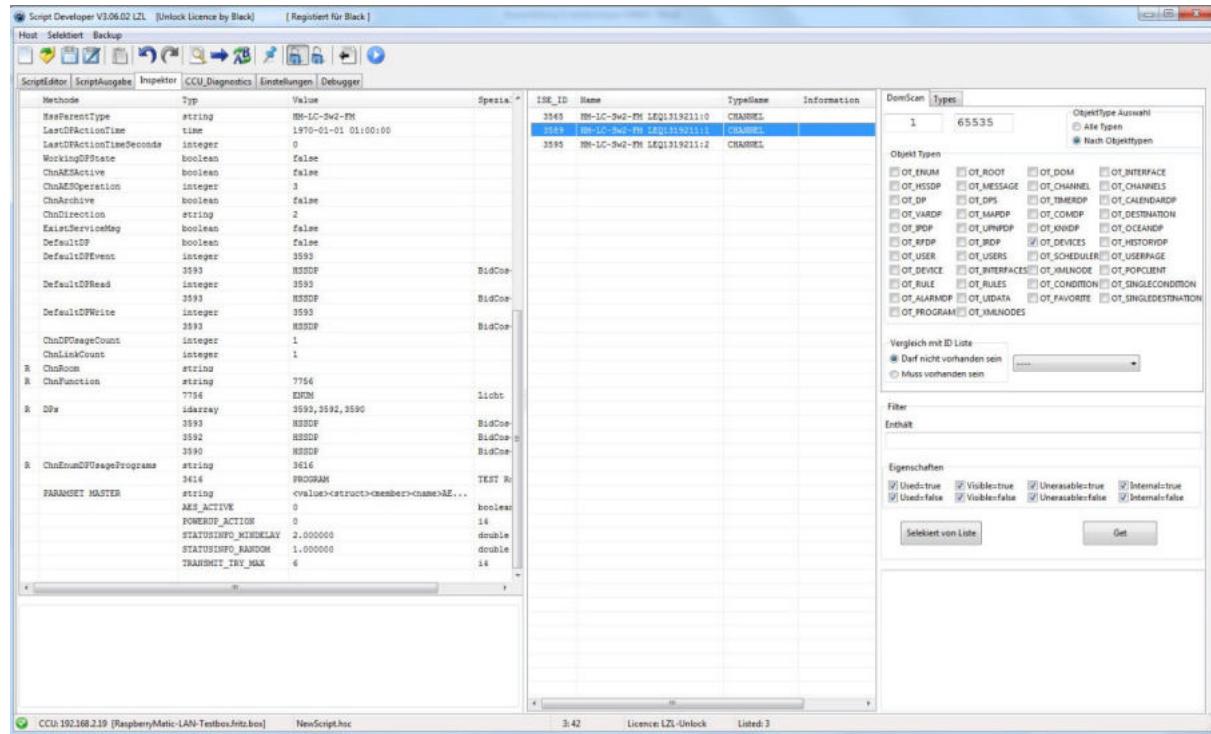
## 4.8 Browsing durch Rekursionsebenen

Seit der Version 3.06.04 verfügt der SDV über einen UNDO/REDO Stack im Inspektor. Dies bedeutet, dass immer, wenn in eine Rekursionsebene gesprungen wird, sich die Einträge im Selektionsfeld gemerkt werden und man über Undo / Redo dann zwischen den Ebenen hin und herspringen kann. Ein GET löscht dabei immer den Undo Stack

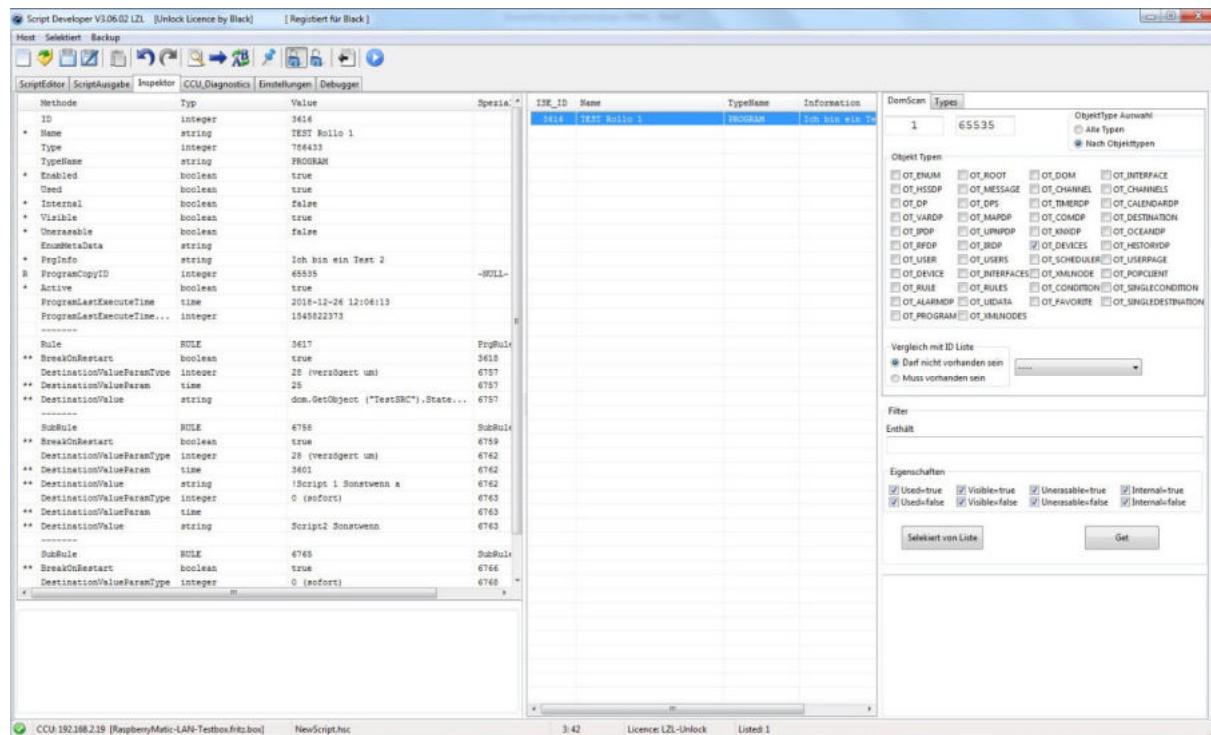
### Hier Beispiel Einsprung über RootDevices

Rekursiv Aufgelöst die EnumIDs und Anwahl des 2 Fach Schaltaktors (Hier zu sehen, Undo ist schon anwählbar geworden)

## Und rekursiv weiter über die Channels des Devices

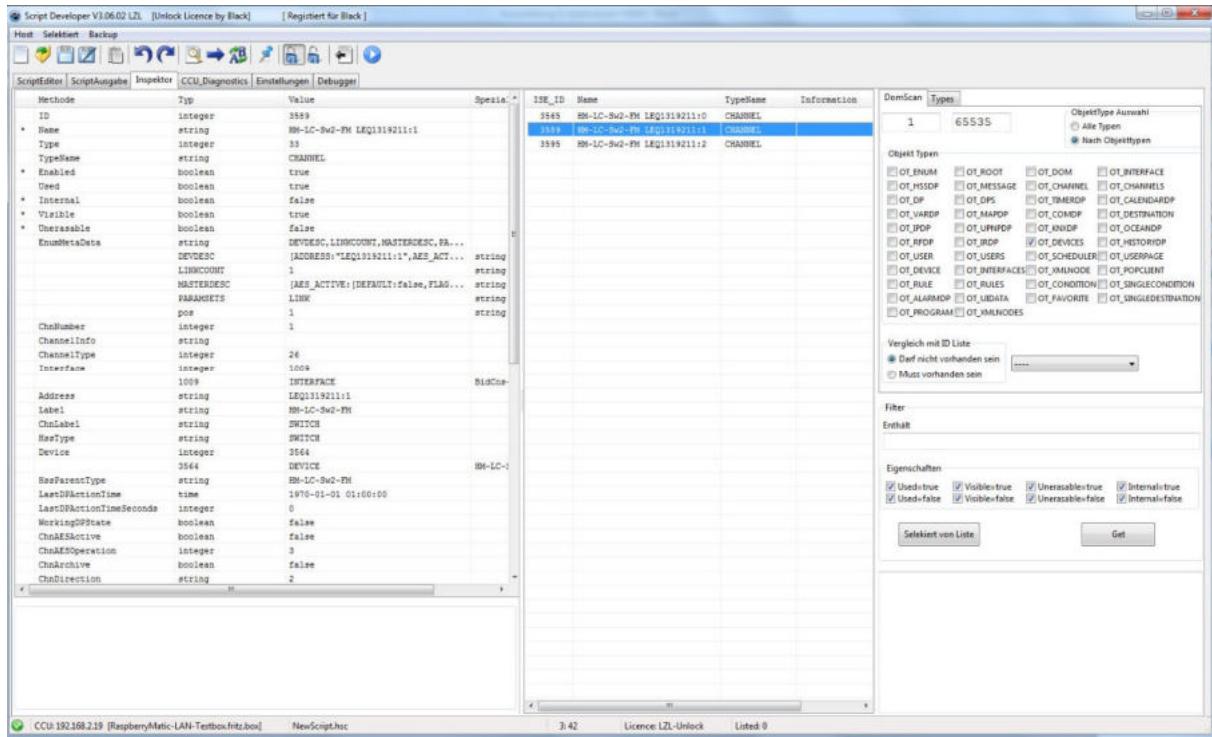


Mal in die Verwendung in dem Programm schauen über ChnEnumDPUsePrograms



Will ich jetzt aber wieder in den Channel zurück, so war dies in der alten Version nur mit Beginn der Selektion von ganz vorne angesagt.

Ab der 3.06.04 bin ich mit Undo eine Rekursionsebene zurück, hier in der Kanalauswahl, ab der ich direkt weitermachen kann



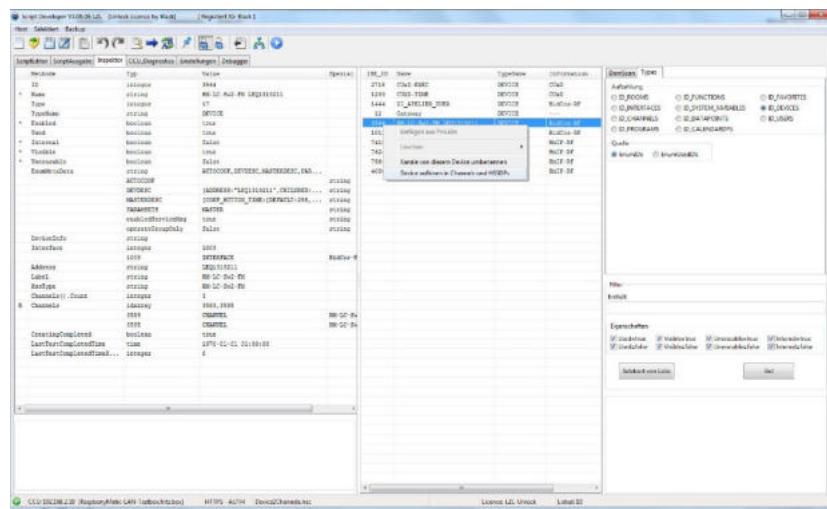
Ab Version 3.06.06 merkt sich der SDV zusätzlich zum Inhalt des Selektionsfeldes auch noch das zuletzt angezeigte Objekt in der Details view und stellt diese Ansicht auch wieder her (So das Objekt noch existent ist)

## 4.9 Auflösen komplexer Objekte

Der SDV wird die Möglichkeit haben, komplexe zusammen gesetzte Objekte aufzulösen und zur Bearbeitung zur Verfügung zu stellen.  
Beispielsweise Devices und auch Programme.

### 4.9.1 Auflösen von Devices

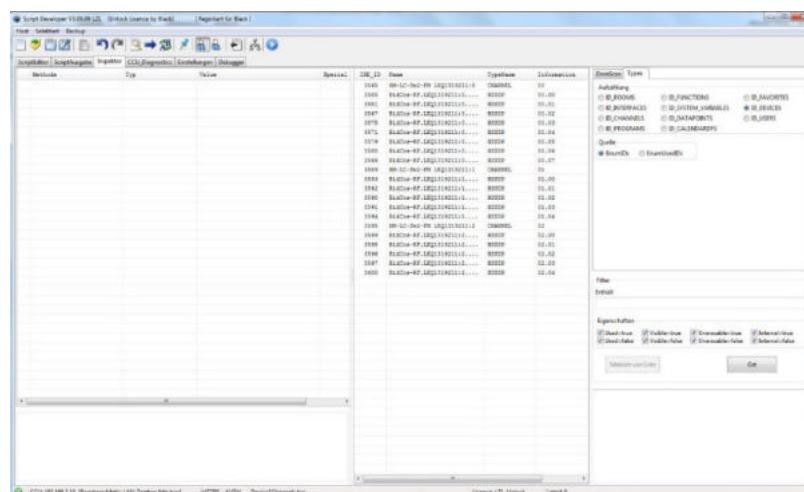
Ein Device besteht ja aus Channels und die Channels wiederum aus Datenpunkten



Wenn das Selektierte Objekt ein Device ist, so hat das PopUp Menü auf der rechten Maustaste nun auch das Feld: Device auflösen in Channels und HSSDPs.

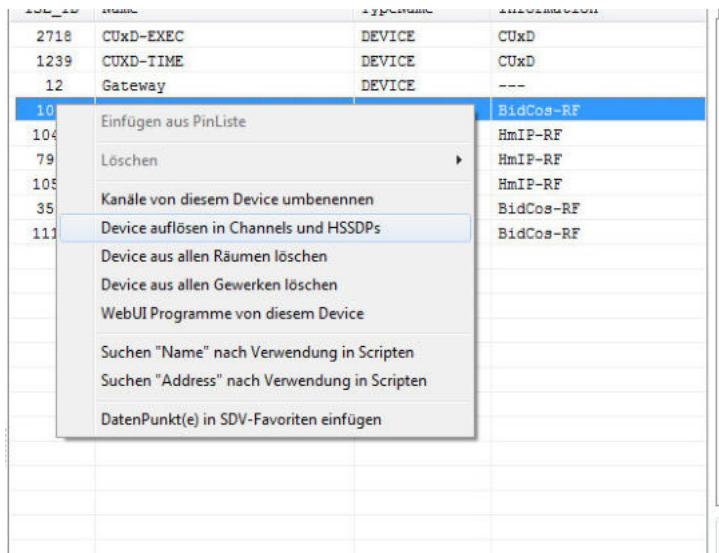
Alternativ über den neuen Menüpunkt . Dieser ist nicht ausgegraut, wenn sich ein Objekt auflösen lässt.

Als Ergebnis erhält man :



Diese Liste lässt sich dann weiter untersuchen mit den schon beschriebenen Arbeitsweisen (Auch Undo /Redo)

Wenn ein Device selektiert wurde, stehen mit Klick rechte Maustaste einige Bearbeitungsmenüs zur Verfügung



Das dargestellte Menü ist abhängig von dem selektierten Objekttyp.

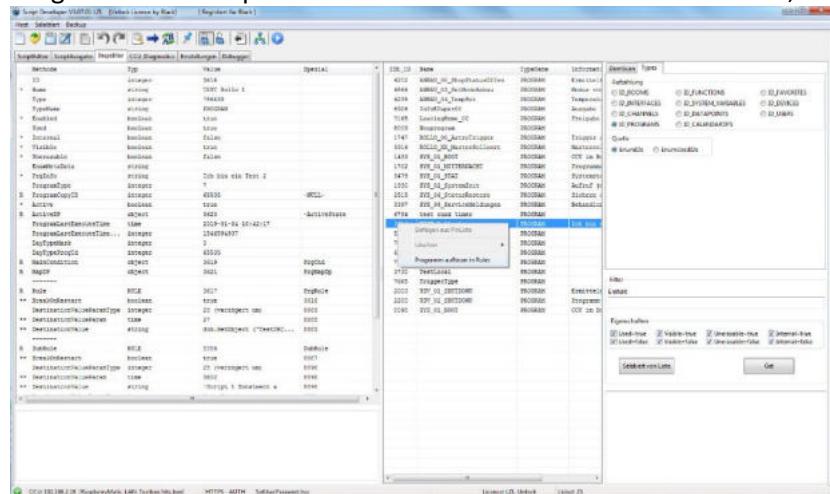
Bei Devices z.b.

Lassen sich alle untergeordneten Kanäle automatisch aus allen Räumen oder aus allen Gewerken entfernen. Dies war früher immer gerne eine Tipporgie.

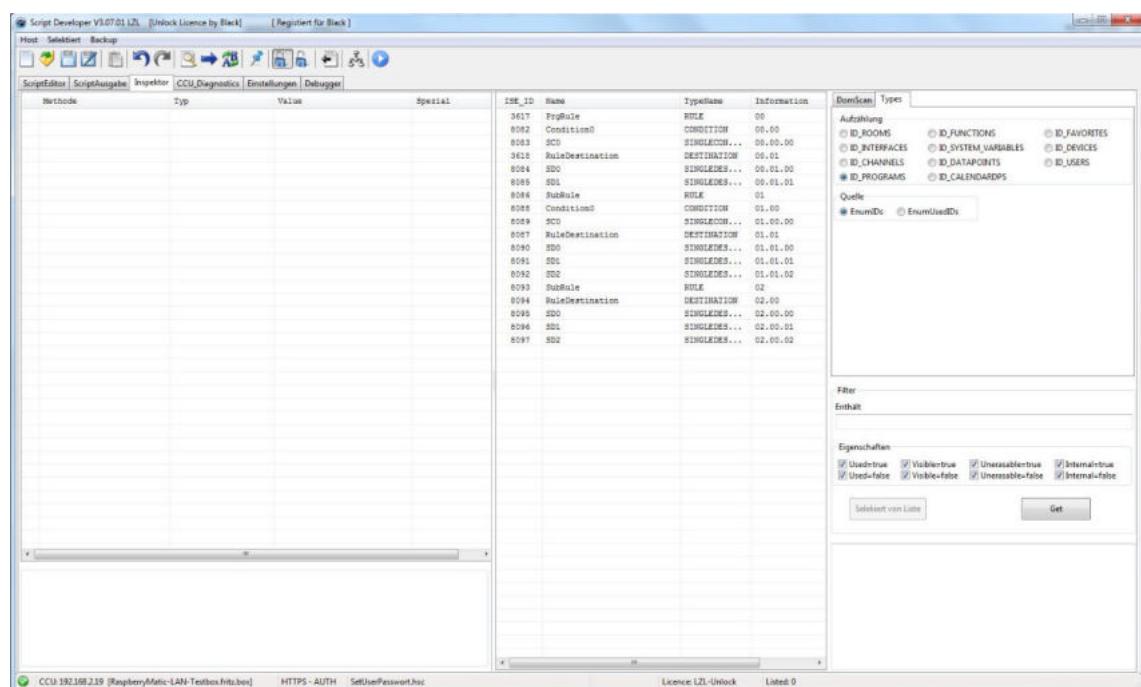
## 4.9.2 Auflösen von Programmen

Ein Programm besteht aus Rules (Regeln bzw den Subrules) und die jeweils aus den Conditions (und ihren untergeordneten SingleConditions sowie den Destinations und den untergeordneten Single Destinations

Aufgelöst wird adäquat zu den Devices: Auswahl über selektieren, dann rechte Maustaste und



Alternativ über den neuen Menüpunkt . Dieser ist nicht ausgegraunt, wenn sich ein Objekt auflösen lässt.



