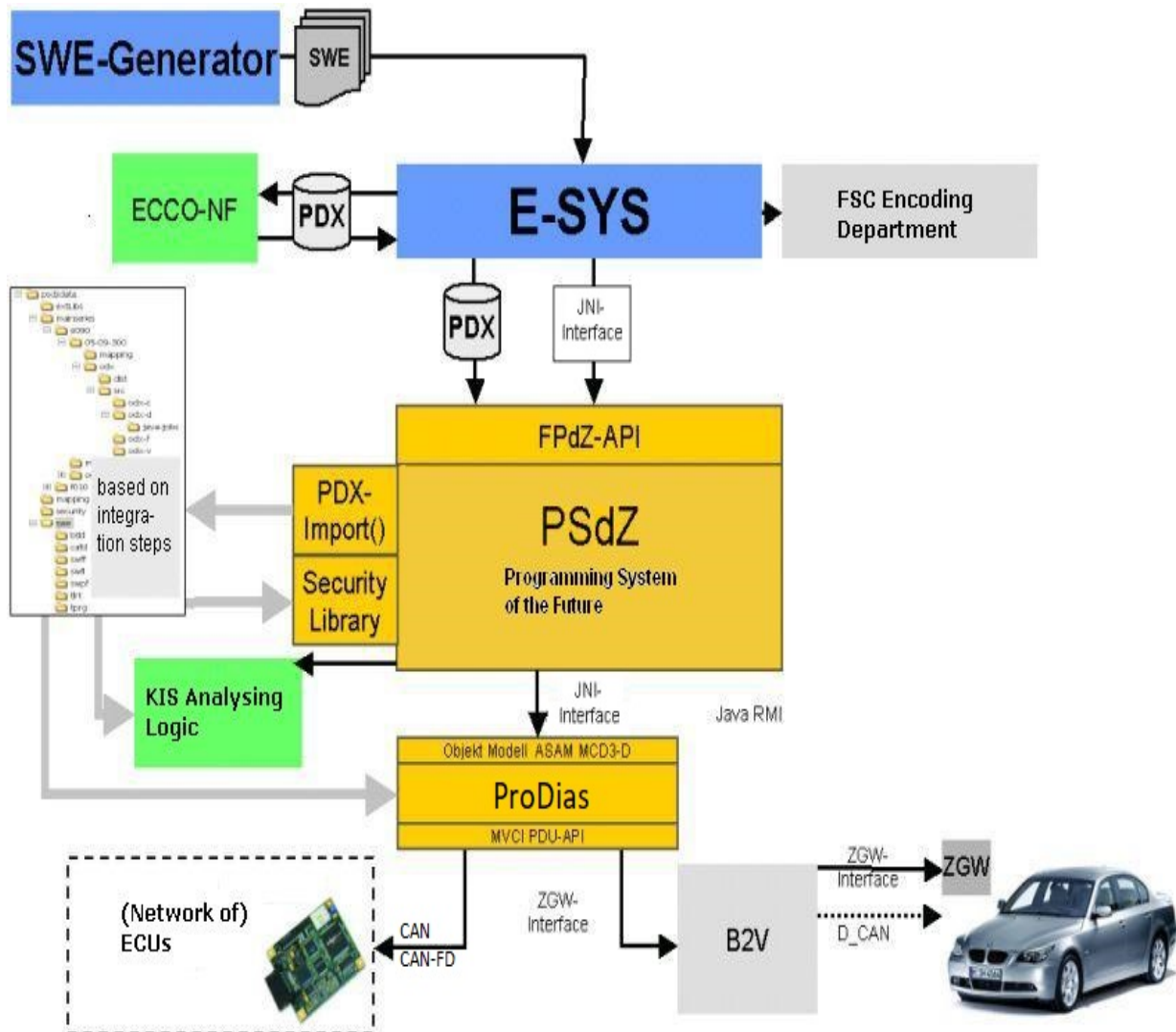


# Introduction

[Top](#) [Next](#)

**Welcome to E-Sys Help System!** With E-Sys you can program / flash Emus and encode them. In addition to that E-Sys offers a bunch of editors and viewers for editing the populating with data necessary for that.

Subsequent we display the architecture of the systems involved in the programming:



E-Sys needs a correct populating with data for programming and encoding. Also necessary is a core program system that provides the services for communication and logistic. For the

populating with data a PDA template is unnecessary with populating of communication (diagnosis, protocol, connection information) for the board net system. The software logistics provides this template.

In this template the developer inserts his software to be transferred in E-Sys into [PDX-Charger](#) in the form of SWEs.

The SWEs are generated from SWE-Generator.

The PSdZ ("Programmierersystem der Zukunft" programming system of the future) provides the services necessary for programming and encoding. For the implementation of the communication services PSdZ uses the D-Server; for the implementation of the logistic services KIS (compatibility and information system [still not completed]) is used. The implementation on the CanCard interface happens directly by the D-Server, the implementation of the communication over Ethernet/ZGW happens by the B2V-Server (BusinessToVehicle).

To realize programming and/or encoding the developer needs a TAL (transaction list) which contains at least one SWDeploy transaction (for programming SWEs) or one CDDeploy transaction (for encoding). Furthermore a FA ("Fahrzeugauftrag" vehicle order) is necessary.

E-Sys provides the [TAL-Editor](#) and [FA/FP-Editor](#) for creation and editing of TALs and FAs, besides the module [TAL-Calculating](#) for calculating a TAL from a pair of actual system installation table ("SVT-Actual") and reference system installation table ("SVT-Target"); SVT abbreviates "System-Verbau-Tabelle". Also in the module [VCM](#) all elements (TAL, FA, SVT-Actual and SVT-Target) of the VCM (Vehicle Configuration Management) can be read.

E-Sys provides an own module [Coding](#) for encoding. Here the codable states are detected and executed in the module [TAL-Processing](#). This coding data can be inspected in E-Sys in [CAF-Viewer](#) (CAF=CodingApplicationFile). The coding data

read out from the ECU can be inspected and changed in the format of a FDL (Function Data List) in [FDL-Editor](#). E-Sys is delivered (Stand 1.0.0) with a populating of data for programming and encoding of the evalboard (diagnosis address 7E) by CANCardX. For programming and encoding by ZGW an adequate populating of data must be obtained from the software logistic and imported in [PDA-Charger](#).

## **Training of vehicle programming and E-Sys handling can be found at the BMW-Intranet:**

[Basic training vehicle-programming](#) (focus for new employees who need to know the basic of the vehicle programming at BMW)

[Advanced training of the E-Sys handling](#) (focus for employees who need a more detailed handling)

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

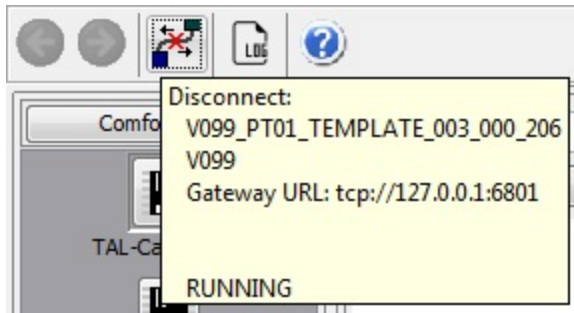
The user interface of E-Sys provides the standard features of a Windows interface so you can be familiar soon with the handling of E-Sys.

The GUI is adapted to the look and feel of Microsoft Outlook. The base elements you use the most time are set forms, dialogs, symbol bars, menus and the buttons for switching between the set forms.

The E-Sys user interface consists of four interaction elements:

- **menu bar:** The discrete functions of the program can be called by menu items. The structure of the menu is divided into function groups (upper menu bar) and sub-functions called by sub menus.
- **symbol bar:** Important functions can be called by special buttons in the symbol bar by a mouse click.
- **navigation bar:** The E-Sys user interface uses some views for data input. These are called by special buttons in the left navigation bar.
- **status bar:** Global data of the application are displayed in the status bar.

In addition, the tooltip displays further information. For example, in the following screen, the tooltip for Connection establishment disconnection is displayed. Here you can find the current connection that was previously displayed in the status bar:



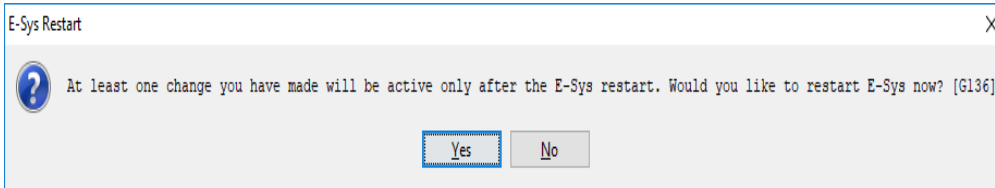
---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

## Options

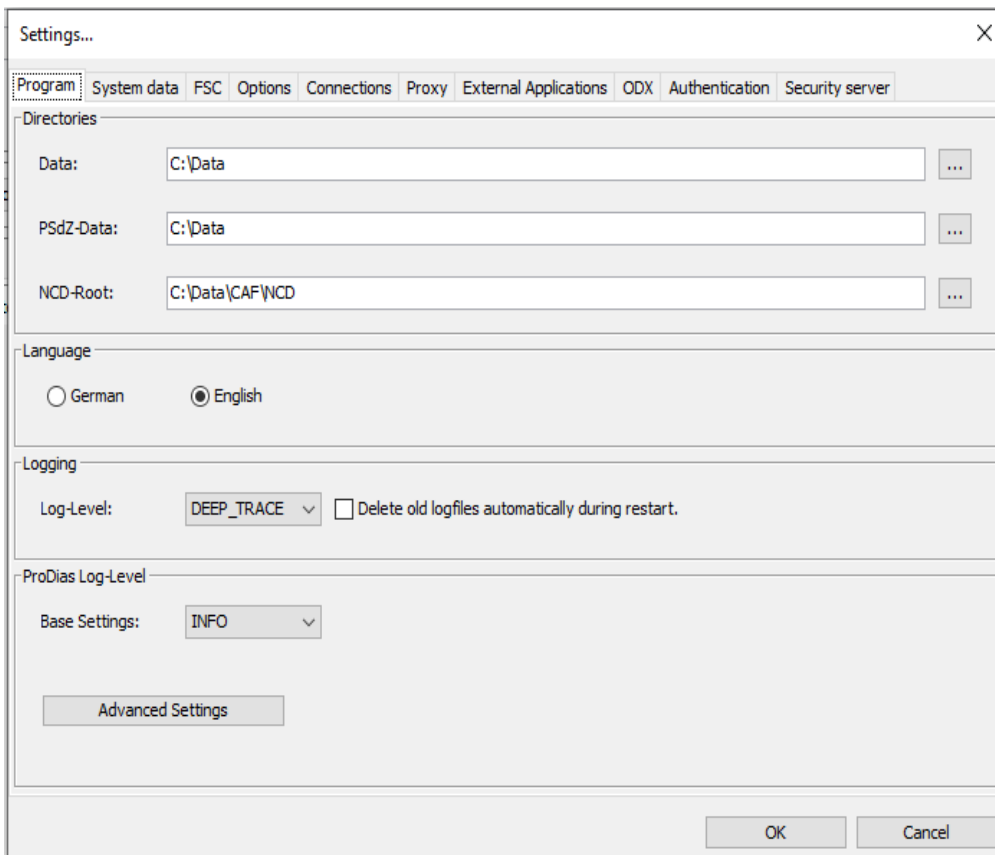
[Top](#) [Previous](#) [Next](#)

In the dialog "Options" the global settings of E-Sys are specified. Some options require a new start of E-Sys to take effect. Then the following dialog appears:



After pressing the "Yes" button E-Sys will be closed and started again with the new settings.

## Program



Directories	
Data	Data directory (Default: C:\Data)
PSdZ-Data	PSdZ-Data directory (Default: C:\Data).

NCD-Root	NCD-Root directory (Default: C:\Data\CAF\NCD).
<b>Language</b>	
German	Set to German language
English	Set to English language
<b>Logging</b>	
Log-Level	Setting of log levels
Delete old log files automatically during restart	During start all old log files are deleted

*Directories:*

- In the **Data** directory, E-Sys stores all relevant data (Log, TAL, executedTAL, SVT, FA, etc.).
- For **PSdZ-Data**, a directory must be entered in which the psdzdata directory is to be created. This is where the contents of the imported containers are stored.
- For **NCD-Root**, a directory must be entered in which the subdirectories "unsigned", "signed" and "default" (delivery state) should be created. This directory stores the net encoding data used for encoding.

*Language:*

Select the language used to represent E-Sys.

*Logging:*

Here you can set the information with which the E-Sys-Log is enriched.

The default setting is "DEEP\_TRACE".

For analysis of errors (programming, coding or e-sys errors), the log level "DEEP\_TRACE" is mandatory.

**OFF:**

The log does not include any additional log output.

**WARN:**

The log also includes WARN editions

**INFO:**

The log also includes INFO and WARN outputs

**FINE:**

The log also includes FINE, INFO and WARN outputs

**DEBUG:**

The log also includes DEBUG, FINE, INFO and WARN outputs

**TRACE:**

The log also includes TRACE, DEBUG, FINE, INFO and WARN outputs

**DEEP\_TRACE:**

The log also includes DEEP\_TRACE, TRACE, DEBUG, FINE, INFO and WARN

outputs

## System data

TesterApplyIdentifier (hex)	Specifies where the ECU was programmed (Systemsupplier, Development, RSU, ...)
ProgrammingDeviceType	Type of tester device.
ProgrammingDeviceSerialNo.	Unique identifier of programming device.
BusPriority	Specifies bus priority.
<b>FingerprintID Settings</b>	
PlantID (dec)	Specifies the identifier of the plant. (SystemSupplier, vehicle plant, HO, ...). The Plant ID (in which plant I am) must be known by the user himself.
SystemsupplierID (dec)	Specifies the Supplier-ID with following meaning: 0...99 (BCD coded e.g. 57 -> Fingerprint 0x0054) 100...153 (not allowed) 154...65535 (Decimal e.g. 154 -> Fingerprint 0x009A)



	Automatically set when selected via the "Systemsupplier List".
Systemsupplier List	Drop-down-List Suppliers Automatically set when selected via the "SystemsupplierID" (if the supplier is known).
DealerID (dec)	Specifies the number of the dealer.
FingerprintID	Fingerprint ID is calculated according to the selected fingerprint ID.
Default-Button	When the default button is set, the default values for the "System Data" tab are restored.

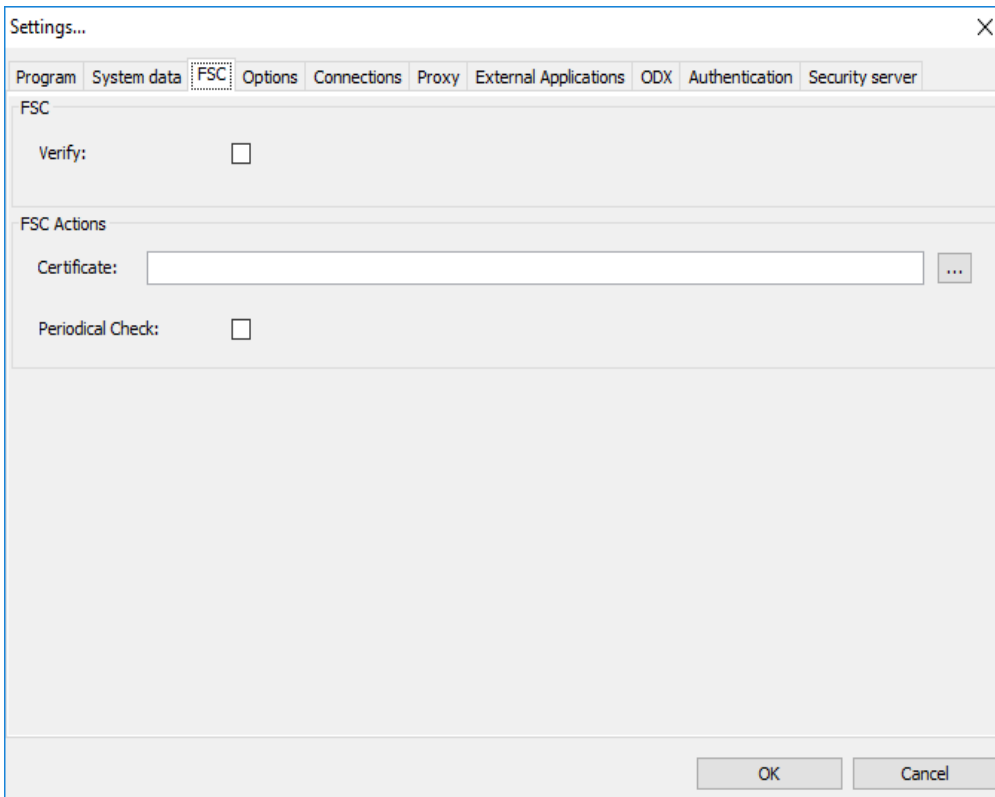
*TesterApplyIdentifier:*

The tester insert ID as well as the FingerprintID are elementary components of the fingerprint, which is written to the ECU before each programming.  
Thus, it is understandable under which UseCase the control unit was last treated.

*FingerprintID Settings:*

Using the radio buttons, the user selects whether a plant ID (PlantID), a supplier ID (SystemsupplierID) or a dealer ID (DealerID) should be transferred to the fingerprint.

**FSC**

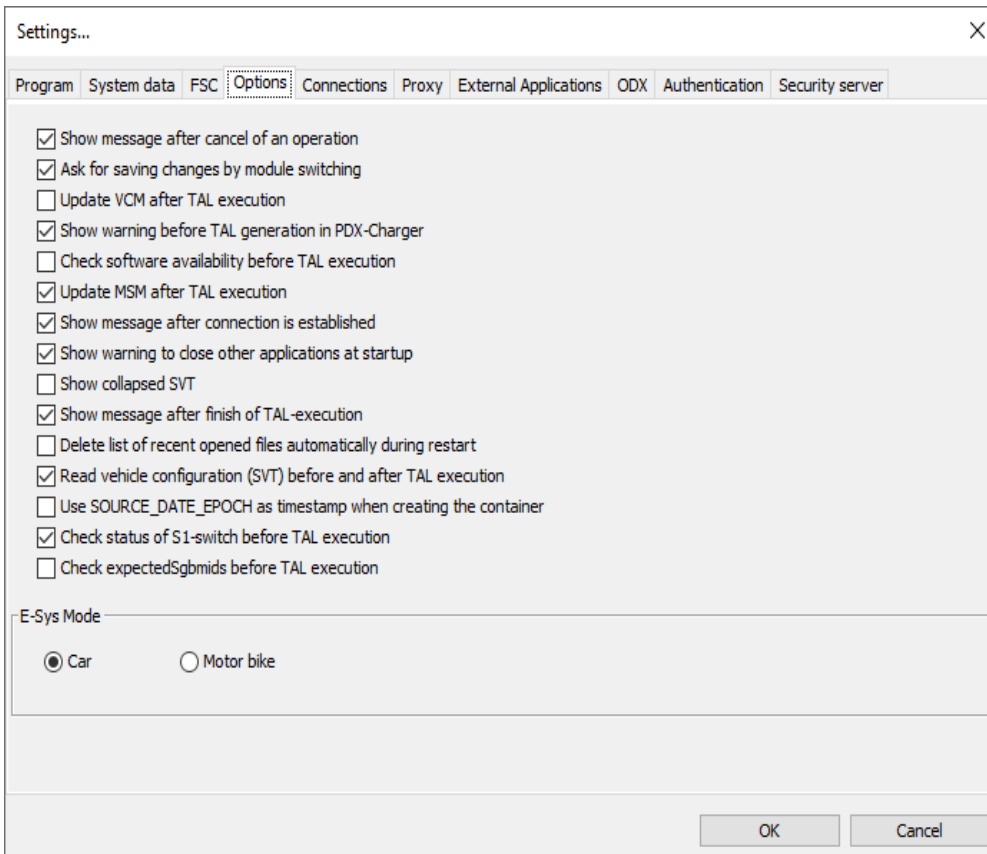


<b>FSC</b>	
Verify	Verify FSC; can be switched off for performance reason. Indicates whether the unlock codes should be validated regularly.
<b>FSC Actions</b>	
Certificate	Specification of a Certificate. In several functions this Certificate is compared with the Certificate in the ECU.
Periodical Check	Before every action the actual state is checked (still not implemented).

*Additional information:*

For more information about FSC (activation codes) an E-Sys, see [comfort-](#) or [expert-](#)mode.

**Options**



Show message after cancel of an operation	Here you can turn on and off the confirmation that an action has been canceled.
Ask for saving changes by module switching	Here you can turn on and off the query to save.
Update VCM after TAL execution	Select whether to update VCM.
Show warning before TAL generation in PDX-Charger	Here the warning can be switched on and off.
Check software availability before TAL execution	Select whether to check whether the SWEs to be used exist before processing the TAL.
Update MSM after TAL execution	Select whether to update MSM (Master-Security Module). If so, the transport keys ( <a href="#">TSL</a> ) are updated.
Show message after connection is established	Here you can switch on and off the confirmation that a connection has been established.
Show warning to close other applications at	Here you can switch on and off the Warning shown at startup.

startup	
Show collapsed SVT	Here you can specify if the SVT is shown collapsed or extended by default.
Show message after finish of TAL-execution	If enabled, a popup window is displayed after TAL-execution indicating the status.
Delete list of recent opened files automatically during restart.	Here you can decide if the list of recent opened files will be deleted after E-Sys restart or not.
Read vehicle configuration (SVT) before and after TAL execution.	If enabled, readSVT will be done before and after the TAL execution and will be logged in the Log File.
Use SOURCE_DATE_EPOCH as timestamp when creating the container	Select whether to use the current system time or the environment variable SOURCE_DATE_EPOCH when creating a PDX container.
Check status of S1-switch before TAL execution	Only relevant if TesterApplyIdentifier=0xF. This check is always carried out for TesterApplyIdentifier != 0xF. In this case, specifies whether the S1 test should be carried out or not. This check verifies that the primary and secondary energy-network are connected.
Check expectedSgbmids before TAL execution	Flag whether the list of expectedSgbmids shall be verified before TAL-execution.
<b>E-Sys Mode</b>	
Car	E-Sys is used for programming/coding car CONTROL units.
Motor bike	E-Sys is used for programming/coding motorcycle control units. (See <a href="#">Motorbike</a> )

**SOURCE\_DATE\_EPOCH:**

The environment variable SOURCE\_DATE\_EPOCH is entered in the META data when the PDX containers are created.

If this value is not set, the current system time is used.

If the value is set, the contents of the environment variable SOURCE\_DATE\_EPOCH is used.

For more information, see: <https://reproducible-builds.org/docs/timestamps/>.

The user is responsible for ensuring that the variable contains the correct value.

If the variable is not set or does not contain a valid timestamp, the current system time is used.

**E-Sys Mode:**

Selection of control units should be treated with cars or motorcycle routines.

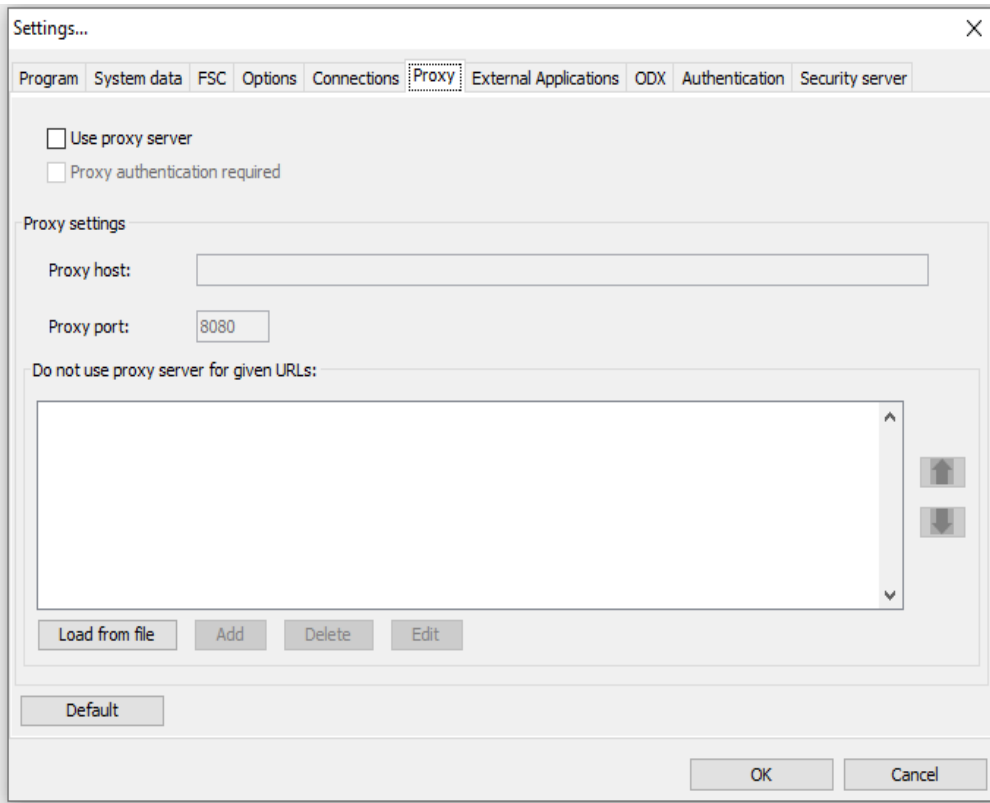
Especially at VCM, different routines are available here.

**Connections**

<b>All connections</b>	
Additional Transmission Timeout [ms]	This time is independent of the selected interface with which the connection to the control unit / vehicle is established.
<b>ICOM connection</b>	
Base Port	Base Port, will be used to calculate the port of the connection using port mapping.
<b>Http Update</b>	
starting Http-Server at	Determines whether the PSdZ internal Http server

Interface Type Ethernet	should be used for flashing over Ethernet.
Server Port	Port on which the Http Update Server is started. 8888 is the number for the PSdZ internal server. The port needs to be 8000, 8080, 8888 or a value between 12000 and 13000
Maximum Http Throughput [Mbit/s]	Maximum Http throughput in Mbit per second that is allowed to the Http server. The input field can either be empty or contain a number between 0,008192 (1024 Byte/s) and 2500,0. Floating point numbers are allowed. The right, non-editable field shows the value converted into bytes per second.
Default-Button	When the default button is set, the default values for the "Connections" tab are restored.

## **Proxy**



Use proxy server	Whether to use a proxy for the backend connections. Subsequent settings can only be edited if the check mark is set here.
------------------	---

Proxy authentication required	Switching on and off the using of authentication for proxy server. The prompt for entering the login data for proxy does not appear until a backend interface is called up.
<b>Proxy settings</b>	
Proxy host	URL of the proxy server
Proxy port	Port of the proxy server
<b>Do not use proxy server for given URLs</b>	
Load from file	Here you can load a txt file with different URLs.
Add	Allows you to add individual URLs.
Delete	This can be used to delete a selected URL (only single selection is possible).
Edit	Can be used to edit a selected URL.
Arrow-Buttons	The arrow buttons can be used to move a marked URL down or up in order.
Default-Button	When the default button is set, the default values for the "Proxy" tab are restored. (URLs will not be deleted!)

*Do not use proxy server for given URLs:*  
The URLs entered here do not use the proxy.

*Load file:*

In the txt-file, the URLs may only be entered in one line and only separated with a semicolon ";". E-Sys does not support another formats of the URLs.

Example:

https://www.google1.de;https://www.bmwgroup.com;https://www.bing2.com

*Add:*

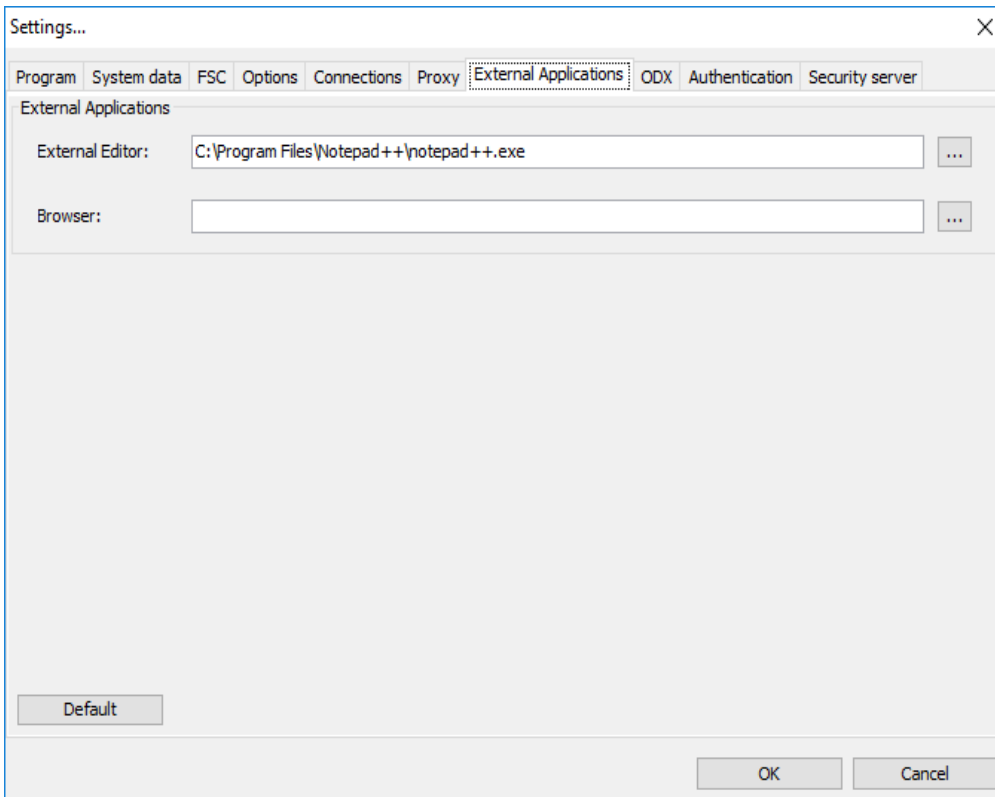
Only one URL may be added at a time.