Die Sortierung über Information erlaubt eine chronologische Sortierung nach Auftreten der Objekte in einem Programm. Diese Liste lässt sich nun in der Detailansicht weiter untersuchen

Script Developer V3.07.01 LZL [Unlock Licence by Black] | Registriert für Black | Host: Selektiert Backup

ScriptEditor ScriptAusgabe Inspector CCU\_Diagnostics Einstellungen Debugger

Methode	Typ	Value	Spezial	IHE_ID	Name	TypeName	Information
ID	integer	8090		3417	ProgRule	RULE	00
Name	string	SDO		8082	Condition0	CONDITION	00.00
Type	integer	6251457		8083	SDO	SINGLECOM...	00.00.00
TypeIDName	string	SINGLEDESTINATION		3418	RuleDestination	DESTINATION	00.01
Enabled	boolean	true		8084	SDO	SINGLEDES...	00.01.00
Used	boolean	true		8085	SD1	SINGLEDES...	00.01.01
Internal	boolean	false		8086	SubRule	RULE	01
Visible	boolean	true		8088	Condition0	CONDITION	01.00
Unreachable	boolean	false		8089	SDO	SINGLECOM...	01.00.00
EnumMetadata	string			8087	RuleDestination	DESTINATION	01.01
DestinationChannel	integer	65535	-NULL-	8090	SDO	SINGLEDES...	01.01.00
DestinationDF	integer	65535	-NULL-	8091	SD1	SINGLEDES...	01.01.01
DestinationParam	integer	20	ivtString	8092	SD2	SINGLEDES...	01.01.02
DestinationParent	integer	8087	DESTINATION	8093	SubRule	RULE	02
DestinationProgram	integer	3416	PROGRAM	8094	RuleDestination	DESTINATION	02.00
DestinationValue	integer	20	ivtString	8095	SDO	SINGLEDES...	02.00.00
DestinationValueType	integer	20	ivtString	8096	SD1	SINGLEDES...	02.00.01
DestinationValueParam	string	'!Script 1 Done wenn a...		8097	SD2	SINGLEDES...	02.00.02
DestinationValueParamType	integer	20	ivtDelay				
DestinationValueParam	time	3602					2019-01-04 01:00

DomScan Types

Aufzählung

- ID\_ROOMS
- ID\_FUNCTIONS
- ID\_FAVORITES
- ID\_INTERFACES
- ID\_SYSTEM\_VARIABLES
- ID\_DEVICES
- ID\_CHANNELS
- ID\_DATAPOINTS
- ID\_PROGRAMS
- ID\_CALENDARIDPS

Quelle

- EnumIDs
- EnumUsedIDs

Filter

Eigenschaften

Used=true  Visible=true  Unreachable=true  Internal=true  
 Used=false  Visible=false  Unreachable=false  Internal=false

Selektiert von Liste

Get

CCU:192.168.2.19 [RaspberryMetric-LAN-Testbox.fritz.box] HTTPS - AUTH SetUserPassword.jsc Licence: LZL-Unlock Listed: 0

## 4.10 Zeitmodule

Zeitmodule lassen sich nun auch in Detailansicht darstellen

Script Developer V3.07.01 LZL [Unlock Licence by Black] | Registriert für Black | Host: Selektiert Backup

ScriptEditor ScriptAusgabe Inspector CCU\_Diagnostics Einstellungen Debugger

Methode	Typ	Value	Spezial	IHE_ID	Name	TypeName	Information
Internal	boolean	false		3409	Zeitmodul	CALENDAROP	
Visible	boolean	true		5971	Zeitmodul	CALENDAROP	
Unreachable	boolean	false		4078	Zeitmodul	CALENDAROP	
EnumMetadata	string						
State	boolean	true					
Time	time	1970-01-01 01:00:00					
TimeSeconds	integer	0					
TimerType	integer	4	ttPeriodic				
CalDuration	integer	0					
CalRepeatTime	time	1970-01-01 01:00:00					
CalRepeatTimeSeconds	integer	0					
Weekdays	integer	0					
Weekdays	integer	0					
Period	integer	60					
Begin	time	2018-01-19 00:00:00					
End	time	1970-01-01 01:00:00					
CalRepetitionCount	integer	0					
SunOffsetType	integer	0	setNone				
Next	time	2019-01-04 22:23:00					
Holidays	boolean	false					
VacationCheck	boolean	false					
VacationCheckType	boolean	false					
BeginSeconds	integer	1516316400					
EndSeconds	integer	0					
DPInfo	string						
DPUusageCount	integer	1					
R_DPUusageProgram	string	1930					
Timestamp	time	2019-01-04 22:22:00	SYB_02_System				
LastValue	boolean	true					
LastTimestamp	time	2019-01-04 22:21:00					
LastTimestampSeconds	integer	1544654860					
LastTriggerID	integer	65535					
LastTriggerTime	time	1970-01-01 01:00:00					

DomScan Types

Aufzählung

- ID\_ROOMS
- ID\_FUNCTIONS
- ID\_FAVORITES
- ID\_INTERFACES
- ID\_SYSTEM\_VARIABLES
- ID\_DEVICES
- ID\_CHANNELS
- ID\_DATAPOINTS
- ID\_PROGRAMS
- ID\_CALENDARIDPS

Quelle

- EnumIDs
- EnumUsedIDs

Filter

Eigenschaften

Used=true  Visible=true  Unreachable=true  Internal=true  
 Used=false  Visible=false  Unreachable=false  Internal=false

Selektiert von Liste

Get

CCU:192.168.2.19 [RaspberryMetric-LAN-Testbox.fritz.box] HTTPS - AUTH SetUserPassword.jsc Licence: LZL-Unlock Listed: 3

## 4.11 Suchen in Scripten nach Variablen, Devices etc

Ab der Version 3.07.02 kann in den vorhandenen Scripten nach dem Vorkommen von Systemvariablen, Geräten, Kanälen, Räumen und Gewerken gesucht werden.

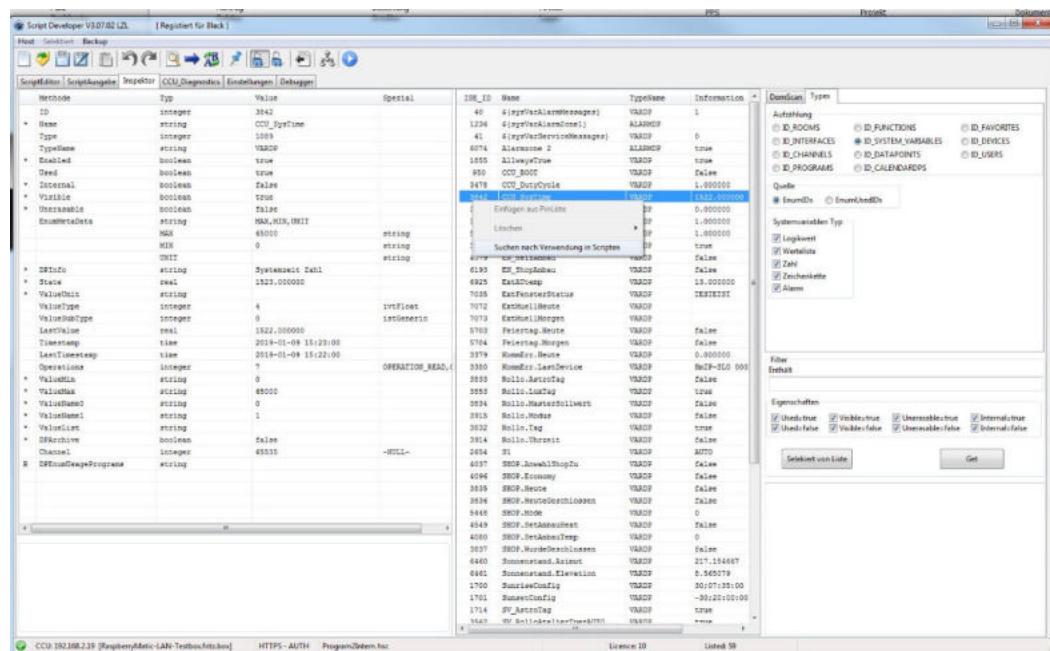
Dazu werden sämtliche SingleConditions, die als Property ein Script enthalten, gesucht und dann via Stringvergleich nach dem oder den Vorkommenden Namen abgesucht.

## Vorgehensweise:

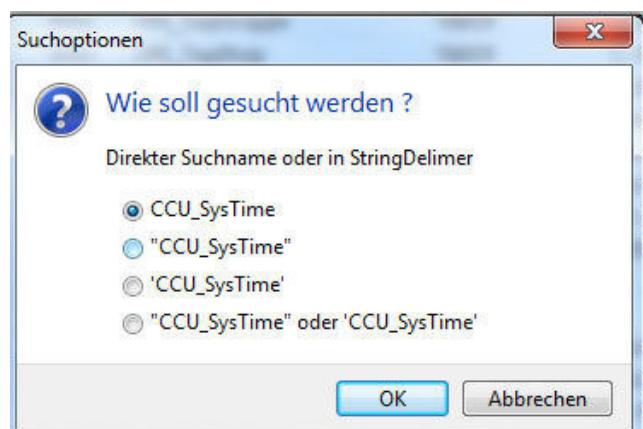
Markieren der oder auch mehrere Suchvariablen (Können Systemvariablen, Devices, Kanäle, Räume und Gewerke sein) Rechte Maustaste und suchen nach Verwendung in Scripten

Es kann hierbei gesucht werden:

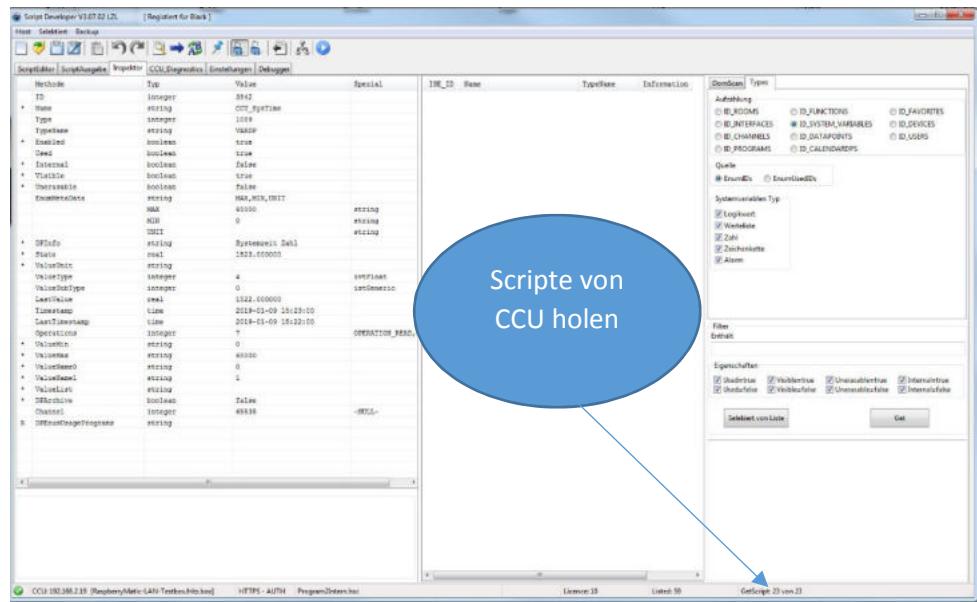
- Suchen "Name" nach Verwendung in Scripten : Suche nach dem Namen (SV, Alarme, Device, Chans)
  - Suchen "Address" nach Verwendung in Scripten : Sucht nach der Seriennummer (Address) von Devices und Channels



Im darauf sich öffnenden Dialog festlegen ob der reine Text oder der Text in StringDelimitern gesucht werden soll

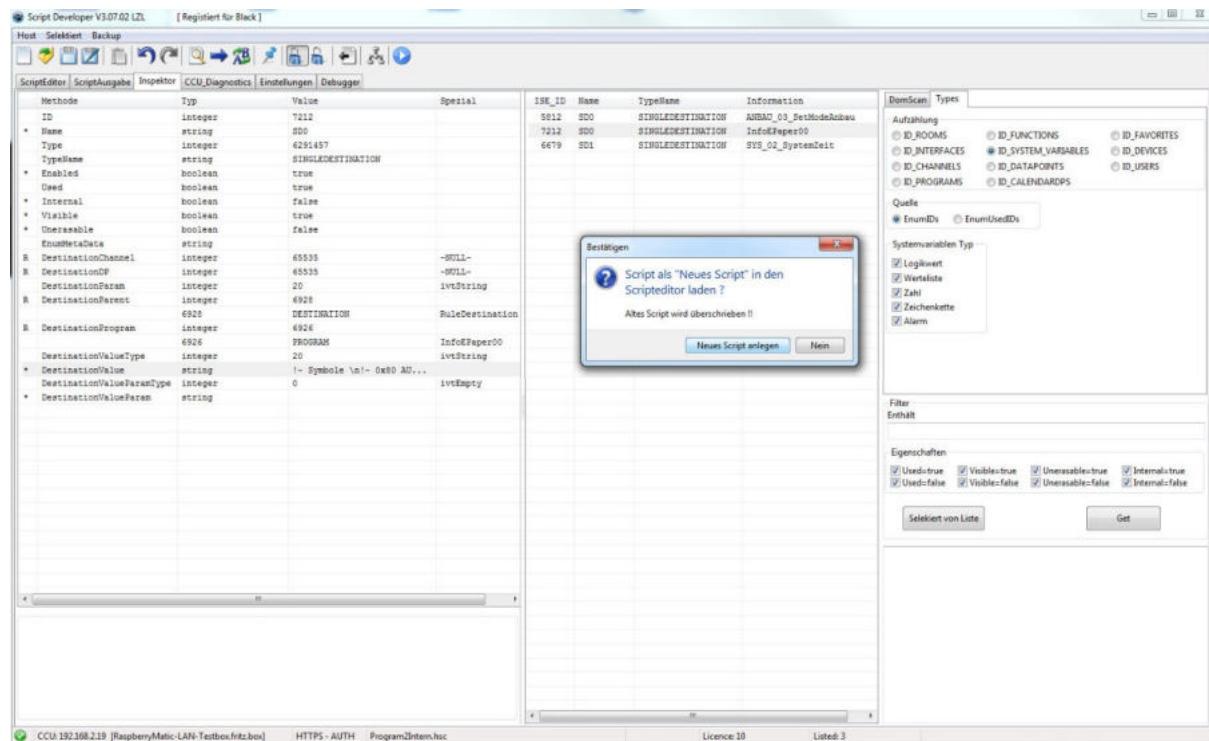


Daraufhin werden erstmal alle Rule in ihre Destinations aufgedröselt und eine Liste angelegt, wie viele Scripte es in diesen SingleDestinations dann gibt. Diese werden in den PC geladen. Das geht recht Fix, um der CCU zwischenzeitlich Zeit für Ihre Aufgaben zu lassen, wird direkt nach dem Empfang PC seitig die Stringanalyse gemacht und die Ergebnisliste aufbereitet.



Wenn fertig, gibt es eine Liste der Singledestinations, die die Scripte enthalten, in welchem der gesuchte Name vorhanden ist. Im Informationsfeld wird auch noch der zu der SingleDestination gehörendem Programm angezeigt:

Das Script lässt sich dann nach Rückfrage öffnen:



Der Editor öffnet sich und es wird direkt zum ersten Vorkommen des zu suchenden Namens gesprungen. bei mehrfachem Vorkommen sind alle Stellen gemäß den MarkUp Einstellungen hervorgehoben. Mit dem Pfeil für Weitersuchen lässt sich dann auch durch den Text nach dem Namen durchsuchen unabhängig von der Hervorhebung. Bei suchen Ersetzen ist als Suchbegriff der zu suchende Name schon vorbesetzt. Ersetzen automatisiert ist nicht vorgesehen, da sollte als letzte Instanz der Mensch das letzte Wort haben. Ein geändertes Skript lässt sich auch direkt wieder auf die CCU in das ursprüngliche Programm hochladen

Script Developer V3.07.02 LZL | Registriert für Block |

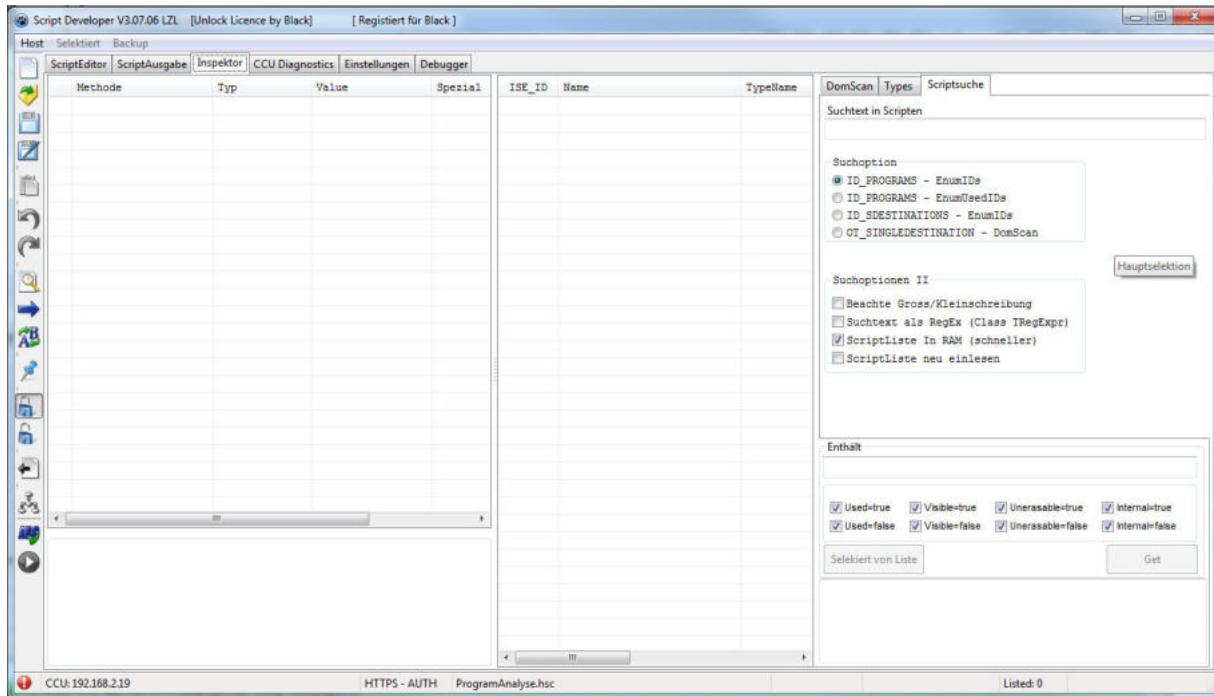
Hilfe Selektion Backup

ScriptEditor ScriptAusgabe Inspector CCU\_Diagnostics Einstellungen Debugger

```
29 |- 0xC1 LANG LANG
30 |- 0xC2 LANG KURZ
31 |- 0xC3 LANG KURZ KURZ
32 |- 0xC4 KURZ
33 |- 0xCE KURZ KURZ
34 |- 0xCF KURZ
35
36 /* Signale
37 |- 0xF0 AUS
38 |- 0xF1 Rotes Blitzen
39 |- 0xF2 Grünes Blitzen
40 |- 0xF3 Orangenes Blitzen
41
42 /* Seriennummer des BN-Disp-EP-#NS5,
43 string seriennummer = "WQ1463971";
44 object o3W = dom.GetObject("ID_SYSTEM_VARIABLES").Get("SVSubmit_DI_EP_Flur");
45
46 /* Variable für die Fenster und Icons
47 /* Nicht verwendete Variablen nicht löschen, sondern auf "" setzen
48 /* Beim Verarbeiten von Objekten kann es vorkommen, dass der Name einer Variablen ein Objekt ist. In diesem Fall kann es vorkommen, dass die Variablen nicht mehr freigegeben werden können. Um dies zu verhindern, müssen Sie die Variablen mit einem anderen Namen umbenennen.
49 object oObj = dom.GetObject("ID_SYSTEM_VARIABLES").Get("ExtFensterStatus");
50 if (oObj.State != "") {
51     smile1= "w1";
52     icon1= "";
53 } else {
54     smile1= oObj.State;
55     icon1= "0x62";
56 }
57
58 string smile3 = dom.GetObject("ID_SYSTEM_VARIABLES").Get("ExtADtemp").State.ToString(1) # " C ";
59 string icon3 = "";
60
61 /* Multistatus nach der Uhrzeit darstellen
62 /* Wenn heute Null und Uhrzeit zwischen 0-8 Uhr Heute Null darstellen
63 /* Andernfalls
64 /* Wenn Morgen null und Uhrzeit zwischen 18 und 24 Uhr.. Morgen Null darstellen
65 integer iSysTime= dom.GetObject("ID_SYSTEM_VARIABLES").Get("ExtSystem").State.ToInt32();
66 string smile1= "";
67 string icon1= "";
68 object oObj = dom.GetObject("ID_SYSTEM_VARIABLES").Get("ExtNullHeute");
69 if ((oObj.State != "") & (iSysTime>20) && (iSysTime<830)) {
70     smile1= oObj.State;
71     icon1= "0x6f";
72 }
73
74
75 $val1$ $this$ $src$ $val2$ $this$ $src$
```

## 4.12 Volltextsuche in Scripten

Ab Version 03.07.07 existiert eine Volltextsuche (ab Level 6)



Suchtext ist selbsterklärend.

Entweder wird nach dem Vorkommen des Textes in dem Script gesucht oder aber, wenn der Haken bei Suchtext als Regex gesetzt wurde, wird dieser Suchtext als regular Expression ausgewertet. (Da hier die Class TRegExpr von Lazarus verwendet wurde... das ganze in POSIX Syntax)

### ID\_Programs – EnumIDs

Es wird in der Aufzählung ID\_PROGRAMS , alle dort gelisteten IDs iteriert und in allen Skripten gesucht, die dort enthalten sind.

### ID\_Programs – EnumUsedIDs

Es wird in der Aufzählung ID\_PROGRAMS , alle dort gelisteten EnumUsedIDs iteriert und in allen Skripten gesucht, die dort enthalten sind.

### ID\_SDESTINATIONS – EnumIDs

Es wird in der Aufzählung ID\_SDESTINATIONS , alle dort gelisteten IDs iteriert und in allen Skripten gesucht, die dort enthalten sind.

### ID\_SINGLEDESTINATION – DomScan

Es wird die gesamte Regadom nach Objecten vom Typ Singledestination durchsucht und in allen Skripten gesucht, die dort enthalten sind. Diese Suchmethode dauert am längsten, findet aber auch Geisterobjekte.

### **Beachte Gross/Kleinschreibung**

Selbsterklärend, ist der Haken gesetzt, muss das Wort genauso in Gross Kleinschreibung vorhanden sein, ansonsten ist Gross Kleinschreibung egal

### **Suchtext als Regex**

Der Suchtext wird als Regulärer Ausdruck interpretiert. .\*Son.t.\* findet alle Script die das Wort Sonst, aber auch z.B. Sonat enthalten. Syntax nachzulesen unter

[https://en.wikipedia.org/wiki/Regular\\_expression#POSIX%20Basic%20Regular%20Expressions](https://en.wikipedia.org/wiki/Regular_expression#POSIX%20Basic%20Regular%20Expressions)

### **Scriptliste in Ram**

Diese Suchoption ist schneller, hierbei wird nur beim ersten Suchlauf die Skriptliste in den Rechner geladen. Wenn mehrere Suchläufe gemacht werden, wird ab dem zweiten Durchlauf mit den Scripten aus dem Ram gearbeitet. Dies ist wesentlich schneller als jedes Mal die Skripte neu von der CCU zu laden.

### **Scriptliste neu einlesen**

Wurden Skripte geändert oder Skriptänderungen zwischenzeitlich auf der CCU gemacht, so sollte dieser Haken gesetzt werden. Beim nächsten Suchlauf wird die einmalig Skriptliste neu aus der CCU geladen

## 4.13 SingleDestinations in ihrer Reihenfolge ändern

Eigentlich ein Wunsch von einem der Tester. Die Reihenfolge der Anweisungen in einem Programm lässt sich abändern.

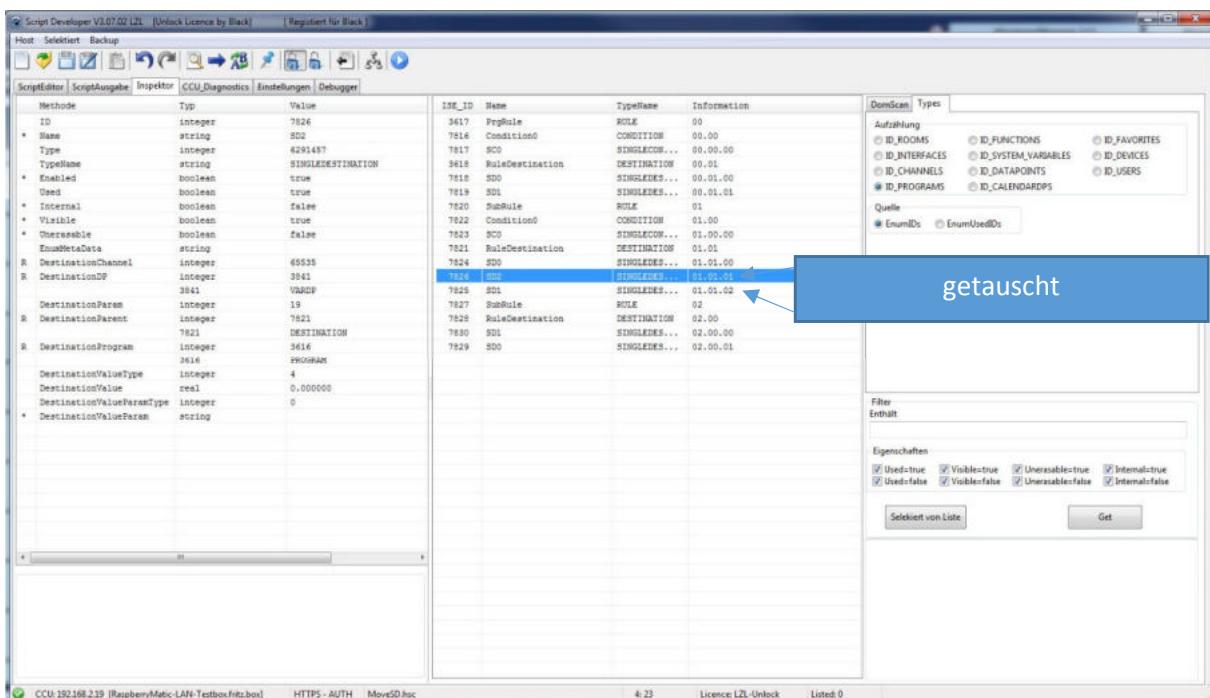
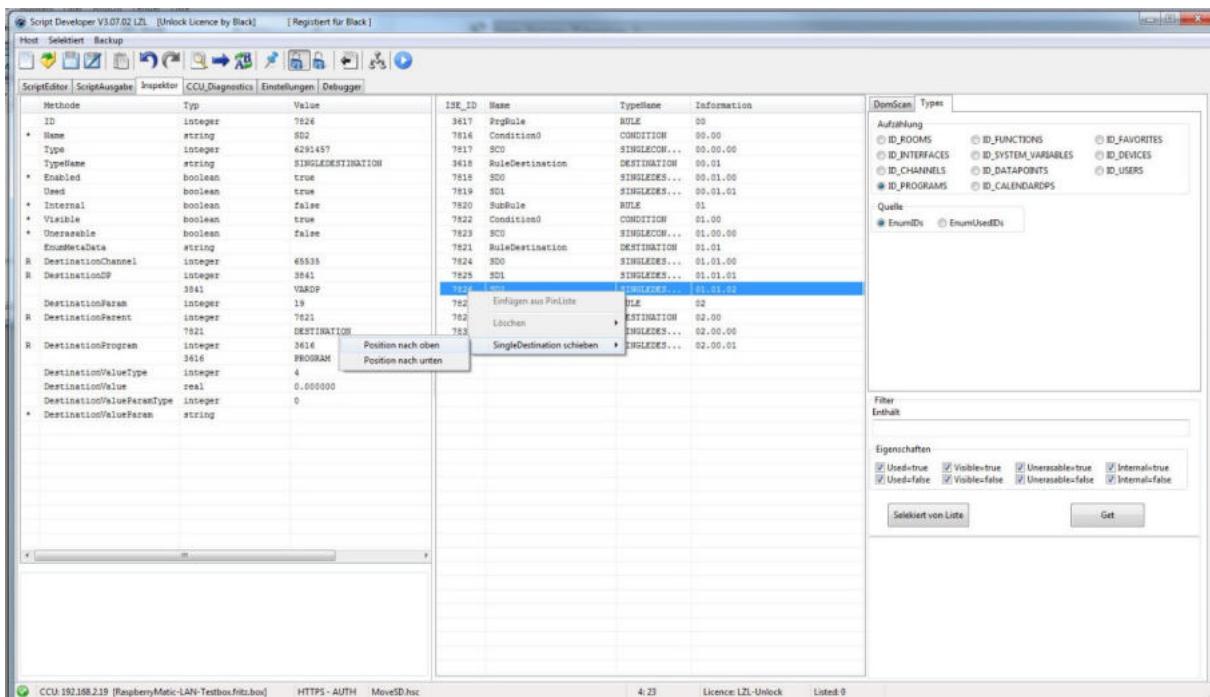
Vorher:

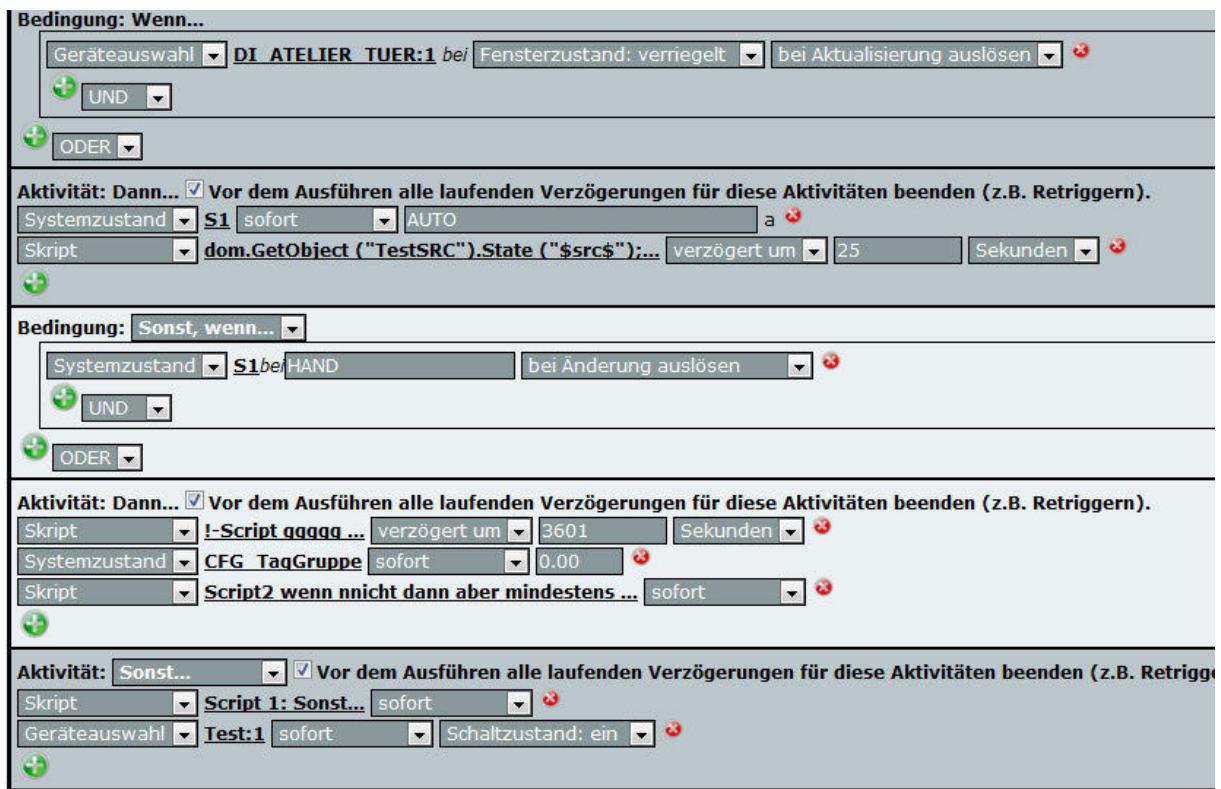
Name	Beschreibung	Bedingung (Wenn...)
TEST Rollo 1	Ich bin ein Test 2	Kanalzustand: DI_ATELIER_TUER:1 bei Fensterzustand: verriegelt bei Aktualisierung auslös
<b>Bedingung: Wenn...</b>		
Geräteauswahl ▼ DI_ATELIER_TUER:1 bei Fensterzustand: verriegelt □ bei Aktualisierung auslösen □		
+ UND □		
+ ODER □		
<b>Aktivität: Dann... ✓ Vor dem Ausführen alle laufenden Verzögerungen für diese Aktivitäten beenden (z.B. Retriggern).</b>		
Systemzustand □ S1 sofort □ AUTO □ a ↗		
Skript □ dom.GetObject("TestSRC").State("\$src\$"); verzögert um □ 25 □ Sekunden □		
+ ↗		
<b>Bedingung: Sonst, wenn...</b>		
Systemzustand □ S1 bei HAND □ bei Aenderung auslösen □		
+ UND □		
+ ODER □		
<b>Aktivität: Dann... ✓ Vor dem Ausführen alle laufenden Verzögerungen für diese Aktivitäten beenden (z.B. Retriggern).</b>		
Skript □ !-Script qqqq... verzögert um □ 3601 □ Sekunden □		
Skript □ Script2 wenn nnicht dann aber mindestens... □ sofort □		
Systemzustand □ CFG_TagGruppe □ sofort □ 0.00 □		
+ ↗		
<b>Aktivität: Sonst... ✓ Vor dem Ausführen alle laufenden Verzögerungen für diese Aktivitäten beenden (z.B. Retriggern).</b>		
Skript □ Script 1: Sonst... □ sofort □		
Geräteauswahl □ Test:1 □ sofort □ Schaltzustand: ein □		
+ ↗		

Das Aussehen im SDV

The screenshot shows the 'ScriptDeveloper' application window. On the left, there's a tree view with nodes like 'Selected', 'Backup', 'ScriptEditor', 'ScriptAusgabe', 'Inspector', 'CCU\_Diagnostics', 'Einstellungen', and 'Debugger'. The main area is a table with columns: 'Beschreibung', 'Typ', and 'Wert'. The table lists various rules and conditions, such as 'ProgRule', 'Condition0', 'SCU', 'RuleDestination', 'SDO', 'SDI', and 'SDO'. A specific row, '7821 RuleDestination', has its 'DESTINATION' value, '01-01-01', highlighted with a yellow background. To the right of the table is a sidebar with sections for 'Bereiche' (Rooms, Functions, System Variables, Devices, Channels, DataPoints, Users, Programs) and 'Aufzählung' (Rooms, Functions, System Variables, Devices, Channels, DataPoints, Users, Programs). Below the table is a 'Filtr' (Filter) section with checkboxes for 'Erweitert' (Advanced) and 'Ermittelt' (Calculated), and a 'Get' button. At the bottom, there's a status bar with 'CCU: 192.168.2.19 [RaspberryPi-LAN-Testbox] 4/23 Licence:L2L-Unlock Listed 0'.

Sieht hier so aus

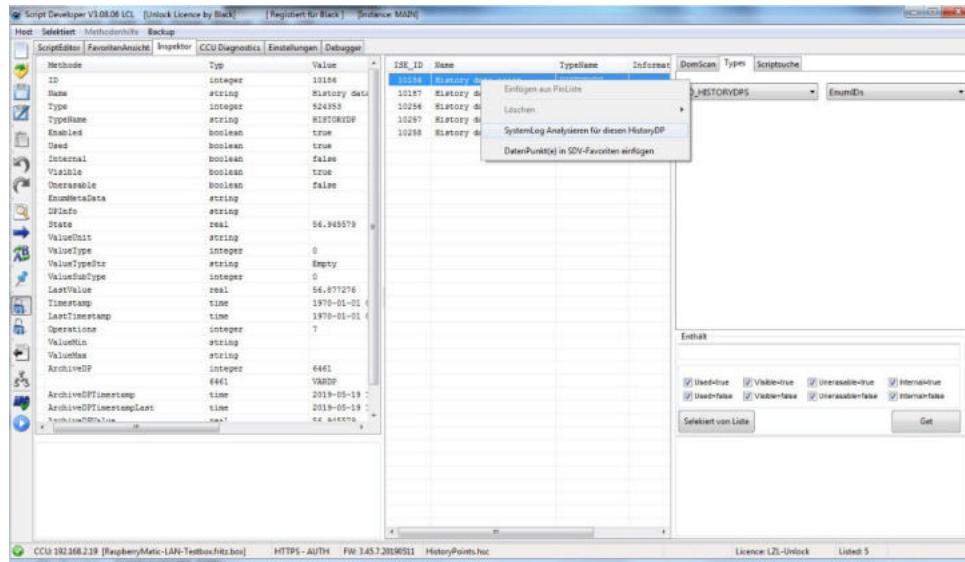




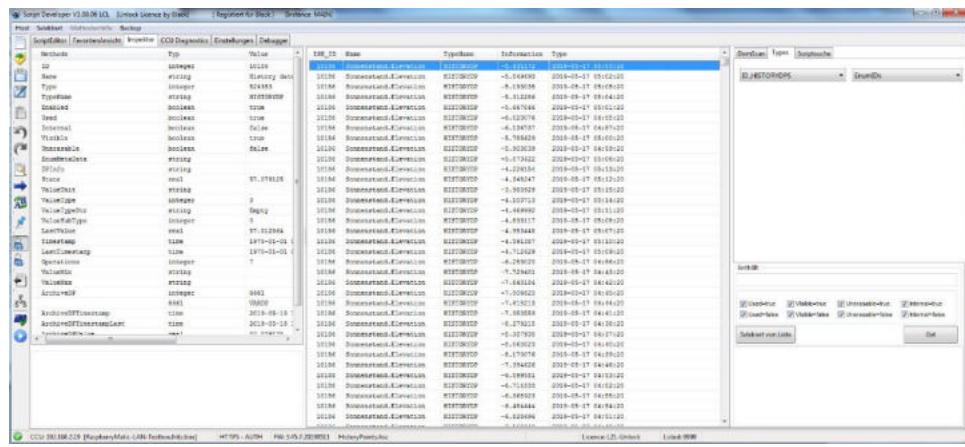
Und hier im WebUI Script

## 4.14 History Data Points und Systemprotokoll

Es besteht die Möglichkeit, History Datapoints im Systemprotokoll zu Selektieren

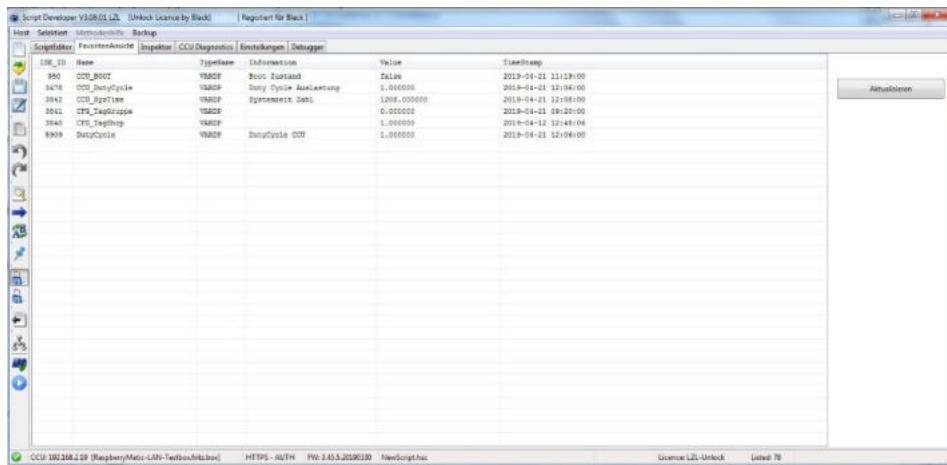


Testweise hat der Punkt hier bei mir fast 10000 Einträge, dauert dann ein paar Sekunden, bis die Werte geholt sind



## 5 Favoritenansicht

Seit der Version V3.08.01 gibt es die Favoritenansicht. Diese soll dazu dienen, einen VariablenPool zusammenzustellen , dessen Werte und Zeitstempel sich übersichtlich darstellen und beobachten lassen.



Aktualisieren führt zum einen Nachladen der aktuellen Werte aus der CCU. In die Favoriten können Systemvariablen, Alarme, Datenpunkte und Programme übernommen werden.

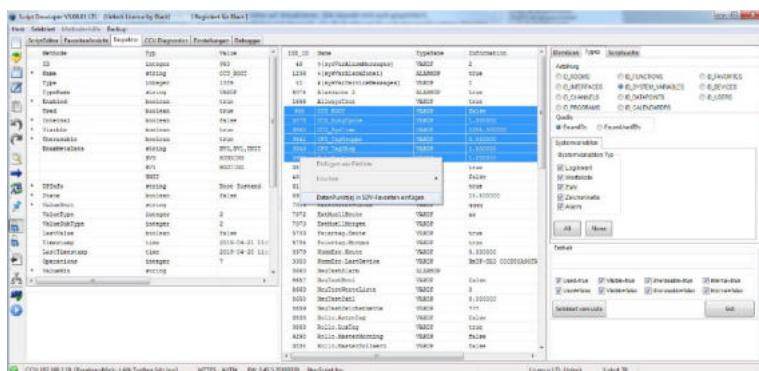
Will man einen Datenpunkt mal eben ändern, einen Doppelklick auf die Zeile der Favoriten, daraufhin wird unter Berücksichtigung des Undo / redo Stacks die Favoriten in die Listenansicht geladen und das angeklickte Objekt zur Bearbeitung in der Detailansicht geöffnet.

Der Inhalt der Favoritenliste wird in der SDV.INI automatisch gespeichert, und steht bei einem Neustart auch wieder zu Verfügung.

Mit Öffnen des Reiters FavoritenAnsicht werden die aktuellen Daten frisch von der CCU geholt

## 5.1 Hinzufügen von Objekten in die Favoriten

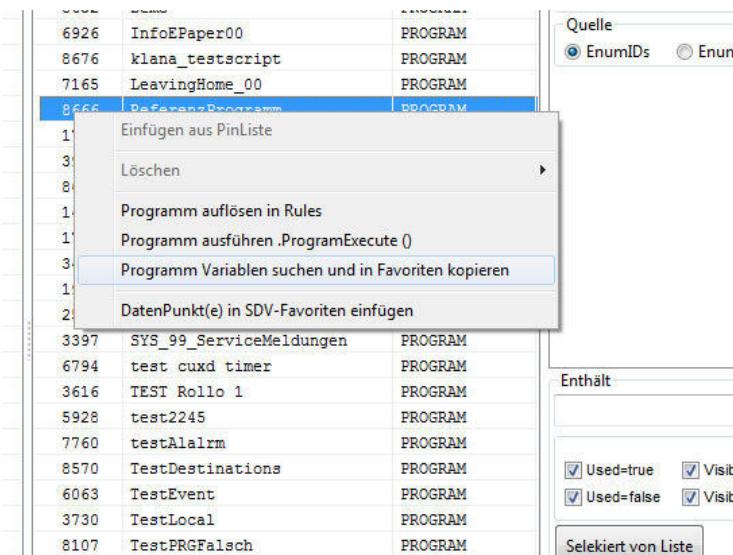
Objekte können aus der Listenansicht des Inspektors in die Favoriten geladen werden. Dazu Objekte Selektieren rechte Maustaste und



Neue Datenpunkte werden immer angehängt. Doppelte IDs werden unterdrückt, zweimal die gleiche Sysvar in den Favoriten geht also nicht.

## 5.2 Verwendete Objekte eines Programmes in die Favoriten laden

Zur Fehlersuche oder zum Testen möchte man öfters alle Werte, die ein Programm beeinflusst oder die von einem Programm beeinflusst werden, im Überblick haben. Dies geht mit dem SDV recht einfach. Dazu in der Listenansicht im Inspektor ein Programm selektieren, rechte Maustaste und „Programm variablen suchen und in Favoriten kopieren“



Darauf löst der SDV das Programmobjekt intern rekursiv auf und schreibt alle gefundenen Systemvariablen und Datenpunkte von Geräten in die FavoritenListe.