Multiple entries (separated with semicolon ";") are invalid.

*Additional information:*

Only BASIC is supported as an authentication method for a proxy.

## **External Applications**



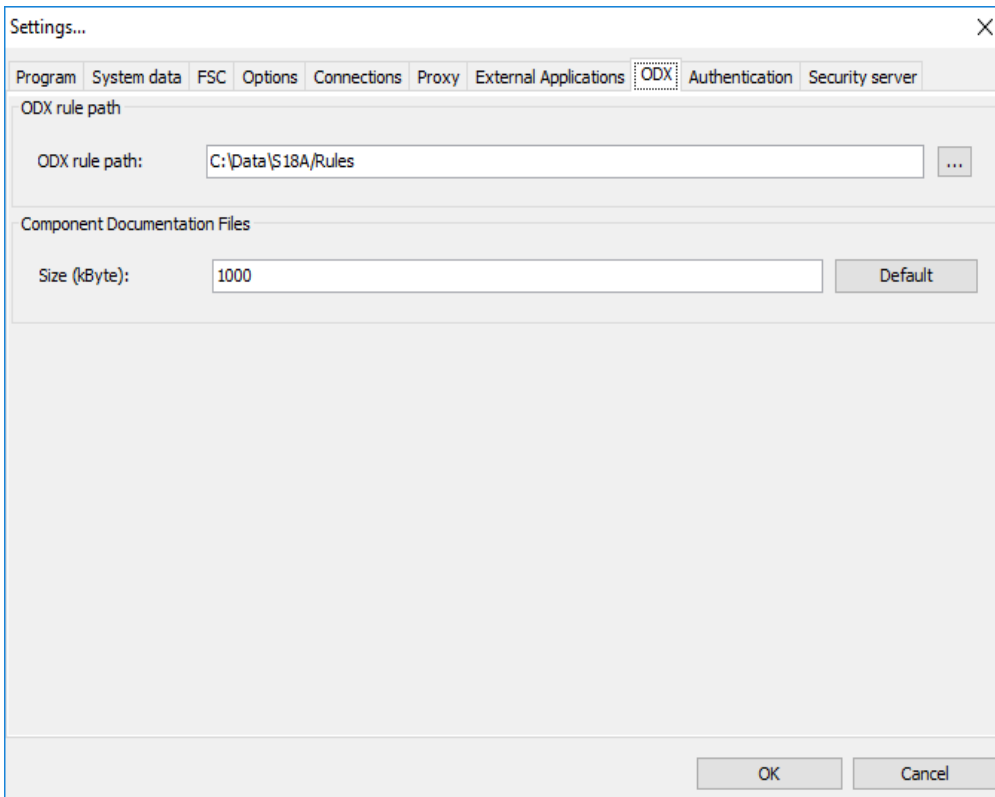
External Applications	
External Editor	Path of the external editor.
Browser	Path of the external browser.
Default-Button	When the default button is set, the default values for the "External Applications" tab are restored. (Both fields are emptied)

*Additional information:*

In the "External Applications" tab, the editor and the browser are indicated to display data.

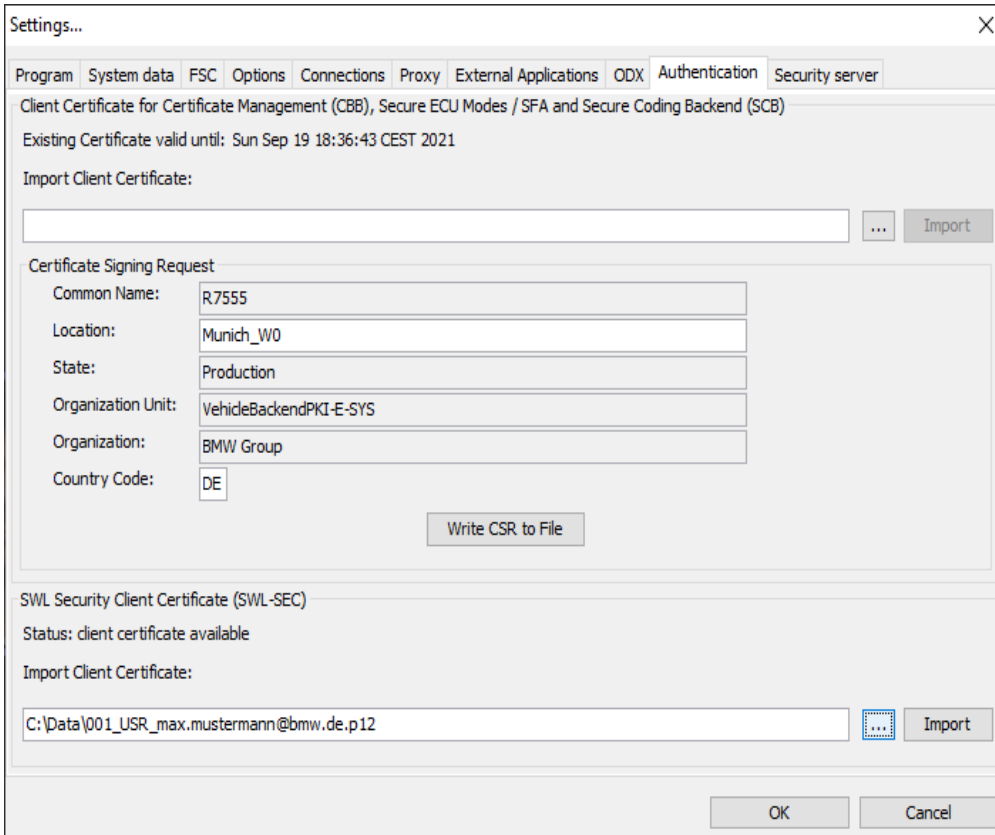
**ODX**





<b>ODX rule path</b>	
ODX rule path	Path of the ODX rule directory.
<b>Component Documentation Files</b>	
Size (kbyte)	Maximal size of the documentation in a component-container.
Default-Button	Pressing the default button restores the default value for the size of the "Component Documentation".

## **Authentication**



<b>Client Certificate for Certificate Management (CBB), Secure ECU Modes / SFA and Security Coding Backend (SCB)</b>	
Import Client Certificate	Import functionality of a client Certificate for connecting to the CBB. Connection to the CBB is required as part of Certificate management in order to calculate bindings for ECUs.
Import Client Certificate	Import functionality of a client Certificate for connecting to the SWL-SEC. Connection to the SWL-SEC is required as part of secure coding in order to sign net coding data.
<b>Certificate Signing Request</b>	
Common Name	Input field (cannot be changed)
Location	Input field (can be customized e.g. "Munich_W0")

State	Input field (cannot be changed)
Organization Unit	Input field (cannot be changed)
Organization	Input field (cannot be changed)
Country Code	Input field (can be customized e.g. "DE")
Write CSR to File	Create a CSR

*Client Certificate for CBB and Secure ECU Modes / SFA:*

Here, the client Certificate is imported.

This is required to communicate with the back-end systems for Certificate management (CBB) as well as for SFA (Secure Feature Activation) from the BMW-Intranet.

*Certificate Signing Request (CSR):*

Here, a client CSR file is generated based on the input fields for the current computer and stored locally (see pop-up window).

This client CSR file can be used to generate a client Certificate through the b2b portal ([see howto at BMW-Intranet](#)).

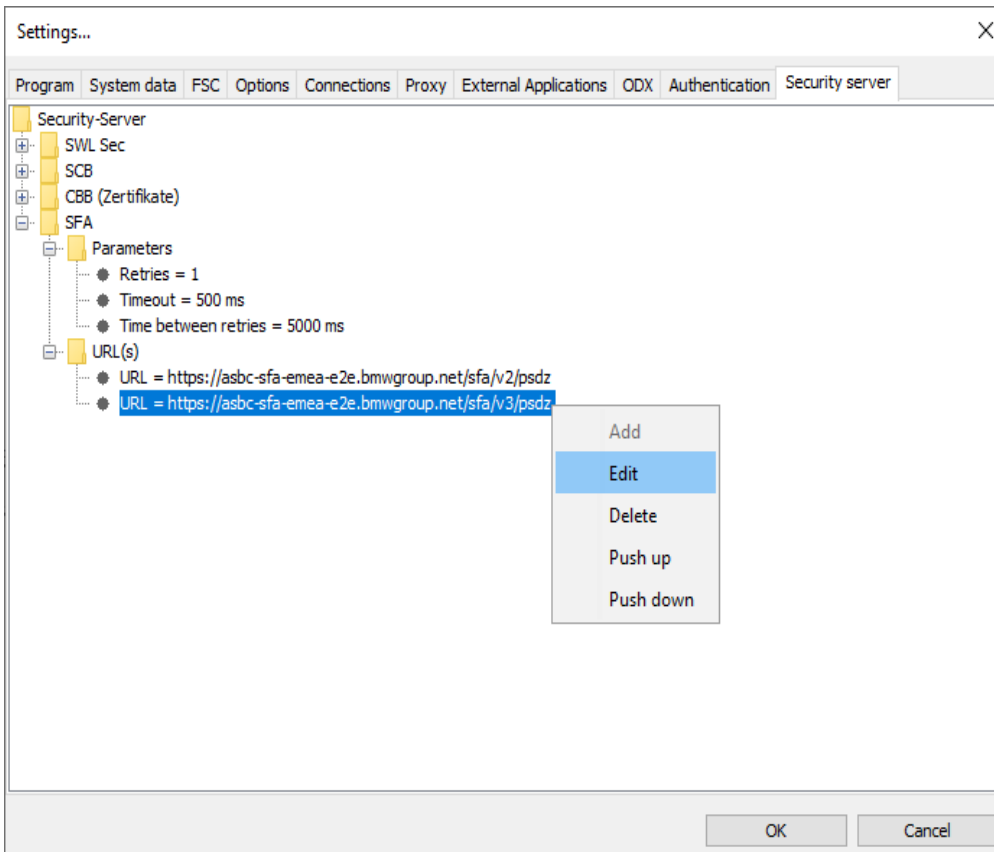
The client Certificate can be imported into E-Sys via "Import Client Certificate".

*Additional information:*

A client Certificate is required to communicate with the CBB (backend for Certificate Management functionality) or the SFA backend.

You can only import a client Certificate if it matches the last generated client CSR file.

**Security server**



<b>Backend Parameters</b>	
Retries	Number of attempts to establish a connection with a backend in the event of an error. If the connection to 1.Server URL fails, repeating >0 will proceed to the next server URL in the list. Range of values: 0-3
Timeout	Time to wait before canceling the request.
Time between retries (ms)	Time to wait before starting a new request to a server URL. Range of values: 2000-20000
<b>URL(s) - Context menu</b>	
Add	Allows you to add individual URLs. Only one URL may be added at a time.
Edit	Can be used to edit a selected URL.
Delete	This can be used to delete a selected URL (only single selection is possible).
Push up	Move a marked URL up in order.
Push down	Move a marked URL down in order.

*Additional information:*

URLs for SWL-SEC and SCB are not displayed and cannot be edited via E-Sys GUI.

The changed values are applied when the Enter key is pressed.

The set values are only transferred to the E-Sys properties when the settings dialog is closed with the "OK" button.

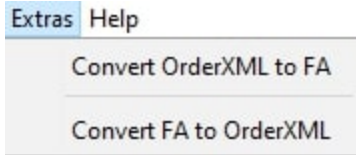
Information on the subject of SFA and SFA backend URL can be found on the BMW intranet [here](#).

Information on Certificate management and CBB URLs can be found on the BMW intranet [here](#).

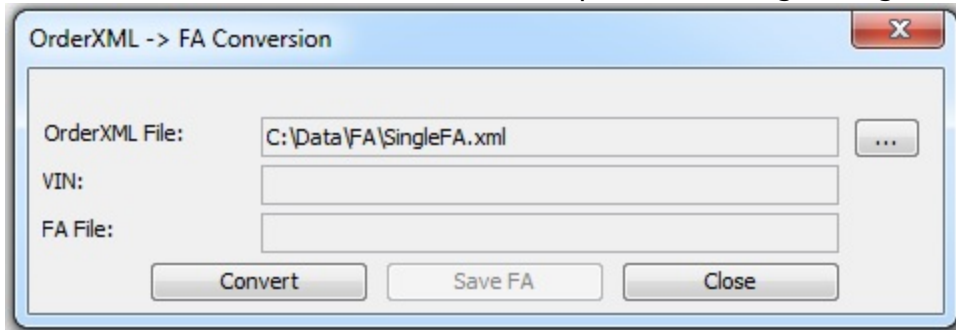
---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

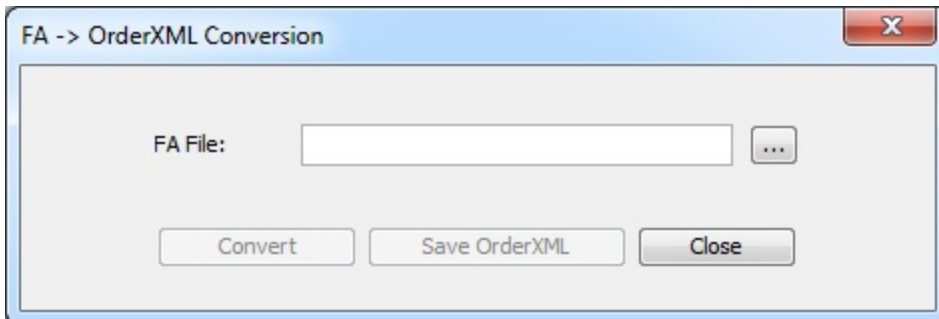
The Extras menu offers functions for converting FA-files:



Menu item 'Convert OrderXML to FA' opens following dialog:



With menu item 'Convert FA to OrderXML' the user is offered following dialog:



---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

[Calculation of TAL](#)

[FSC](#)

[Certificate-Management](#)

[SFA / ECU-Modes](#)

[KDS](#)

[ANFLASH](#)

[SecOC](#)

[ProgrammingProtection](#)

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

The module "Calculation of TAL" calculates a TAL (transaction list) from a pair of actual system installation table ("SVT-Actual") and reference system installation table ("SVT-Target"). This TAL can be executed in the module TAL-Processing. SVT-Actual specifies the actual state or installation of the system i.e. which hardware unities and which software unities reside in the ECU/vehicle. SVT-Target specifies the state of the ECU/vehicle to be reached after the processing of the TAL.

By clicking the checkbox "Use data backup" and specification of the backup directory the individual information of the former exchanged ECU can be taken-up into the TAL creation process.

The backup data is created in mask [TAL-Processing](#).





Comfort Mode

TAL-Calculating

FSC

Certificate Management

SFA

KDS

Anflash

Expert Mode

Editors & Viewers

Data Handling

External Applications

Personal view

Vehicle Order

Read Load Save Edit

Vehicle Profile

SVT

- SVT
  - ECUs (1)
    - CCU2 (3a)
      - HVEL\_0000598F\_004\_003\_002
      - HVEL\_0000598F\_004\_003\_002
      - MBELR-01 [3A0001]
      - CFI-06 [3A0002]
        - BTLD\_0000808E\_001\_000\_000

SVT Actual

File Name: C:\01\_FP\SVT\_SMAC.xml

Read (ECU) ind. smart actuators Read (ECU) without smart actuators Read (VCM)

Load Save Edit

SVT Target

KIS

I-Step (shipm.):  Calculation Strategy

I-Step (target):   Single Flash  Construction Progress

Complete Flash

Calculate

SVK Preview

I-Step: S18A-18-11-400

Alternative I-Step:

Diagnostic Addresses:

BNTN Variants:

Download

Calculation Strategy

via Diagnostic Addresses

via Vehicle Order

via Vehicle Order and BNTN-Variants

SVT-Update

PDX-Container:  ... Update SVT

File Name:

Read (VCM) Load Save Edit

HW-IDs from SVTactual

TAL-Filter

File Name:

Load Edit Reset

TAL

Use data backup

ID-Backup ID-Restore Directory:  ...

Include ECUs from SVTtarget only  with Mirror-Protocol

TAL:

TAL calculation Save Edit Execute

SVT filter

All  SVT Reset

Actual state Target state Identical state S SMAC Hardware difference

Login Backend SFA: Logged out SWL-Sec: Logged out

## **GUI Elements**

<b>Vehicle data</b>	
Vehicle Order	File containing a list of vehicle orders. For selecting a certain vehicle order, this item needs to be activated by using the context menu item 'Activate FA'. If the list of vehicle orders contains only one element, this order will be automatically activated as soon as the file is loaded.
Read	Read current vehicle order from vehicle.
Load	Load a FA from file system.
Save	Save the current FA.
Edit	Load current vehicle order to FA/FP editor.
Vehicle Profile	The Vehicle profile will be automatically generated from an activated Vehicle Order and displayed in a tree structure.
<b>SVT Actual</b>	The actual SVT
Read (ECU) incl. Smart Actuators	Read SVT from ECU (functionally) including Smart Actuators
Read (ECU) without Smart Actuators	Read SVT from ECU (functionally) excluding Smart Actuators
Read (VCM)	Read SVT from vehicle.
Load	Load a SVT from file system.
Save	Save the SVT-Actual.
Edit	Edit in the corresponding editor.
<b>SVT Target</b>	The reference SVT.
Read (VCM)	Read SVT-Target from VCM
Load, Save	Load/save an SVT-Target from/to file system.
File Name	The name and path of the SVT file.
Edit	Edit the SVT-Target in the corresponding editor.

HW-IDs from SVTactual	HWEL, HWAP und HWFR elements from SVTactual will be copied to the Emus in the target SVT.
<b>SVT Target (KIS calculation)</b>	
I-Step (shipm.)	Available shipment integration steps.
I-Step (target)	Available target integration steps.
Calculation strategy	Possible strategy to calculate the SVT (Single Flash   Complete Flash   Construction Progress).
Calculate	Calculates an SVT-Target by integration step and vehicle order (and SVT_Actual if exists).
<b>SVT Target (SWL-Cockpit)</b>	
I-Step	Text field for the I-Step of the SVT-Target (mandatory)
Alternative I-Step	optional field for an alternative I-Step
Diagnostic Addresses	comma separated list of diagnostic addresses (in hex-format)
BNTN Variants	comma separated list of BNTN-Variants
Calculation strategy	Possible strategy to calculate the SVT (via Diagnostic Addresses   via Vehicle Order   via Vehicle Order and BNTN-Variants).
Download	Sends request to SWL-Cockpit and receives a calculated SVT-Target.
<b>SVT Target (PDA-Update)</b>	
Load PDA-Container	Load PDA-Container for updating SVT
Update SVT	Start SVT-Update process
<b>TAL-Filter</b>	
File Name	Filename of the TAL-Filter
Load	Opens a file dialogue to load an existing

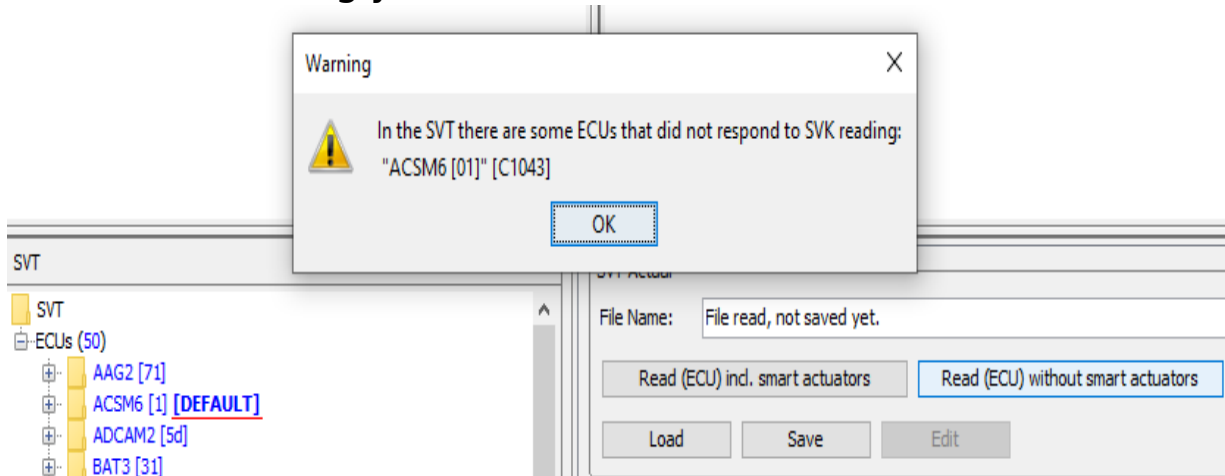
	TAL-Filter from a file. The settings from the TAL-Filter will then be visible in the SVT-View, the file path is displayed in the text field. An already loaded/opened filter will be discarded.
Edit	Edit in the corresponding editor. This button is only active if there is an active filter.
Reset	Deletes the active filter together with all corresponding symbols in the SVT-View. This button is only activated if there is an active filter.
<b>TAL Individual data recovery</b>	
ID-Backup	Calculate the IDR-Backup TAL using all necessary data.
ID-Restore	Calculate the IDR-Restore TAL using all necessary data.
Directory	The backup directory that contains the individual data. When calculating IDR-Restore TAL, the specification is mandatory.
<b>TAL</b>	
Emus from SVTsoll	If checked only Emus from the target SVT will be considered for the TAL generation.
with Mirror-Protocol	If checked the TAL with mirror protocol actions will be calculated.
TAL Calculation	Calculate the normal TAL or TAL with mirror protocol actions using all necessary data.
Save	Save the generated TAL.
Edit	Edit in the corresponding editor.
Execute	Transfer the TAL into module TAL-Processing.
<b>SVT-Filter</b>	

Filter	Filter the SVT tree in according to the selected transaction.
SVT Reset	Reset SVT-Actual/Soll and delete SVT tree.

### **Displaying of Default Entries in SVT**

If a SVT is read out from the vehicle, it may not be possible to determine the current software status (SVK = system installation identifier) of some ECUs. Such ECUs are included in the SVT as default entries.

If default ECUs are contained in the SVT read out, this is announced to the user in a dialog and the affected ECUs are marked accordingly in the SVT tree.



### **Displaying of Smart Actuators**

If a SVT gets loaded or read, which includes Smart Actuators, so these Smart Actuators will get displayed inside the related Smart Actuator Master Ecu Node. SVK's from a Smart Actuator Master ECU are getting highlighted by a red "S" instead of a dot.

### **SVT-Update**

Within section 'SVT-Update' it is possible to update all versions of the SGBMIDs included in the Target-SVT.

With button 'Update SVT' E-Sys will start a check for each SGBMID contained in the SVT and search for versions of that SGBMID in the loaded PDA-Container.

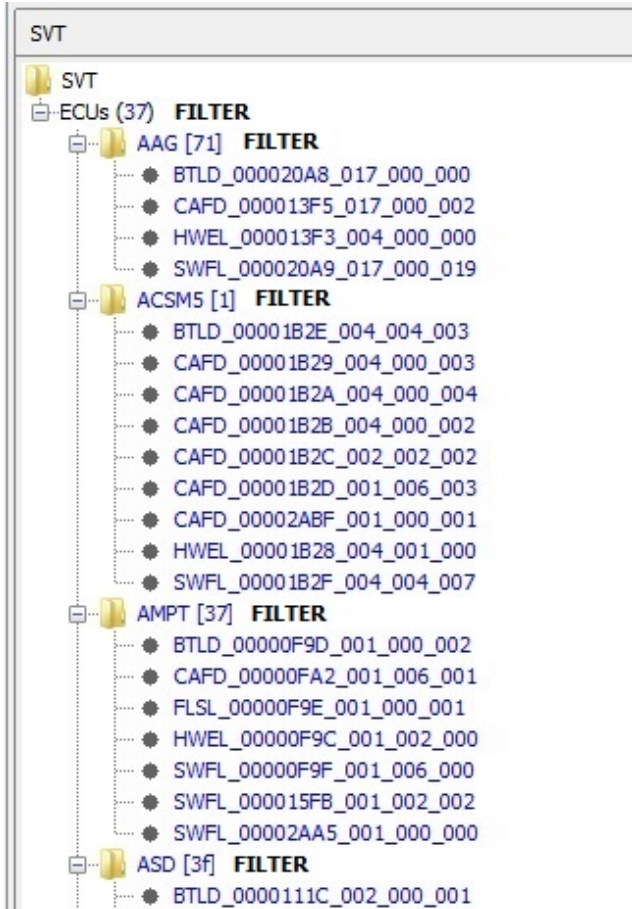
If a different version of a SGBMID was found, the version in the SVT will be updated by the one from the PDA-container.

### **Reading Vehicle Order and VIN**

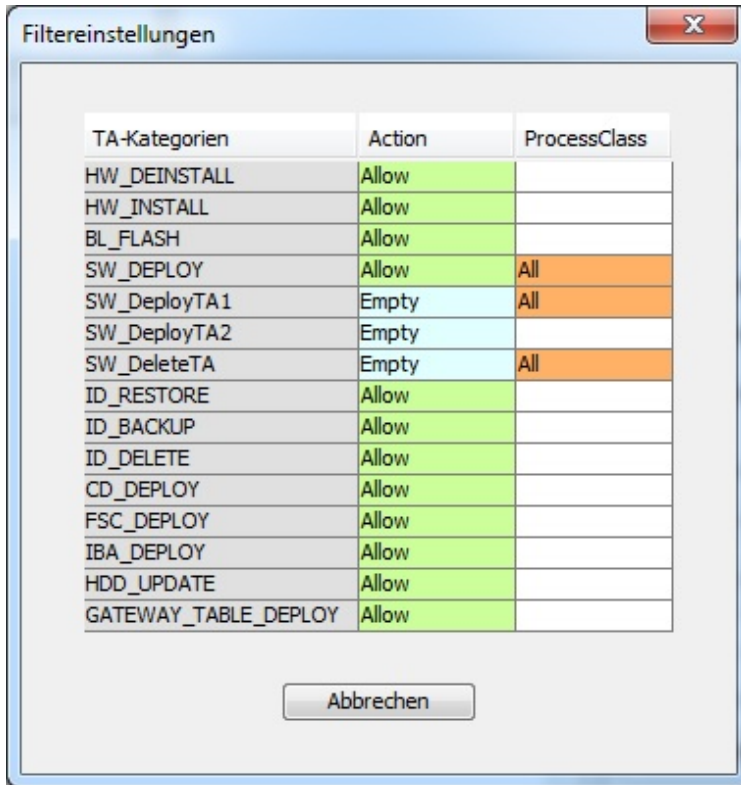
After reading the vehicle order the VIN will always be read too. The VIN will be shown in the element ""FAList/FA/FZAuftrag/Header/Vinlong". The vehicle order and the VIN will be read from the master ECU. If an error occurs they will be read from the backup ECU. In the status bar is shown where the vehicle order and the VIN were actually read from.

### **Displaying TAL-Filter settings**

If filter settings have been defined for the root element or any of the ECU elements in the SVT tree view, these elements will be displayed with an additional filter symbol as soon as the TAL-Filter file is loaded.



A detailed figure showing all filter settings of an element will be displayed via the context menu entry 'Filter Settings'.



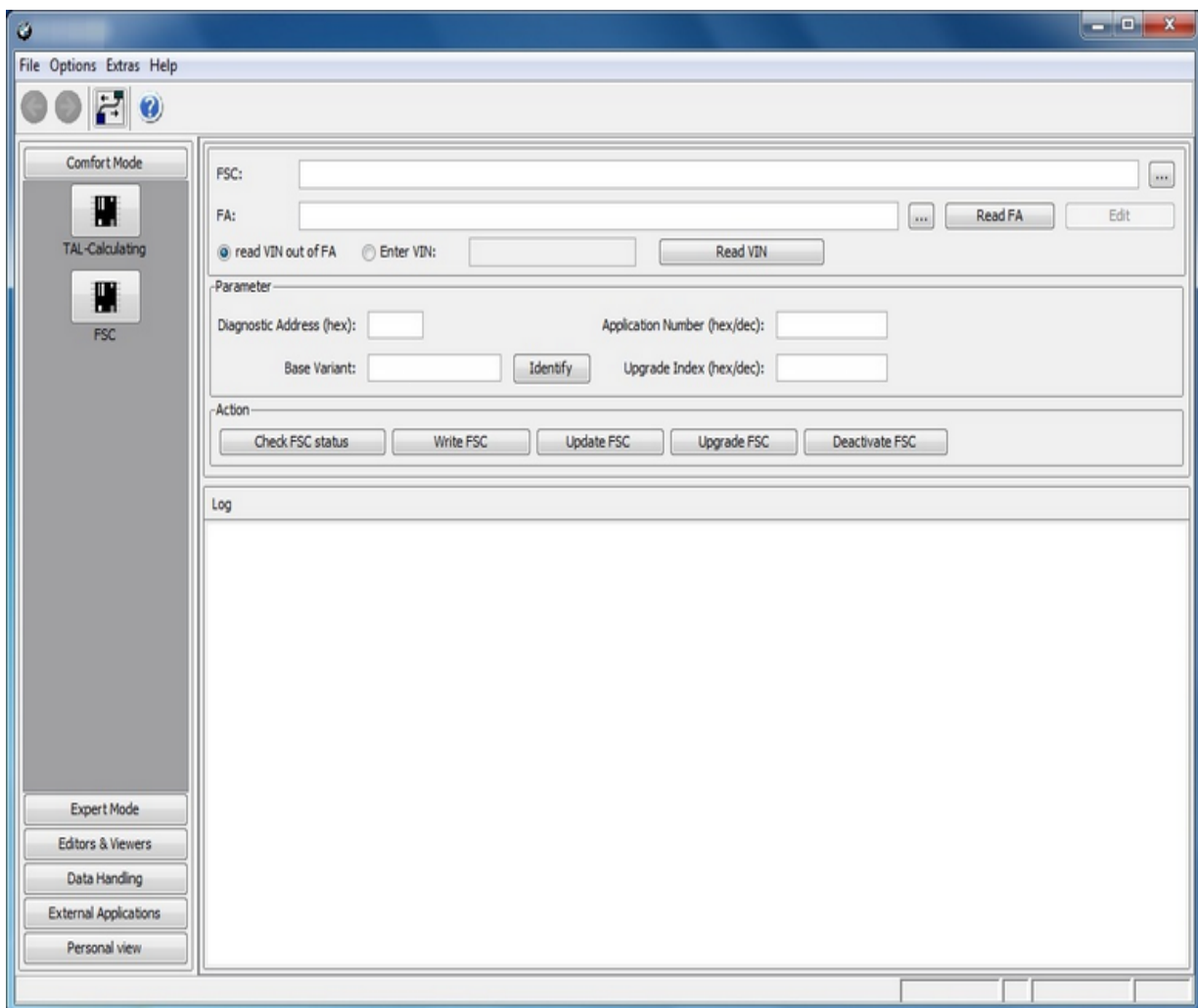
---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.