## 5.3 Favoritenliste löschen

In die Favoriten reinklicken, rechte Maustaste – Favoritenliste löschen

Es wird natürlich nur die interne Favoritenliste des SDV gelöscht, auf der CCU wird da nicht gelöscht oder verändert.

## 5.4 Einschränkungen

Die Favoritenliste arbeitet IseID orientiert. Wird von der CCU ein Object gelöscht, beispielsweise eine Systemvariable, so wird mit dem nächsten Aktualisieren diese auch von der Favoritenansicht gelöscht, da der ISEID Bezug nun ins Leere greift.

Beim Arbeiten mit 2 CCU's bedeutet das:

Ein Wechsel der CCU als Host im SDV führt immer auch zu einer komplett anderen ISEID Liste. So wird es dann vorkommen, dass die favoritenliste auf einmal leer sein wird, wenn der Host gewechselt wird.

## 6 Kleine Helper im Alltag

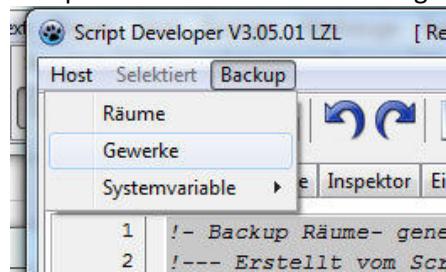
### 6.1 Backups

Von relevanten Objekten können Backups gemacht werden. Diese ersetzen KEIN richtiges SystemBackup an der CCU !!!  
Vielmehr dienen diese im Falle eines Umzuges von einem alt System auf ein Neusystem als Hilfestellung, wenn man das alte Systembackup nicht benutzen will (Loswerden von in den Jahren angesammelten Leichen), oder aber ein inkonsistentes System.  
Ab Version 3.09.03 besteht die Möglichkeit, jeder CCU ein separates Backup Verzeichnis zuzuordnen  
Damit werden Verwechslungen vermieden, wenn man mit 2 CCU arbeitet. (Beide CCUs haben ein jeweils eigenes Backup-Verzeichnis)

z.B.

```
[CCU1]
IP=192.168.2.19
USERNAME=ScriptAdmin
PASSWORD=xxxxxxxxxxxxxxxxxxxx
USEHTTPS=true
SSHUSERNAME=root
SSHPW=xxxxxxxxxxxxxxxxxxxx
BACKUPDIR=C:\MTH\Homematic\Backups\CCU_19\
```

Den passenden Lizenzlevel vorausgesetzt, findet sich die Backups hier:



Devices müssen VORHER manuell umgezogen worden sein über ablernen und neu anlernen.  
Und die Geräte müssen, damit die Backups von Räumen und Gewerken sinnig arbeiten können, wieder ihre „alten“ Namen haben.

Siehe dazu auch die passende EQ3 Dokumentation. Der SDV legt keine neuen Devices oder Direktverbindungen an.

#### 6.1.1 Räume

Auf dem Bestandssystem wird ein Restore Programm erzeugt und lässt sich anschließend auf dem PC abspeichern. Der vorgeschlagene Dateiname ist dabei backup\_Rooms\_ + datum und Uhrzeit der Generierung.

Auf dem „Neusystem“ lässt sich dieses Programm über den SDV via Scripteditor dann starten.  
Dabei passiert folgendes:

Es wird geprüft, ob dort schon ein Raum mit dem Namen „XX“ existiert. Wenn ja, gut, wenn nein, wird dieser Raum neu anlegt, mit Namen und Beschreibung versehen und in ID\_ROOMS eingehängt.

Waren dem alten Raum Kanäle zugeordnet, so versucht der SDV nun diese Kanäle des Altsystems über ihren Kanalnamen zu identifizieren. Ist dieses erfolgreich, so wird dieser Kanal dem Raum hinzugefügt.

### 6.1.2 Gewerke

Auf dem Bestandssystem wird ein Restore Programm erzeugt und lässt sich anschließend auf dem PC abspeichern. Der vorgeschlagene Dateiname ist dabei backup\_Functions\_ + datum und Uhrzeit der Generierung.

Auf dem „Neusystem“ lässt sich dieses Programm über den SDV via Scripteditor dann starten.

Dabei passiert folgendes:

Es wird geprüft, ob dort schon ein Gewerk mit dem Namen „XX“ existiert. Wenn ja, gut, wenn nein, wird dieses Gewerk neu anlegt, mit Namen und Beschreibung versehen und in ID\_FUNCTIONS eingehängt.

Waren dem alten Gewerk Kanäle zugeordnet, so versucht der SDV nun diese Kanäle des Altsystems über ihren Kanalnamen zu identifizieren. Ist dieses erfolgreich, so wird dieser Kanal dem Gewerk hinzugefügt.

### 6.1.3 Systemvariablen

Der komplizierteste Part.

Auf dem Bestandssystem wird ein Restore Programm erzeugt und lässt sich anschließend auf dem PC abspeichern. Der vorgeschlagene Dateiname ist dabei backup\_Sysvars\_ + Datum und Uhrzeit der Generierung.

Auf dem „Neusystem“ lässt sich dieses Programm über den SDV via Scripteditor dann laden.

Hierbei können noch folgende Einstellungen in dem Programm Kopf vorgenommen werden:

```
----- Scriptausgabe -----
!-      Backup SystemVariablen vom 06.12.2018 13:21:02
!-      Erstellt mit Script Developer V3.04 by Black 2018
!----- Diese Zeilen Anpassen -----
boolean bcreate= true; !- Anlegen, wenn noch nicht existierte
boolean bupdate= true; !- Wert Updaten, wenn vorhanden und gleicher Typ
boolean barchive= false; !- false: immer restore mit DPArchive (false), true: restore mit
altem Wert
```

bcreate:

true: wenn die Systemvariable noch nicht existiert wird diese angelegt und in ID\_SYSTEM\_VARIABLES eingehängt.

False: wenn die Systemvariable noch nicht existierte, wird auch nix gemacht.

bupdate:

true: wenn die Systemvariable schon existierte und diese den gleichen Typ hat, wird der State wert aus dem Backup in die variable geschrieben. Wenn nicht der gleiche Typ- passiert nix

false: wenn die Systemvariable schon existiert- wird nix gemacht

barchive: (nur bei Neuanlage)

true: beim Restore wird die Archiv Option der Systemvariable aus dem Backup genommen.

False: es wird immer ohne Archiv Option angelegt beim Restore.

Der SDV unterscheidet dabei von sich aus zwischen Alarm und Systemvariable. Bei Alarm wird nicht der Zustand (AllsArmed) verändert. Heisst: bei Neu Anlage sind die Alarne immer scharf, auch wenn dieser Alarm vorher im Alt System über AIArm (false) unscharf geschaltet wurde !

Zugeordnete Channels werden ebenfalls versucht zu rekonstruieren, so sich der Kanal über den alten Kanalnamen identifizieren lässt (s.a. Räume und Gewerke)

### 6.1.4 Devices und Kanäle

Bei diesem Backup werden die Namen der Kanäle und Geräte gesichert. Die Identifikation erfolgt später über das Interface und die Seriennummer, die der Kanäle durch Durchiterieren und Vergleich mit ChnNumber Methode.

Hilfreich beim Umzug von einem System auf ein anderes System. Nachdem die Geräte abgelernt und am neuen System MANUELL !!!! angelernt wurden, kann das Restore Programm die alten Namen anhand der Seriennummern wiederherstellen. Anschließend können die Raum / Gewerk und Systemvariablen Restore gemacht werden.

### 6.1.5 Backup Programme

Über diesen Punkt kann von einem oder mehreren WEB-UI Programmen Backups gezogen und als Datei (JSON bzw Ausführbares HM-Script) abgespeichert werden. Mit diesen Programmen lässt dich ein WEB-Ui Programm vollständig wiederherstellen. Dabei arbeitet der restore nicht ISEID bezogen, mit dem Restore Programm lässt sich ein WebUI Programm auch auf einem komplett anderen System wiederherstellen, so natürlich die verwendeten Datenpunkte in symbolischer Adressierung und vom richtigen Typ vorhanden sind:

Für die Programme gibt es die Möglichkeit, den Backupnamen in der INI festzulegen.

```
[LAST]
...
BACKUPNAME=$NAME_BACKUP_$FW_$DATE_$TIME
```

Bedeutung der Abkürzungen

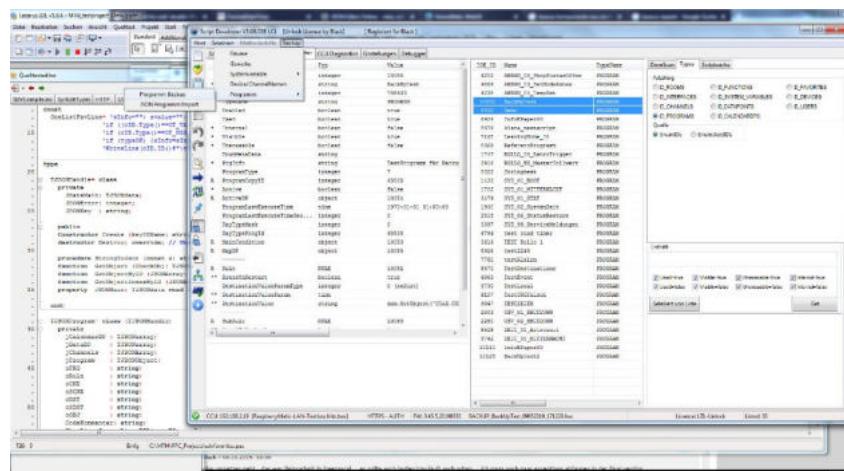
\$NAME: der Name des Programmes wird eingesetzt

\$FW: Firmwareversion

\$DATE: Datum in der Form JJJJMMTT

\$TIME: Zeit in der Form HHMMSS

Programme in der Listenauswahl selektieren, eins oder mehrere...



Danach in dem sich öffnenden Dialog das Verzeichnis Auswählen, wo das Backup hin gespeichert werden soll:

Der Programmname heisst 'BACKUP\_ + ProgrammName + \_ + TTMMYYYY\_HHMMSS.hsc bzw .json danach werden die Backups der ausgewählten Programme angelegt.

werden Objekte selektiert, die keine Programme sind, so werden diese natürlich ignoriert.

So sollte es aussehen, wenn alles durchlief:

boolean	false		8666	ReferenzProgramm
string			1747	ROLLO_00_AstroTrigger
string	TestProgramm für Backup		3916	ROLLO_XX_MasterSollwert
integer	7		8622	Stringtest
integer	65535		1433	SYS_01_BOOT
boolean	false		1702	SYS_01_MITTERNACHT
object	10054		3479	SYS_01_STAT
time				stemZeit
integer				statusRestore
integer				serviceMeldungen
integer				timer
object				o 1
object				m
RULE				nations
boolean	true		6063	TestEvent
integer	0 (sofort)		3730	TestLocal
time			8107	TestPRGFalsch
string	dom.GetObject("CUxD.CU")		9047	TESTZEITM
RULE	10063		2003	USV_01_SHUTDOWN
boolean	false		2280	USV_02_SHUTDOWN
time			8528	ZEIT_00_Astrozeit

ab V8.03.04 wird auch noch die Laufzeit für das Backup in Sekunden angezeigt. In der unteren Infozeile erfolgt Anzeige ganz rechts, welches Programm von wie vielen gerade im Backupprozess ist. Danach liegen im Verzeichnis 2 Files.

das hsc kann z.b. in den SDV geladen werden und dort ausgeführt werden.

Der Kopf sieht immer so aus:

die ersten Zeilen können auch Händisch angepasst werden, wenn man den Weiß, was man tut.

Programmname ist selbsterklärend, die Einträge unter verwendete Kanäle und verwendete Systemvariablen können verändert werden. (beispielsweise Rollo Programm, wenn man einen neuen Aktor benutzt... oder Hm auf HMIP umbaut. es muss nur der alte auf den neuen Kanal geändert werden, das ID zusammenbasteln Gedöns macht dann das Programm. wo was substituiert wurde steht auch als Test im Programm. Hier sieht man es besser, Der SDV trägt in die Kommentarzeilen unter der Tabelle ein, welcher Datenpunkt wie über Kanal - HSSID zusammengesetzt wurde.

Zum Restore.

Es wird überprüft, ob das Programm schon existiert... natürlich Abbruch.  
auch ob alle verwendeten Datenpunkte (Sysvars Kanäle oder DPs) vorhanden sind und auch den  
richtigen Typ haben. Originale Sysvar BOOL und auf dem neuen System sysvar String erkennt der SDV  
und bricht vor der Generierung ab.

Bei einem Erfolgreichen Durchlauf sieht die Scriptausgabe so aus:

```
Backup erstellt vom SDV V3.08.03B ICL am 09.05.2019 19:43:41
CCU ProgrammName: "BackUpTest"
CCU ProgrammInfo: "TestProgramm für Backup Restore"
-----
Test auf Existenz referenzierter Channels
Test auf Existenz und Typkonsistenz referenzierter Systemvariablen
    OK.... SYSVAR TP1
    OK.... SYSVAR TestVar
Test auf Existenz und Typkonsistenz referenzierter Geräte-Datenpunkte
Restore Programm von Program "BackUpTest" erfolgreich durchgelaufen
----- Script Variablen -----
```

Im Hinblick z.B auf Gerätetausch... Gerät selektieren rechte Maustaste und dann WEBUI Programme von diesem Gerät.

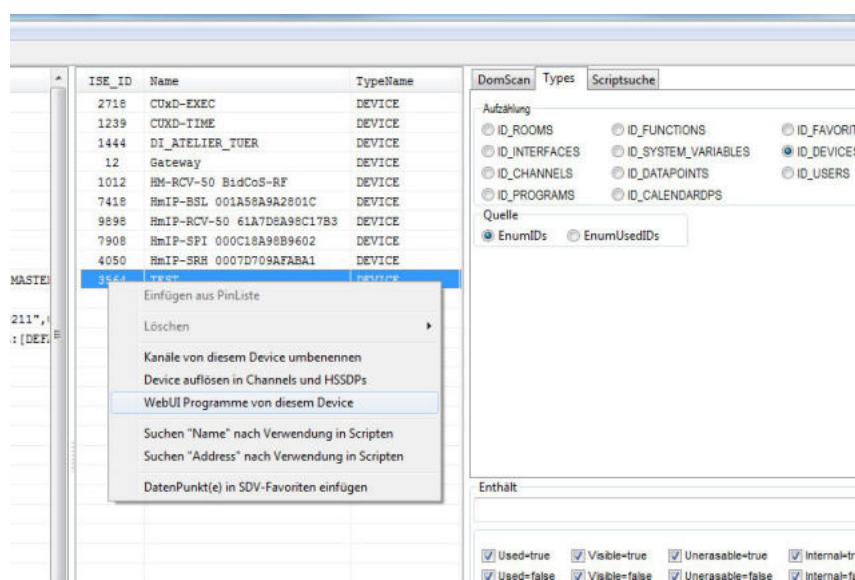
Funktion ist unspektakulär kann die WebUI auch (habt nur programmiert, weil ein Device keine idarray hat, wo die verwendeten Programme drinstehen)

Diese gefundenen Programme kann man nun selektieren, und backup machen.

Im Editor bearbeitet lassen sich nun z.b. die Kanalbezeichner ändern (auch HM auf HMIP z.B.).

dann die gefundenen Programme löschen und die neuen Restoren.. Gerätetausch von HM-nach HMIP (Das neue Gerät mit den Kanälen MUSS natürlich vorhanden sein und die Datenpunkte auch den gleichen Namen (z.B. level) und Typ haben..)

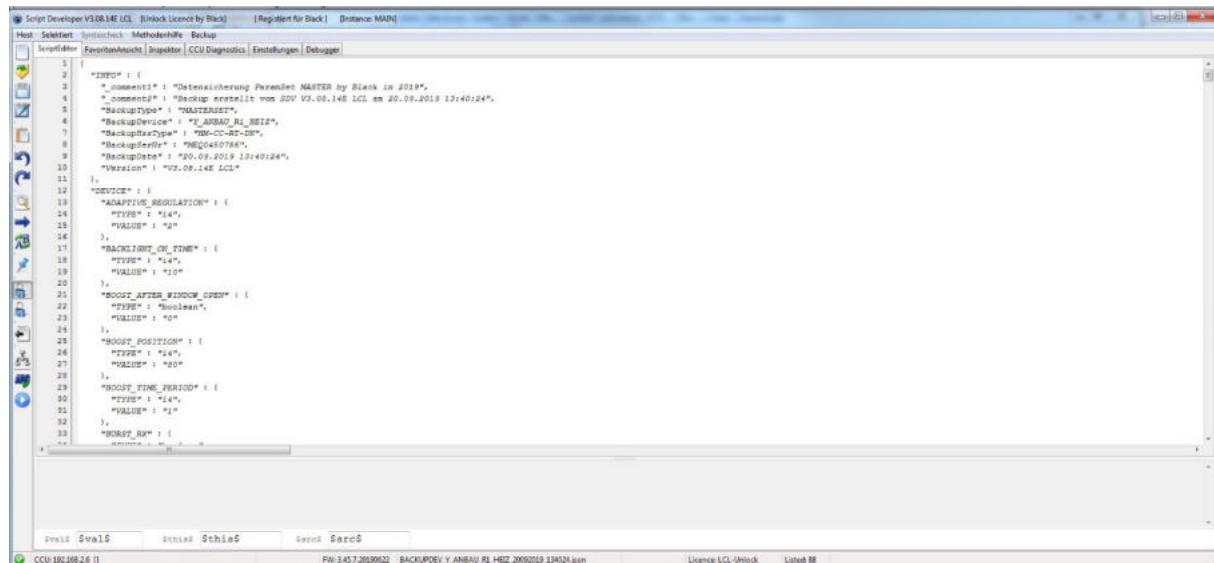
natürlich müssen die neuen Kanal Bezeichner richtig eingegeben worden sein, sonst gibt das Shit in - Shit out Prinzip



## 6.1.6 Backup Masterset

Es ist möglich, von einem Gerät (oder von mehreren Selektierten) ein Backup zu erstellen. Dazu werden das oder die Geräte Selektiert, dann Backup – Paramset Master von Device

Dabei wird von allen selektierten Geräten eine JSON Datei von den Geräteeinstellungen angelegt. Dieses JSON kann in den Editor geladen und dann ausgeführt werden.



The screenshot shows the Script-Developer V3.09 LCL software interface. The main window displays a JSON configuration file for a backup master set. The JSON code is as follows:

```
1 "JSON": {
2   "_comment": "Datensicherung Parameter MASTER by Black im 2019",
3   "_comment": "Backup erstellt vom SDD V3.08.1#6 LCL am 20.09.2019 13:40:24",
4   "BackupType": "MasterSet",
5   "BackupName": "TE_MASTER_SL_MST2",
6   "BackupFileType": "MEQ0450788",
7   "BackupUpdate": "20.09.2019 13:40:24",
8   "Version": "V3.08.1#6 LCL"
9 },
10 "DEUTOP": {
11   "ADAPTIVE_REGULATION": {
12     "TYPE": "1#",
13     "VALUE": "0"
14   },
15   "BACKLIGHT_ON_TIME": {
16     "TYPE": "1#",
17     "VALUE": "10"
18   },
19   "BOOST_AFTER_WINDOW_OPEN": {
20     "TYPE": "Boolean",
21     "VALUE": "0"
22   },
23   "BOOST_POSITION": {
24     "TYPE": "1#",
25     "VALUE": "20"
26   },
27   "BOOST_TIME_PERIOD": {
28     "TYPE": "1#",
29     "VALUE": "1"
30   },
31   "BRIGHT_RK": {
32     "comment": "BRIGHT_RK"
33   }
34 }
```

At the bottom of the interface, there is a toolbar with icons for File, Open, Save, Print, and Help. Below the toolbar, a status bar shows the path "CCU:1021682.6 []", the file name "FW:3.09.7.2019022 BACKUPDEV\_Y\_ANBAU\_RS\_H02\_20092019.134024.json", and the license information "Licence: LCL-Unlocked".

Dabei werden die Einstellungen des JSON an das Gerät übertragen. Wenn man weiß, was man tut, kann man einzelne Werte auch manuell verändern.

## 6.1.7 Backup Linkset

Es ist möglich, von einem Gerät (oder von mehreren Selektierten) ein Backup der zu dem Gerät definierten Deirektverknüpfungen zu erstellen. Dazu werden das oder die Geräte Selektiert, dann Backup – Paramset Link von Device

Dabei wird von allen selektierten Geräten eine JSON Datei von den zu den Geräten definierten Direktverknüpfungen angelegt. Dieses JSON kann in den Editor geladen und dann ausgeführt werden. Dabei werden die Direktverknüpfungen angelegt über AddLink und anschließend die Werte der Expertenparameter, so vorhanden, geladen

Script Developer V3.09.01-SP2 ICL [Unlock Licence by Black] [Registert für Black] [Instance: MAIN]

Host Selektiert Syntaxcheck Methodenhilfe Backup

ScriptEditor FavoritenAnsicht Inspektor CCU Diagnostics Einstellungen Debugger

```

63 // 3. Direktverknüpfung
64
65 WriteLine ("3. Direktverknüpfung");
66 WriteLine ("-----");
67 string stdErr=""; string stdOut="";
68 string rDevName="HmIP-WTH-2 000A9709A0DA37:4";
69 string rDevHssType="HEATING_SHUTTER_CONTACT_RECEIVER";
70 WriteLine ("Channel Empfänger: " # rDevName);
71 WriteLine ("Channel Empfänger: " # rDevHssType);
72 string sDevName="HmIP-SRH 000D709AFABA1:1";
73 string sDevHssType="ROTARY_HANDLE_TRANSCEIVER";
74 WriteLine ("Channel Sender : " # sDevName);
75 WriteLine ("Channel Sender : " # sDevHssType);
76 object rChan=channels.Get (rDevName); object sChan=channels.Get (sDevName);
77 if (!rChan) (WriteLine ("ERROR - Empfängerkanal # " + rDevName + " gibt es nicht! --> Abbruch"); quit);
78 if (!sChan) (WriteLine ("ERROR - Senderkanal # " + sDevName + " gibt es nicht! --> Abbruch"); quit);
79 if (rChan.HssType () != rDevHssType) (WriteLine ("ERROR - Empfängerkanal # " + rDevName + " hat falschen HssType () --> Abbruch"); quit);
80 if (sChan.HssType () != sDevHssType) (WriteLine ("ERROR - Senderkanal # " + sDevName + " hat falschen HssType () ! --> Abbruch"); quit);
81 object oIFace=dom.GetObject (rChan.Interface ());
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
379
380
381
382
383
384
385
386
387
388
389
389
390
391
392
393
394
395
396
397
398
399
399
400
401
402
403
404
405
406
407
408
409
409
410
411
412
413
414
415
416
417
418
419
419
420
421
422
423
424
425
426
427
428
429
429
430
431
432
433
434
435
436
437
438
439
439
440
441
442
443
444
445
446
447
448
449
449
450
451
452
453
454
455
456
457
458
459
459
460
461
462
463
464
465
466
467
468
469
469
470
471
472
473
474
475
476
477
478
479
479
480
481
482
483
484
485
486
487
488
489
489
490
491
492
493
494
495
496
497
498
499
499
500
500
501
502
503
504
505
506
507
508
509
509
510
511
512
513
514
515
516
517
518
519
519
520
521
522
523
524
525
526
527
528
529
529
530
531
532
533
534
535
536
537
538
539
539
540
541
542
543
544
545
546
547
548
549
549
550
551
552
553
554
555
556
557
558
559
559
560
561
562
563
564
565
566
567
568
569
569
570
571
572
573
574
575
576
577
578
579
579
580
581
582
583
584
585
586
587
588
589
589
590
591
592
593
594
595
596
597
598
599
599
600
601
602
603
604
605
606
607
608
609
609
610
611
612
613
614
615
616
617
618
619
619
620
621
622
623
624
625
626
627
628
629
629
630
631
632
633
634
635
636
637
638
639
639
640
641
642
643
644
645
646
647
648
649
649
650
651
652
653
654
655
656
657
658
659
659
660
661
662
663
664
665
666
667
668
669
669
670
671
672
673
674
675
676
677
678
679
679
680
681
682
683
684
685
686
687
688
689
689
690
691
692
693
694
695
696
697
698
699
699
700
701
702
703
704
705
706
707
708
709
709
710
711
712
713
714
715
716
717
717
718
719
720
721
722
723
724
725
726
727
728
729
729
730
731
732
733
734
735
736
737
738
739
739
740
741
742
743
744
745
746
747
748
749
749
750
751
752
753
754
755
756
757
758
759
759
760
761
762
763
764
765
766
767
768
769
769
770
771
772
773
774
775
776
777
778
779
779
780
781
782
783
784
785
786
787
788
789
789
790
791
792
793
794
795
796
797
797
798
799
799
800
801
802
803
804
805
806
807
808
809
809
810
811
812
813
814
815
816
816
817
818
819
819
820
821
822
823
824
825
826
827
828
829
829
830
831
832
833
834
835
836
837
838
839
839
840
841
842
843
844
845
846
847
848
849
849
850
851
852
853
854
855
856
857
858
859
859
860
861
862
863
864
865
866
867
868
869
869
870
871
872
873
874
875
876
877
878
879
879
880
881
882
883
884
885
886
887
888
889
889
890
891
892
893
894
895
896
897
897
898
899
899
900
901
902
903
904
905
906
907
908
909
909
910
911
912
913
914
915
916
917
917
918
919
920
921
922
923
924
925
926
927
928
929
929
930
931
932
933
934
935
936
937
938
939
939
940
941
942
943
944
945
946
947
948
949
949
950
951
952
953
954
955
956
957
958
959
959
960
961
962
963
964
965
966
967
967
968
969
969
970
971
972
973
974
975
976
977
978
979
979
980
981
982
983
984
985
986
987
987
988
989
989
990
991
992
993
994
995
996
997
997
998
999
999
1000
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1009
1010
1011
1012
1013
1014
1015
1016
1016
1017
1018
1019
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1089
1090
1091
1092
1093
1094
1095
1096
1097
1097
1098
1099
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1188
1189
1189
1190
1191
1192
1193
1194
1195
1196
1197
1197
1198
1199
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1209
1210
1211
1212
1213
1214
1215
1216
1217
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1288
1289
1289
1290
1291
1292
1293
1294
1295
1296
1297
1297
1298
1299
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1309
1310
1311
1312
1313
1314
1315
1316
1317
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1348
1349
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1388
1389
1389
1390
1391
1392
1393
1394
1395
1396
1397
1397
1398
1399
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1409
1410
1411
1412
1413
1414
1415
1416
1417
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1438
1439
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1448
1449
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1488
1489
1489
1490
1491
1492
1493
1494
1495
1496
1497
1497
1498
1499
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1509
1510
1511
1512
1513
1514
1515
1516
1517
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1538
1539
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1548
1549
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1588
1589
1589
1590
1591
1592
1593
1594
1595
1596
1597
1597
1598
1599
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1609
1610
1611
1612
1613
1614
1615
1616
1617
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1638
1639
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1648
1649
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1688
1689
1689
1690
1691
1692
1693
1694
1695
1696
1697
1697
1698
1699
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1709
1710
1711
1712
1713
1714
1715
1716
1717
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1738
1739
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1748
1749
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1788
1789
1789
1790
1791
1792
1793
1794
1795
1796
1797
1797
1798
1799
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1809
1810
1811
1812
1813
1814
1815
1816
1817
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1838
1839
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1848
1849
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1878
1879
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1888
1889
1889
1890
1891
1892
1893
1894
1895
1896
1897
1897
1898
1899
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1909
1910
1911
1912
1913
1914
1915
1916
1917
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1938
1939
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1948
1949
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1978
1979
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1988
1989
1989
1990
1991
1992
1993
1994
1995
1996
1997
1997
1998
1999
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2038
2039
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2048
2049
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2078
2079
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2088
2089
2089
2090
2091
2092
2093
2094
2095
2096
2097
2097
2098
2099
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2109
2110
2111
2112
2113
2114
2115
2116
2117
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2138
2139
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2148
2149
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2178
2179
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2188
2189
2189
2190
2191
2192
2193
2194
2195
2196
2197
2197
2198
2199
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2209
2210
2211
2212
2213
2214
2215
2216
2217
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2238
2239
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2248
2249
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2278
2279
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2288
2289
2289
2290
2291
2292
2293
2294
2295
2296
2297
2297
2298
2299
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2309
2310
2311
2312
2313
2314
2315
2316
2317
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2338
2339
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2348
2349
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2378
2379
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2388
2389
2389
2390
2
```

## 6.1.8 Backup Gerät von der WebUI

Beim Entfernen eines Gerätes werden ja alle WebUI Bezüge getilgt. Nach dem Neuanlernen sind ja alle Einträge in Räume, Gewerke, Favoriten und sehr ärgerlich, in den Programmen weg.  
Mit dem neuen Menüpunkt besteht die Möglichkeit, die WebUI Einträge, die zu dem Gerät gehörten wieder vollständig zu rekonstruieren.

Master und Linksets werden hierbei nicht angepackt, dazu gibt's die separaten Backupmöglichkeiten.

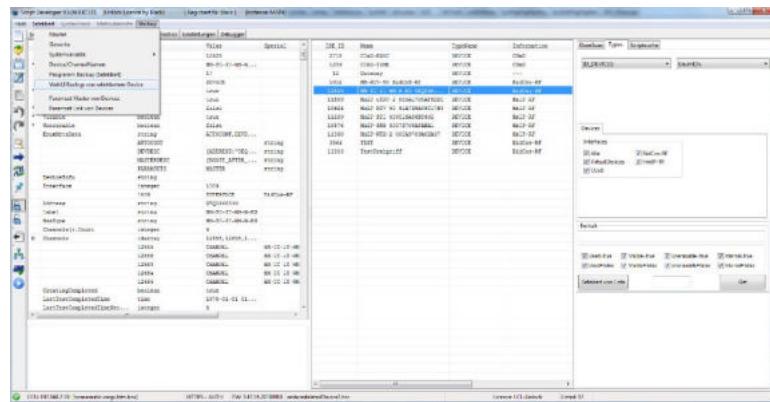
Es ist hierbei sogar möglich, der WebUI ein Baugleiches IP Gerät unterzuschieben und zu rekonstruieren, dafür muss nach dem anlernen des anderen Gerätes nur der Gerätename auf den alten Namen geändert werden.

Ansonsten ist der Identifiziermechanismus

Seriенnummer, wenn nicht gefunden --> Gerätename, wenn auch nicht gefunden Error

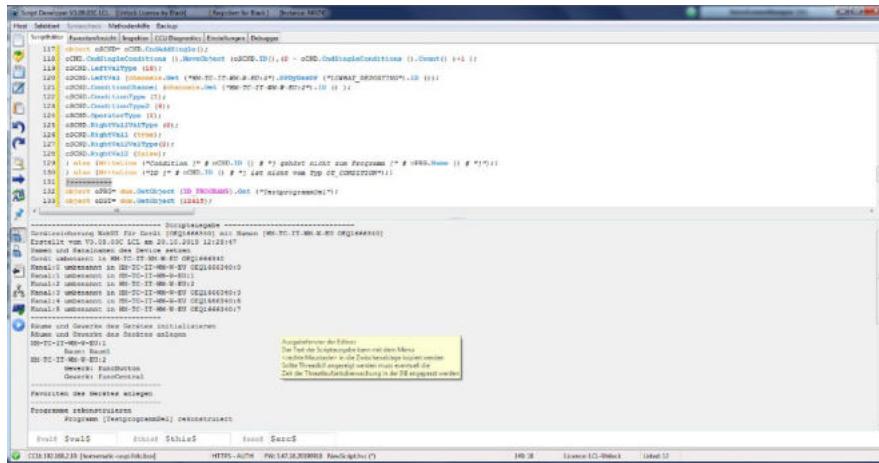
Vorgehensweise:

Gerät selektieren



Im folgenden Dialog das Backup Verzeichnis festlegen. Nach Durchführung liegt das generierte Restoreprogramm als Datei vor und liegt ebenfalls in der Windows Zwischenablage. Nach dem Löschen des Gerätes sieht ein Programm, welches das Gerät beinhaltete, ein wenig zerrupft aus:

Das generierte Programm kann nun auf der CCU unter Script ausführen oder im SDV in den Scripteditor geladen und dort ausgeführt werden  
Die Statusausgabe verrät auch was alles angepasst und geändert wurde



- Das Gerät wird anhand seiner Seriennummer identifiziert, wenn das fehlschlägt, anhand des Gerätenamens
- Der Gerätename und die Namen aller Kanäle des Gerätes werden auf den Namen aus der Sicherung geändert.
- Alle Räume und Gewerke, denen es beim Anlernen eventuell zugewiesen wurde werden entfernt
- die Kanäle werden allen Räumen und Gewerken gemäß der Sicherung zugeordnet.
- In den Favoriten werden die Kanäle des Geräte wieder richtig (an die ursprüngliche Stelle) eingefügt
- In allen Programmen, in denen das Gerät bzw Datenpunkte der Kanäle des Geräte verwendet wurden und wo diese Bezüge (SingleConditions/SingleDestinations) des Gerätes beim Löschen des Gerätes auch gelöscht wurden, werden diese auch wieder an der richtigen Stelle rekonstruiert. (Zum Einsatz kommen hier auch Mechanismen aus dem schon lange laufenden Programme-Backup)

Auch das Programm schaut anschließend wieder so aus wie vor dem Löschen des Gerätes

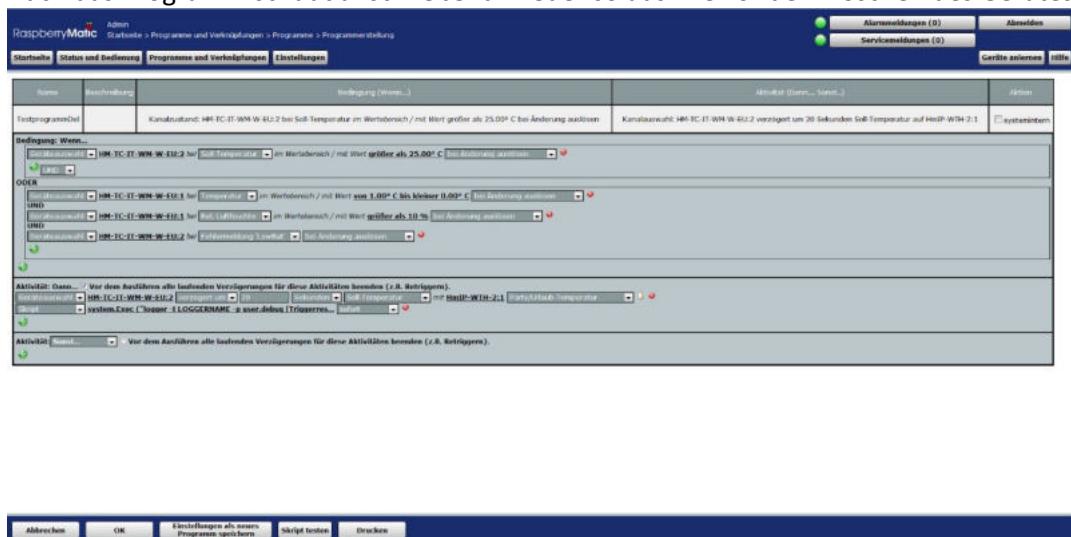


Diagramme kann er nicht rekonstruieren und wird er auch nicht können, da ich die Diagrammfunktion der CCU gelinde gesagt selber suboptimal finde, ich diese selber nicht nutze (bei mir läuft IOBroker), ich mir also auch die Zeit schenke, dafür etwas zu programmieren.

## 6.2 Umbenennen von Kanälen von Geräten

Wer hatte nicht schon alles die Freude, z.B. an einem neu angelernten IP Gerät mit 14 Kanälen die Namen neu zu vergeben. Dies geht nun schneller.

Das Device wird selektiert und der Name der Device geändert.

Anschliessend rechte maustaste auf das Device in der Listendarstellung und Punkt auswählen:

ISE_ID	Name	TypeName	Information
2718	CUxD-EXEC	DEVICE	CUxD
1239	CUXD-TIME	DEVICE	CUxD
1444	DI_ATELIER_TUER	DEVICE	BidCos-RF
12	Gateway	DEVICE	---
3564	HM-LC-Sw2-FM LEQ1319211	DEVICE	BidCos-RF
1012	HM-RCV-50 BidCoS-RF	DEVICE	BidCos-RF
74	Einfügen aus PinListe		HmIP-RF
76	Löschen		HmIP-RF
75			HmIP-RF
40	Kanäle von diesem Device umbenennen		HmIP-RF

Rückfrage mit Ja bestätigen und die Kanäle werden so benannt:

Device: DeviceName

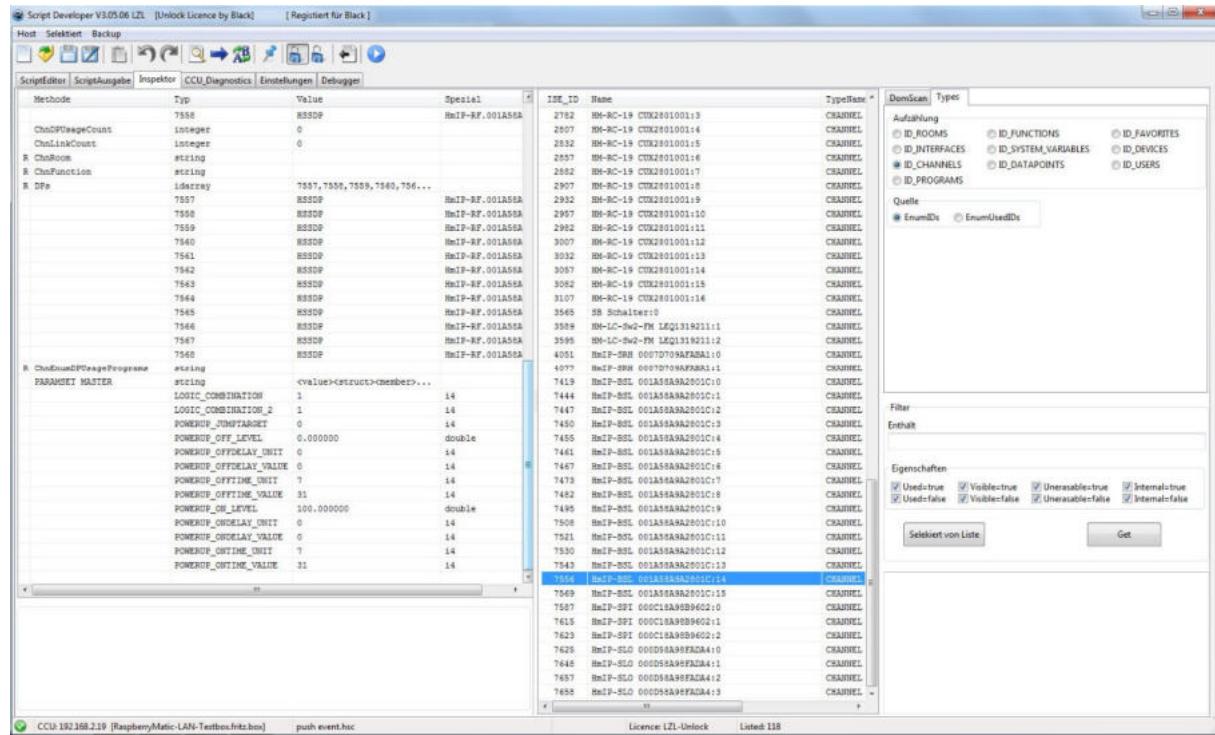
Kanal0 : DeviceName:0

Kanal1 : Devicename:1

Etc...

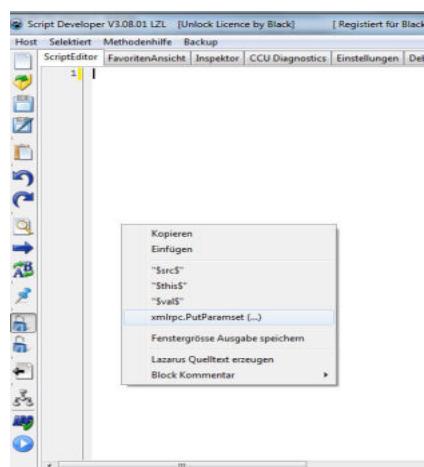
## 6.3 Paramset Master

Bei Device, Kanälen, die einen Paramset Master haben, wird dieser mit angezeigt (wenn in der Sicht konfiguriert und mind. Level 6)

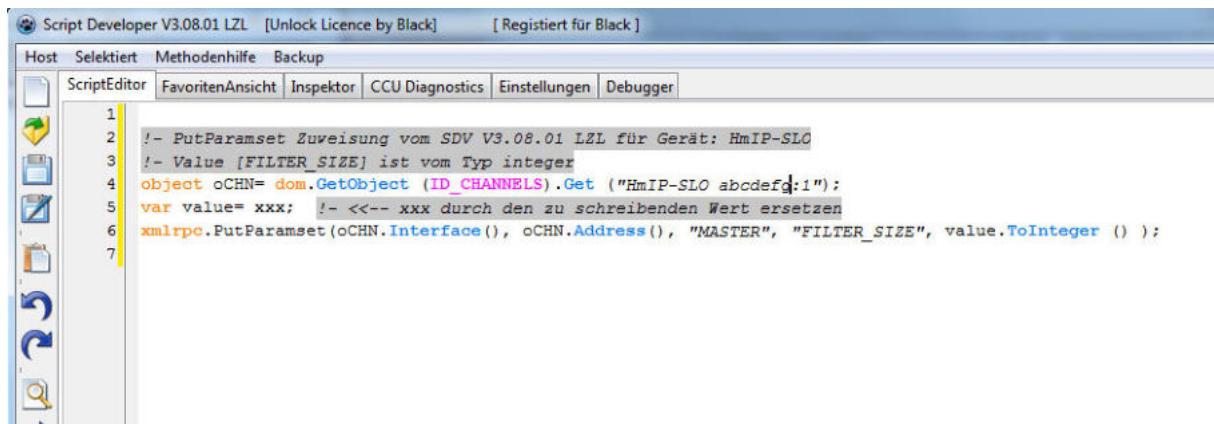


Die Werte lassen sich dann über PutParamset verändern in einem Script

Ab Version V3.08.01. Einzelne Parameter des Mastersatzes lassen sich nun auch markieren in der Detailansicht. Im Editor lässt sich nun Automatisiert der Code für die Manipulation dieses Masterparameters erzeugen.