The module FSC serves for inquiring the FSC's state and for writing of FSC into ECU. In addition to that it's possible to (de)activate FSC's already existing in the ECU. Only an activated FSC can really be used.

Corresponding to the topic FSC there are other masks in E-Sys: FSC's can be displayed and edited in **FSC-Editor** easily. In expert mode **FSC-Extended-Module** is available in addition to that.



## GUI Elements

FSC File	The FSC to be written.

FA File	the vehicle order corresponding to the ECU.
Diagnosis Address	of ECU
Software-ID	Identifier of software
Base Variant	Name of ECU variant
Upgrade Index	of FSC
Poll FSC State	The State of FSC in the ECU is inquired.
Write FSC	The FSC loaded into E-Sys is written to the ECU.
Update FSC	The written FSC (to ECU) is checked and activated.
Upgrade FSC	Combines write FSC and update FSC.
Deactivate FSC	Deactivate FSC in ECU (set to invalid)

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

### **ATTENTION**

The functionalities provided in this module which include direct server access require to have several preconditions to be fulfilled:

- The system which runs E-Sys must be in a BMW network segment in which the backend systems (CBB) are available.
- There must be a valid E-Sys client Certificate imported into E-Sys.
- At least one URL of a valid backend system (CBB) must be configured in E-Sys.

A more detailed set of instructions can be found here:

[HOWTO](#) (only available within BMW network).

For further questions please contact the support service:

[support-security-systemfunktionen@bmw.de](mailto:support-security-systemfunktionen@bmw.de)

### **Introduction:**

As an enabler for new functions and to increase the security of existing functions, individual Certificates are required in the treatment of vehicles. From SP2018, individual Certificates and their binding to the respective vehicle must be installed both in the factory and in the service. In addition, bindings between ECU devices must be distributed. This functionality is provided by this module.

### **Overview Certificate Management Module at Comfort-Mode:**

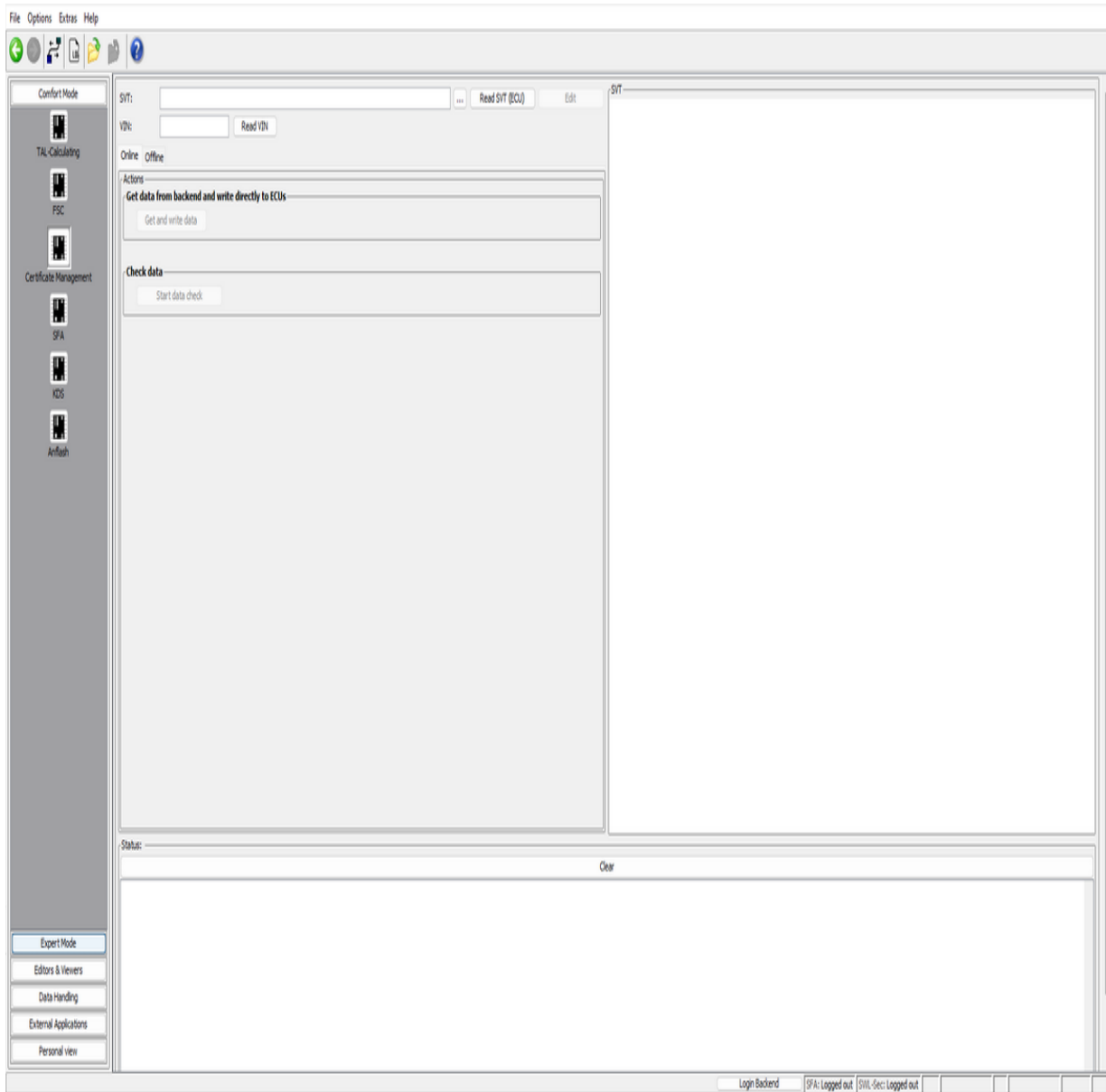
The comfort Certificate management module now offers easier usability of the standard Certificate management functions.

For this purpose, you always need an SVT, which contains CertEnabled ECUs. [ **Note:** This is recognizable for every ECU on the CertEnabled = "true" attribute in the ecuStatusInfo. The Emus that are CertEnabled are shown in the SVT-View on the right.]

In this mode, the entire vehicle is always treated. Therefore, no filters (blacklist or whitelist) can be used.

This module has 2 tabs, which are explained in the following sections:

### **Online:**



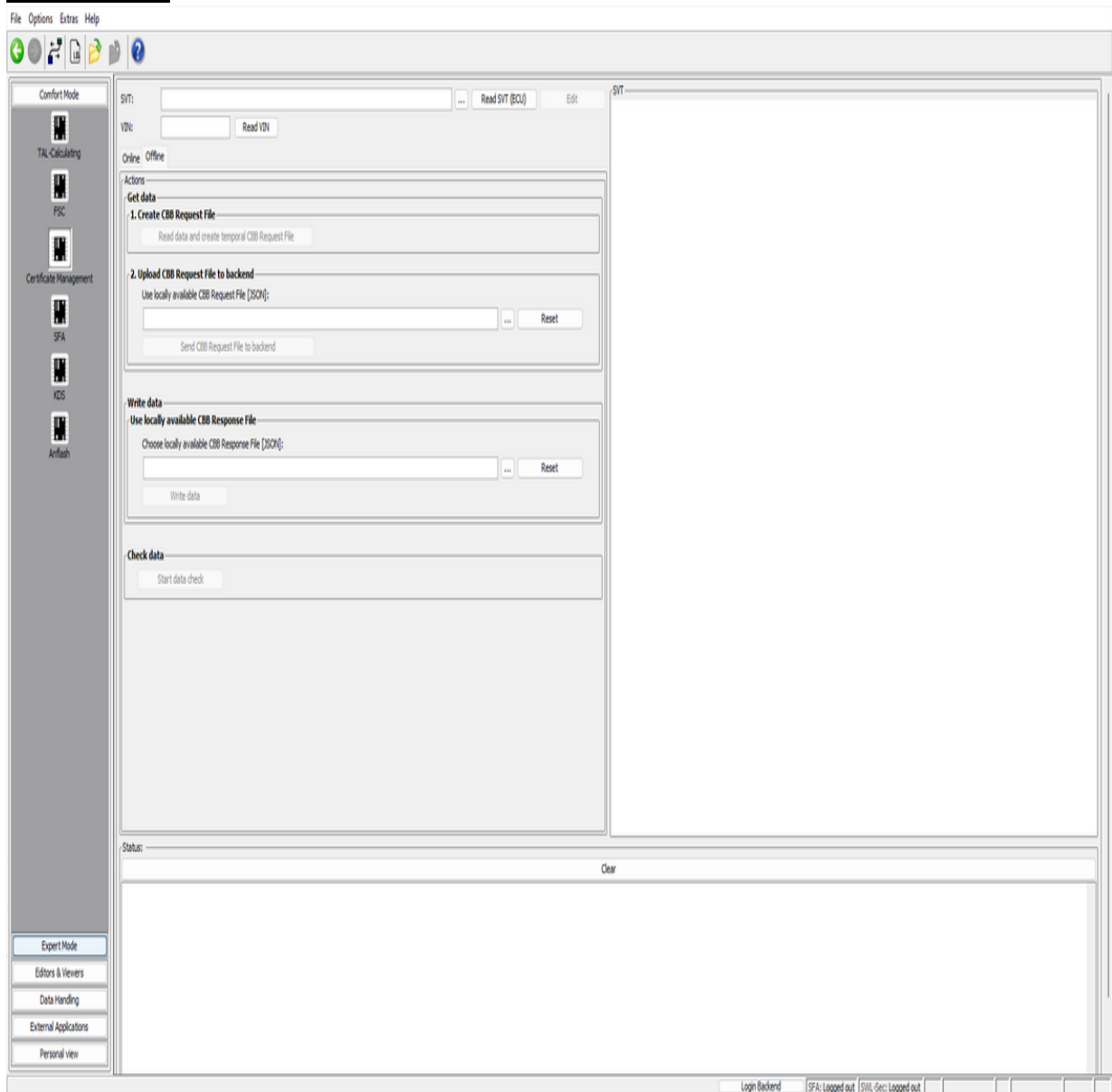
## **Write Data:**

Gets data from the backend and writes it directly into the vehicle.

## **Check Data:**

Starts the check routine of the ECUs and fetches its results after a certain period of time to ensure that all results can be fetched. The tree view will be enriched with the results of the verification.

## **Offline:**



## **Get Data:**

### **Step 1: Create CSR Request File**

Creates a file in JSON format with the individual Certificate requests (CertificateRequest) that the user either sends with E-Sys to the Backend (only possible if tester is connected to BMW-intranet) or use the B2B Interface.

How to get access to the B2B-Interface is described at the [HOWTO](#) as well.

### **Step 2: Upload Request File To Backend**

Uploads the selected request file to the backend system (CBB) and stores the response file locally.

### **Write Data:**

Writes the locally stored response file into the vehicle.

### **Check Data:**

Starts the check routine of the Emus and fetches their results after a waiting time which is the maximum of the Emus to ensure all final check results can be fetched. The tree view will be enriched with the check results.

### **TLS:**



Comfort Mode

TAL-Calculating

FSC

Certificate Managem...

SFA

KDS

Anflash

SecOC

Expert Mode

Editors & Viewers

Data Handling

External Applications

Personal view

SVT:  ... Read SVT... Edit

VIN:  Read VIN

Online Offline TLS

Actions

**Add TLS Certificate**

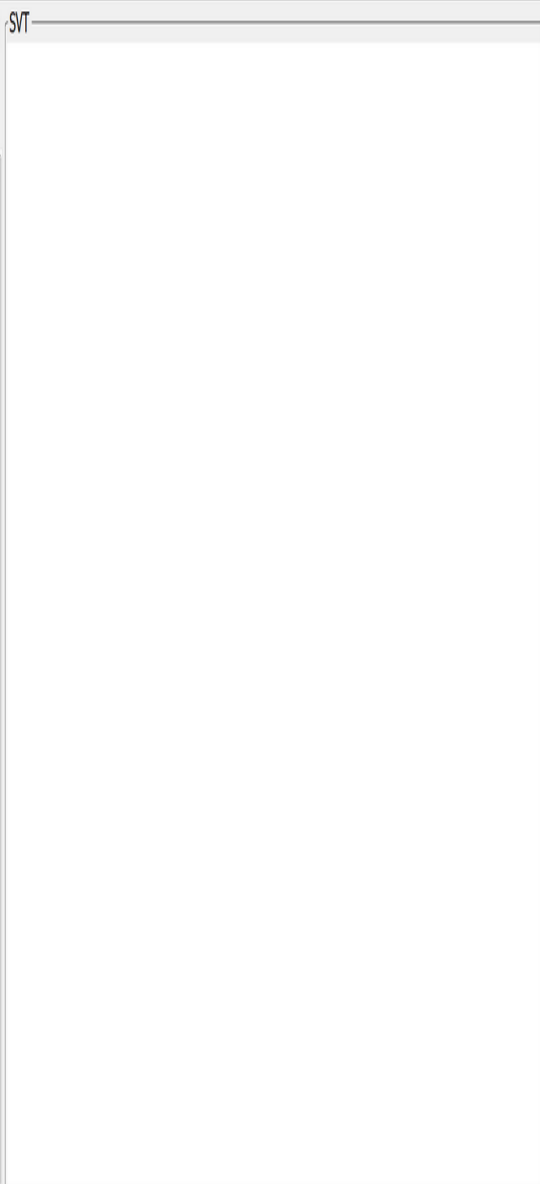
Use locally available TLS Certificate [crt]:  
 ... Reset

Use locally available TLS Certificate Key [key]:  
 ... Reset

Import Certificate

**Manage Certificates**

Check Certificates Delete expired Certificates



Status:

Clear



### **Add TLS Certificate:**

Adds the content of the selected certificate file and key file to the key store. Both files needs to contain the vehicle VIN and it needs to be the same in both vehicles.

### **Manage Certificates:**

#### **Check Certificates:**

Prints all TLS certificates' VIN and expiration date to the logs.

#### **Delete Expired Certificates:**

Deletes all expired certificates from the TLS Key Store.

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

### **Introduction:**

The module empowers the user on the one hand to switch all Emus, which supports Secure ECU Modes, between the three states (Engineering, Plant, Field).

On the other hand the user can activate/deactivate and read the current secure feature status of all Emus, which are supporting these system function.

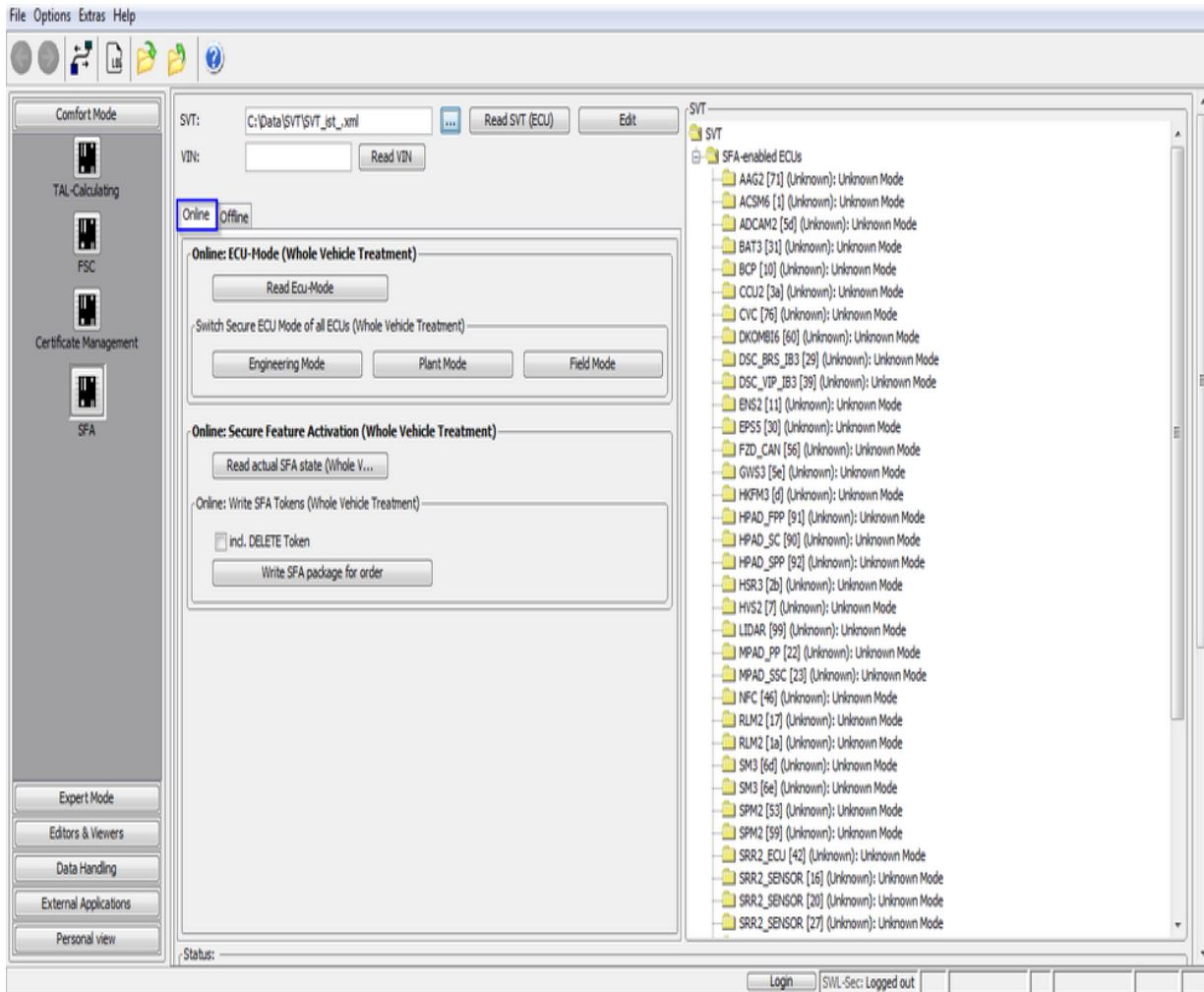
The Modes Plant and Engineering are only accessible with a valid secure token an the ECU has to be in den Field-Mode.

To switch back into the Field-Mode is no token necessary.

All Emus that are contained in the svt are visualized on the right side in the svt-tree-view.

[See 'Backend / SFA' if extended rights on SFA-Backend are required](#)

### **Online-tap:**

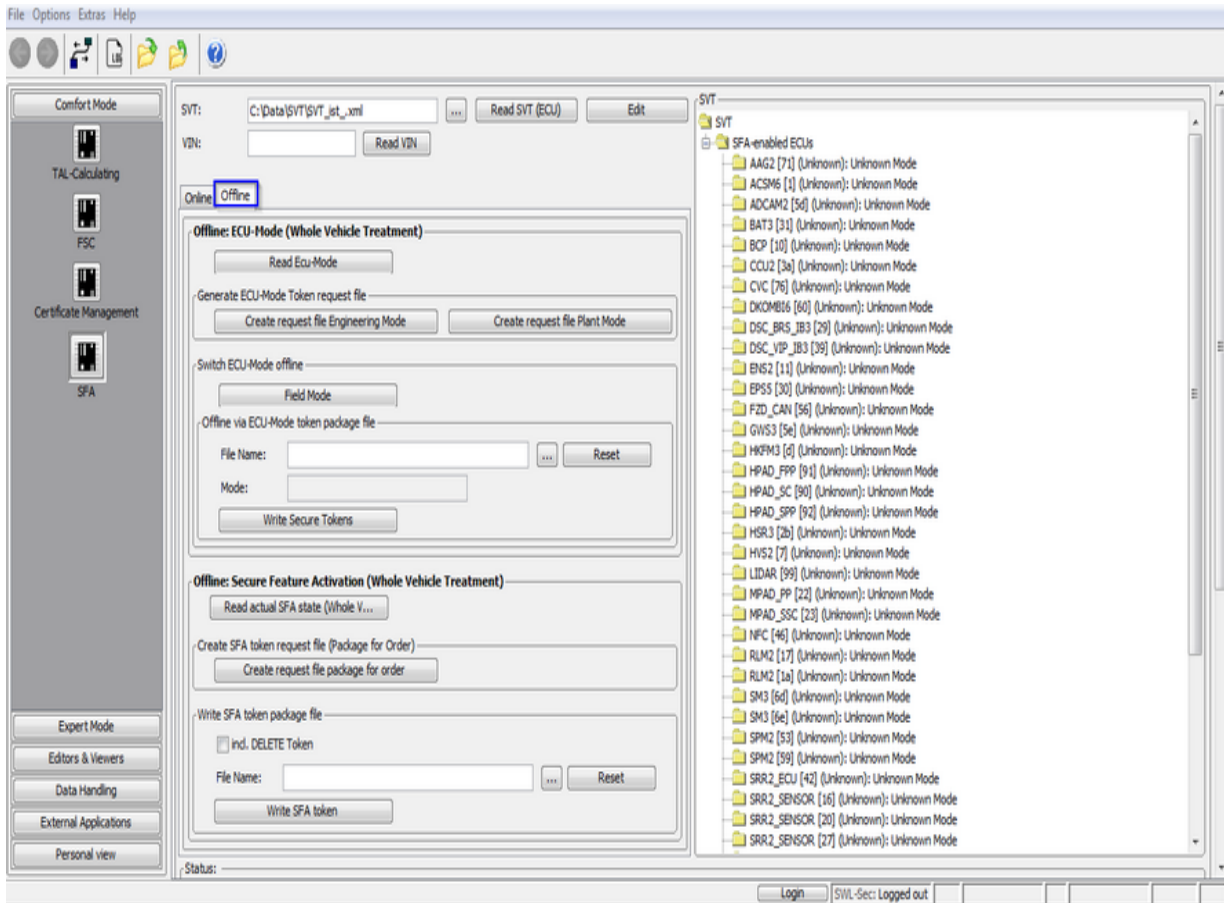


	Description:	Hints:
Read / load SVT	Generates the current SVT of the entire vehicle. In the svt-tree view are only the Emus of the SVT displayed, that support SFA	
Read / Edit VIN	Read the VIN out of the vehicle. Alternative has the user the option to edit the VIN manually.	
Read ECU-Mode	Read the current ECU-Mode of the entire vehicle. The actual	

	ECU-Mode is shown in the svt-view behind each base variant.	
Switch Mode: Plant Mode	Generates a Plant-Mode-Token for all Emus, that provides ECU-Mode, sending the request to the Backend and writing the response at the Emus. The Emus are finally in the Plant-Mode	BMW-intranet connection is necessary
Switch Mode: Engineering Mode	Generates an Engineering-Mode-Token for all Emus, that provides ECU-Mode, sending the request to the Backend and writing the response at the Emus. The Emus are finally in the Engineering-Mode	BMW-intranet connection is necessary
Switch Mode: Field Mode	All Emus are switching into the Field-Mode	
Read SFA status	Read the current SFA-Status of each ECU is shown in the svt-tree view.	
Write SFA package for order token-set	Generates at the backend a new secure token set. In case of new calculation is active the tokens for the already active features, that should be	BMW-intranet connection is necessary

	<p>active based on the VIN, are going also to be generated.</p> <p>If the user wants only the features active, that are calculated to be active by the backup-System, "incl. DELETE Token" has be active. These means that it is possible that features which was active before are now deleted. To reactivate these features a new generated token is required.</p>	

**Offline-tap:**



	Description:	Hints:
Read / load SVT	Generates the current SVT of the entire vehicle. In the svt-tree view are only the Emus of the SVT displayed, that support SFA	
Read / Edit VIN	Read the VIN out of the vehicle. Alternative has the user the option to edit the VIN manually.	
Read ECU-Mode	Read the current ECU-Mode of the entire vehicle. The actual ECU-Mode is shown in the svt-view behind each base variant.	
Switch Mode: Field Mode	All Emus are switching into the Field-Mode	
Generates request	Generates a request token set for the Engineering Mod for all Emus and saved	

token set for Engineering Mode	local.	
Generates request token set for Plant Mode	Generates a request token set for the Plant Mod for all Emus and saved local.	
Write Secure Tokens:	Writes the chosen token package file for the ECU mode. The ECU-Mode of the token set is shown.	
Read SFA state	Read the current SFA-Status of each ECU is shown in the svt-tree view.	
Generate "package for Order" request	Generates a request file and save it locally. These file has to be sent manually to the backend where the secure token set ist generated.	
Write SFA token	Write the chosen SFA token set file. During the writing of the token set has the user the option do delete all additional features (feature is not included in the package for order token set).	

**Color representation of SFA features in the SVT tree view:**

In order to determine the status of all SFA features (activated, deactivated etc.) for each ECU shown in the SVT tree view, the button "Read actual SFA state" must be activated.

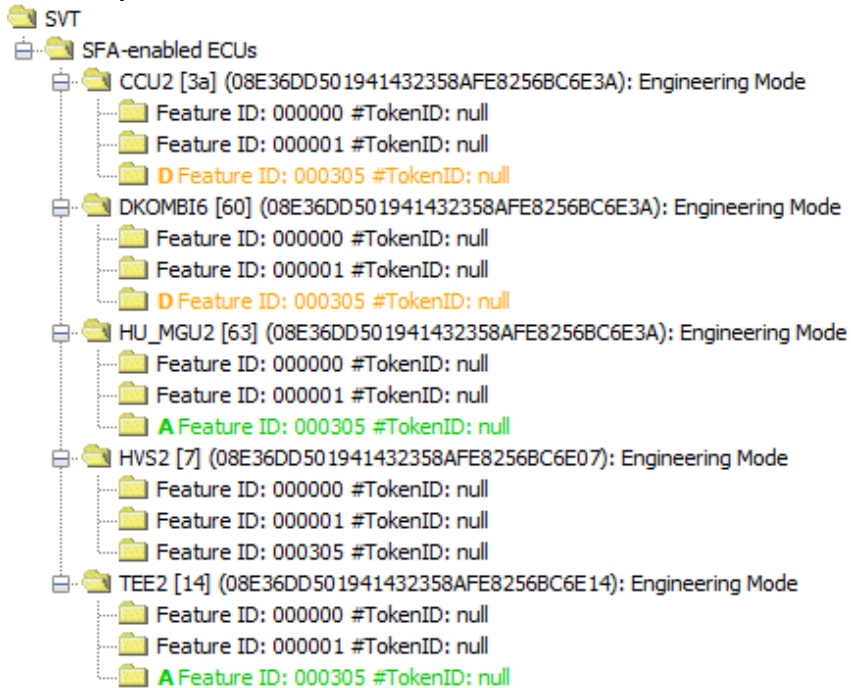
This triggers communication with every SFA-capable ECU and its features are read out.

The SVT tree structure is then updated in such a way that each ECU node can be opened up and all available features

can be displayed.

Each feature is highlighted in color in the SVT tree view.

### Example:



Note: a character is added to the beginning of the Feature ID string to help color blind users identify the state

### Color marks and characters meaning:

**Green** and character A - the Feature\_ID is **activated**

**Orange** and character D - the Feature\_ID is **deactivated** or **expired**

**Red** and character E - the Feature\_ID is **incorrect**

**Black** and No character - the Feature\_ID is initially **not active**, i.e. no secure token available

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved.



This message will not appear if you compile this help file with the registered version of Help & Manual.

## **Introduction:**

KDS (anti-theft protection 2.0) based on the secure token sets (see SFA / Secure ECU-Mode).

The functionalities provided in this module which include direct server access require to have several preconditions to be fulfilled:

- The system which runs E-Sys must be in a BMW network segment in which the backend systems (SFA-Backend) are available.
- There must be a valid E-Sys client Certificate imported into E-Sys.
- At least one URL of a valid backend system (SFA-Backend) must be configured in E-Sys.

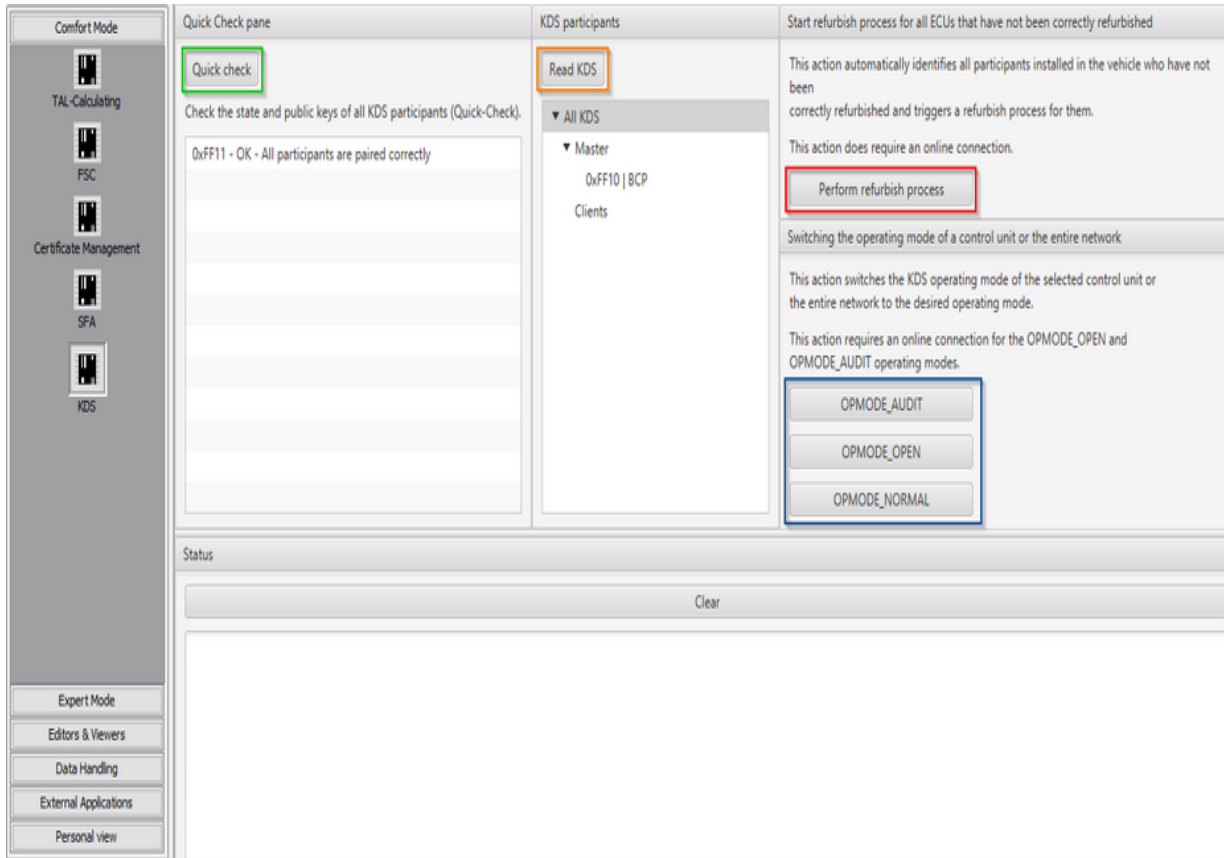
A more detailed set of instructions can be found here:

[HOWTO](#) (only available within BMW network).

## **Notifications:**

These module supports only the processing of the entire vehicle.

For more KDS functions or the selection of specific Emus please switch to the expert-mode ([KDS - Extended](#))



- **Quick check (marked green):**

Starts the KDS system check. The result is visualized in the GUI and has one of the following values:

KDS quick check results
MASTER_OK_CLIENT_OK
MASTER_OK_CLIENT_INVALID
MASTER_INVALID_CLIENT_OK
MASTER_INVALID_CLIENT_INVALID
ERROR_CLIENT_NOT_PAISED

- **Read KDS (marked orange):**

Read the entire KDS system and visualize the system. The system is separated into the KDS-master and the KDS-clients tap.

- **Perform refurbish process (marked red): [Only with**

## **BMW-Intranet connection possible]**

Identifies all participants in the KDS system, that have not been correctly paired and is pairing the entire KDS system.

- **Switch the operation mode (marked blue):**

Change the operation mode for the chose Emus. If no ECU is selected the entire KDS system is going to switch.

The switch into the modes AUDIT and OPEN requires a BMW-Intranet connection and the valid client-backend-Certificate (E-Sys [options](#) / authentication)

---

**This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.**

The Anflash module offers the possibility to perform a complete flash cycle in a fully automated way.

The structure of the required configuration files is described in a separate manual. The required master-configuration file can be created or edited with the [ANFLASH-Editor](#).

A prerequisite for the execution of the Anflash process is a valid vehicle connection. In addition, a check is made before the Anflash process to ensure that all the required configuration files exist.

After a successful start, all ECU configs defined in the master config are processed sequentially in the order of their position numbers.

During each execution, the current process step is displayed.

The Anflash process can be aborted during execution via the Stop button.

At the end of the execution, both the overall status and the status of individual ECU configurations are displayed.

In addition, report files are generated.

This is done on two levels, firstly for each executed ECU-Config and secondly for the entire Anflash process.

The report files are stored in the E-Sys Data directory below the "Anflash-Reports" directory.

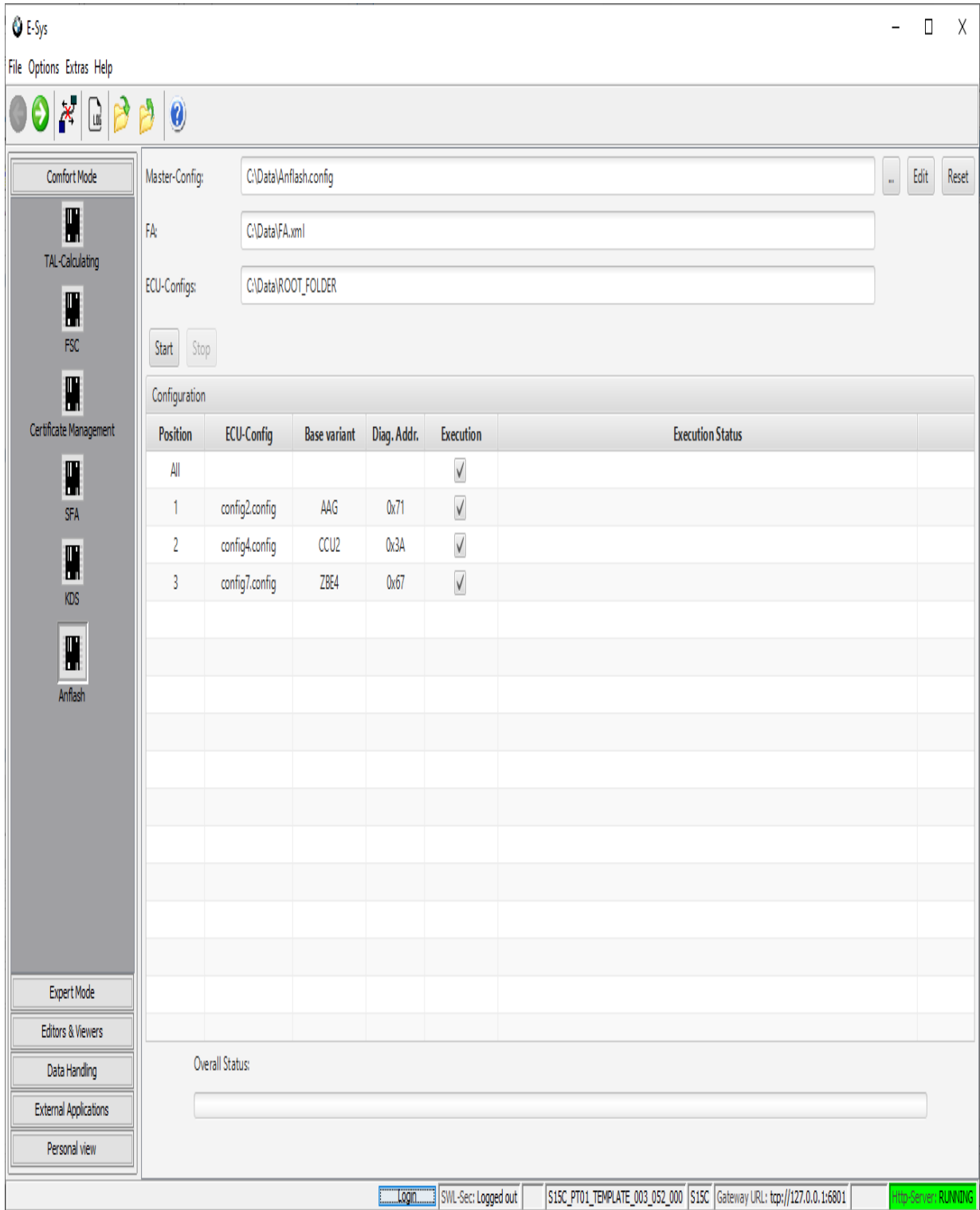
## **Note:**

Before starting the Anflash process, a BAT/HAF verification is carried out.

It will be checked whether a BAT/HAF control unit is installed.

If this is the case, the status of switch S1 is determined.

When the switch is open, a message appears informing the user about possible risks for the Anflash process.



## GUI Elements

Master-Config	
Textfield	Absolute path to the Master-Config

Load	<p>Opens a file dialogue to load an existing Master-Config from a file. The ECU-Configs from the Master-Config will then be visible in the Configuration-View, the file path is displayed in the text field.</p> <p>An already loaded/opened file will be discarded.</p>
Edit	<p>Edit in the corresponding editor.</p> <p>This button is only active if there is an file is open and no Anflash process has been started.</p>
Reset	<p>Unload the active filter together with all corresponding symbols in the Table-View.</p> <p>This button is only active if there is an file is open and no Anflash process has been started.</p>
<b>FA</b>	
Textfield	<p>Absolute path to the vehicle order (FA).</p> <p>An absolute path is always displayed, even if it is specified as a relative path in the configuration file.</p>
<b>ECU-Configs</b>	
Textfield	<p>Absolute path to the ROOT directory where all ECU-configuration files are located.</p> <p>An absolute path is always displayed, even if it is specified as a relative path in the configuration file.</p>
<b>Execution</b>	
Start	Start of Anflash process.
Stop	<p>Abort the Anflash process.</p> <p>PDX-Import: The current import process must be completed.</p> <p>TAL-Execution: After the current processed TALline the processing is aborted..</p>

<b>Configuration table</b>	
Columns	List the existing ECU-Configs for the loaded Master-configuration file.
Rows	List the Data of each ECU-Config. The Execution on ECU-Config can be activated or deactivated by checkbox.
<b>Statusanzeige</b>	
Status label	Visualization of the currently step of Anflash process. Example: ECU-STATUS CCU2 [0x3a]: PDX-Import (1/3) - PT01_Container_1.pdx At the end of the execution, the overall status is displayed.
Progress bar	Visualization of progress of an Anflash process. In the case of a TAL Execution, the remaining duration is also displayed.

The screenshot shows a software interface with a 'Start' and 'Stop' button at the top left. Below the buttons is a 'Configuration' section containing a table with the following data:

Position	ECU-Config	Base variant	Diag. Addr.	Execution	Execution Status
All				<input checked="" type="checkbox"/>	
1	SRR_43.config	SRR	0x43	<input checked="" type="checkbox"/>	In progress ...
2	SRR_44.config	SRR	0x44	<input checked="" type="checkbox"/>	

Below the table, the text 'ECU-Status SRR [0x43]: PDX-Import (1/1) - PT01\_template.003\_055\_000.pdx' is displayed above a progress bar that is approximately 50% filled with blue. At the bottom of the interface, there is a status bar with the following elements: 'Logout', 'SSL-Enc. CERTIFICATE', '\$15C\_PT01\_TEMPLATE\_003\_052\_000', '\$15C', 'Gateway URL: tcp://127.0.0.1:6801', and 'Http-Server: RUNNING'.

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

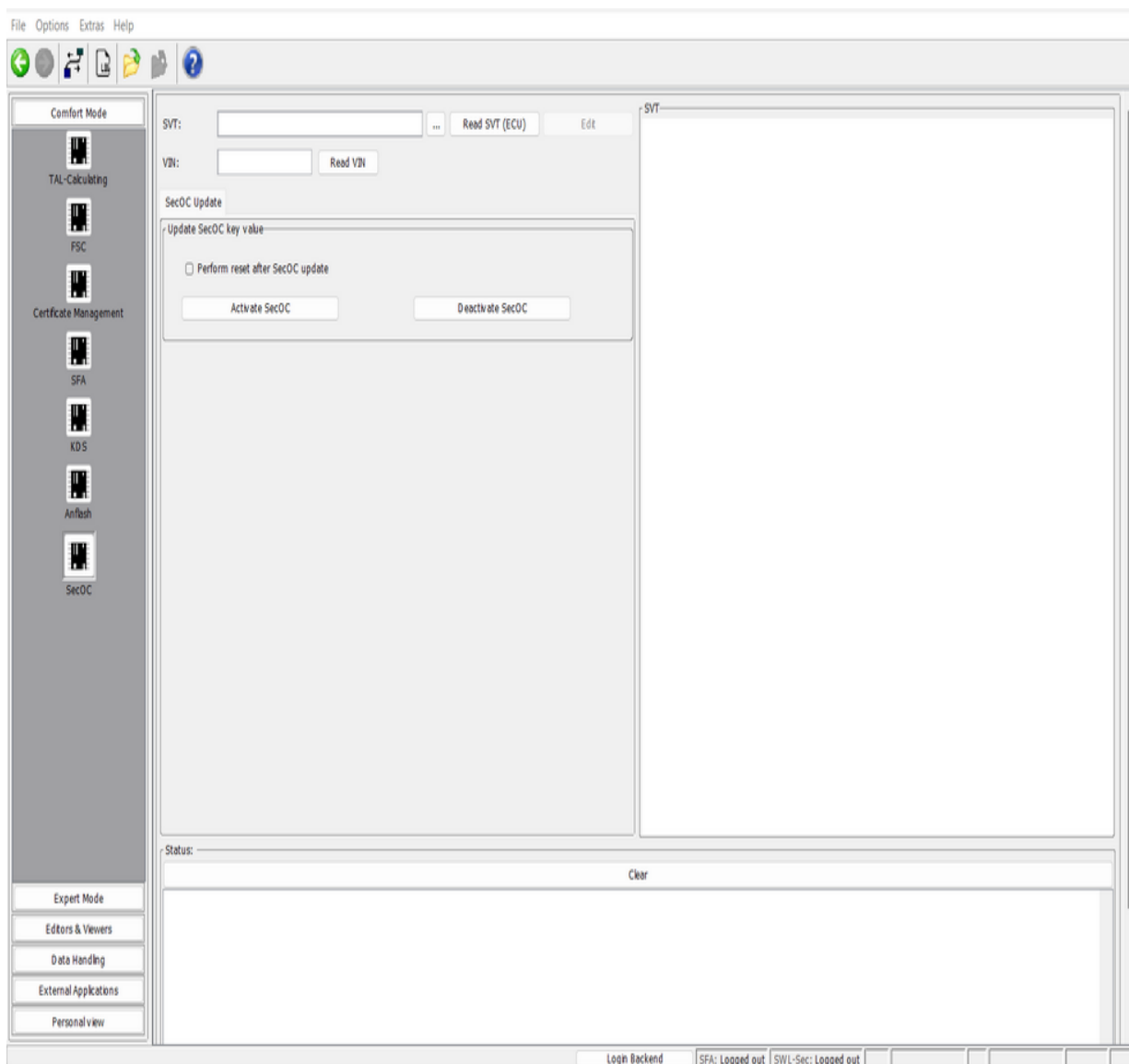


The SecOC module enables the updating of the Security Onboard Communication (SecOC) for a specified group of Electronic Control Units (ECUs). Moreover, it provides the capability to update and reset all ECUs following the update.

The ECUs available for selection are determined by an SVT that can be directly accessed through the connection or imported.

Upon reading the SVT, you have the option to select a single ECU or multiple ECUs simultaneously from a provided set.

Once the ECU(s) is chosen, you can activate or deactivate the SecOC keys by selecting the corresponding button. Additionally, you have the ability to perform a reset on all selected ECUs following the update.



## **GUI Elements**

Read / load SVT	Generates the current SVT of the entire vehicle. In the svt-tree view are only the Emus of the SVT displayed.
Read / Edit VIN	Read the VIN out of the vehicle. Alternative has the user the option to edit the VIN manually.
Reset ECU(s)	Allow to reset to all selected ECU(s) following the update.
Activate SecOC	Allow to activate SecOC keys to all selected ECU(s)
Deactivate SecOC	Allow to deactivate SecOC keys to all selected ECU(s)
Tree View	Allow to select the ECU(s).

This help file was created with an unregistered evaluation copy of Help & Manual.  
© EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

The Programming Protection module allows the user to read the programming protection status for a set of selected ECUs.

The ECUs available for selection are determined by an SVT that can be directly accessed through the connection or imported.

Upon reading the SVT, you have the option to select a single ECU or multiple ECUs simultaneously from a provided set. If no selection is done, all ECUs displayed at the SVT tree will be considered.

Once the ECU(s) is chosen, you can query the respective programming status by selecting the corresponding button.



- Comfort Mode
  - TAL-Calculating
  - FSC
  - Certificate Management
  - SFA
  - KDS
  - Anflash
  - SecOC
  - Programming Protection
- Expert Mode
- Editors & Viewers
- Data Handling
- External Applications
- Personal view

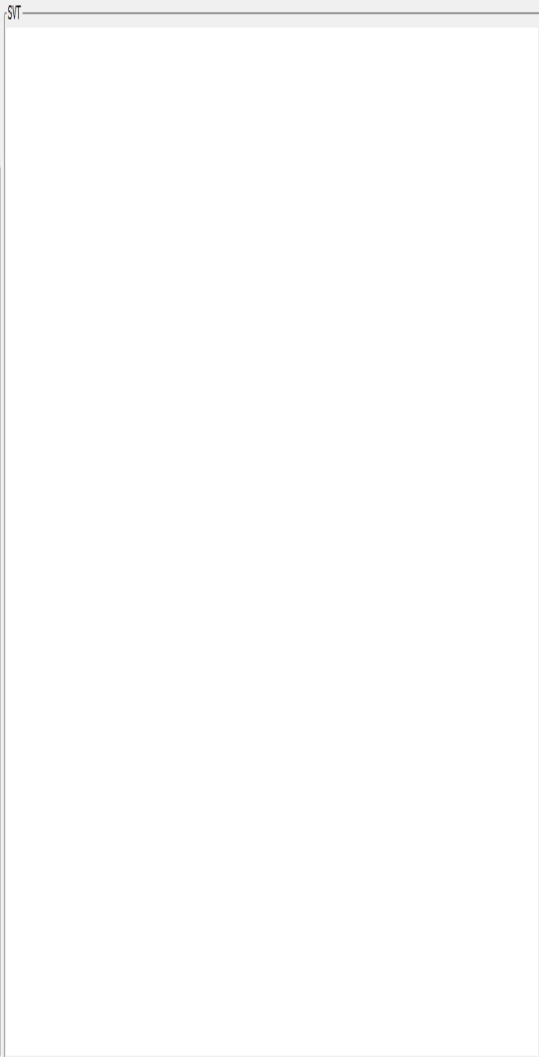
SIV:  ... Read SIV (ECU) Edit

VIN:  Read VIN

Programming protection status

Read programming protection status

Read programming protection status



Status:

Clear



## **GUI Elements**

Read / load SVT	Generates the current SVT of the entire vehicle. In the svt-tree view are only the Emus of the SVT displayed.
Read / Edit VIN	Read the VIN out of the vehicle. Alternative has the user the option to edit the VIN manually.
Read programming protection status	Allows to read the programming protection status to all selected ECU(s)
Tree View	Allows to select the ECU(s).

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

[FA/FP-Editor](#)

[TAL-Editor](#)

[TAL-Filer-Editor](#)

[SVT-Editor](#)

[SWESEQ-Editor](#)

[FSC-Editor](#)

[FDL-Editor](#)

[ANFLASH-Editor](#)

[CAF-Viewer](#)

[LOG-Viewer](#)

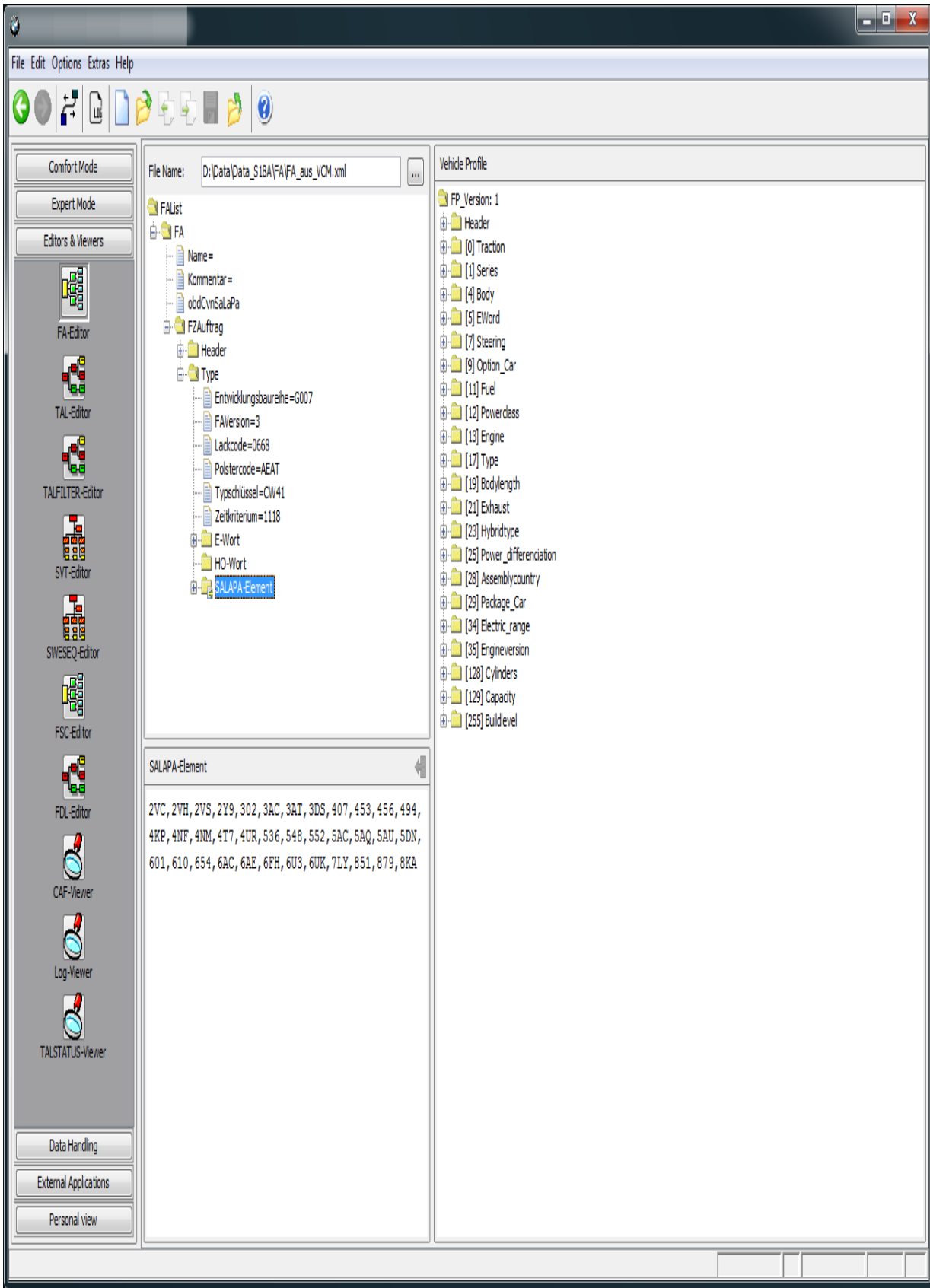
[TALSTATUS-Viewer](#)

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

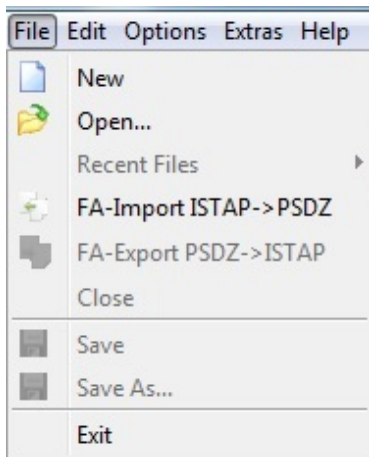
The module FA/FP-Editor makes for display and edition of FAs (vehicle jobs) and the correspondent FPs (vehicle orders). The FA designates a discrete state of installation of SWEs and HWEs for a vehicle. You can derive the vehicle profiles from the vehicle order.

An ECU or a vehicle can contain/save exactly one vehicle order. During the communication with an ECU or with an vehicle over PSdZ always only a single vehicle order is exchanged. But in the local file system a vehicle order always is saved within a list consisting of any number of vehicle jobs marked by an unique ID.

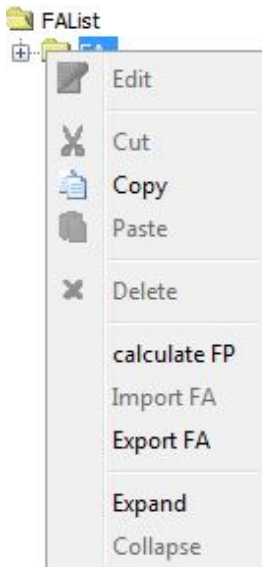




## **File Menu**



## **Context Menu**



## **Specifics**

- The editors subdivides in two areas. On the left side is the FA list's view. If you select in the left window a FA by the context menu you retrieve the FP. The calculated FP is displayed on the right hand side of the window.
- To display a single FA you have to create an empty FA list and import by context menu the desired FA.
- A single FA can be imported also by context menu.

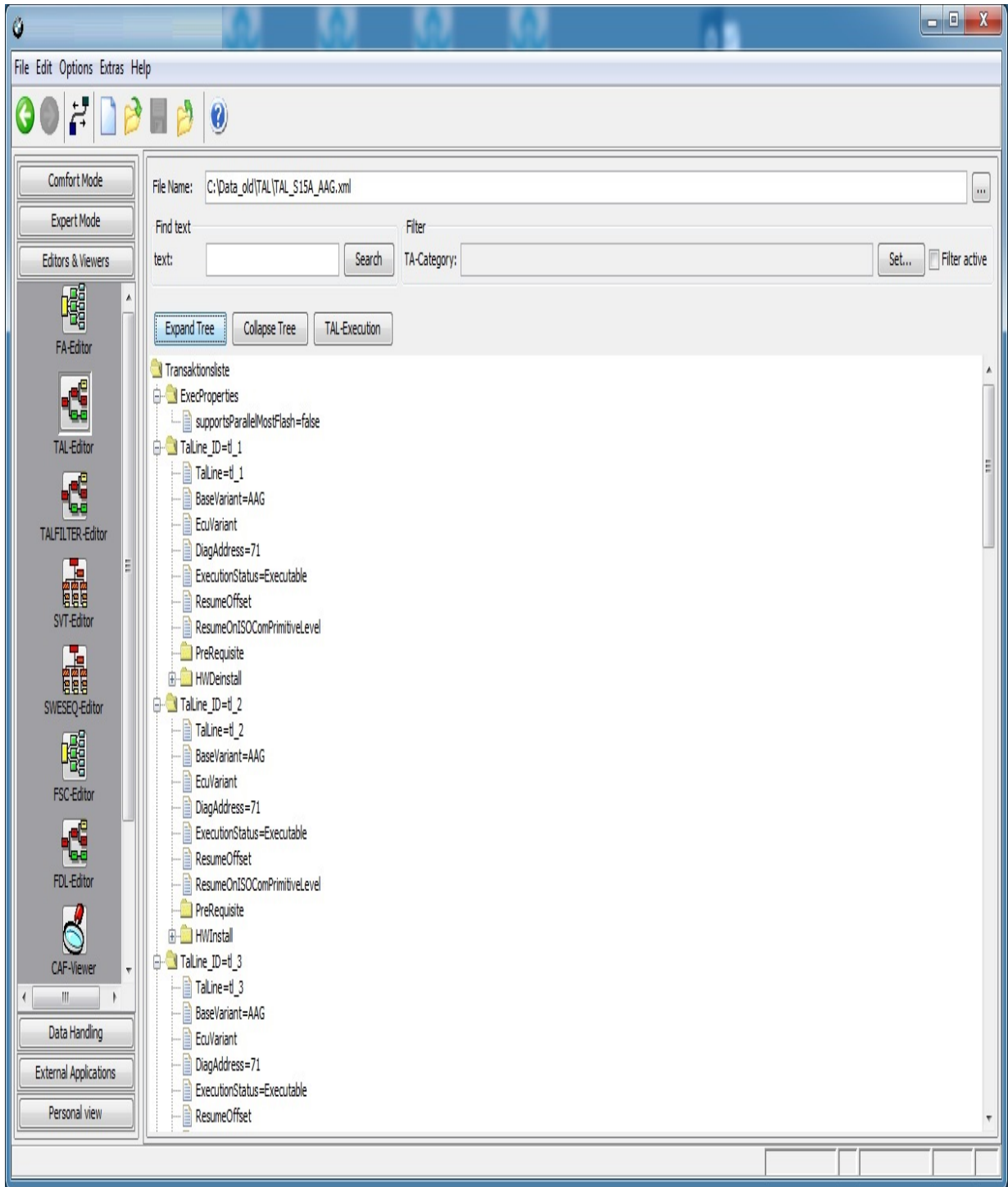
- A single FA can be exported also in ISTAP-Format using the File menu.
- A single ISTAP-Format FA can also be imported using the File menu.