Dies erzeugt nun automatisiert den richtigen Code, in Abhängigkeit von: welches Gerät ist es (HMIP, HM, CUXD o.ä) und dem erwarteten Datentyp.

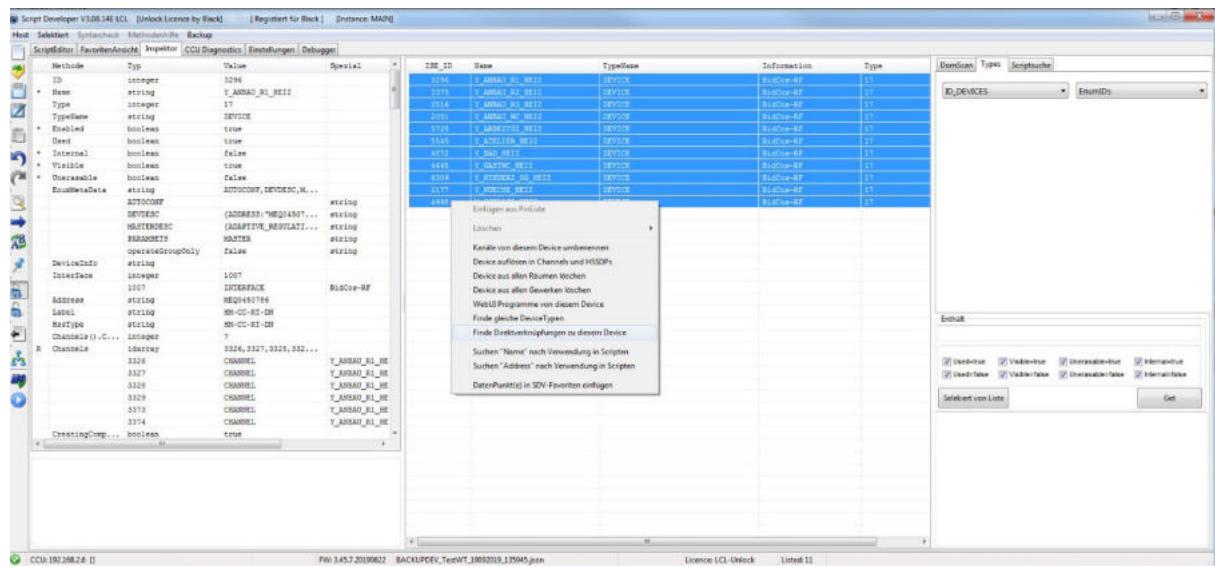


The screenshot shows the interface of Script Developer V3.08.01 LZL. The title bar reads "Script Developer V3.08.01 LZL [Unlock Licence by Black] [Registert für Black]". The menu bar includes "Host", "Selektiert", "Methodenhilfe", and "Backup". Below the menu is a toolbar with icons for file operations like Open, Save, and Print. The main area is a "ScriptEditor" tab, which contains the following code:

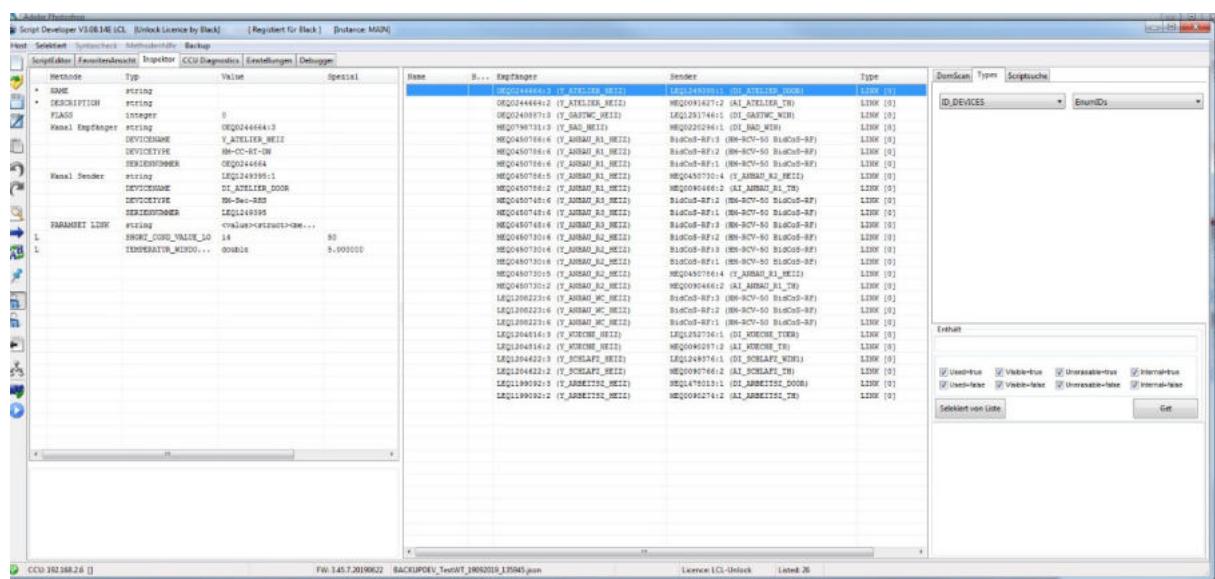
```
1  !- PutParamset Zuweisung vom SDV V3.08.01 LZL für Gerät: HmIP-SLO
2  !- Value [FILTER_SIZE] ist vom Typ integer
3  object oCHN= dom.GetObject (ID_CHANNELS).Get ("HmIP-SLO abcdefg:1");
4  var value= xxxx; !- <<- xxxx durch den zu schreibenden Wert ersetzen
5  xmirpc.PutParamset(oCHN.Interface(), oCHN.Address(), "MASTER", "FILTER_SIZE", value.ToInteger () );
```

## 6.4 Linkset eines Gerätes

Ist ein Device markiert, so liegt auf der rechten Maustaste in der Listenübersicht der Menüpunkt „Finde Direktverknüpfungen zu diesem Device“



Die markierte Liste wird abgearbeitet, ist das markierte Element ein Device, so wird dazu versucht, die zugehörigen Linksets aufzulösen.

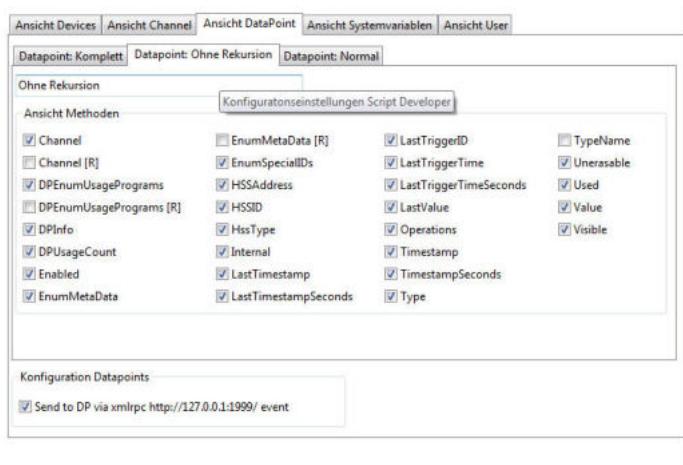


Die Linksets können angewählt werden, Sortiert werden nach Name, Beschreibung, Empfänger und Sender. Gleichzeitig werden die Expertenparameter, so vorhanden unter Paramset Link dargestellt.

Diese können auch markiert werden und im Editor kann für diese Parameter automatisch der Code zum Beschreiben generiert werden.

## 6.5 Rega Push auf Datenpunkte via Rega event

Rega Push: Damit lassen sich Datenpunkte innerhalb der Rega verändern, die sich normalerweise nicht verändern lassen. Z.B. Batteriefehler eines Netzaktors ^^. Setzt voraus, dass dieser Haken im der Konfiguration Datenpunkte gesetzt wurde:



Des weiteren braucht es in der Sicht Freigabe auf TypeName (SDV prüft auf HSSDP) und anklicken von Value. (Level 7 braucht es dafür auch)

Hat der DP die Eigenschaft Write, wird ein ganz normales State (xxx) ausgeführt. Hat er diese nicht, macht der SDV bei Wertänderung von sich aus ein Rega Event auf den DP.

(Wenn die Freigabebedingungen passen)

## 6.6 Querverweise zu Datenpunkten in WebUI-Programmen

Es ist möglich, zu HSSDP, VarDP und AlarmDPs selektiv die Verwendung in WEBUI Programmen zu suchen.

The screenshot shows the 'Script Developer V3.09.00A LCL' interface. A context menu is open over a data point in the main list, with the option 'Querverweise zu Datenpunkten' selected. This opens a new window titled 'Querverweise zu Datenpunkten'. Inside, there's a table of data points and a search bar at the top right. Below the table, there are several checkboxes for selecting system variables: 'Logonwert', 'Name', 'true', 'Zeilenzusta...', and 'Alarm'. At the bottom of the dialog, there are 'Add' and 'Close' buttons.

Als Trigger: der Datenpunkt muss im Bedingungsteil als „bei Änderung“ oder „bei Aktualisierung“ verwendet werden.

Nur prüfen: der Datenpunkt muss im Bedingungsteil als „nur prüfen“ verwendet werden

Als Egal wie in Bedingungen: der Datenpunkt muss im Bedingungsteil vorkommen

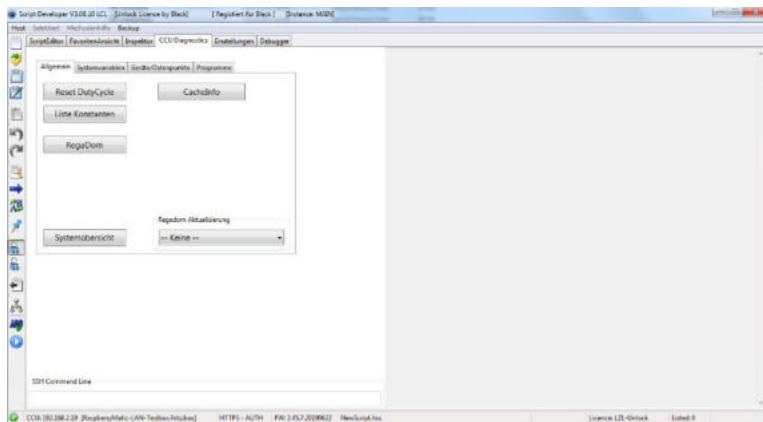
Im Zuweisungsteil: Der Datenpunkt wird im Dann oder Sonst Teil verwendet

Egal wo: Der Datenpunkt muss nur irgendwo im WebUI Programm vorkommen

Als Ergebnis erscheint in der Listenauswahl die Programme, wo der selektierte Datenpunkt unter der angewählten Bedingung vorkommt.

The screenshot shows the 'Script Developer V3.09.00A LCL' interface with the search results for 'SYS\_01\_BOOT'. The results table has columns: ISE\_ID, Name, TypeName, and Information. Two entries are listed: '8632 Demo PROGRAM DemoTest' and '1433 SYS\_01\_BOOT PROGRAM CCU im Boot'. To the right of the results table, there are two buttons: 'DomScan' and 'Types'. Below the results table, there is a section labeled 'Systemvariablen'.

## 7 Diagnosebild



### 7.1 Allgemein

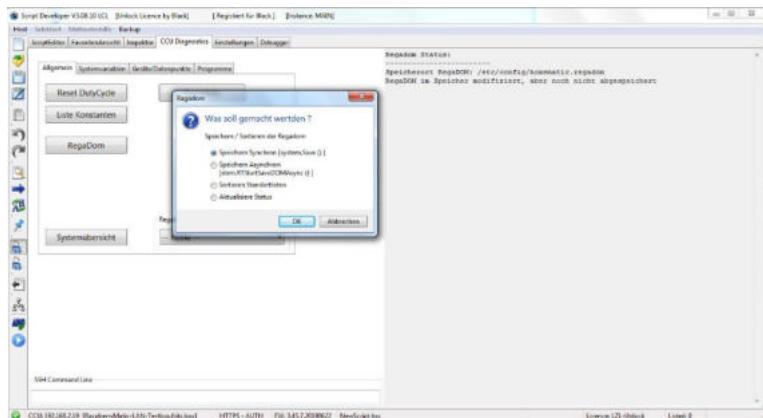
Reiter für Allgemeine Einstellungen

#### 7.1.1 Liste Konstanten

Alle Systemkonstanten der Rega werden dargestellt

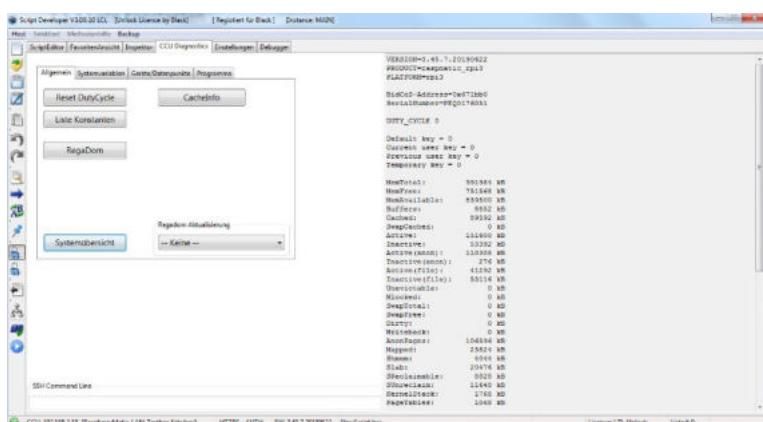
#### 7.1.2 RegaDom

Status der Rega abrufen und Speichern anstoßen



#### 7.1.3 Systemübersicht

Allgemeine Information über die CCU

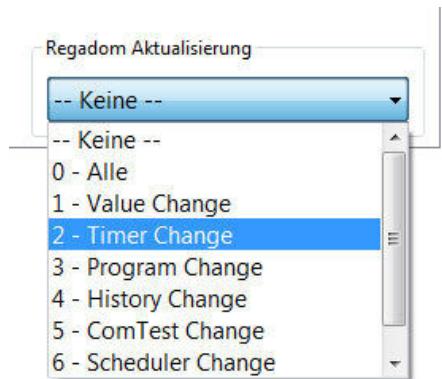


#### 7.1.4 CacheInfo

Darstellung des CacheInhaltes der CCU

#### 7.1.5 Regdom Aktualisierung

Anstossen von dom.RTUpdate (). Auswahl für die ComboBox



Hier auch Schnellstatus Programme mit gesetzter Copy ID. Ist nach Neustart zentrale immer noch der Wert <>0... Geisterprogramm

## 7.1 SSH Funktionalität

Wenn auf der CCU / Raspberrymatik SSH freigegeben ist, ist es möglich, ab Level 5 eine rudimentäre SSH Funktionalität zu nutzen.

Was ist dazu nötig ?

In der SDV Ini müssen die nötigen Schlüssel eingegeben sein.

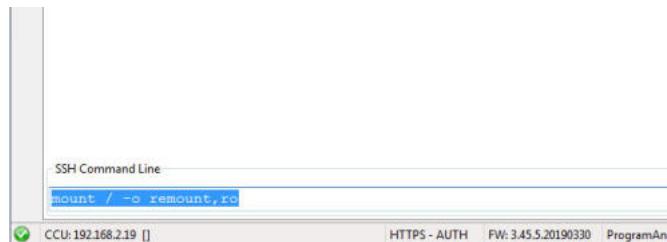
```
[CCU1]
IP=192.168.2.xx
USERNAME=xxxx
PASSWORD=yyyy
USEHTTPS=true
SSHUSERNAME=root      Bestandsnutzer: diese schlüssel hinzufügen
SSHPW=sshpasword

[CCU2]
IP=192.168.2.xx
USERNAME=
PASSWORD=
USEHTTPS=false
SSHUSERNAME=root
SSHPW=

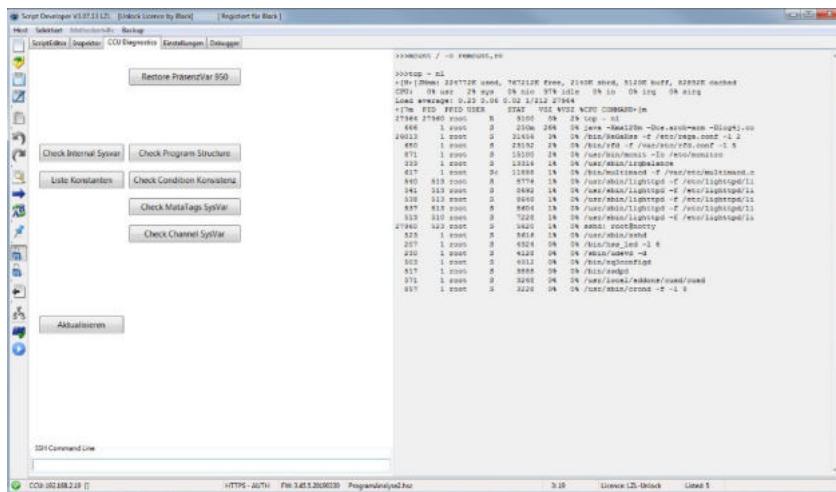
[HOSTCCU]
IP=192.168.2.xx
USERNAME=
PASSWORD=
USEHTTPS=False
SSHUSERNAME=root
SSHPW=
```

Ab der 3.08.xx kommt der SDV wieder ohne das PLink aus, hier wird dies nun über CUxD realisiert.

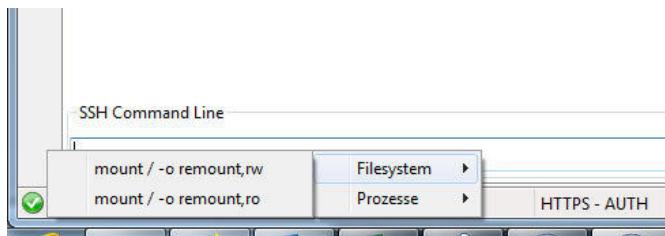
In dieser Zeile lassen sich nun einfache SSH Kommandos ausführen.



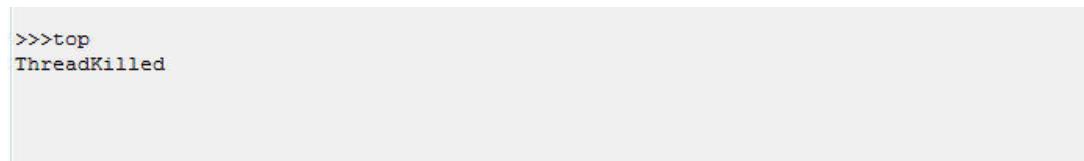
Dies soll und ist aber kein Ersatz für eine putty Konsole, eher gedacht für ich muss mal eben schnell einen Befehl ausführen oder man hat nicht die Befehlsfolgen im Kopf sondern benutzt die Hilfe. Programme dort zu starten oder z.b. top zusehen geht nicht und ist auch nicht geplant, das kann putty ganz klar besser. zweimal Rad erfinden ist auch nicht nötig.



Einige Funktionen sind als Hilfe mit rechter Maustaste auf die Kommandozeile programmiert, wird bei Bedarf von mir noch erweitert, Vorschläge dazu sind auch willkommen.



Programme dort zu starten oder z.b. top zusehen geht nicht und ist auch nicht geplant, das kann putty ganz klar besser. zweimal Rad erfinden ist auch nicht nötig. Also eine Ausführung von top rennt nach 10 Sekunden in den Thread Timeout Kill.



### 7.1.1 SSH Realisierung (CUxD ab 2.3.1 oder plink.exe)

INI Schlüssel SSHLINK=false

bei false versucht er den Zugriff über die CUxD Seite. Wenn der Zugriff nicht Freigeschaltet ist über

USERACCESS=1+

USERLOGIN=

Wichtig ist das + !!!!!!!

rennt er dabei gegen Access denied.

Das Kleingedruckte: USERACCESS=1+ ist Sicherheit wie alte Version, also keine.

Wer das nicht möchte, kann den damals beschriebenen alten Weg gehen, putty suite runterladen, plink exe ins SDV Verzeichnis kopieren, cmd unter Windows öffnen und einmal im SDV Verzeichnis ausführen:

```
plink -ssh root@IPderCCU -pw SSHPASSWORT top - n1
```

Wenn es der erste Aufruf war, die Frage nach dem Zertifikat bejahen. danach geht der Aufruf auch headless mit dem SDV.

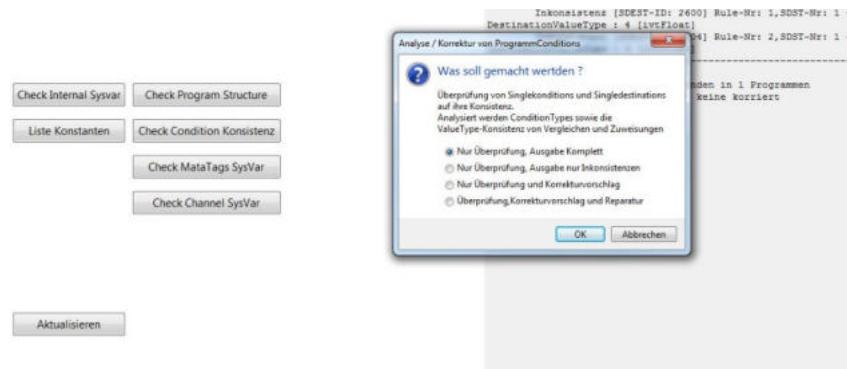
damit ist aber wieder Windows im Spiel.