---

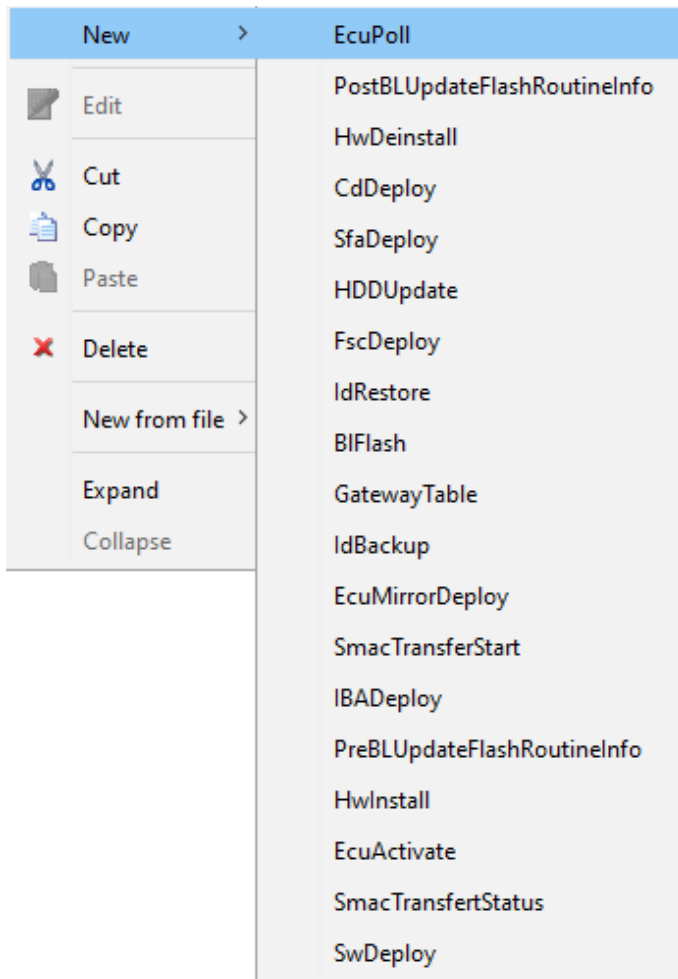
This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

With the TAL-Editor you can display and edit transaction lists (TALs). A TAL is a model year plan to transform an ECU, a system of Emus or the entire vehicle from the state corresponding to the actual system installation table to the state corresponding to the reference system installation table. The PSdZ generates the TAL from a pair of SVT-Actual and SVT-Target.

The TAL contains a list of actions sorted by ECU name, e.g. dismount HWE, program SWE, save individual data, etc. After processing this list the vehicle is in the state SVT-Target.



## Context-Menu

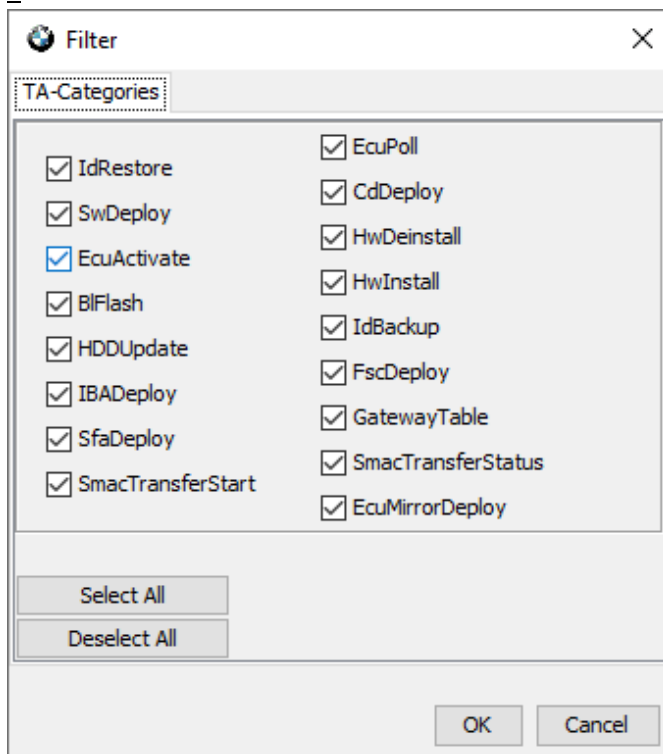


## **GUI-Elements**

Expand Tree	Open the whole tree up to the first level
Collapse Tree	Close the whole tree including sub trees.
TAL-Execution	Transfer TAL to module TAL-Processing and open it
Find Text	Allows searching for text in the TAL tree view. By pressing the "Search" button, all detected text elements will be marked one after another in the tree view.
Filter	By using checkbox "filter active", you en/disable the filter. By the button "Set..." you switch to a dialog for choosing the particular elements to be

affected by the filter.

## **Filter-Settings**



---

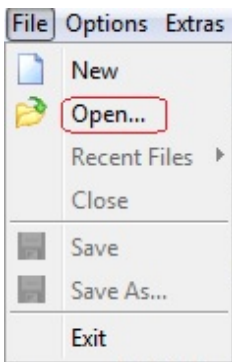
**This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.**

Module TALFILTER-Editor offers the possibility to display and modify TAL-Filters.

## **Creation of new TAL-Filter**

There are two ways of creating a new filter file:

### 1. with File Menu



This will create an empty TAL-Filter with a content like the one shown below:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<talfilter>
  <allEcu/>
</talfilter>
```

### 2. with SVT-File

-

File Name: File generated, not saved yet.

Svt: C:\Data\SVT\SVT\_Test1.xml

ID-Base	Set All to...	Set All to...	Set All to...	Set All to...	Specify...	Specify...	hwDeinstal	hwInstall	ecActivate	ecPol	ecMirrorDeploy	biFlash	svDeploy	smacTra...	smacTra...	idRestore	idBackup	cdDeploy	fsDeploy	SftDeploy	lbaDeploy	hddUp...	gatew...
ecuAll	Empty	Allow	Prohibit	Force	svDeploy	ecMirrorDeploy	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty
Dx10 [16]	Empty	Allow	Prohibit	Force	svDeploy	ecMirrorDeploy	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty
Dx08 [8]	Empty	Allow	Prohibit	Force	svDeploy	ecMirrorDeploy	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty
Dx6F [111]	Empty	Allow	Prohibit	Force	svDeploy	ecMirrorDeploy	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty
Dx56 [86]	Empty	Allow	Prohibit	Force	svDeploy	ecMirrorDeploy	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty
Dx07 [7]	Empty	Allow	Prohibit	Force	svDeploy	ecMirrorDeploy	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty
Dx76 [66]	Empty	Allow	Prohibit	Force	svDeploy	ecMirrorDeploy	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty
Dx76 [118]	Empty	Allow	Prohibit	Force	svDeploy	ecMirrorDeploy	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty
Dx28 [40]	Empty	Allow	Prohibit	Force	svDeploy	ecMirrorDeploy	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty
Dx3A [58]	Empty	Allow	Prohibit	Force	svDeploy	ecMirrorDeploy	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty
Dx43 [67]	Empty	Allow	Prohibit	Force	svDeploy	ecMirrorDeploy	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty
Dx44 [68]	Empty	Allow	Prohibit	Force	svDeploy	ecMirrorDeploy	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty
Dx46 [70]	Empty	Allow	Prohibit	Force	svDeploy	ecMirrorDeploy	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty
Dx67 [103]	Empty	Allow	Prohibit	Force	svDeploy	ecMirrorDeploy	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty
Dx30 [48]	Empty	Allow	Prohibit	Force	svDeploy	ecMirrorDeploy	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty
Dx11 [17]	Empty	Allow	Prohibit	Force	svDeploy	ecMirrorDeploy	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty
Dx14 [20]	Empty	Allow	Prohibit	Force	svDeploy	ecMirrorDeploy	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty
Dx20 [32]	Empty	Allow	Prohibit	Force	svDeploy	ecMirrorDeploy	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty

After selecting an existing SVT file, a new TAL-Filter will be created when pressing button 'Calculation'. This TAL-Filter will contain empty entries for all Emus defined in the SVT file.

## **Editing and Displaying TAL-Filter settings**

As soon as a new TAL-Filter has been created or an existing filter has been loaded, its filter settings will be displayed in a table structure:





For modifying the filter settings the user is offered several options:

### 1. Set All to...

By using one of the buttons 'Empty', 'Allow', 'Prohibit' or 'Force' all Transaction Categories of an ECU can be set to the same value.

Additionally these four buttons located in the first row ('ecuAll'), will set all Transaction Categories of all Emus to the specified value.

### 2. Select Action

Each Transaction category of every ECU can separately be set to one of the following action values:

- Empty
- Allow
- Prohibit
- Force

#### *Note:*

For Transaction category swDeploy it is possible to specify user defined settings for swDeployTA, swDeleteTA and smacDeployOnMasterTA.

Therefore an additional action value ('user defined') will be available for that category.

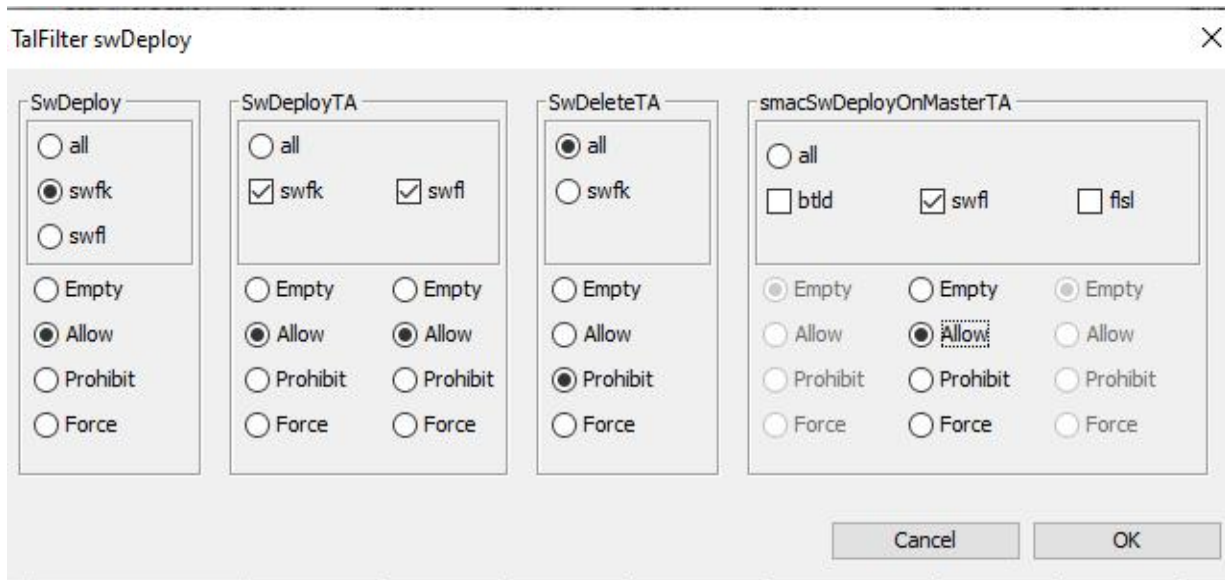
For Transaction category ecuMirrorDeploy it is possible to specify user defined settings for ecuMirrorDeployTA and smacEcuMirrorDeployOnMasterTA.

Therefore an additional action value ('user defined') will also be available for that category.

### 3. swDeploy

As mentioned before, it is possible to define additional filter settings for category swDeployTA, swDeleteTA and smacSwDeployOnMasterTA.

This can be done by opening an additional dialog via the swDeploy button:



### 4. ecuMirrorDeploy

As mentioned before, it is possible to define additional filter settings for category ecuMirrorDeployTA and smacEcuMirrorDeployOnMasterTA.

This can be done by opening an additional dialog via the ecuMirrorDeploy button:

EcuMirrorDeploy	EcuMirrorDeployTA	smacEcuMirrorDeployOnMasterTA
<input checked="" type="radio"/> all <input type="radio"/> btd <input type="radio"/> swfi <input type="radio"/> swfk <input type="radio"/> fsl <input type="radio"/> ibad <input type="radio"/> navd <input type="radio"/> entd  <input checked="" type="radio"/> Empty <input type="radio"/> Allow <input type="radio"/> Prohibit <input type="radio"/> Force	<input type="radio"/> all <input checked="" type="checkbox"/> btd <input checked="" type="checkbox"/> swfi <input type="checkbox"/> swfk <input checked="" type="checkbox"/> fsl <input checked="" type="checkbox"/> ibad <input type="checkbox"/> navd <input checked="" type="checkbox"/> entd  <input type="radio"/> Empty <input type="radio"/> Empty <input type="radio"/> Empty <input type="radio"/> Empty <input type="radio"/> Empty <input type="radio"/> Empty <input type="radio"/> Empty <input checked="" type="radio"/> Allow <input checked="" type="radio"/> Allow <input type="radio"/> Allow <input checked="" type="radio"/> Allow <input checked="" type="radio"/> Allow <input type="radio"/> Allow <input checked="" type="radio"/> Allow <input type="radio"/> Prohibit <input type="radio"/> Prohibit <input type="radio"/> Prohibit <input type="radio"/> Prohibit <input type="radio"/> Prohibit <input type="radio"/> Prohibit <input type="radio"/> Prohibit <input type="radio"/> Force <input type="radio"/> Force <input type="radio"/> Force <input type="radio"/> Force <input type="radio"/> Force <input type="radio"/> Force <input type="radio"/> Force	<input type="radio"/> all <input type="checkbox"/> btd <input checked="" type="checkbox"/> swfi <input checked="" type="checkbox"/> fsl  <input type="radio"/> Empty <input type="radio"/> Empty <input type="radio"/> Empty <input type="radio"/> Allow <input checked="" type="radio"/> Allow <input checked="" type="radio"/> Allow <input type="radio"/> Prohibit <input type="radio"/> Prohibit <input type="radio"/> Prohibit <input type="radio"/> Force <input type="radio"/> Force <input type="radio"/> Force

---

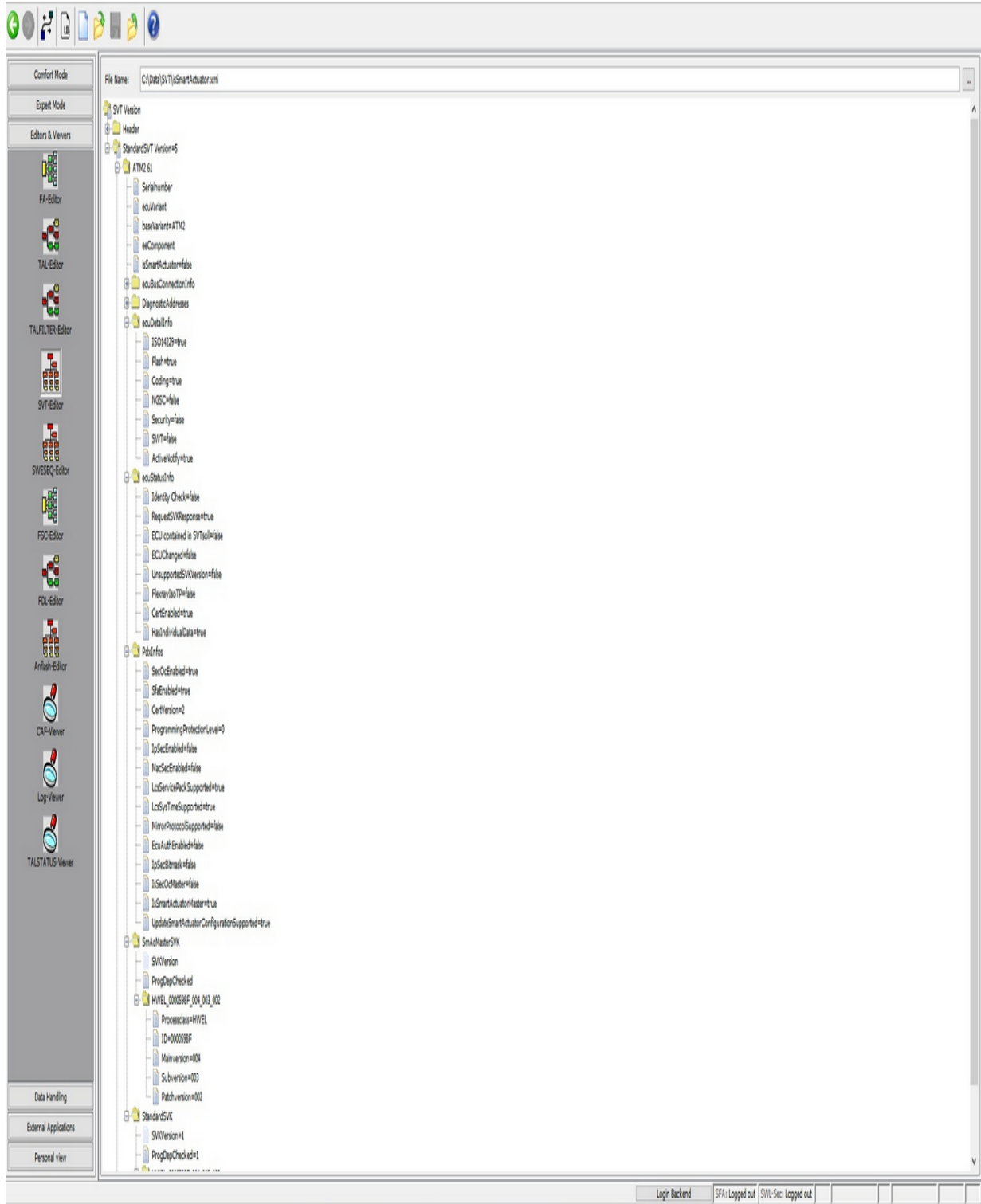
This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

With SVT-Editor you can display and edit SVTs (system installation tables). An SVT reflects a discrete state of a ECU or vehicle.

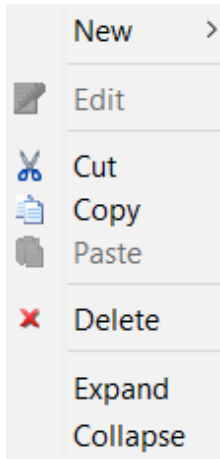
You have to differentiate between an SVT-Actual which describes the actual state of the vehicle and an SVT-Target which describes the desired state of a vehicle. Trivially the actual state is given and the reference state is created by the user or can be imported from the KIS.

The PSdZ can generate a TAL based on SVT-Actual and SVT-Target. This TAL contains the action list to transform the vehicle from actual to reference state.

**Hint:** Whereas an SVT contains the state for a vehicle in the SVK (system installation identifier) only the state of a single ECU is recorded. Therefore an SVT consists at least of one SVK. Since an SVK can be displayed as an SVT with a single SVK no separate SVK-Editor exists.

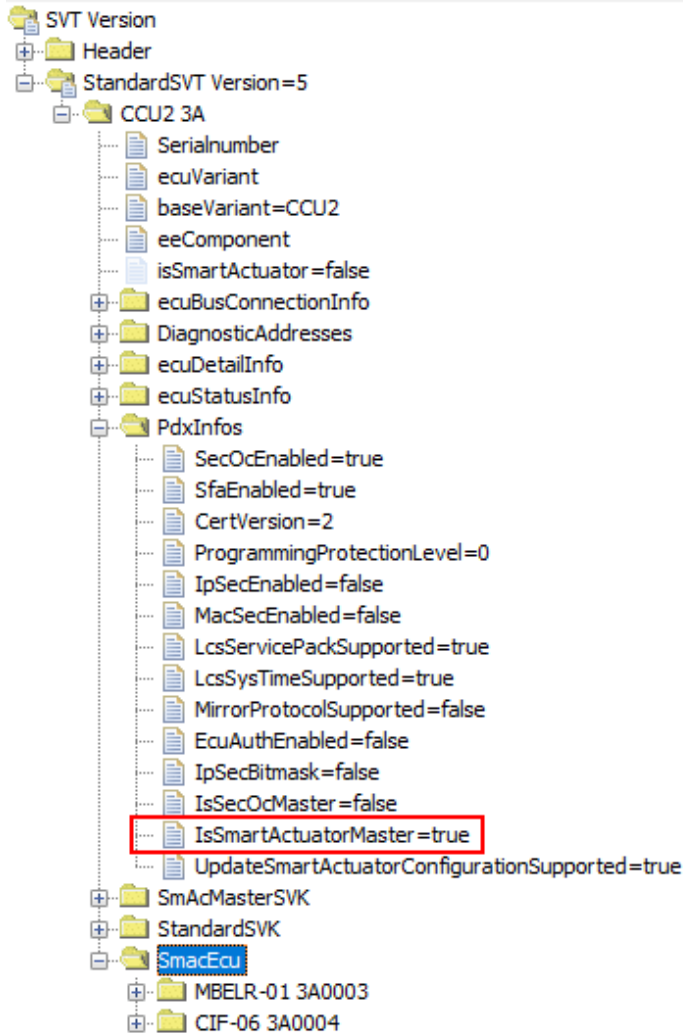


## Context-Menu



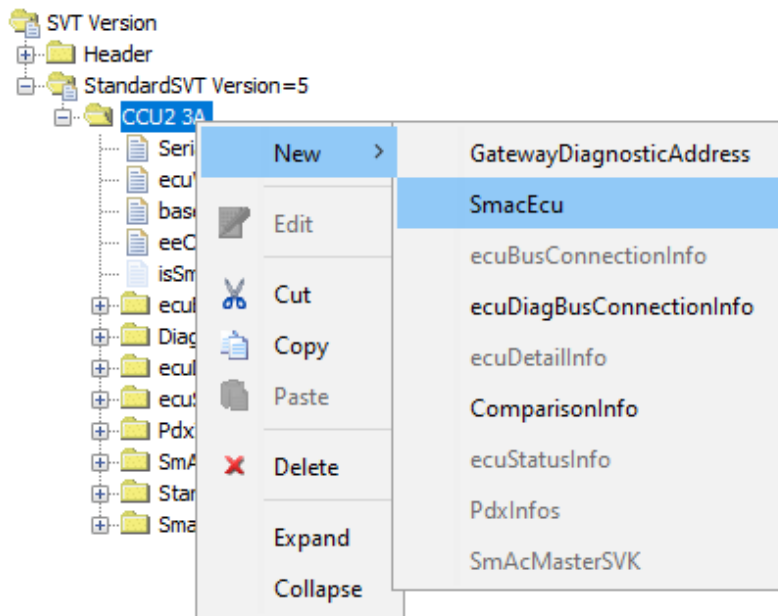
## **Smart Actuators in SVT**

– Although Smart Actuators (SMACs) in SVT are on the same level as normal ECUs, they are represented in a more user-friendly way in the SVT Editor. All SMACs are assigned to the related Smart Actuator Master (SMAC Master) to allow a clearer representation of the mapping.

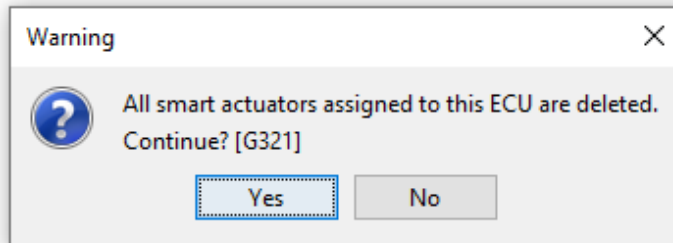
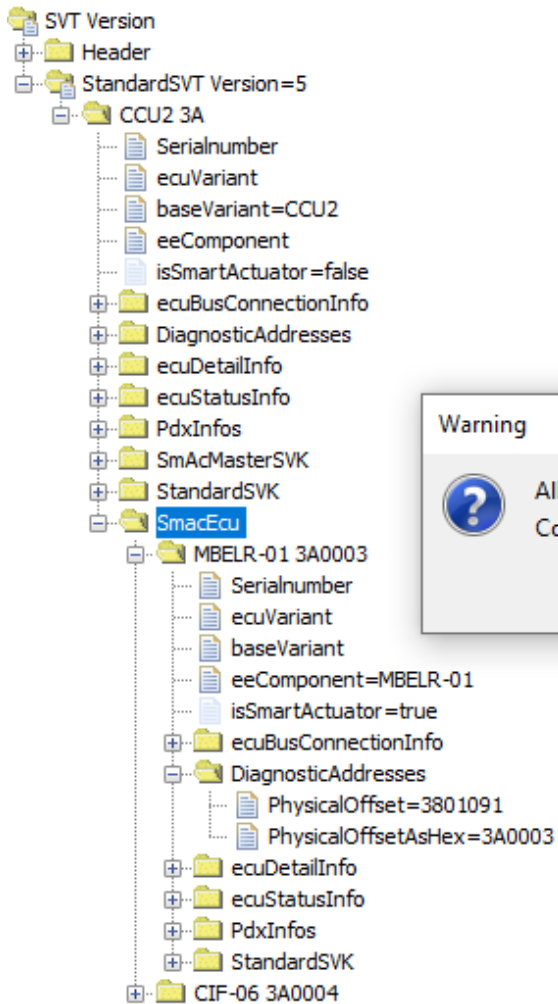


Adding new SMAC nodes is done via the item in the context menu of the respective SMAC master. The new SMAC node is automatically assigned a valid diagnostic address when it is added (the first 2 digits correspond to the diagnostic address of the SMAC master).





If a change to the master results in SMACs being removed from the SVT, a confirmation prompt appears via a dialog to the user, who can either confirm or cancel this.



---

This help file was created with an unregistered evaluation copy of Help & Manual.  
© EC Software. All rights reserved. This message will not appear if you compile  
this help file with the registered version of Help & Manual.

Module SWESQ Editor offers displaying and modification of a SWE Sequence.

E-Sys

File Edit Options Extras Help

File Name: C:\E-Sys\_Data\S18A\Etc\sweseq.xml

- SWESEQ
  - Version
  - EcuDependencies
    - EVALBOA...
      - BootloaderId=01020304
      - Ecu
        - BaseVariant=EVALBOARD
        - PhysicalOffset=7E
      - PreConditions
        - 21222324
      - Dependors
        - 11121314

Comfort Mode

Expert Mode

Editors & Viewers

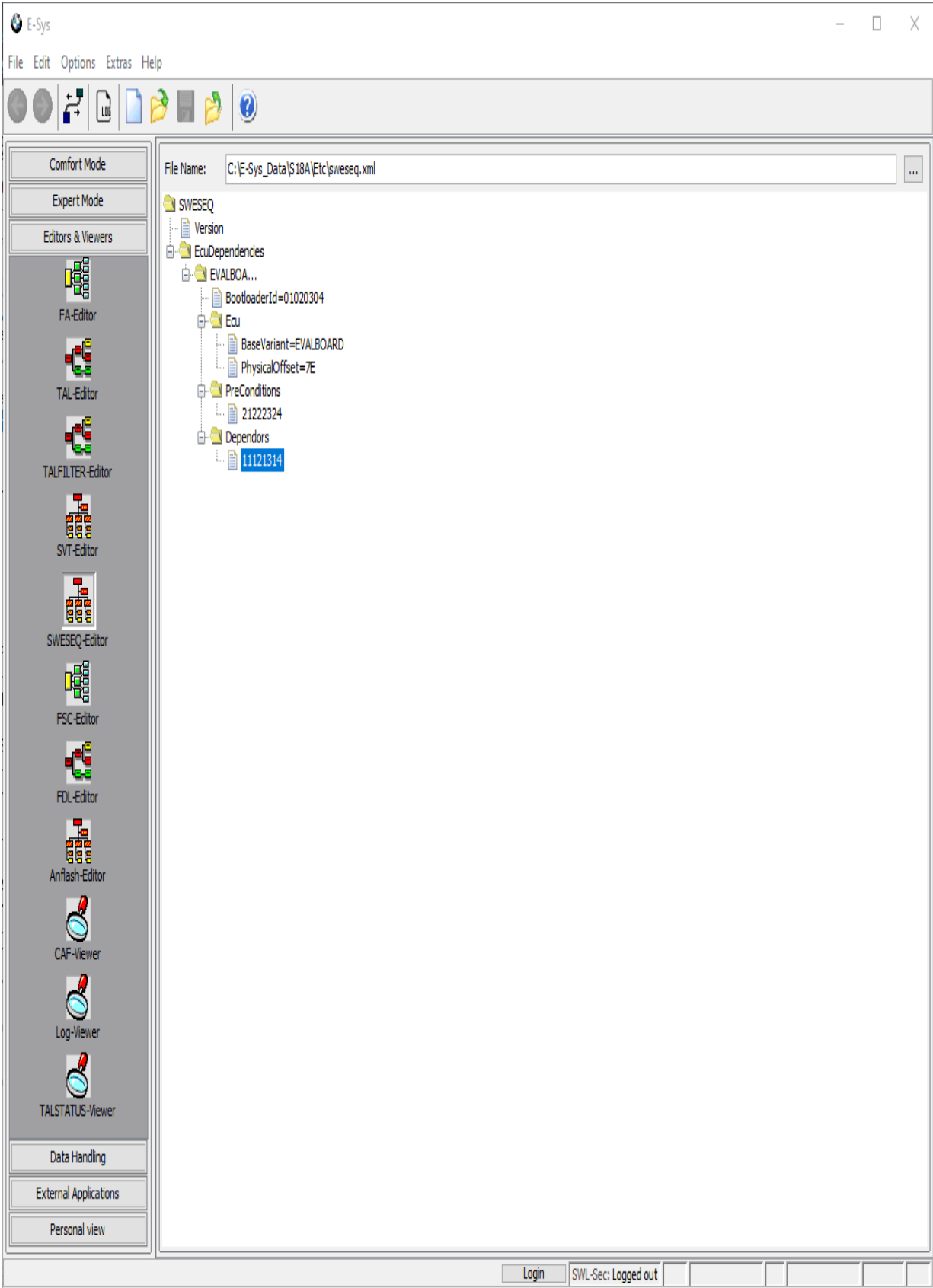
- FA-Editor
- TAL-Editor
- TALFILTER-Editor
- SVT-Editor
- SWESEQ-Editor
- FSC-Editor
- FDL-Editor
- Anflash-Editor
- CAF-Viewer
- Log-Viewer
- TALSTATUS-Viewer

Data Handling

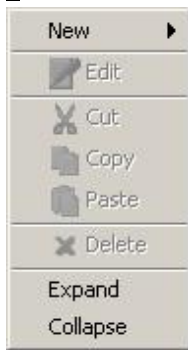
External Applications

Personal view

Login SWL-Sec: Logged out



## **Context Menu**



The menu options offers insertion of a new element, editing, cut, copy, paste, deletion, expanding or collapsing of an element within the SWE sequence.

## **GUI Elements**

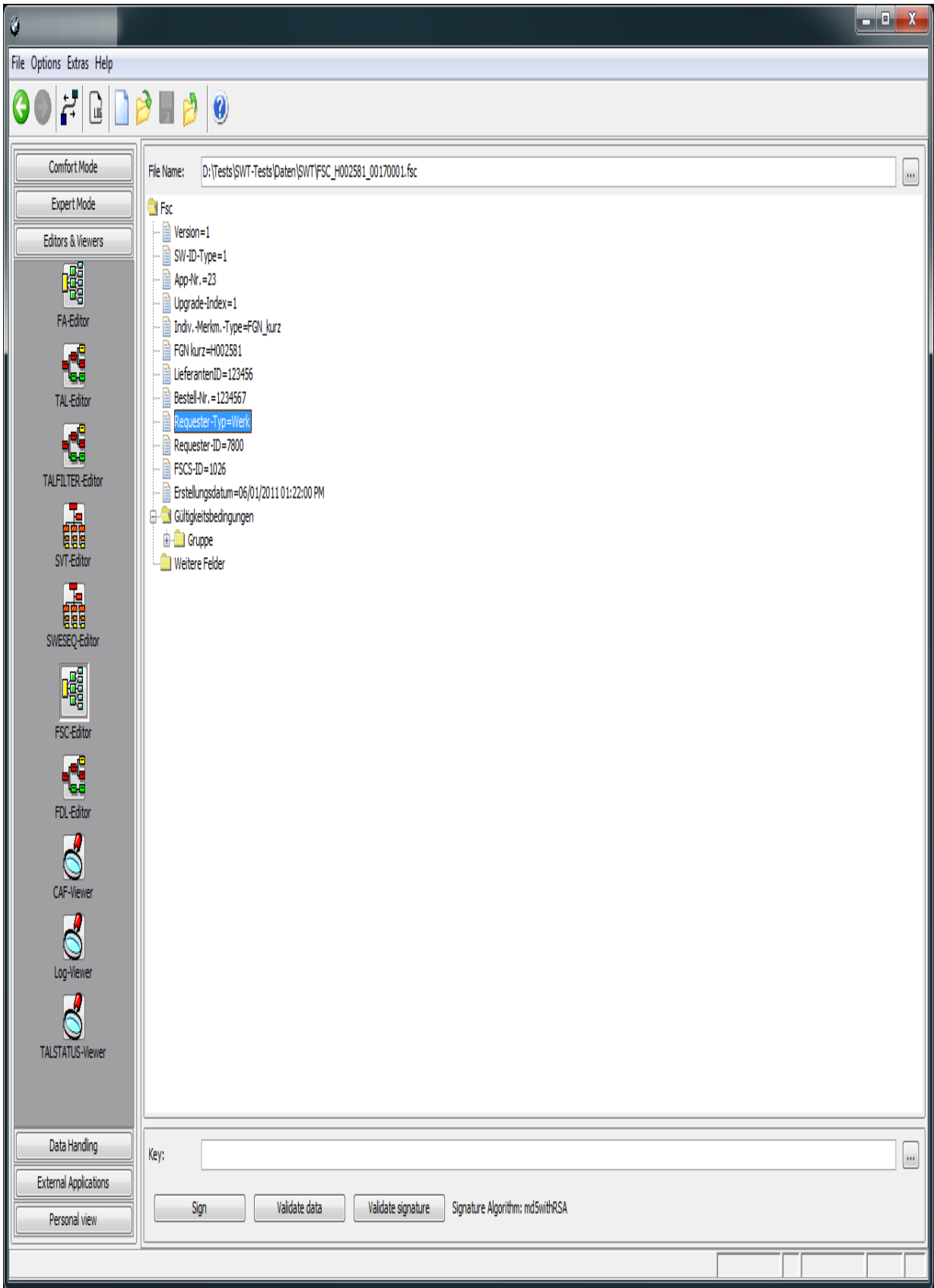
File Name    Currently opened file.  
" ... "        Button for opening a SWESEQ  
                 file.

---

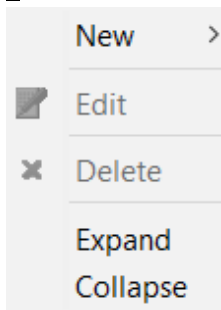
This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

Module FSC Editor serves for display and modification of FSC's.

Corresponding to the topic FSC there are other masks in E-Sys: Module [Freischaltcode](#) (FSC) serves for inquiring the FSC State, for writing and (de-)activating of FSC. In expert mode the [FSC-Extended-Module](#) is available in addition to that.



## **Context Menu**



The menu options serves for selection of a line to be edited. Or for insert of a new line or deletion of a line within the FSC.

## **GUI Elements**

FSC-File	Selection of the FSC to be displayed and (eventually) modified.
" ... "	Button for specifying the path to the FSC file.

---

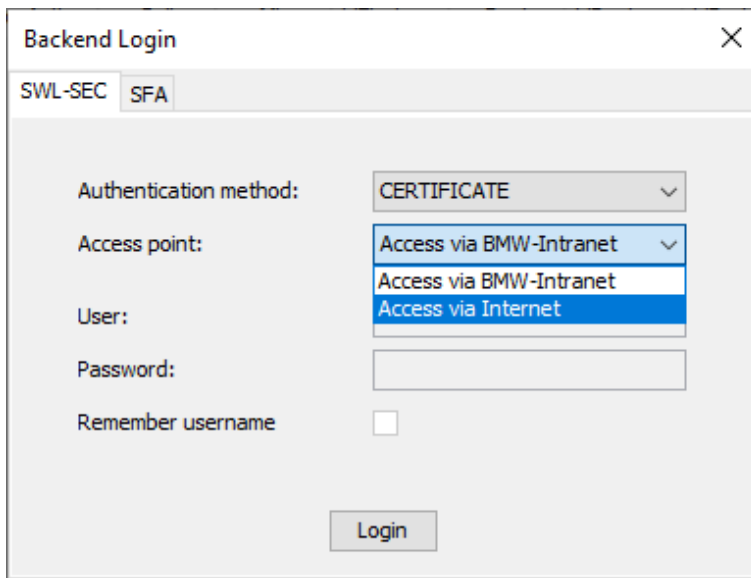
This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.



The FDL-Editor displays the netto-coding-file (NCD) in a visual format the FDLs (function data lists).

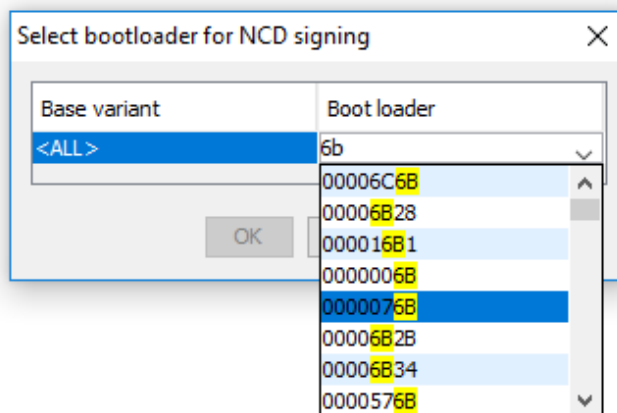
The NCDs can be edited within these module.

Before the file is saved, it is automatically signed. Here can be required the access data for the SWL-Security Backend (see secure coding) and the BTLD number.



The 'Backend Login' dialog box is shown with the 'SWL-SEC' tab selected. It contains the following fields and options:

- Authentication method: CERTIFICATE (dropdown)
- Access point: Access via BMW-Intranet (dropdown menu is open showing 'Access via BMW-Intranet' and 'Access via Internet')
- User: (text input field)
- Password: (password input field)
- Remember username:
- Login button



The 'Select bootloader for NCD signing' dialog box is shown. It features a table with the following data:

Base variant	Boot loader
<ALL>	6b
	00006C6B
	00006B28
	000016B1
	0000006B
	0000076B
	00006B2B
	00006B34
	0000576B

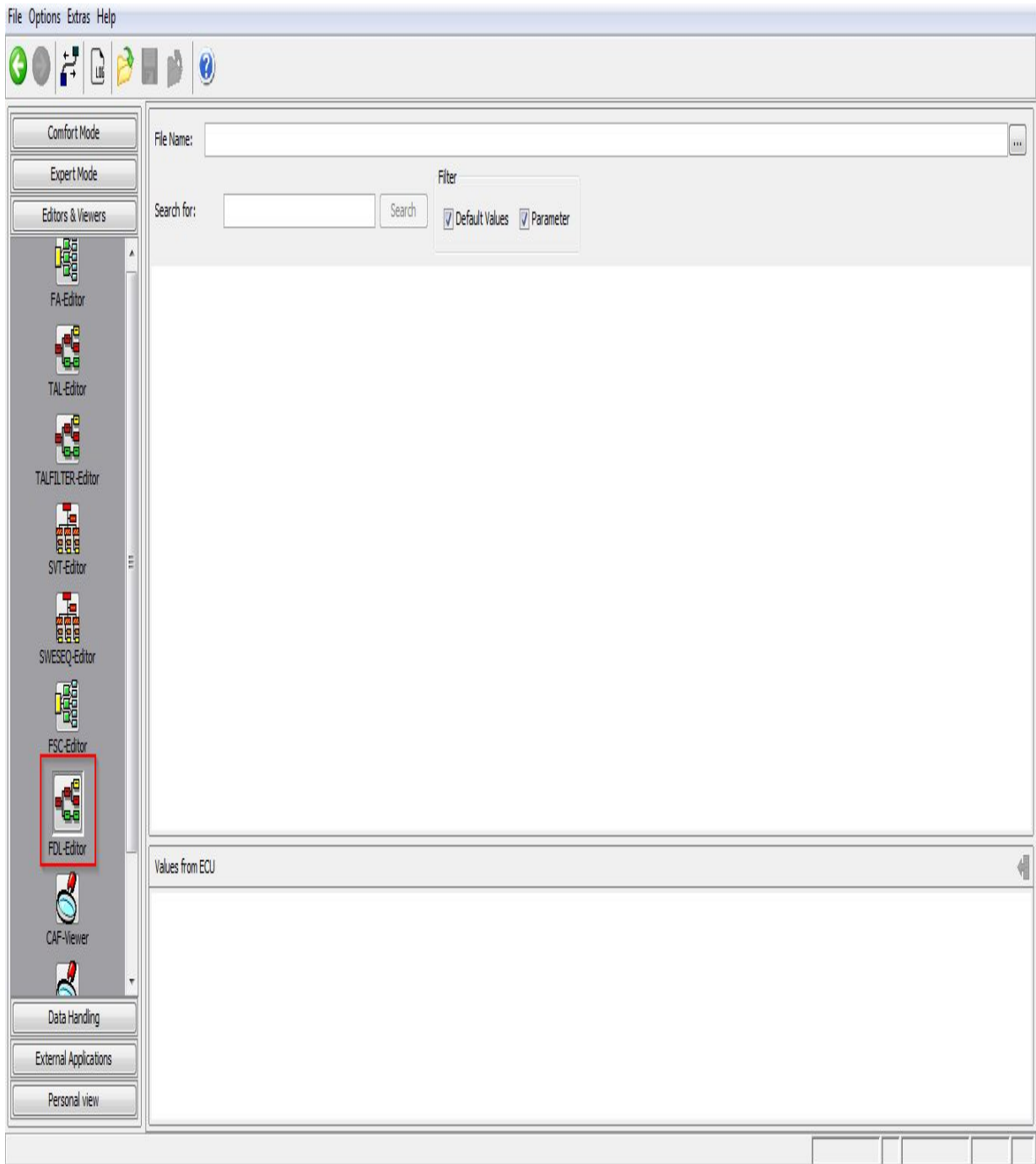
An 'OK' button is located at the bottom left of the dialog.

Signing of NCDs for Coding 2 or Coding 3 is only possible if the code verification stamp (CPS) in the NCD file is valid.

A function data list is in principle a CAF (Coding Application File) which is aggregated with the data residing actually in

the ECU.

A CAF exists prior to ECU coding as data record. But a FDL is generated by readout of data from a ECU and the input of the result in the corresponding CAF.



-

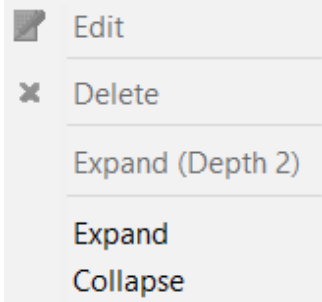
## **File-Menu**

- Save as FWL...

The coding data is stored as a readable function value list (FWL).

-

## **Context-Menu**



-

## **GUI-Elements**

1. Search Function Name searches the specified text in the whole tree. If there is a match the tree is expanded to the place of finding. You can continue the search with button F3.

Filter filters corresponding sub trees

-

---

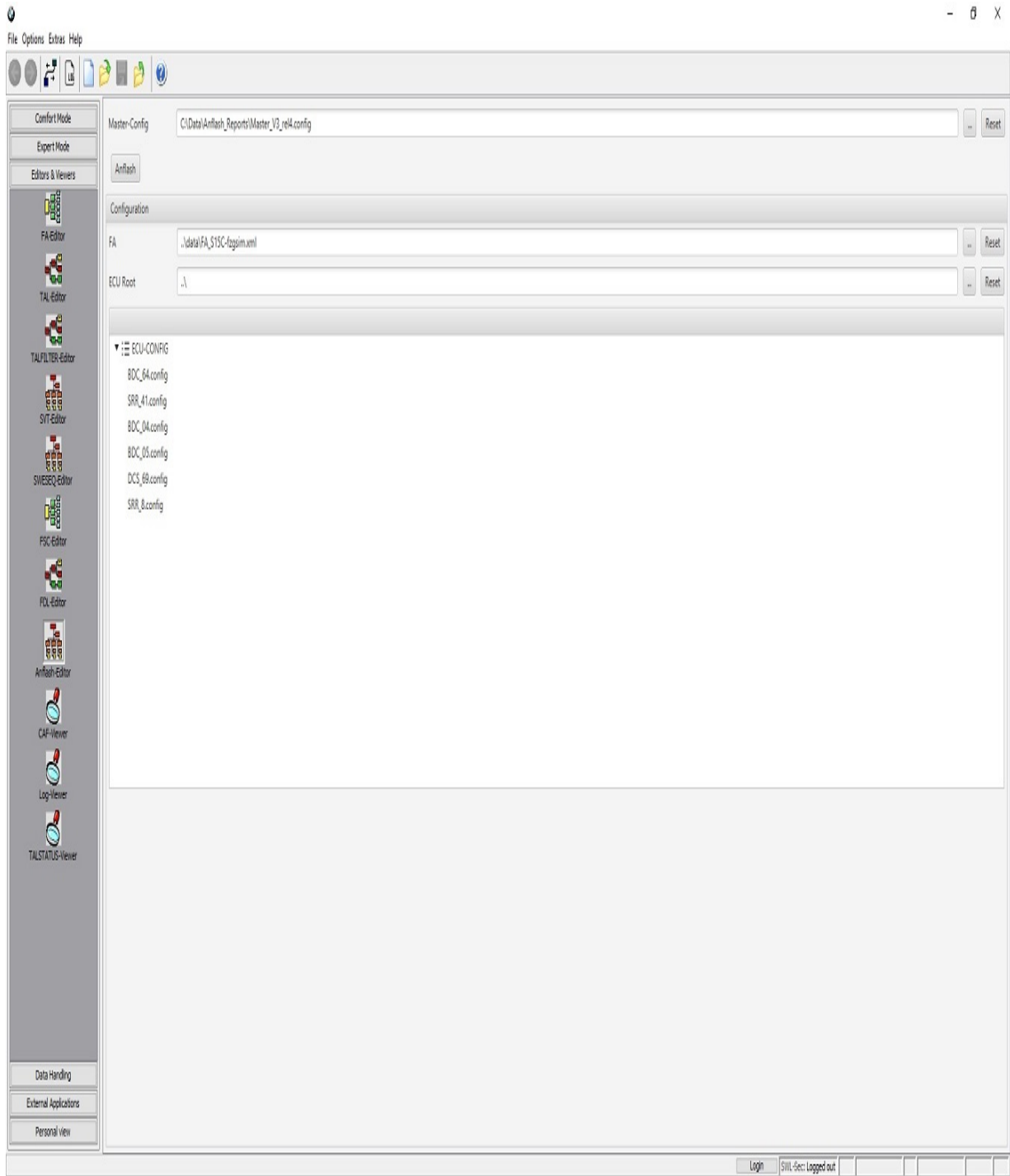
**[This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.](#)**

Module Anflash-Editor offers the possibility to display and modify Master configuration files.

The structure of the required configuration files is described in chapter 2.1 of the operation manual.

In the operation manual, the usage of the Anflash-Editor will be explained in detail.

By pressing the 'Anflash' button, the currently loaded master-configuration file can be opened and executed in the [ANFLASH](#) module.



## GUI Elements

<b>Master-Config</b>	
Textfield	Absolute path to the Master-configuration file.

Load	Opens a file dialogue to load an existing Master-Config from a file.
Reset	Closes the active file together with all corresponding elements in the Editor-View. This button is only active if a file has been opened before.
Anflash	Opens the currently loaded master-configuration file in module ANFLASH.
<b>FA</b>	
Textfield	Absolute or relative path to the vehicle order (FA). Relative paths can be entered manually.
Load	Opens a file dialogue to load a vehicle order file.
Reset	Resets the content of the text field back to the value, that is defined in the master-configuration file.
<b>ECU Root</b>	
Textfield	Absolute or relative path to the ROOT directory where all ECU-configuration files are located. A relative path entry can be entered manually.
Load	Opens a file dialogue for selection a folder.
Reset	Resets the content of the text field back to the EcuRoot entry, that is defined in the master-configuration file.
<b>ECU-CONFIG</b>	
Tree View	List the names of the ECU-Configs as defined in the Master-configuration file.

---

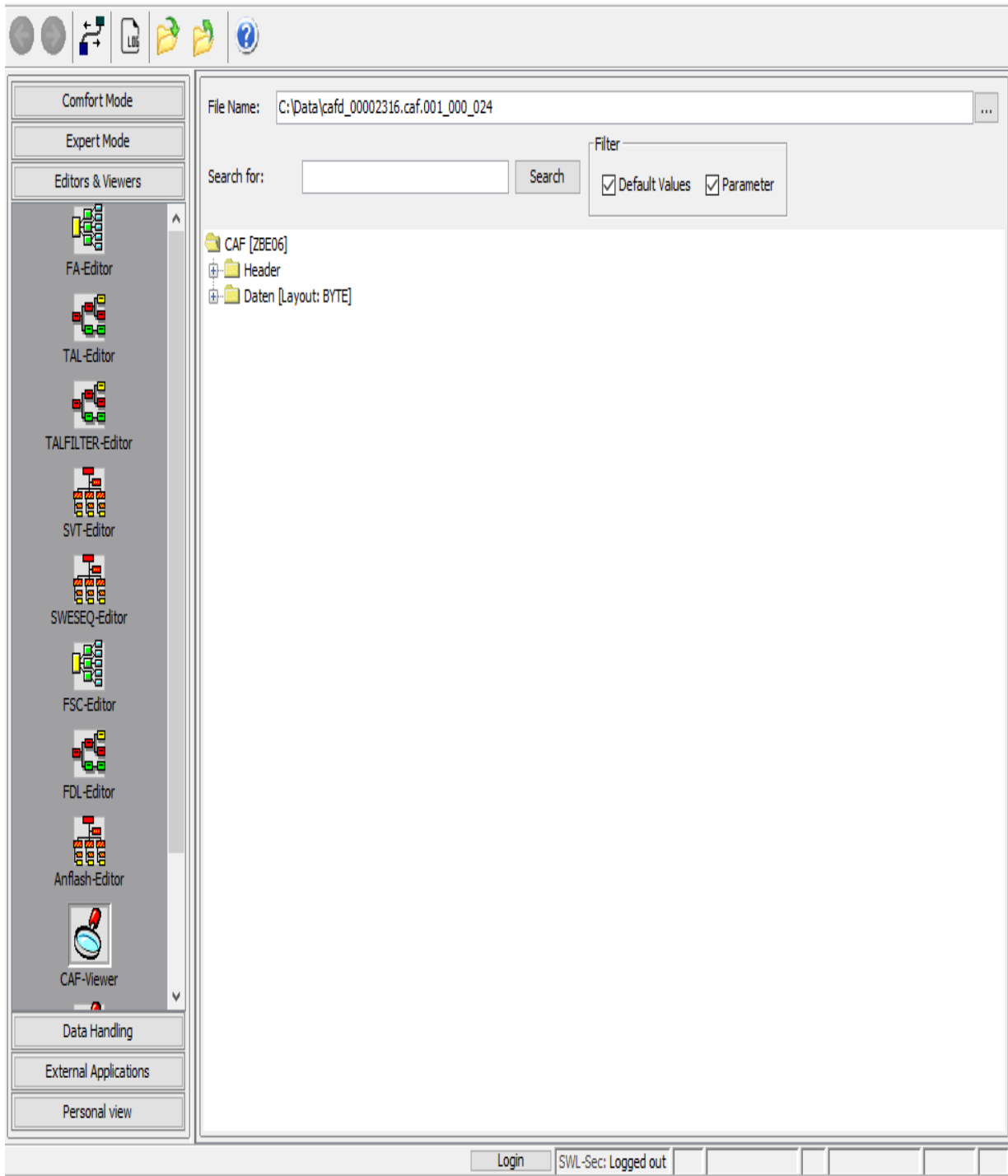
This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved.

This message will not appear if you compile this help file with the registered version of Help & Manual.

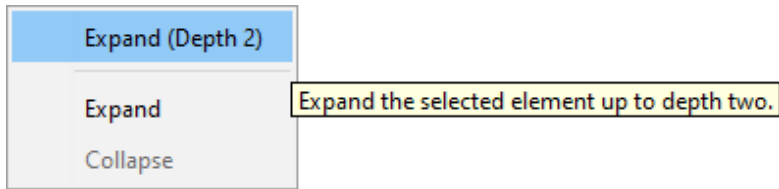
The CAF-Viewer views CAFs (Coding Application Files). A CAF file is a special SWE (software unity) that contains the coding data of a single ECU. All coding data of a single ECU can be distributed on multiple CAFs. As well the entire application software of a ECU can be distributed on multiple SWEs.

The coding data is divided in discrete coding data groups within the CAF file. Every coding data group contains at least one function which has at least one parameter.





## Context Menu



-

## **GUI Elements**

-

Search Function Name	searches the specified text in the whole tree. If there is a match the tree is expanded to the place of finding. You can continue the search with button F3.
----------------------------	--

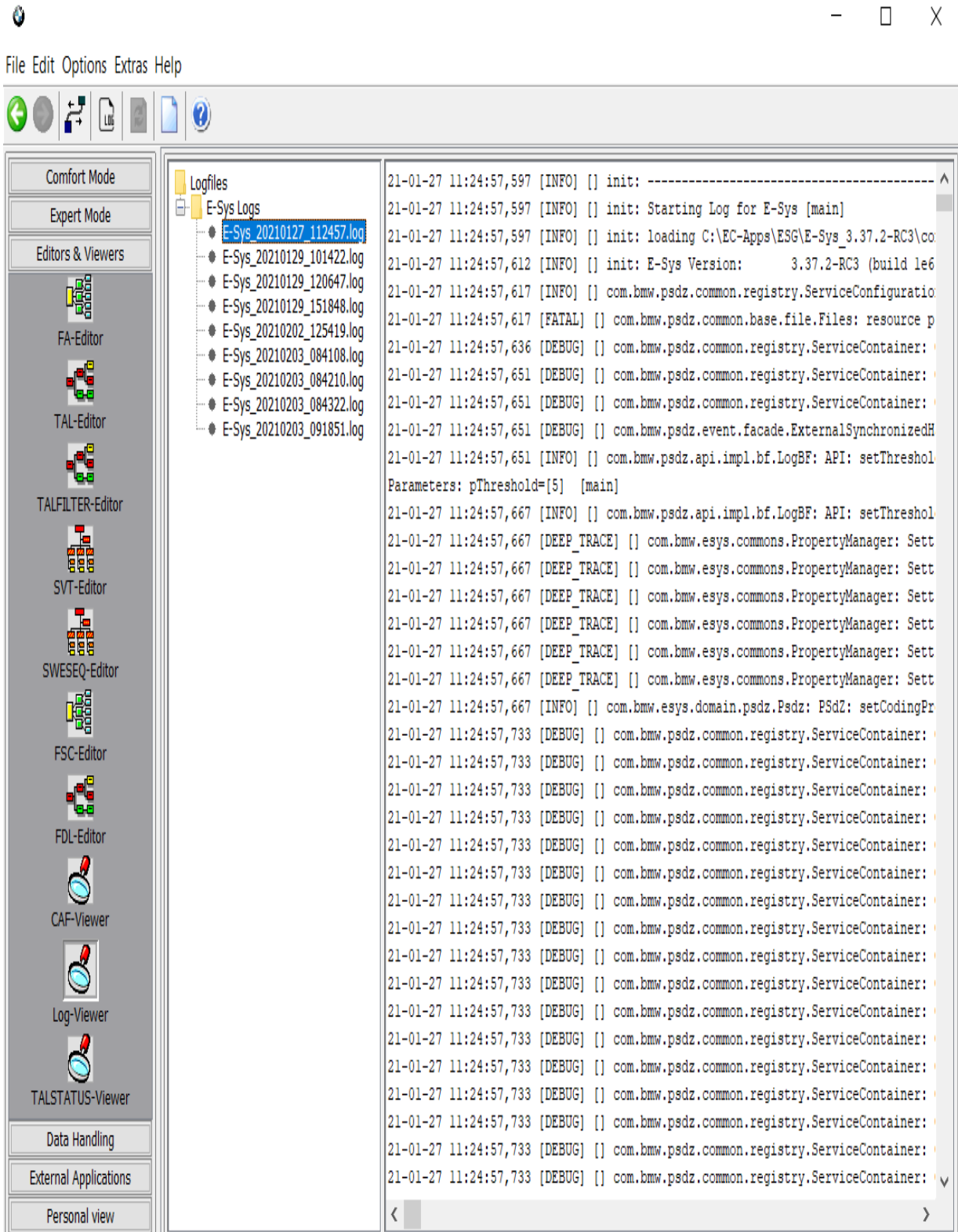
filter	filters corresponding sub trees
--------	---------------------------------

-

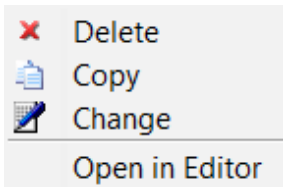
---

**This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.**

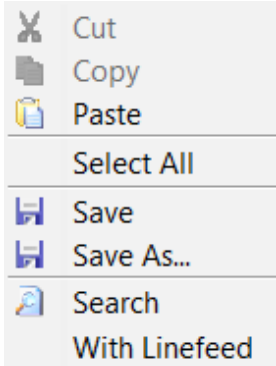
The log file viewer makes for view log files and protocols as well as manage them.



## Context-Menu (left side)



### **Context-Menu (right side)**



### **Specifics**

- You have to define a view before viewing log files. Do this on the root on the left side by the context menu. Specify name, path and extension on this occasion.
- The extension can be specified in the form of a regular expression. E.g.: `*.*`, `*.log`
- In the view (right window) you have discrete possibilities of editing.

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

**Note:**

**For Executed TALs that contain Mirror Deploy transactions, no graphical representation of the TAL execution process is currently possible.**

**Module TALSTATUS-Viewer offers the view of an executed TAL in a table with colored status**

**By double clicking on colored status field of the table, the executed TAL is displayed in an editor pane and the corresponding TAL line is marked.**

File Options Extras Help

TAL: D:\Data\E-Sys\_Data\_3.30.0\ExecutedTAL\G07\_TAL\_bl\_sw\_cd\_gwrb\_executed\_20170328\_134035\_FinishedWithError.xml

ID - Basic	Execution Context		Generation Context		XML	Timing							
	hwDeinstall	hwInstall	blFlash	swDeploy			idRestore	idBackup	idDelete	cdDeploy	fsDeploy	baDeploy	hddUpdate
AAG_71			Finished	Finished				Finished					
ACSM5_01			Finished	Finished				Finished					
ATM_61			FinishedWithError	NotExecutable				NotExecutable					
BDC_BODY3_40			FinishedWithError	NotExecutable				NotExecutable					
BDC_GW3_10			Finished	Finished								Finished	
CHC_38			Finished	Finished				Finished					
DKOMB14_60			FinishedWithError	NotExecutable				NotExecutable					
DME_BAC2_12			FinishedWithError	NotExecutable				NotExecutable					
DSC_BRS_IB_29			Finished	Finished				Finished					
DSC_VIP_IB_39			Finished	Finished				Finished					
EGS7_18			Finished	Finished				Finished					
EMARS_04			Finished	Finished				Finished					
EMARS_25			Finished	Finished				Finished					
EPS4_30			Finished	Finished				Finished					
FRR2_21			Finished	Finished				Finished					
FZD2_56			Finished	Finished				Finished					
GWS2_5E			Finished	Finished				Finished					
HKA2_78			Finished	Finished				Finished					
HKFM2_0D			Finished	Finished				Finished					
HSR2_2B			Finished	Finished				Finished					
HU_MGU_63			FinishedWithError	NotExecutable				NotExecutable					
ICAM3_06			Finished	Finished				Finished					
IHKA4_78			Finished	Finished				Finished					
KAFAS4_5D			Finished	FinishedWithError				NotExecutable					
LMV2_19			Finished	Finished				Finished					
NIV1_57			Finished	Finished				Finished					
PCU_2E			Finished	Finished				Finished					
RAM_37			FinishedWithError	NotExecutable				NotExecutable					
RSE_MGU_26			FinishedWithError	NotExecutable				NotExecutable					
SAS2_23			Finished	Finished				Finished					
SM2_69			FinishedWithError	NotExecutable				NotExecutable					
SM2_6A			FinishedWithError	NotExecutable				NotExecutable					
SM2_6D			Finished	Finished				Finished					
SM2_6E			Finished	Finished				Finished					
SPM_53			Finished	Finished				Finished					
SPM_59			Finished	Finished				Finished					
SRR_08			Finished	Finished				Finished					
USS_2C			Finished	Finished				Finished					
NDP_76			Finished	Finished				Finished					

Editors & Viewers

- FA-Editor
- TAL-Editor
- TALFILTER-Editor
- SVT-Editor
- SWESQ-Editor
- FSC-Editor
- FDL-Editor
- CAF-Viewer
- Log-Viewer
- TALSTATUS-Viewer

Data Handling

External Applications

Personal view

**The execution context tab provides information about the tal execution environment, including connection parameters, versions, execution parameters, fingerprint information and operating system information.**



TAL: C:\01\_FP\05\_Data\18A\_Rack\ExecutedTAL\_tal\_ncd\_coding1\_executed\_20200323\_174654\_FinishedWithWarnings.xml

State Execution Context Generation Context XML Timing

Connectionparameter

Vin: WBA000110HSVPG08

mcProjectName: S15C\_PT01\_TEMPLATE\_003\_027\_000

VehicleInfo: S15C

Baureihe:

BauStufe:

AccessLink: EthernetAccessLinkImpl [bus=ETHERNET, ip=127.0.0.1, port=6801]

Interface: Ethernet

executionConfig

Parallel:  True TaMaxRepeat: 0

UseFlaMode:  True UseProgCounter:  False

UseAEP:  False CodingType: FDL

Mileage: 0 ManageRoE:  False

ProgMode:  True ExpectedSgmid/ValidationActive:  True

HddUpdateURL:

ECU Blacklist	ECU Whitelist	UDS-Fallback
DKOMBI4_0x60	BDC_GW3_0x10	DKOMBI6_0x60
		WAVE_0x61
		UNKNOWN_0xFF

Versions

Application: E-Sys 3.36.0-RC3-SNAPSHOT 7e529e0

API: 5.10.00-RC3-2020-03-11T16:04:54Z

Core: 5.10.00-RC3-2020-03-11T16:04:54Z

CodeSysLib: 001.012.000

MCD: 003.005.001

JobLib: 005.027.002

KisAL: unknown

KisWB: unknown

JavaRTE: 1.8.0\_121 64bit

TemplateGlobal: 003.027.000

TemplateProject: 003.027.000

Fingerprint

TesterServiceID: FF OrganisationID: 4d2

ProgDeviceSerial: 1000 ProgDeviceType: 1

System

OS: Windows 10: 10.0 OS-Architecture: amd64 Hostname: R7555

Comfort Mode

Expert Mode

Editors & Viewers

- FA-Editor
- TAL-Editor
- TALFILTER-Editor
- SVT-Editor
- SWESQ-Editor
- FSC-Editor
- FDL-Editor
- CAF-Viewer
- Log-Viewer
- TALSTATUS-Viewer

Data Handling

External Applications

Personal view

Login SWL-Sec: Logged out

**The generation context tab provides information about the tal generation environment, including connection parameters, versions, generation parameters and operating system information.**

File Options Extras Help

TAL: D:\Data\Data\_S18A\_FzgSim\ExecutedTAL\TAL\_gesamt\_executed\_20170727\_122032\_FinishedWithError.xml

State Execution Context Generation Context XML Timing

Connectionparameter

Vin:

mczProjectName: S18A\_PTO1\_TEMPLATE\_003\_000\_214

VehicleInfo: S18A

Baureihe:

Baustufe:

AccessLink: EthernetAccessLinkImpl [bus=ETHERNET, ip=|127.0.0.1, port=50160]

Interface: Ethernet

generationConfig

BlackList Used:  False Target Setup:  True

WhitelList Used:  False ConfigParams Used:  False

Backup Data:

Vin:

FSEQ:

```
S18A
000.000.056
Patrice PICARD
Update new XML-Structure, CSM4HU, BATPCU2SGE4DME
20170514-120000
```

CSEQ:

```
S18A
000.000.040
Patrice PICARD
Update new XML-Structure
20170326-120000
```

Versions

API: 5.05.00-M2-RC1-2017-07-06T09:50:18Z

Core: 5.05.00-M2-RC1-2017-07-06T09:50:18Z

CodeSysLib: 001.012.000

MCD: 003.003.000

JobLib: 005.012.000

KisAL: 005.007.002

KisWB: 000.000.213

JavaRTE: 1.8.0\_92 32bit

TemplateGlobal: 003.000.214

TemplateProject: 003.000.214

System

OS: Windows 7: 6.1 OS-Architecture: x86 Hostname: R6696

Comfort Mode

Expert Mode

Editors & Viewers

FA-Editor

TAL-Editor

TALFILTER-Editor

SVT-Editor

SIVSEQ-Editor

FSC-Editor

FDL-Editor

CAF-Viewer

Log-Viewer

TALSTATUS-Viewer

Data Handling

External Applications

Personal view

**By double clicking on colored status field of the table, the executed TAL is displayed in an editor pane and the corresponding TAL line is marked.**

File Options Extras Help

Comfort Mode  
Expert Mode  
Editors & Viewers

FA-Editor  
TAL-Editor  
TALFILTER-Editor  
SIVT-Editor  
SIVSEQ-Editor  
FSC-Editor  
FDL-Editor  
CAF-Viewer  
Log-Viewer  
TALSTATUS-Viewer

Data Handling  
External Applications  
Personal view

TAL: D:\Data\E-Sys\_Data\_3.30.0\ExecutedTAL\tal\_delta\_executed\_20170420\_233524\_FinishedWithError.xml

State Execution Context Generation Context XML Timing