Wer weder CuxD den Access setzen möchte noch plink benutzen will, der kann den SDV nutzen bis auf die SSH Funktionalität.

## 7.2 Programm Condition Konsistenz.

Durch häufiges Ändern, oder auch internen Fehlern in der WEB-UI kann es vorkommen, dass eigentlich in sich richtig ausschauende Programme in Ihren Objektstrukturen logische Fehler aufweisen und nicht mehr triggern oder ansonsten unlogisches Verhalten aufweisen. Der bekannteste Fehler z.B. ist ein Programm mit einem Stringvergleich welches aber reproduzierbar nicht triggern wird, weil die WebUi aus dem bei Strings eigentlich vorgesehenem ConditionType 1 (bei) eine 6 (ist kleiner bis gleich) macht.

Als Vorstufe zu der Erweiterung Program backups kann der SDV nun mir bekannte logische Inkonsistenzen erkennen, aufzeigen, und wenn möglich, einen Korrekturvorschlag nennen und diesen auf Anweisung auch ausführen.



Auswahl ist selbsterklärend.

Stufe 4 (mit Reparatur sollte immer erst nach Stufe 3 (Überprüfung mit Korrekturvorschlag) benutzt werden. Hier werden die Fehler aufgeführt und ein Lösungsvorschlag aufgezeigt. Wenn dieser schlüssig ist, gut, kann man den SDV drauf loslassen, wenns arg schräg aussieht kann oder muss man die Korrektur manuell machen.

```
ProgrammAnalyse Konsistenz SingleConditions & Singledestinations
Black in April 2019
-----
Analyse Program: Alarmtest [ID: 8528]
    Inkonsistenz [SCND-ID: 8567] Rule-Nr: 1,CND-Nr: 1, SCND-Nr: 1 -- LeftVal: 2 [ivtBinary], RightVal: 4 [ivtFloat]
        Ursache Boolvergleich ivtBinary Links mit nicht ivtFloat Wert rechts
        Umrechnen von rechts Wert 15.000000 [ivtFloat] in true [ivtBinary]
        Kann vom SDV korrigiert werden
-----
Analyse Program: Stringtest [ID: 8622]
    Inkonsistenz [SCND-ID: 9258] Rule-Nr: 1,CND-Nr: 1, SCND-Nr: 1 -- LeftVal: 20 [ivtString], RightVal: 16 [ivtInteger]
        Ursache Stringvergleich ivtString Links mit nicht ivtInteger Wert rechts
        Umrechnen von rechts Wert 25 [ivtInteger] in 25 [ivtString]
        Kann vom SDV korrigiert werden
-----
Analyse Program: USV_01_SHUTDOWN [ID: 2003]
    Inkonsistenz [SDEST-ID: 2600] Rule-Nr: 1,SDST-Nr: 1 -- DestinationDP : 2 [ivtBinary], DestinationValueType : 4 [ivtFloat]
    Inkonsistenz [SDEST-ID: 2604] Rule-Nr: 2,SDST-Nr: 1 -- DestinationDP : 2 [ivtBinary], DestinationValueType : 4 [ivtFloat]
-----
Geprüft: 37 Programme
Dabei wurden 4 Inkonsistenzen gefunden in 3 Programmen
Von diesen 4 Inkonsistenzen wurden keine korrigiert
```

In diesem Beispiel werden gefundene Inkonsistenzen aufgezeigt, die Ursache dargelegt und die Lösung und ob der SDV diese, wenn er mit Reparatur gestartet wird, diese korrigieren kann.

Wenn ja, wird im einem Korrekturlauf diese Inkonsistenzen beseitigt.

Der SDV verändert dabei Programmobjekte..  
AUF JEDEN FALL VORHER EIN BACKUP MACHEN !

Es gibt kein UNDO, das geht nur über Restore

```
ProgrammAnalyse Konsistenz SingleConditions & Singledestinations
Black in April 2019

Analyse Program: Alarmtest [ID: 8528]
Inkonsistenz [SCND-ID: 8567] Rule-Nr: 1,CND-Nr: 1 -- LeftVal: 2 [ivtBinary], RightVal: 4 [ivtFloat]
Ursache Boolvergleich ivtBinary Links mit nicht ivtFloat Wert rechts
Umrechnen von rechts Wert 15.000000 [ivtFloat] in true [ivtBinary]
Kann vom SDV korrigiert werden
Korrigiert durch RightValValType (ivtBinary)

-----
Analyse Program: Stringtest [ID: 8622]
Inkonsistenz [SCND-ID: 9258] Rule-Nr: 1,CND-Nr: 1 -- LeftVal: 20 [ivtString], RightVal: 16 [ivtInteger]
Ursache Stringvergleich ivtString Links mit nicht ivtInteger Wert rechts
Umrechnen von rechts Wert 25 [ivtInteger] in 25 [ivtString]
Kann vom SDV korrigiert werden
Korrigiert durch RightValValType (ivtString)

-----
Analyse Program: USV_01_SHUTDOWN [ID: 2003]
Inkonsistenz [SDEST-ID: 2600] Rule-Nr: 1,SDST-Nr: 1 -- DestinationDP : 2 [ivtBinary], DestinationValueType : 4 [ivtFloat]
Inkonsistenz [SDEST-ID: 2604] Rule-Nr: 2,SDST-Nr: 1 -- DestinationDP : 2 [ivtBinary], DestinationValueType : 4 [ivtFloat]

-----
Geprüft: 37 Programme
Dabei wurden 4 Inkonsistenzen gefunden in 3 Programmen
Von diesen 4 Inkonsistenzen wurden 2 korrigiert
```

Ein nochmaliger Lauf zeigt diese 2 korrigierbaren nun nicht mehr als inkonsistent an.

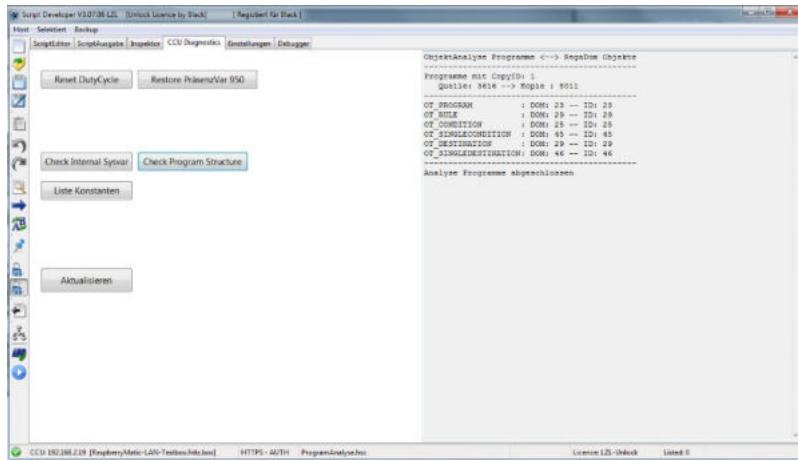
```
ProgrammAnalyse Konsistenz SingleConditions & Singledestinations
Black in April 2019

Analyse Program: USV_01_SHUTDOWN [ID: 2003]
Inkonsistenz [SDEST-ID: 2600] Rule-Nr: 1,SDST-Nr: 1 -- DestinationDP : 2 [ivtBinary], DestinationValueType : 4 [ivtFloat]
Inkonsistenz [SDEST-ID: 2604] Rule-Nr: 2,SDST-Nr: 1 -- DestinationDP : 2 [ivtBinary], DestinationValueType : 4 [ivtFloat]

-----
Geprüft: 37 Programme
Dabei wurden 2 Inkonsistenzen gefunden in 1 Programmen
Von diesen 2 Inkonsistenzen wurden keine korrigiert
```

## 7.3 Programmanalyse

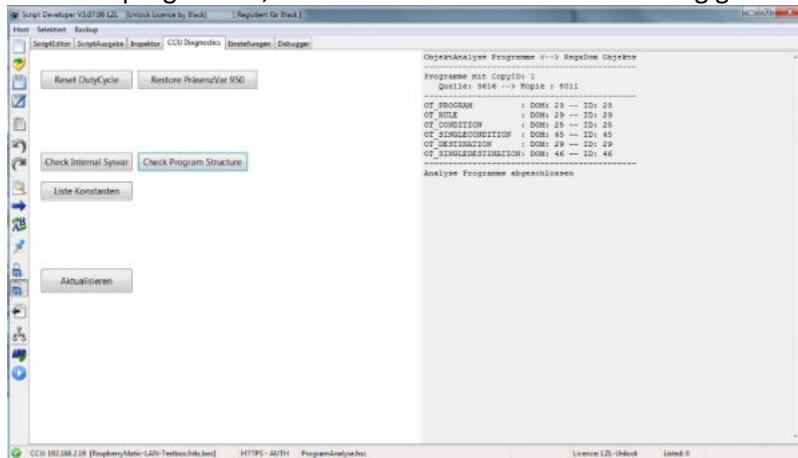
Mittels der Programmanalyse kann die Konsistenz der Programmobjekte überprüft werden. So sieht der Aufruf auf, wenn die Prüfroutine erfolgreich über die Programme gelaufen ist.



Keine Einträge in ProgrammCopyID, die weiteren Tests ergeben eine Übereinstimmung zwischen dem DomScan und der Analyser der Programme

### Mögliche Fehlerbilder:

#### 1. Geisterprogramme, Leichen welche beim Editieren übrig geblieben sind.



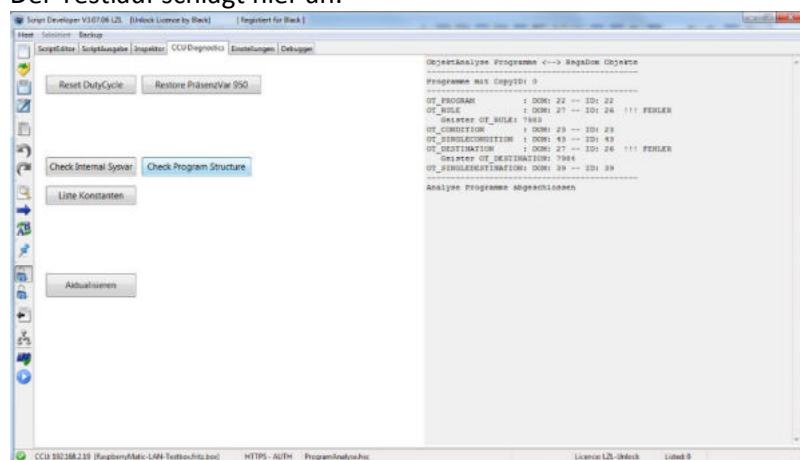
Hier wird das ursprüngliche Programm und die (Geisterkopie) daraus angezeigt. Der erste Versuch wäre: einen Reboot zu versuchen und zu überprüfen, ist der Eintrag des Geisterprogrammes weg oder nicht. Wenn nicht, kann manuell versucht werden, die Objekte der Kopie (nicht des Quellprogrammes) zu löschen

#### 2. Programmfragmentreste

In der Regadom tummeln sich Reste ehemaliger Programme, welche aber nicht mehr in der Struktur unter ID\_PROGRAMS auftauchen.

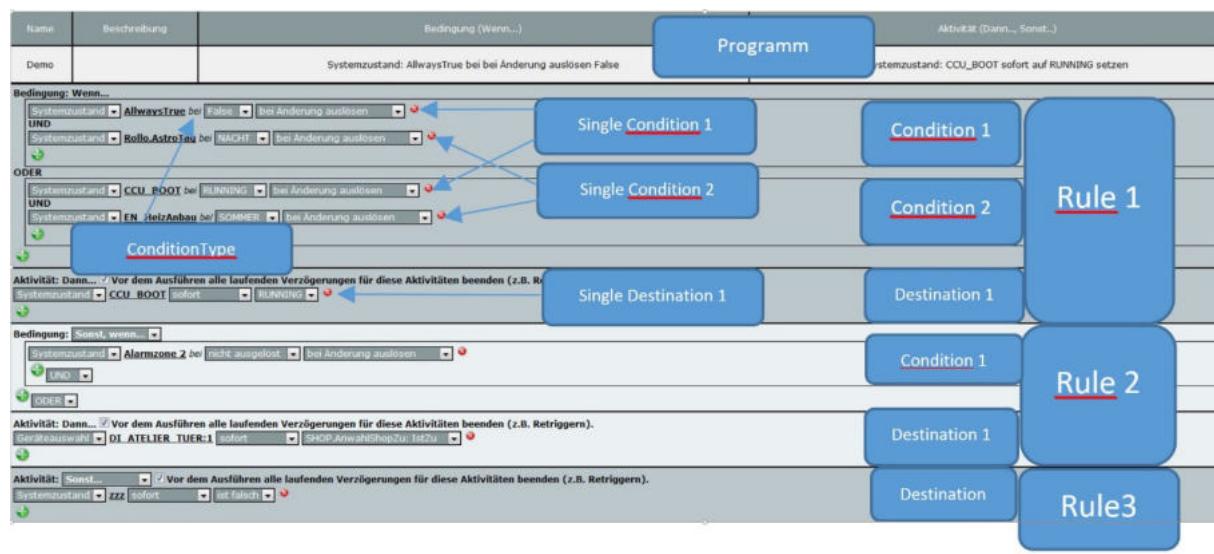
Provoziert habe ich es hier in dem Testfall, in dem ich ein RULE Object mal händisch angelegt habe.

Der Testlauf schlägt hier an:

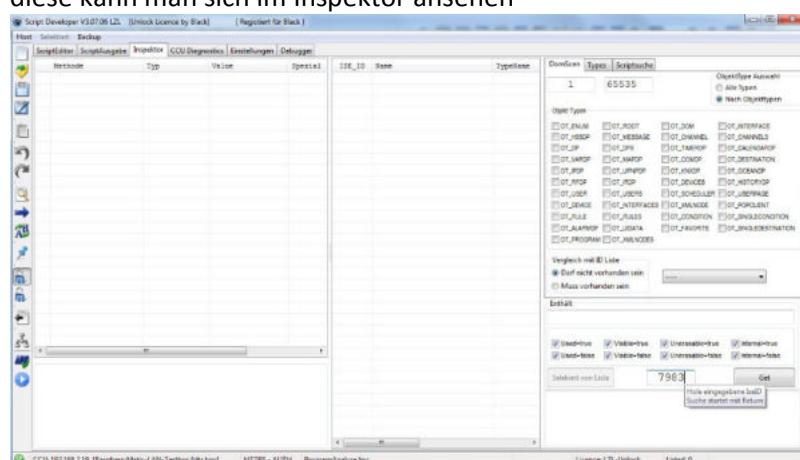


Es wird unter den Objecten OT\_RULE und OT\_DESTINATION angezeigt, dass sich in der Rega 2 Objekte tummeln, welche keinen Bezug zu dem Inhalt eines Programmobjektes haben. (2 deshalb, weil ein dom.CreateObject (OT\_RULE) auch gleichzeitig das zu der Rule gehörende OT\_DESTINATION Object erzeugt)

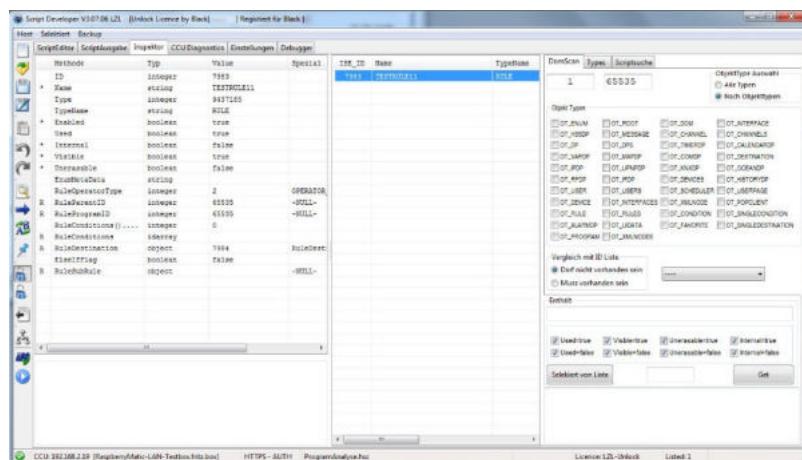
Ein Programm ist auf der CCU so Aufgebaut:



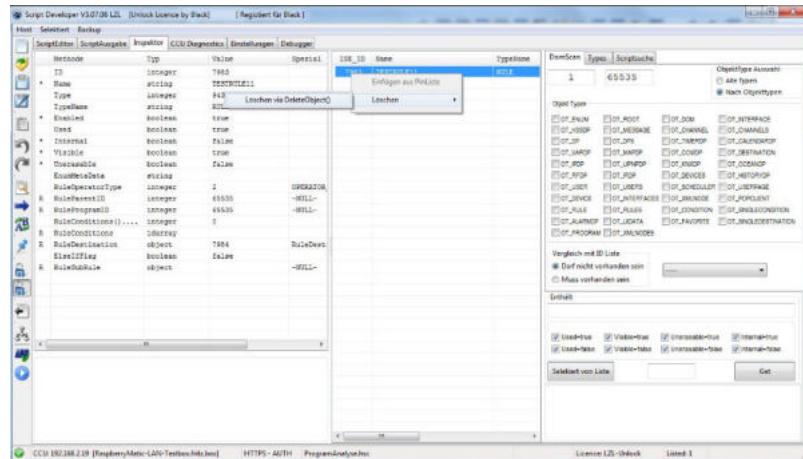
Die isID des gefundenen Objektes wird ausgegeben  
diese kann man sich im Inspektor ansehen



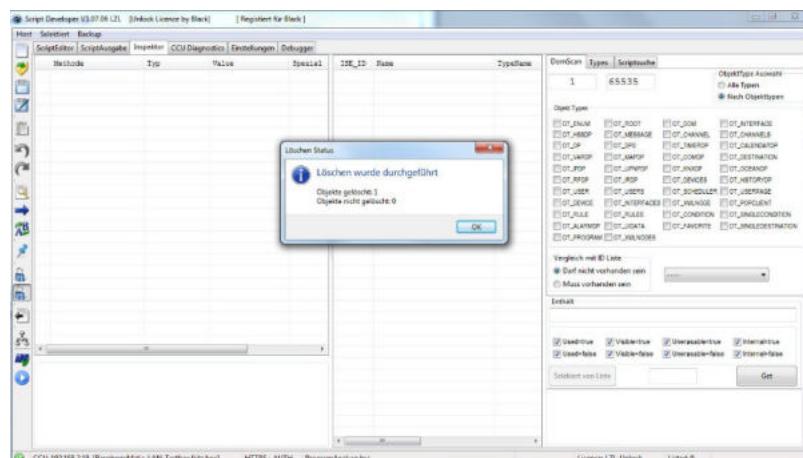
Dazu die gefundene ID in die Suchmaske eintragen und Enter drücken



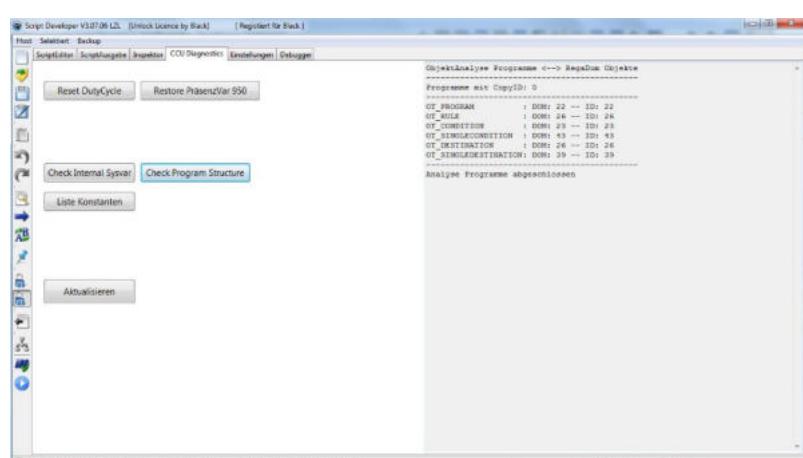
Man sieht, ein Object ohne Bezüge. es kann also entfernt werden. Löschen Freigeben, Programmunterobjekte müssen auch freigeben sein zum löschen, dann



es wurde gekillt



Die abschliessende Kontrolle zeigt:



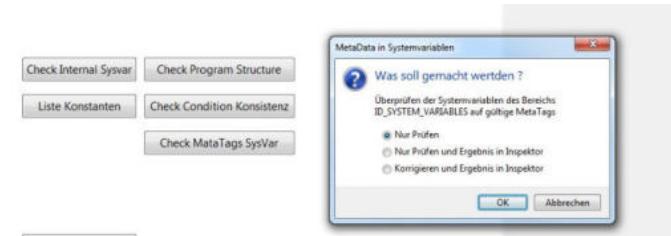
Es ist weg.

**Und immer die Erinnerung: vor Löschen immer ein Backup machen, bei löschen gibt es kein Undo.**

## 7.4 Systemvariablen Metadatenkonsistenz

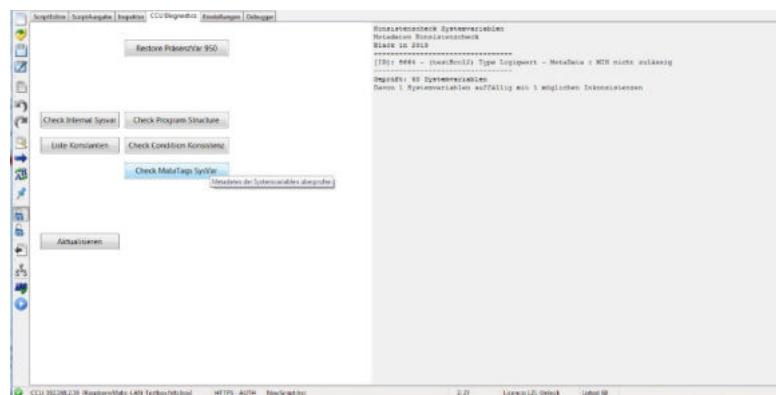
Dieser Menüpunkt überprüft die Systemdaten auf Konsistenz von dem Typ, welcher durch ValueType und ValueSubtype spezifiziert ist und den Einträgen unter EnumMetadata.

Druck erzeugt folgendes Menü:



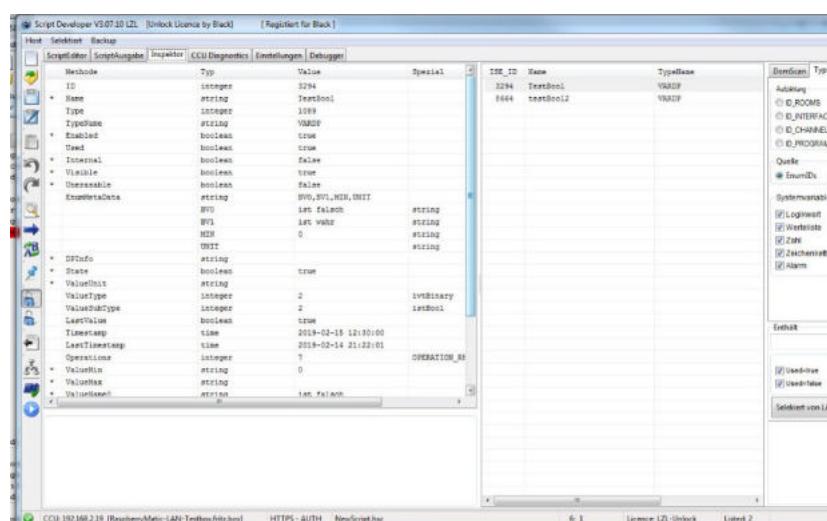
### Punkt 1: nur Prüfen

Bringt eine Ausgabe in der Form, wie das Script, welches ich im Forum Online gestellt hatte. Mehr nicht. Er tut selber nix.



### Punkt 2: Nur Prüfen und Ergebnis in Explorer

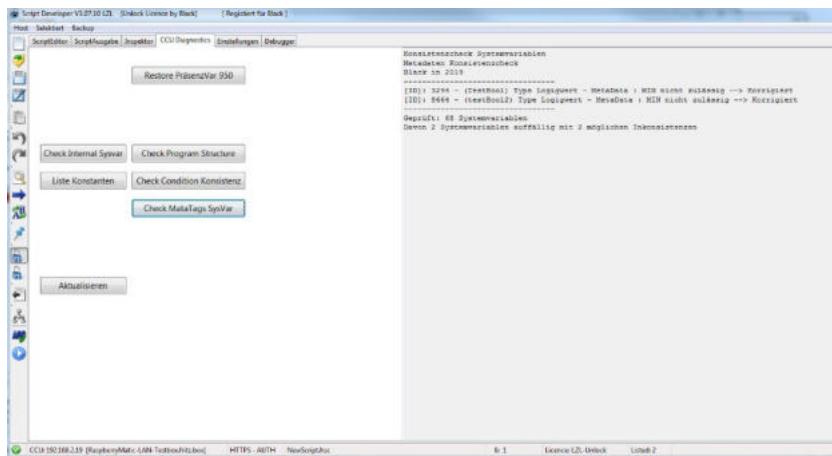
Genau wie Punkt 1, es wird eine Überprüfung gemacht, zusätzlich wird das Ergebnis noch in die Auswahlliste des Inspektors geladen. Dort kann überprüft und gegenbenenfalls auch manuell korrigiert werden.



### Punkt 3: Korrigieren und Ergebnis in den Inspektor

Hier wird automatisch korrigiert und das Ergebnis dann in den Inspektor geladen.

VORHER AUF JEDENFALL EIN BACKUP MACHEN. Es gibt kein Undo hierbei.

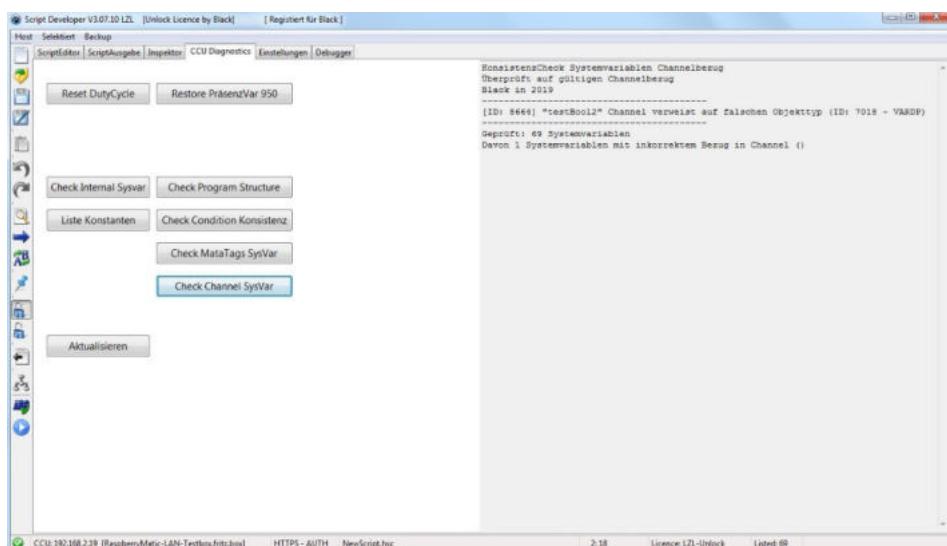
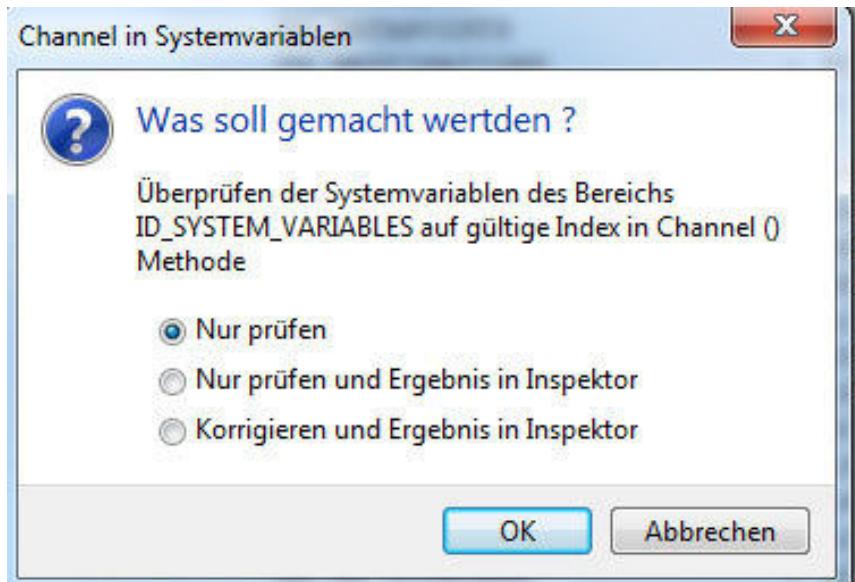


Ein nochmaliger Durchlauf bringt dann die Fehlerfrei-Nachricht.

```
Konsistenzcheck Systemvariablen
Metadaten Konsistenzcheck
Black in 2019
-----
-----
Geprüft: 68 Systemvariablen
Keine Auffälligkeiten festgestellt
```

## 7.5 Systemvariablen Test auf NirvanaChannelBezüge

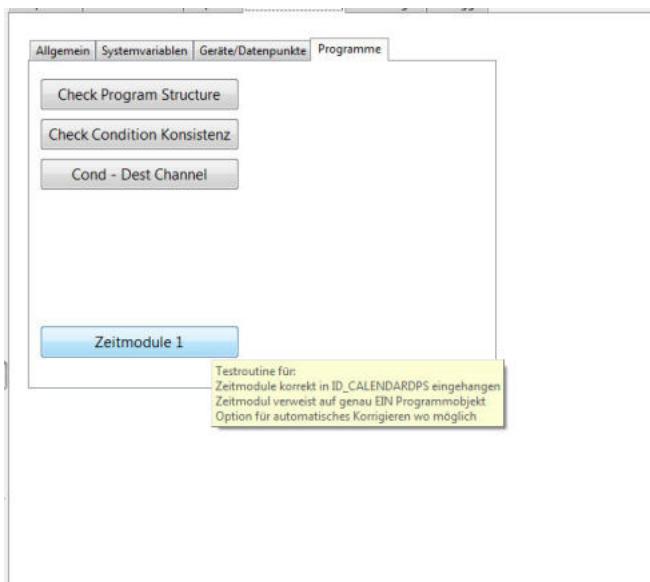
Es wird überprüft, ob die Methode Channel () entweder Leer ist (ID\_ERROR) oder auf einen gültiges Channel Objekt verweist. Wenn nicht, handelt es sich um einen NirvanaChannelBezug.  
Hier auch: Auswahl was gemacht werden soll



Die Auswahl Korrigieren bereinigt dann diese automatisch und lädt die veränderten Systemvariablen in den Inspektor.

```
KonsistenzCheck Systemvariablen Channelbezug
Überprüft auf gültigen Channelbezug
Black in 2019
-----
[ID: 8664] "testBool2" Channel verweist auf falschen Objekttyp (ID: 7018 - VARDP) --> korrigiert
-----
Geprüft: 69 Systemvariablen
Davon 1 Systemvariablen mit inkorrekttem Bezug in Channel ()
```

## 7.6 Test und Korrektur Zeitmodule



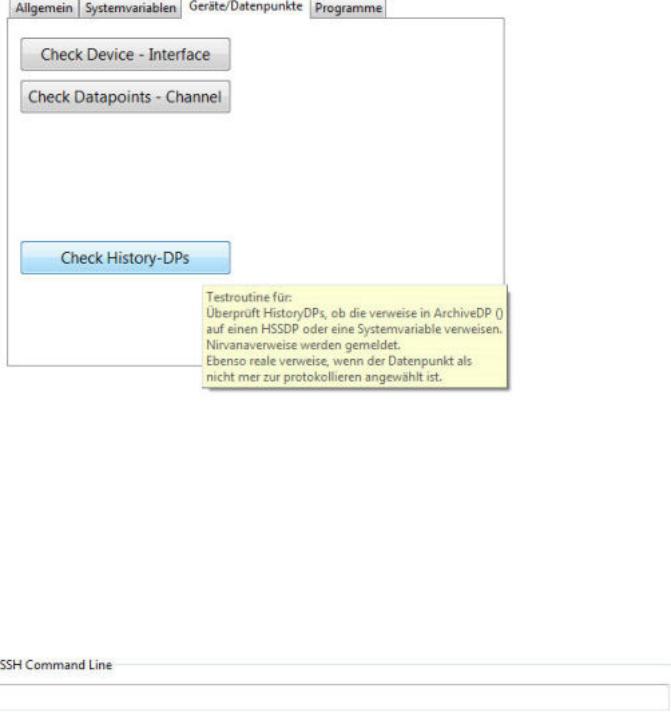
Seit Version 3.08.10 ist das Menü CCU Diagnostics in Reitern strukturiert. Unter Programme findet sich nun die Testoption Zeitmodule 1 . Hierbei wird geprüft auf korrektes Einhängen des Zeitmodules in ID\_CALENDAR DPS, ob jedem Zeitmodul genau ein Programmobjekt zugeordnet ist. Korrektur kann da erfolgen wo möglich. Auswahl erfolgt nach anklicken des Buttons „Zeitmodul 1“



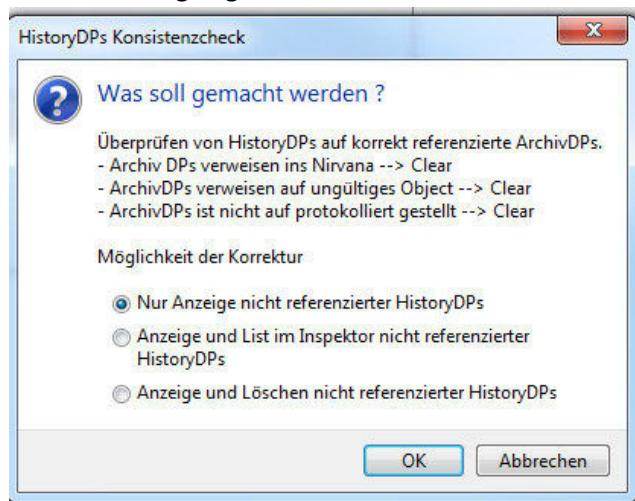
## 7.7 Test und Korrektur HistoryDPs.

HistoryDPs sind eine wunderbare Gelegenheit um immensen Datenmüll in der RegaDOM zu hinterlassen. So wird beispielsweise für bei Anwahl eines Channels auf protokolliert für jeden einzelnen Datenpunkt des Channels ein HistoryDP angelegt. Wird das Gerät irgendwann mal gelöscht oder die Aufzeichnung deaktiviert, so bleibt – richtig gedacht – der HistoryDP erhalten. Kann man so machen, muss man aber nicht.

Mit dem Punkt



Lassen sich verwaiste oder auch falsche HistoryDPs aufspüren. Die Art der Bearbeitung wird beim Anklicken festgelegt.



Nur Anzeige:

Listet die gefundenen „Inkonsistenzen“ rechts im Ausgabefenster auf. Es wird aber noch nix korrigiert oder verändert

Anzeige und List im Inspektor

Listet die gefundenen „Inkonsistenzen“ rechts im Ausgabefenster auf. Zusätzlich werden die gefundenen IDs in Listauswahl des Inspektors geladen und können dort mit dem Inspektor untersucht werden. Es wird aber noch nix korrigiert oder verändert.

Anzeige und Löschen

Wie unter Punkt 1 werden die gefundenen Inkonsistenzen rechts im Ausgabefenster gelistet, gleichzeitig wird korrigiert (nicht referenzierte DPs werden gelöscht).

Lauf alles in Ordnung:

```
Analyse History DPs
Black in August 2019
-----
Es wurden 11 HistoryDPs überprüft
Dabei wurden keine Auffälligkeiten festgestellt
```

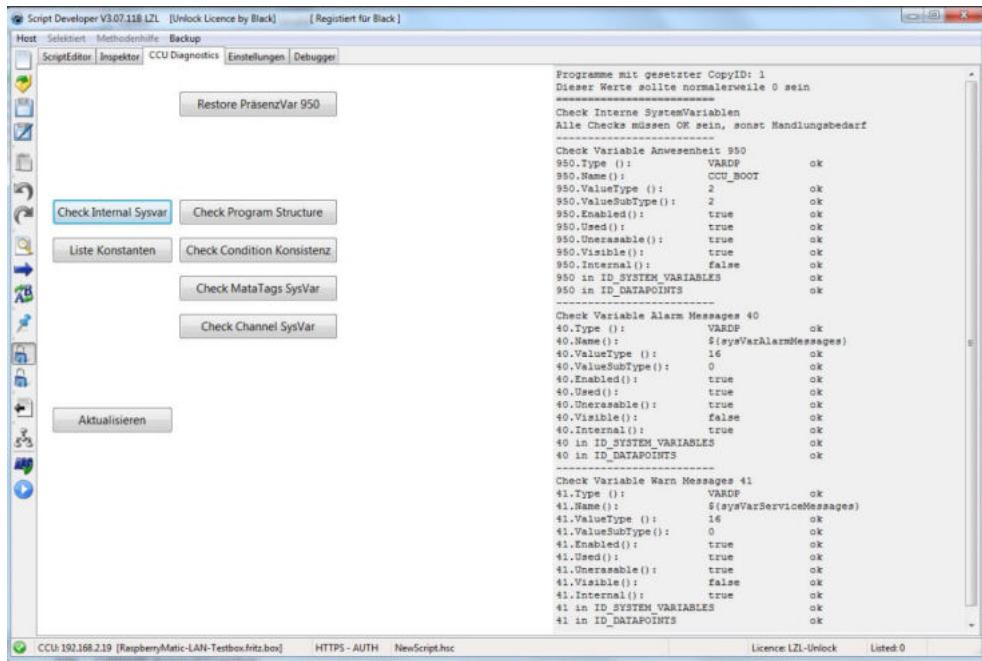
Ein HistoryDP verweist auf eine Systemvariable, bei der Protokollierung allerdings abgewählt wurde:

```
Analyse History DPs
Black in August 2019
-----
HistoryDP 11208 verweist in ArchiveDP [6461] auf Systemvariable mit DPArchive ()=false. Empfehlung löschen
-----
Es wurden 12 HistoryDPs überprüft
Dabei wurden 1 Inkonsistenz(en) festgestellt
Davon wurden 0 Inkonsistenz(en) korrigiert
```

Wenn Korrektur angewählt würde, würde der HistoryDP gelöscht werden vom SDV.

## 7.8 Check Internal Sysvar

Hierbei werden die Systeminternen auf Konsistenz geprüft (ID 950 Anwesenheit, ID 40 und 41 , Warnung und Alarm)



Alles Tests sollten OK melden, bei Fehler besteht Handlungsbedarf.

## 8 License Disclaimer

Mit diesem Programm werden 2 DLLs des OpenSSL Project mitgeliefert

Original Lizenztext:

### LICENSE ISSUES

---

The OpenSSL toolkit stays under a dual license, i.e. both the conditions of the OpenSSL License and the original SSLeay license apply to the toolkit. See below for the actual license texts. Actually both licenses are BSD-style Open Source licenses. In case of any license issues related to OpenSSL please contact openssl-core@openssl.org.

### OpenSSL License

---

```
* =====
* Copyright (c) 1998-2016 The OpenSSL Project. All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
*
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in
*    the documentation and/or other materials provided with the
*    distribution.
*
* 3. All advertising materials mentioning features or use of this
*    software must display the following acknowledgment:
*    "This product includes software developed by the OpenSSL Project
*    for use in the OpenSSL Toolkit. (http://www.openssl.org/)"
*
* 4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to
*    endorse or promote products derived from this software without
*    prior written permission. For written permission, please contact
*    openssl-core@openssl.org.
*
* 5. Products derived from this software may not be called "OpenSSL"
*    nor may "OpenSSL" appear in their names without prior written
*    permission of the OpenSSL Project.
*
* 6. Redistributions of any form whatsoever must retain the following
*    acknowledgment:
```

\* "This product includes software developed by the OpenSSL Project  
\* for use in the OpenSSL Toolkit (<http://www.openssl.org/>)"  
\*  
\* THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS'' AND ANY  
\* EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE  
\* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR  
\* PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR  
\* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,  
\* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT  
\* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;  
\* LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)  
\* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,  
\* STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)  
\* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED  
\* OF THE POSSIBILITY OF SUCH DAMAGE.  
\* ======  
\*  
\* This product includes cryptographic software written by Eric Young  
\* (eay@cryptsoft.com). This product includes software written by Tim  
\* Hudson (tjh@cryptsoft.com).  
\*  
\*  
\* Original SSLeay License

-----  
\* Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com)  
\* All rights reserved.  
\*  
\* This package is an SSL implementation written  
\* by Eric Young (eay@cryptsoft.com).  
\* The implementation was written so as to conform with Netscapes SSL.  
\*  
\* This library is free for commercial and non-commercial use as long as  
\* the following conditions are aheared to. The following conditions  
\* apply to all code found in this distribution, be it the RC4, RSA,  
\* Ihash, DES, etc., code; not just the SSL code. The SSL documentation  
\* included with this distribution is covered by the same copyright terms  
\* except that the holder is Tim Hudson (tjh@cryptsoft.com).  
\*  
\* Copyright remains Eric Young's, and as such any Copyright notices in  
\* the code are not to be removed.  
\* If this package is used in a product, Eric Young should be given attribution  
\* as the author of the parts of the library used.  
\* This can be in the form of a textual message at program startup or  
\* in documentation (online or textual) provided with the package.  
\*  
\* Redistribution and use in source and binary forms, with or without  
\* modification, are permitted provided that the following conditions  
\* are met:  
\* 1. Redistributions of source code must retain the copyright  
\* notice, this list of conditions and the following disclaimer.  
\* 2. Redistributions in binary form must reproduce the above copyright  
\* notice, this list of conditions and the following disclaimer in the

\* documentation and/or other materials provided with the distribution.

\* 3. All advertising materials mentioning features or use of this software

\* must display the following acknowledgement:

\* "This product includes cryptographic software written by

\* Eric Young (eay@cryptsoft.com)"

\* The word 'cryptographic' can be left out if the routines from the library

\* being used are not cryptographic related :-).

\* 4. If you include any Windows specific code (or a derivative thereof) from

\* the apps directory (application code) you must include an acknowledgement:

\* "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"

\*

\* THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS'' AND

\* ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE

\* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE

\* ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE

\* FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL

\* DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS

\* OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)

\* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT

\* LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY

\* OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF

\* SUCH DAMAGE.

\*

\* The licence and distribution terms for any publically available version or

\* derivative of this code cannot be changed. i.e. this code cannot simply be

\* copied and put under another distribution licence

\* [including the GNU Public Licence.]

\*/