```
<patchVersion>005</patchVersion>
</expectedSGBMIDs>
</cdDeploy>
</talLine>
<talLine cause="ta_39" startTime="20170420-233503.365" endTime="20170420-233503.365" diagAddress="63" base/variant="HJ_1872" status="NotExecutable" id="t_32">
  <preparation id="prep_32" cause="ta_39" status="NotExecutable"/>
  <authentication id="auth_32" cause="ta_39" status="NotExecutable"/>
  <finalization id="final_32" cause="ta_39" status="NotExecutable"/>
  <cdDeploy status="NotExecutable" id="tg_32" cause="ta_39">
    <cdDeployTA status="NotExecutable" id="ta_60" cause="ta_39">
      <sgmid>
        <processClass>CAFD</processClass>
        <id>00001EF6</id>
        <mainVersion>006</mainVersion>
        <subVersion>011</subVersion>
        <patchVersion>017</patchVersion>
      </sgmid>
    </cdDeployTA>
  </cdDeploy>
  <expectedSGBMIDs>
    <processClass>HWEL</processClass>
    <id>00002479</id>
    <mainVersion>001</mainVersion>
    <subVersion>002</subVersion>
    <patchVersion>001</patchVersion>
  </expectedSGBMIDs>
  <expectedSGBMIDs>
    <processClass>BTLD</processClass>
    <id>00001FFD</id>
    <mainVersion>002</mainVersion>
    <subVersion>009</subVersion>
    <patchVersion>059</patchVersion>
  </expectedSGBMIDs>
  <expectedSGBMIDs>
    <processClass>SIVFL</processClass>
    <id>00002718</id>
    <mainVersion>002</mainVersion>
    <subVersion>009</subVersion>
    <patchVersion>059</patchVersion>
  </expectedSGBMIDs>
  <expectedSGBMIDs>
    <processClass>SIVFL</processClass>
    <id>00002719</id>
    <mainVersion>002</mainVersion>
    <subVersion>009</subVersion>
    <patchVersion>059</patchVersion>
  </expectedSGBMIDs>
  </cdDeploy>
</talLine>
```

**Instead of an executed tal also a tal can be opened in order to get a quick overview about the transactions which will be executed.**

File Options Extras Help

ExecTAL: C:\BMW\Testing\Data\_E-Sys\_3\_27\_0\_M3-RC4\_M43674\TAL\TAL\_G11FV\_RF\_150423.xml

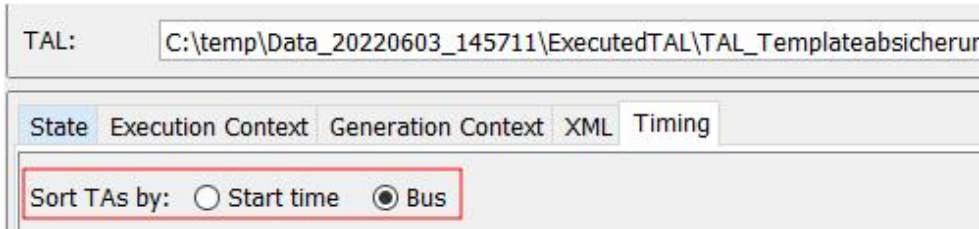
State Context XML

ID - Basic	hwDenstall	hwInstall	bFlash	swDeploy	idRestore	idBackup	idDelete	cdDeploy	fsdDeploy	ibaDeploy	hddUpdate	gatewayTableDe...
AAG			Executable	Executable				Executable				
ACSM5			Executable	Executable				Executable				
AMP_TOP18			Executable	Executable				Executable				
ATM			Executable	Executable				Executable				
BDC_BODY2			Executable	Executable				Executable				
BDC_GW2			Executable	Executable								Executable
DKOMB12			Executable	Executable				Executable				
DME_BAC2			Executable	Executable				Executable				
DSC3			Executable	Executable				Executable				
EGS			Executable	Executable				Executable				
EMARS			Executable	Executable				Executable				
EPS2			Executable	Executable				Executable				
FLM			Executable	Executable				Executable				
FRR2			Executable	Executable				Executable				
FZD2			Executable	Executable				Executable				
GWS2			Executable	Executable								
HKFM2			Executable	Executable				Executable				
HSR2			Executable	Executable				Executable				
HU_NB72			Executable	Executable				Executable		Executable		
ICAM3			Executable	Executable				Executable				
IHKA4			Executable	Executable				Executable				
KAFAS3			Executable	Executable				Executable				
LEM			Executable	Executable				Executable				
LMV2			Executable	Executable				Executable				
NVI			Executable	Executable				Executable				
PCU			Executable	Executable				Executable				
RS			Executable	Executable				Executable				
RSE_HIGH_NB72			Executable	Executable								
SAS2			Executable	Executable				Executable				
SM2			Executable	Executable				Executable				
SPM			Executable	Executable				Executable				
USS			Executable	Executable				Executable				
VDP			Executable	Executable				Executable				
ZBE4			Executable	Executable				Executable				

Data Handling  
External Applications  
Personal view

**In the Timing tab the timing sequence is displayed. If the executed tal is loaded for the first time in the viewer, a jpg file is generated and displayed. If the jpg file is already in the same directory as the executed tal, it is displayed only. With the mouse wheel one can zoom the picture.**

**The Timing tab can be sorted either by start time or by bus.**

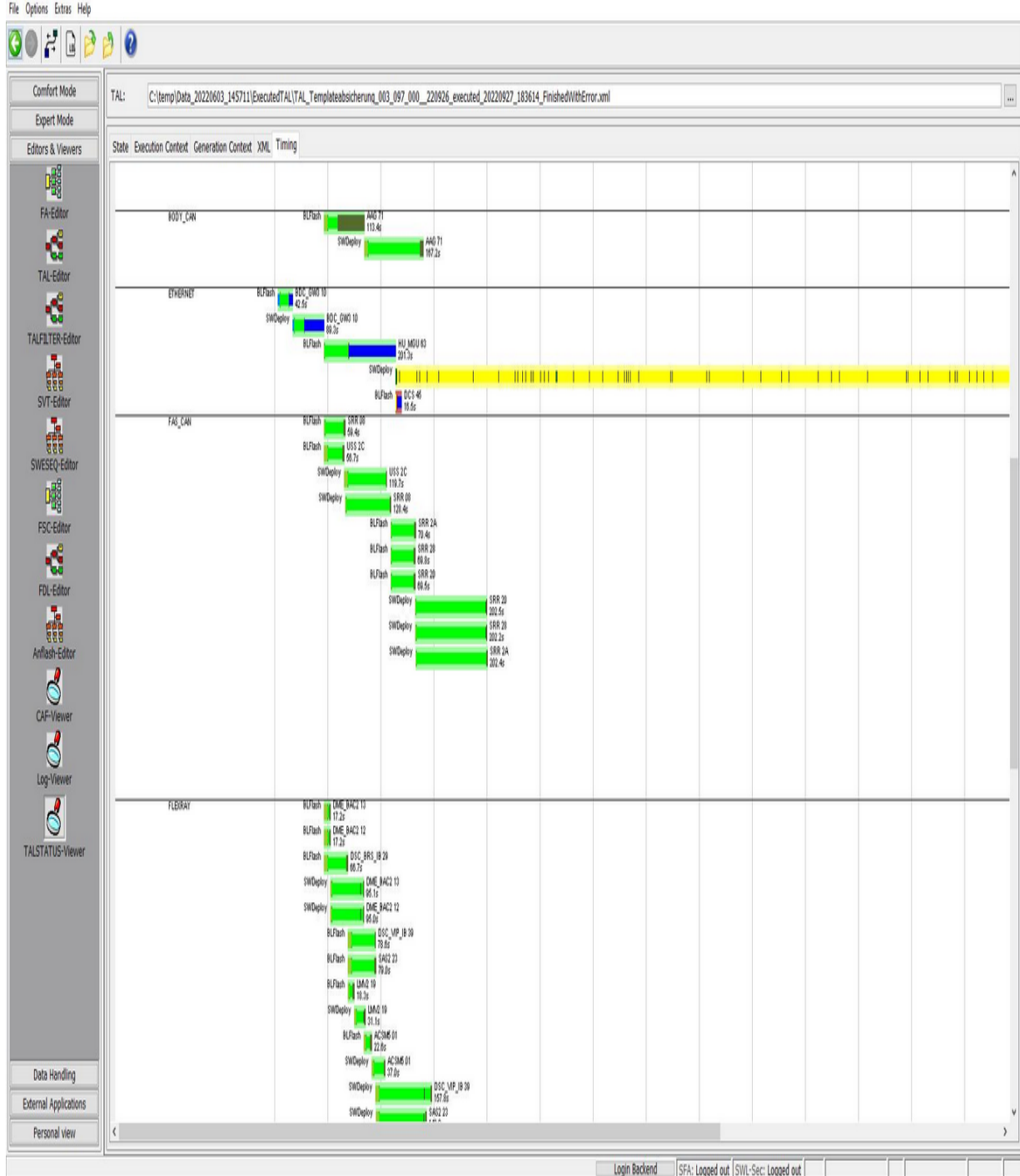


Sorted by start time:





Sorted by bus:



This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

[TAL-Processing](#)

[VCM](#)

[Coding](#)

[Coding-Verification](#)

[NCD-Preparation](#)

[FSC-Extended](#)

[TSL Update](#)

[NAV/ENT Update](#)

[OBD-CVN](#)

[Certificate-Management](#)

[SFA / ECU-Mode-Extended](#)

[KDS-Extended](#)

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

In the module TAL-Processing a TAL is processed.

A TAL is a model year plan with actions that are to be processed on a single ECU, a system of Emus or the entire vehicle. It doesn't count where the TAL is obtained from: The TAL can be loaded from the file system or adopted from another E-Sys module.

A TAL consists at least of one TAL line. There is maximal one TAL line for every ECU and transaction category.

If you try to execute a TAL with assembly or disassembly of hardware a message pops up and you can operate to save the information of the ECU by the menu Extras->Individualdatenrettung that is specific for this individual ECU. Information that can't be retrieved on the HO network nor from other Emus of the same type.

These files are saved in a backup directory. Now the HW can be exchanged and the information added to the TAL in [Calculation of TAL](#) mask.

See there further information.

### **Notes:**

- BAT/HAF verification

Before starting the TAL processing, a BAT/HAF verification is carried out. It will be checked whether a BAT/HAF control unit is installed. If this is the case, the status of switch S1 is determined.

When the switch is open, a message appears informing the user about possible risks for the TAL-execution process.

- **Programming protection**

To minimize security risks, for example due to old software versions, a secure token is required for flash operations requiring authorization.

For the EMUs requiring authorisation, the token can be determined automatically under certain circumstances (see chapter [SFA / LCS / Secure ECU Modes](#))

the process for Program Protection Basic requires no action from the user.

Without a configured backend, however, or in the case of Programming Protection Plus, the necessary token must be requested with a token request in the SFA backend or from the CBB.

The request can be created with the button "Create request file" (Basic: JSON format; Plus: XML format) and then uploaded to the respective

backend for authentication. The backend will generate a signed secure token from the request..

As soon as the signed token is available, it can be loaded via the "..."  
button to be used in the TAL. button to be available for TAL processing.

**Notes:**

- It is possible to generate a token for Program Protection Plus or to use a signed token, even if the ECU UID could not be read from all ECUs. If at least one ECU has responded to the request (ECU-UID), the process is continued.

To use this option the following entry must be contained in  
Esys.properties: TalExecution.param.flashProtectionPlusOptimistic=true

- When loading a token file, the content of the file is not checked for validity. This happens only when a TAL processing is started!

TAL: C:\temp\Data\_20220603\_145711\TAL\TAL\_PROG\_PROTECT.xml ... Edit

SVT Target: C:\temp\Data\_20220603\_145711\SVT\SVT\_funktional\_20230128\_153133.xml ... Read SVT (ECU) Edit

FA: C:\temp\Data\_20220603\_145711\FA\FA.xml ... Read FA (VCM) Edit

Programming Protection Basic (Field-Mode): Create request file ... Reset

Programming Protection Plus (Plant-Mode): Create request file ... Reset

Programming Tokens: Create unsigned Token ... Reset

Authentication method for Mirror-Protocol (Programming Token mandatory)

Execution in Plant-Mode: ... Reset

Signed Tokens: ... Reset

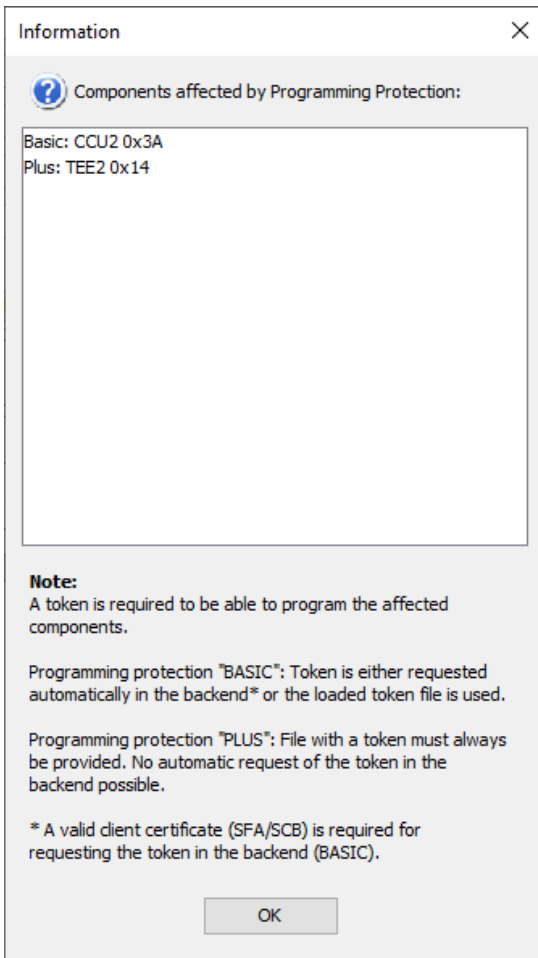
SFA Programming Token: ... Reset

Without authentication

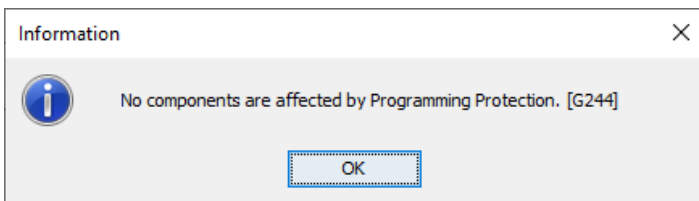
read VIN out of FA  Enter VIN: WBA0000150TSVPG11 Read VIN Check VIN

Start Stop Check software availability **Check Programming Protection** Details

The type of programming protection that may be necessary can be determined via the button "Check programming protection". If corresponding tokens are necessary, they are displayed in the following dialog.



If no components are affected, only the following information is displayed:



### · **Mirror-Protocol**

E-Sys is able to handle DualMemoryClients with RSUMirrorProtocol. The prerequisite for this is that the client certificate and the URL for access to the SFA backend are set up (see also chapter [SFA / LCS / Secure ECU Modes](#)).

E-Sys determines double memory ECUs independently on the basis of the TAL and takes them into account in the further process if ecuMirrorDeploy TAs exist for them.

For TAL processing, the following authentication options must be selected:

- Signed Token
- SFA Programming Token
- Processing in plant mode
- No authentication

An unsigned Programming Token is required for all authentication methods.

## **Signed Token**

Because some ECUs do not require the token to be signed, it is possible to use an unsigned token as a signed token. In these cases, the user does not need to load the signed token – the field for loading it can be left empty – **only** the authentication method “Signed Token” needs to be selected.

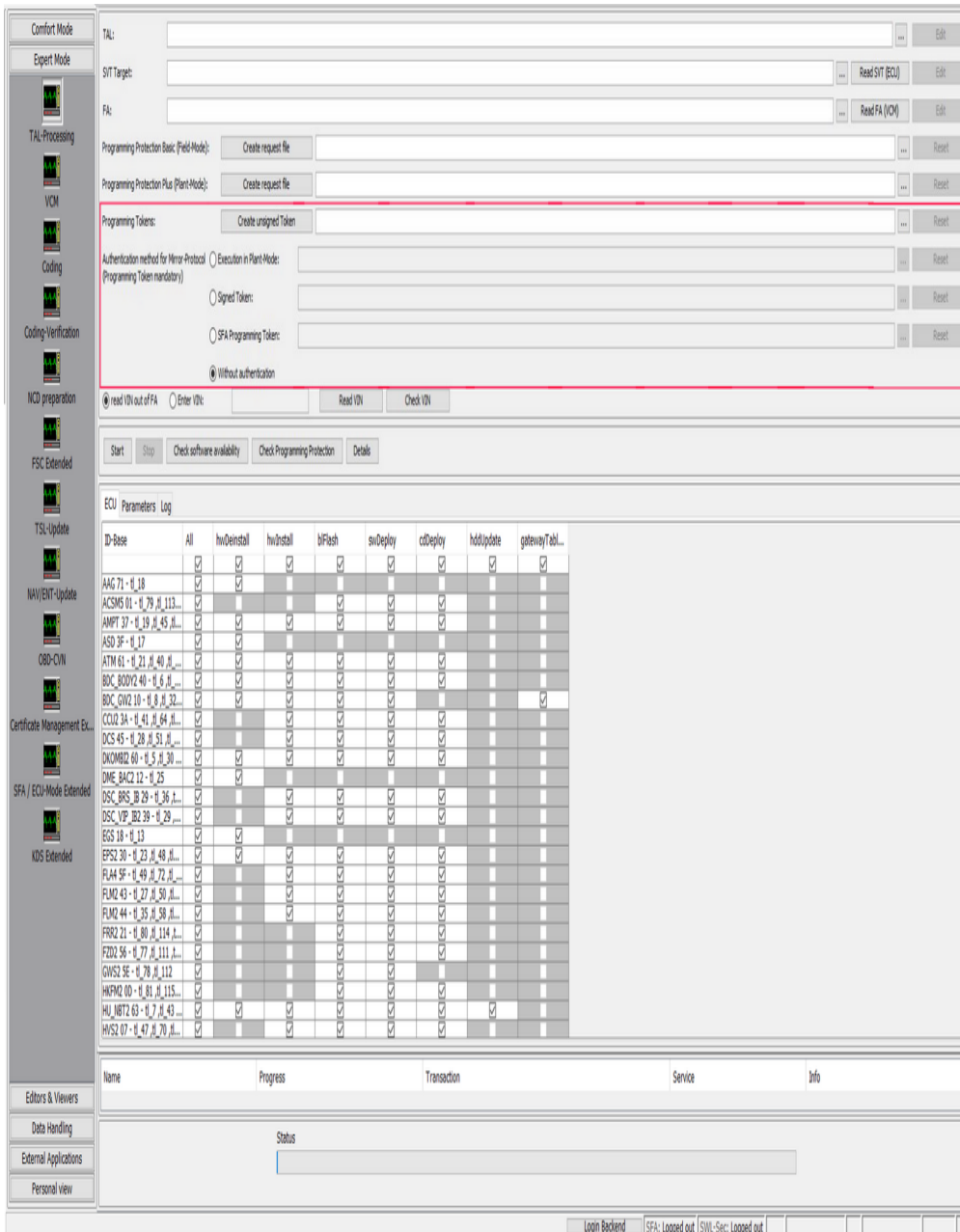
For these cases, the process of generating the unsigned token can be automated. So, when the authentication method is “Signed Token” and no file containing the unsigned token has been provided, an unsigned token will be automatically generated before TAL processing begins. This automatic process creates the following two files in the default directory for programming tokens (e.g., C:\Data\ProgrammingTokens):

- ProgrammingToken\_YYYYMMDD\_HHMMSS.xml

-

ProgrammingToken\_YYYYMMDD\_HHMMSS\_UnsignedMirrorProtocolToken.json  
Where YYYYMMDD\_HHMMSS is the timestamp when the files were created.

It is important to note that if the user provides both unsigned and signed files, those will be the ones used.



## GUI Elements

TAL	Using the three-dot-button a file dialog is started to choose a TAL file. The file path will be shown in the text field. With the "Edit" button you can change to the TAL-Editor to make some changes to the TAL.
SVT	Using the three-dot-button a file dialog is started to choose a target SVT file. The file path will be shown in the text field. With the "Edit" button you can change to the SVT-Editor to make some changes to the SVT.



	<p>The SVT will be written to the VCM before the flashing the Emus.</p> <p>Pressing button "Read SVT (ECU)" the SVT will be generated by reading the content of the connected Emus.</p> <p>The SVT is optional.</p>
FA	<p>Using the three-dot-button a file dialog is started to choose a vehicle order file. The file path will be shown in the text field.</p> <p>With the "Edit" button you can change to the FA-Editor to make some changes to the vehicle order.</p> <p>Pressing the "Read FA (VCM)" button the vehicle order will be read from the ECU.</p> <p>An FA is optional for flashing but mandatory for coding.</p>
Programming Protection Basic	<p>The button "Create request file" can be used to create a token request in JSON format for the SFA backend.</p> <p>The "..." button is used to load a signed secure token response. The path is displayed in the text field after loading.</p>
Programming Protection Plus	<p>The button "Create request file" can be used to create a token request in XML format for the CBB.</p> <p>The "..." button is used to load a signed secure token response. The path is displayed in the text field after loading.</p>
Programming Tokens	<p>Using the button "Generate token" a file with unsigned programming tokens will be created. The file name is displayed in the text field. This file must be signed outside of E-Sys.</p> <p>Using the three-dot-button the file dialog is started to choose a file with signed Programming Tokens. The file path is displayed in the right text field.</p> <p>A signed Programming Token is required to execute a TAL with Mirror-Deploy actions!</p>
read VIN out of FA	If selected the VIN will be taken from the vehicle order for the TAL execution.
Enter VIN	if the selected the VIN will be taken from the text field for the TAL execution.
Read VIN	Read the VIN.
Start	Start of TAL processing. This button turns to "Pause", after another click (disabled for parallel programming) the processing pauses and button turns to "Resume". The next click of this button continues the processing of the TAL.
Stop	After the current processed TALline the processing is

	aborted.
Check software availability	Checks if all software required for TAL execution is available.
<b>ECU</b>	
rows	List the existing transaction categories for the loaded TAL.
columns	List all transactions for each control unit. If a control unit does not have a transaction for a category, the corresponding cell is displayed in gray. If the executionStatus of the transaction differs from "executable", the corresponding executionStatus is displayed. An executable transaction can be activated or deactivated by checkbox.



- Comfort Mode
- Expert Mode
- TAL-Processing
- VCM
- Coding
- Coding-Verification
- NCD preparation
- FSC Extended
- TSL-Update
- NAV/ENT-Update
- OBD-CVIN
- ificate Management Extended
- FA / ECU-Mode Extended
- KDS Extended

TAL:  ... Edit

SVT Target:  ... Read SVT (ECU) Edit

FA:  ... Read FA (VCM) Edit

Programming Protection Basic (Field-Mode): Create request file  ... Reset

Programming Protection Plus (Plant-Mode): Create request file  ... Reset

Programming Tokens: Create unsigned Token  ... Reset

Authentication method for Mirror-Protocol (Programming Token mandatory)

Signed Token:  ... Reset

SFA Programming Token:  ... Reset

Execution in Plant-Mode:  ... Reset

Without authentication

read VIN out of FA  Enter VIN:  Read VIN Check VIN

Start Stop Check software availability Check Programming Protection Details

ECU Parameters Log

Parallel Programming

1 Repeat in case of Error

Check Programming Counter

Deactivate HTTP-Transmission

Auto-fill InstalledECUList in TAL before execution

Overwrite existing InstalledECUList

Merge with existing InstalledECUList

0 Kilometers for fingerprint Read from GWSZ

Deactivate ResponseOnEvent (RoE) during TAL-Execution

SMAC: force SWEs transmission

Execute optimized bootloader flash

Deactivate TAS

Use local NCD files

...

Enable delta SWE flash

Programming mode UDS Fallback:

Activate programming mode for switchable ECUs

Diag-Address	ID-Base	Blacklist	Whitelist

Name	Progress	Transaction	Service	Info
Status				

Programming mode UDS Fallback Programming Protection Status Check

Activate programming mode for switchable E...

Diag-Address	ID-Base	Blacklist	Whitelist
[0x10]	BDC_GW3_10	<input type="checkbox"/>	<input type="checkbox"/>
[0x40]	BDC_BODY3_40	<input type="checkbox"/>	<input type="checkbox"/>
[0x60]	DKOMBI4_60	<input type="checkbox"/>	<input type="checkbox"/>
[0x61]	ATM2_61	<input type="checkbox"/>	<input type="checkbox"/>
[0x63]	HU_MGU_63	<input type="checkbox"/>	<input type="checkbox"/>
[0x67]	ZBE4_67	<input type="checkbox"/>	<input type="checkbox"/>
[0x71]	AAG_71	<input type="checkbox"/>	<input type="checkbox"/>
[0x7]	HVS2_7	<input type="checkbox"/>	<input type="checkbox"/>

Programming mode UDS Fallback Programming Protection Status Check

Diag-Address	ID-Base	Prevent UDS-Fallback
[0x10]	BDC_GW3_10	<input type="checkbox"/>
[0x40]	BDC_BODY3_40	<input type="checkbox"/>
[0x60]	DKOMBI4_60	<input type="checkbox"/>
[0x61]	ATM2_61	<input type="checkbox"/>
[0x63]	HU_MGU_63	<input type="checkbox"/>
[0x67]	ZBE4_67	<input type="checkbox"/>
[0x71]	AAG_71	<input type="checkbox"/>
[0x7]	HVS2_7	<input type="checkbox"/>

Reset

Programming mode UDS Fallback Programming Protection Status Check

Diag-Address	ID-Base	Programming Protection Status
[0x10]	BDC_GW3_10	<input type="checkbox"/>
[0x40]	BDC_BODY3_40	<input type="checkbox"/>
[0x60]	DKOMBI4_60	<input type="checkbox"/>
[0x61]	ATM2_61	<input type="checkbox"/>
[0x63]	HU_MGU_63	<input type="checkbox"/>
[0x67]	ZBE4_67	<input type="checkbox"/>
[0x71]	AAG_71	<input type="checkbox"/>
[0x7]	HVS2_7	<input type="checkbox"/>

Reset

## Parameters

Parameters for selection of special features for TAL processing.

Parallel Programming	Specification if the Emus should be flashed parallel. This setting is not active for a DIRECT connection.
Repeat	Definition of the number of repetitions in case of error.
Programming Counter	Specification if the Programming Counter should be activated.
HTTP-Transmission	Specification if the HTTP-Transmission should be activated.
InstalledECUList	Specification if the InstalledECUList in TAL should be auto-filled before execution.
Kilometers	Definition of the Mileage in Kilometers for fingerprint.
ResponseOnEvent	Specification if the ResponseOnEvent during TAL-Execution should be deactivated. This setting is not active for a DIRECT connection.
Optimizable bootloader flash	Specification if the optimizable bootloader flash should be performed or is not allowed. If activated the system is performing an check which of the software (SWFL, SWFK) that has to be falshed, based on TAL, is after the bootloader-flash already installed. After these check only the needed, not already installed, software is going to be flashed.
Enable delta SWE Flash	Specification if the Enable SWE Flash should be activated
Use local NCD files	Specification if to use the pre-signed and locally available NCD data. <u>Notes:</u> - in this directory a subfolder is searched with the VIN as name - NCD files must be signed with the VIN from the used FA
SMAC: force SWEs transmission	Specifies whether all SWEn for SMAC should always be transmitted. If deactivated, only required SWEn are transferred. SWEn that are already present and valid on the smart actuator are skipped.
Tab "Programming mode"	Definition of a black / white list for the mode switching of individual control units. Whitelist -> Control unit must be switched. Blacklist -> Control unit must not be switched. This setting is not active for a DIRECT connection.
Tab "UDS fallback"	If the HTTP data transfer of the software packages to the ECU fails, the UDS protocol is used as fallback. In this table, the fallback mechanism can be deactivated for the selected ECU.

	If an error now occurs in the HTTP transfer, the flash process for the selected ECU is aborted with error.
Tab "Programming Protection Status Check"	Choose a set of Ecu's that should execute the read status programming protection during TAL Processing.
<b>Log</b>	
Clear	Clear the log window.
Events	Specification if the events should be displayed in the logging.
EventType	Type of the logged events are specified.
<b>Progress</b>	
Table	Visualization of the currently executed transactions and services for the appropriate ECU (Name).
Progress Bar	Visualization of progress in TAL processing.

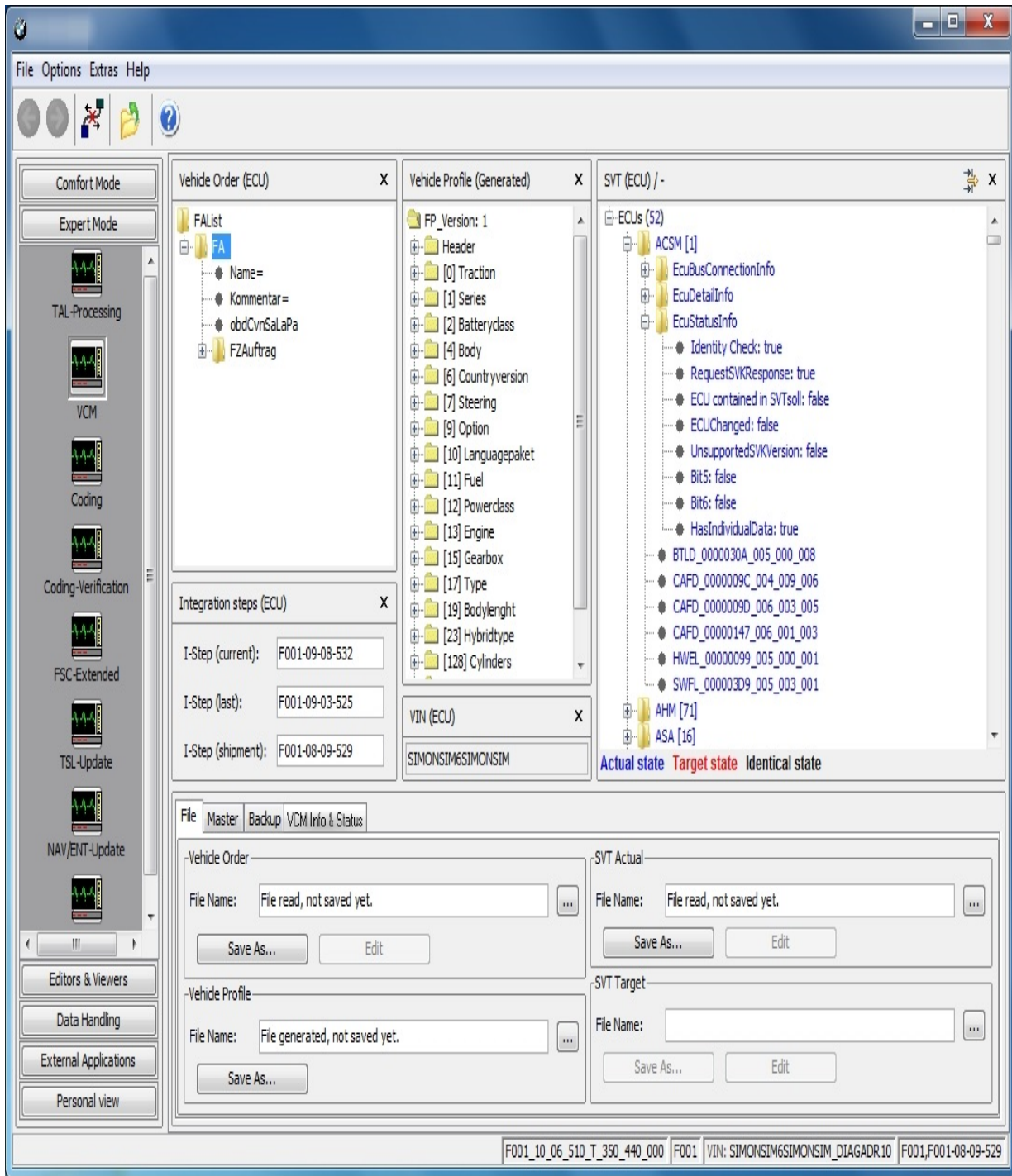
### **Reading Vehicle Order and VIN**

After reading the vehicle order the VIN will always be read too. Also the VIN could be read independently by pressing the button "Read VIN". The vehicle order and the VIN will be read from the master ECU. If an error occurs they will be read from the backup ECU. In the status bar is shown where the vehicle order and the VIN were actually read from.

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.


The Module VCM provides functions of the VCM (Vehicle Configuration Management). The VCM is a component in the vehicle that records discrete states and information of the vehicle. The VCM is no ECU but a function of the vehicle. VCM's functions are addressed by a special (reduced) vehicle info specification thus during the begin of a session the application communicates first with VCM and afterwards addresses the Emus. The module VCM has been extended significantly in its functionality for integration step 3.0.0. Besides the present reliable functions now also control functions to backup partner CAS are possible. The VCM functionality itself is provided actually by the ZGW ECU.



## GUI Elements

Vehicle Order (source)	Display window for FA-List with specification of source.
✕	Clear display window for the Vehicle Order (FA) and delete correspondent



	data in the RAM.
Vehicle Profile (source)	Display window for Vehicle Profile (FP) with specification of source.
x	Clear display window for FP
SVT (source)/(source)	Display window for SVT Actual and SVT Target with specification of sources.
x	Clear display window for SVTs.
Integration Steps	Read/write elements show data for current, last and shipment.
x	Clear display window for the integration steps.
VIN (source)	Display of VIN
x	Clear display window for the VIN.
	Provide masking discrete elements of the displayed SVT in display window.

## **Control Elements in "File" tab**

Three-dot-Button	Load respective data (FA, FP or SVT) from file system.
Save as	Save respective data (FA, FP or SVT) into file system.
Edit	Open the respective editor and load current data (FA or SVT) to provide it for display.



-

**Control Elements in Sub mask "SVT Filter"**

check box "Only SVK"	Determines if filter "Only SVK" is set. If filter is set only the type of SVKs displayed in the combo box, are displayed in the SVK window - but not further information like ECU info or BUS info.
belonging combo box	Contains all SVKs and the entry "all".
check box "Ecu Info"	Determines if filter "Ecu Info" is set.
belonging combo box	Contains all ECU Info (like e.g. ActiveNotifyEnabled, IdentityCheck or UnsupportedSVKVersion )
check box "Bus Info"	Determines if filter "Bus Info" is set.
belonging combo box	Contains all bus variants of LIN, MOST, CAN, Flexray.

-

-

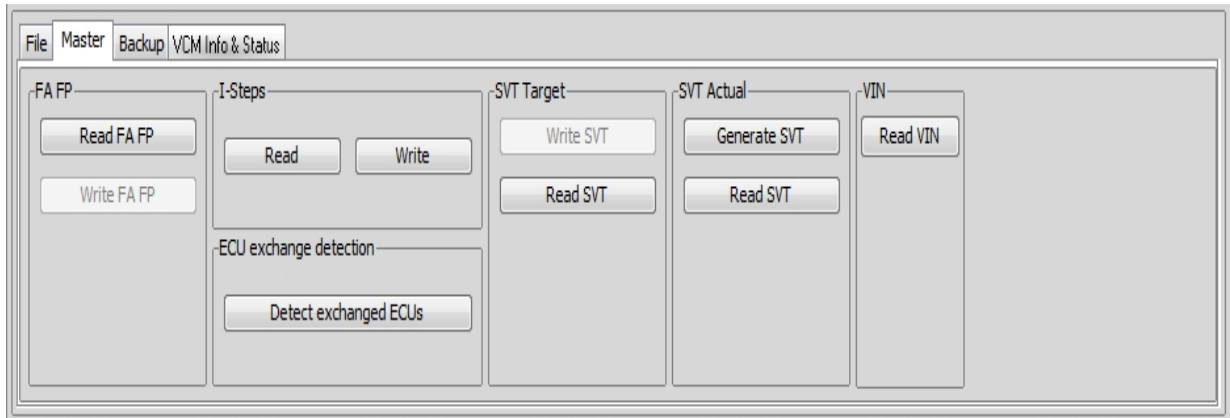
-

**Control Elements in "Master" tab**

FA FP: Read FA FP	Read FA, FP and VIN from VCM and display information.
FA FP: Write FA FP	Write current displayed FA and FP to VCM.
Update VIN	Write VIN displayed in input form to VCM.
I-Steps: Read	Read I-step current, last and shipment from VCM and

	display information.
I-Steps: Write	Write I-steps displayed in the mask (current, last and shipment) to VCM.
ECU exchanged detection: Detect exchanged Emus	Start routine for detection of exchanged Emus. The result are the bold marked Emus.
SVT Target: Write SVT	Write current SVT target to VCM.
Read SVT	Read SVT target from VCM and display it.
SVT Actual: Generate SVT	Determine SVT from returned SVKs of the discrete Emus.
SVT Actual: Read SVT	Read SVT actual from VCM and display it..
VIN: read VIN	Read VIN from master.

-



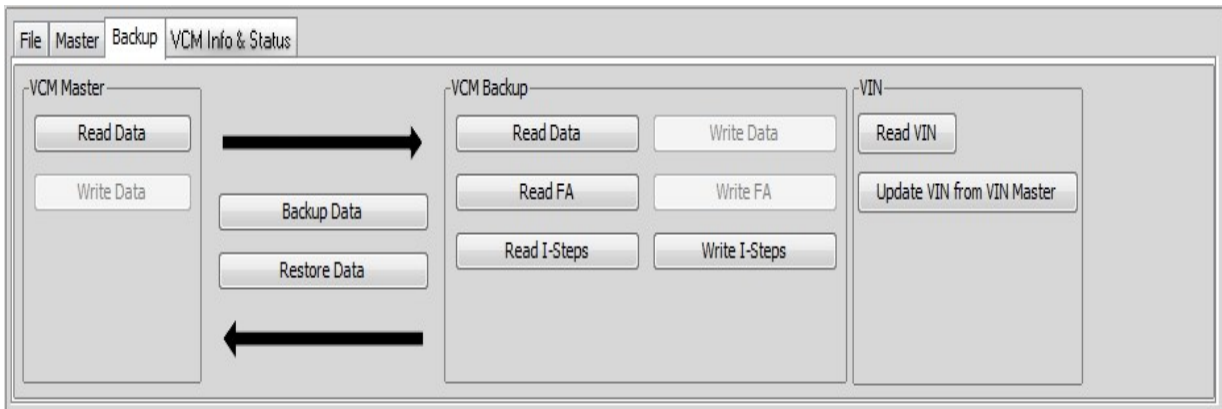
-

### **Control Elements in "Backup" tab**

VCM Master: Read Data	Read and display all backup relevant data (FA, FP, SVT Target, I-Steps) from VCM.
VCM Master: Write Data	Write all displayed data (FA,

	FP, SVT Target, I-Steps) to VCM.
Backup Data	Master backup function: Read and display all backup relevant data (FA, FP, SVT Target, I-Steps) from VCM and write it to Backup Partner (normally CAS).
Restore Data	Master backup function: Read and display FA, FP, SVT target and I-Steps from CAS and write to VCM.
VCM Backup: Read Data	Read and display FA, FP, SVT Target and I-Steps from CAS.
VCM Backup: Write Data	all backup relevant and displayed data (FA, FP, SVT target, I-Stufen) are written to the backup partner (usually CAS)
VCM Backup: Read FA	Read and display FA and VIN from CAS.
VCM Backup: Write FA	Write displayed FA to CAS.
VCM Backup: Read I-Steps	Display I-Steps read from Backup Partner (normally CAS).
VCM Backup: Write I-Steps	Write displayed I-Steps to Backup Partner (normally CAS).
VIN: Read VIN	Read VIN from backup.
VIN: Update VIN from VIN Master	Update VCM backup VIN with VCM master VIN.

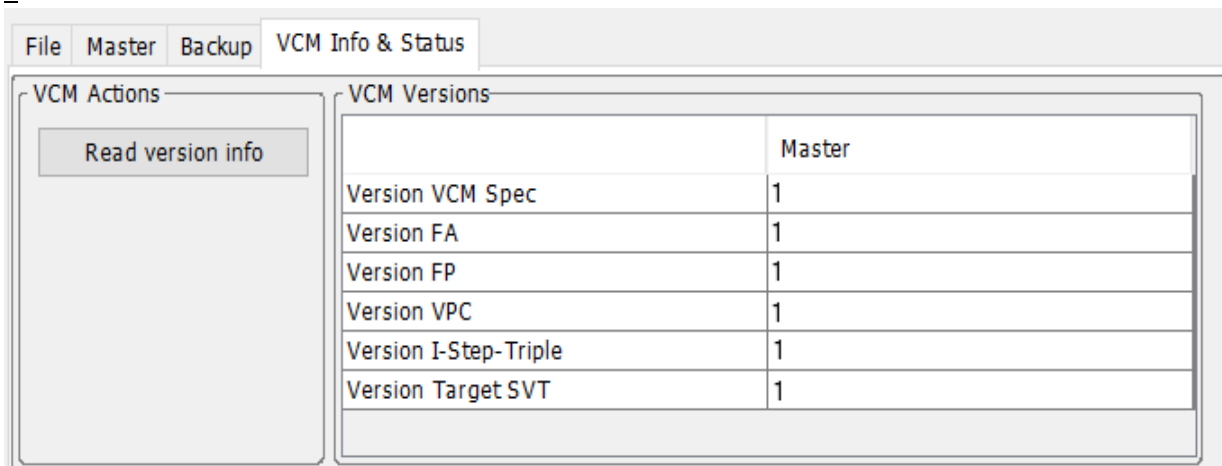
-



## Special Topics

- ECU exchange detection: A routine compares actual serial numbers of the connected Emus with a reference list. Differing serial numbers belong to exchanged Emus. These are marked as bold.
- Interplay ZGW (VCM) and Backup Partner (CAS)

## Control Elements in "VCM Info & Status" tab



VCM Actions: Read version info

Read and update VCM versions

## Special Topics

- Reading of version informations is only supported by VCM SP25 and newer.

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

With module Coding it's possible to select one or multiple Emus for coding without creating a TAL manually for this

# purpose.

File Options Extras Help

The screenshot shows a software application window with a menu bar (File, Options, Extras, Help) and a toolbar. The interface is divided into several sections:

- Left Panel:** A vertical sidebar with mode selection buttons: "Comfort Mode" and "Expert Mode". Below these are icons and labels for various diagnostic functions: "TAL-Processing", "VCM", "Coding", "Coding-Verification", "NCD preparation", "FSC-Extended", "TS-Update", "NAV/ENT-Update", "OBD-CIN", "Certificate Management Extended", and "SFA / ECU-Mode Extended". At the bottom of this sidebar are buttons for "Editors & Viewers", "Data Handling", "External Applications", and "Personal view".
- Top Section:** Contains a "Vehicle Order" input field and a "Vehicle Profile" area. Below the "Vehicle Order" field are "Read", "Load", "Save", and "Edit" buttons.
- Bottom Section:** A large area labeled "SVT" (Software Verification Tool) for comparison. To its right is a control panel with sections for:
  - SVT Actual:** File Name input and "Read (VCM)", "Read (ECU)", "Load", "Save", "Edit" buttons.
  - SVT Target:** "KIS" section with "I-Step (shipm.:" and "I-Step (target):" dropdowns, a "Calculate" button, and "Calculation Strategy" options: "Single Flash" (radio button) and "Construction" (radio button, selected).
  - File Name:** Input field and "Read (VCM)", "Load", "Save", "Edit" buttons.
  - Buttons:** "HW-IDs from SVTactual" and "Detect CAF for SVE".
  - Coding:** "Code", "Read Coding Data", "Code NCD", "Code Default Values", and "Read CPS" buttons.
  - Checkbox:** "Parallel TAL-Execution".
  - SVT filter:** "All" dropdown and "SVT Reset" button.

At the bottom of the main workspace, there is a status bar with a legend: "Actual state" (blue), "Target state" (red), "Identical state" (green), "Hardware difference" (green with lightning bolt), and "NCD" (orange). The bottom right corner features a "Login" button and a status indicator "SWL-Sec: Logged out".



- The SVTs recorded in red letters in the picture above indicate the reference system state of an ECU. The blue ones indicate the actual state of the ECU. Black entries indicate SVTs with identical reference and actual state. If the hardware of an ECU is different, this is indicated by the arrow symbol.

- **GUI Elements**

<b>Vehicle Data</b>	
Vehicle Order	File containing a list of vehicle orders. For selecting a single vehicle order, this order needs to be activated by using the context menu item 'Activate FA'. If the list of vehicle orders contains only one element, this element will be automatically activated as soon as the file is loaded.
Read	Read out the actual vehicle order from the vehicle.
Load	Open a vehicle order from the file system.
Save	Opens a file dialog to save the vehicle order.
Edit	Change to the FA-Editor and open the vehicle order file.
Vehicle Profile	The Vehicle profile will be automatically generated from an activated Vehicle Order and displayed in a tree structure.
<b>SVT Actual</b>	
Read (VCM)	read out the vehicle order from the vehicle.
Read (ECU)	read out the vehicle order functionally
Load	Open a vehicle order from the file system.
Save	Opens an file dialog to save the SVT.
Edit	Change to the SVT-Editor and open the SVT
<b>KIS/SVT target</b>	
I-Step (shpim.)	Available shipment integration steps

I-Step (target)	Available target integration steps
Calculation strategy	Possible strategy to calculate the SVT (Single Flash   Complete Flash   Construction Progress).
File Name	The path of the SVT file.
Calculate	Calculates an SVT-Target by integration step and vehicle order (and SVT-Actual if exists).
Read (VCM)	Read SVT-Target from VCM
Load, Save	Load/save an SVT-Target from/to file system.
Edit	Edit in the corresponding editor.
HW-IDs from SVTactual	HWEL, HWAP und HWFR elements from SVTactual will be copied to the Emus in the target SVT.
Detect CAF for SWE	<i>not supported yet</i>
<b>Coding</b>	
Code	Code the ECU.
Read Coding Data	Read the coding data from ECU.
Code NCD	Code the ECU with the selected NCD.
Code Default Values	Code the ECU to the default state.
Read CPS	Read out the CPS.
Parallel Execution	TAL- If checked coding will be executed parallel.
Stop error	TAL on if checked the coding will stop if one error occurs.
<b>Filter</b>	
Filter	filter SVT tree by selected process class
SVT Reset	reset SVT/SVK-Ist/Soll and delete SVT tree

## **Reading Vehicle Order and VIN**

After reading the vehicle order the VIN will always be read too and put into the vehicle order (FAList/FA/FZAuftrag/Header/Vinlong). The vehicle order and the VIN will be read from the master ECU. If an error occurs they will be read from the backup ECU. In the status bar is shown where the vehicle order and the VIN were actually read from.

### **Note:**

Before starting the coding process (Code, Code NCD or Code Default Values), a BAT/HAF verification is carried out. It will be checked whether a BAT/HAF control unit is installed. If this is the case, the status of switch S1 is determined. When the switch is open, a message appears informing the user about possible risks for the coding process.

For SVT Target Calculation through KIS, an I-Level container (PDX) with KIS data must be prior imported. The FA corresponding to the container series, must be imported as well.

---

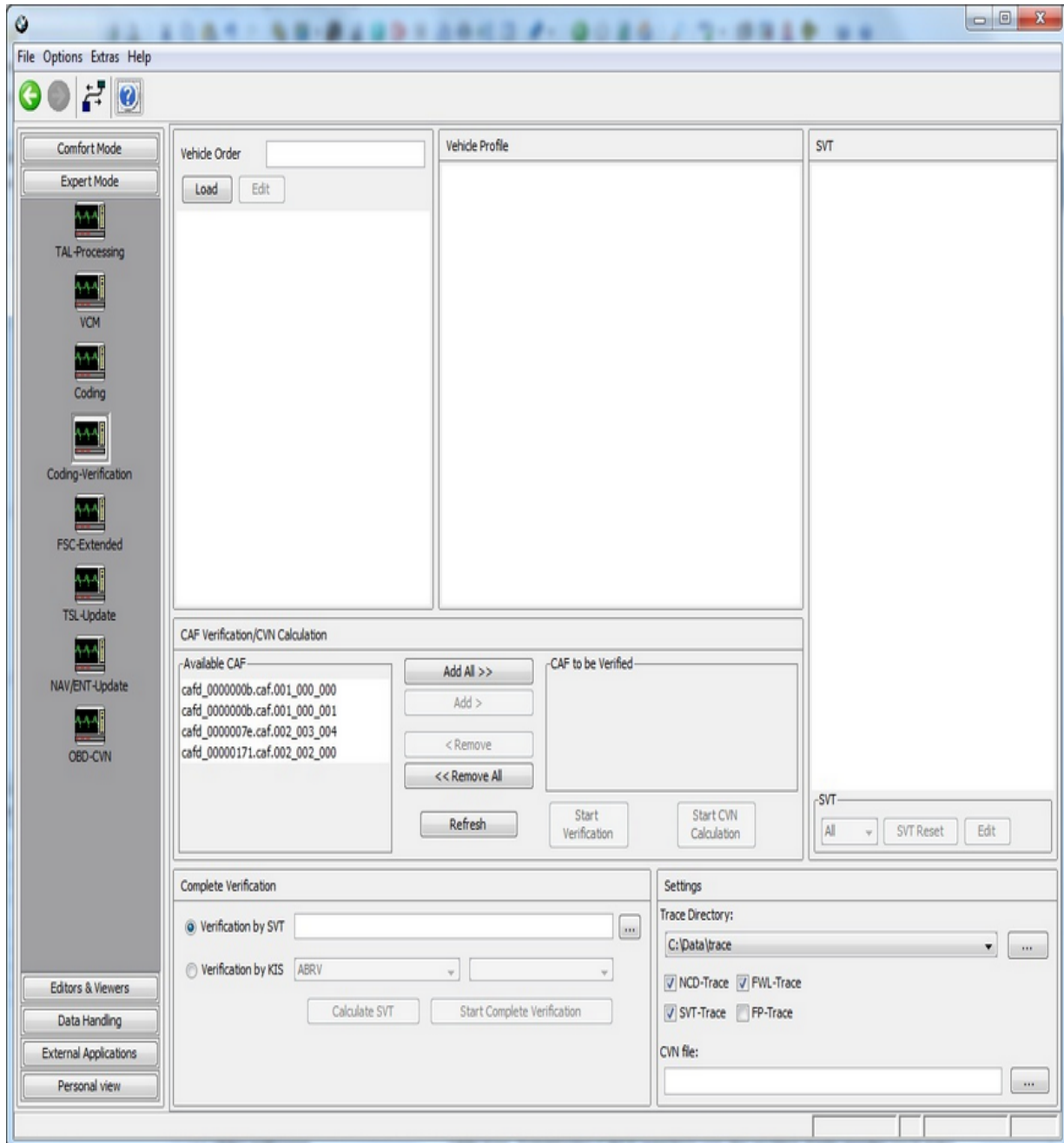
This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

# Coding-Verification

[Top](#) [Previous](#) [Next](#)

In module Coding-Verification is checked offline, if the discrete parts ( FA, CAF, SVT) match together. In detail there are verification of CAF, entire verification by SVT and by KIS (in the latter also an SVT is created).

The CVN calculation computes the Sub CVNs of CAF files and saves the result in a file.



-

-

## **GUI Elements**

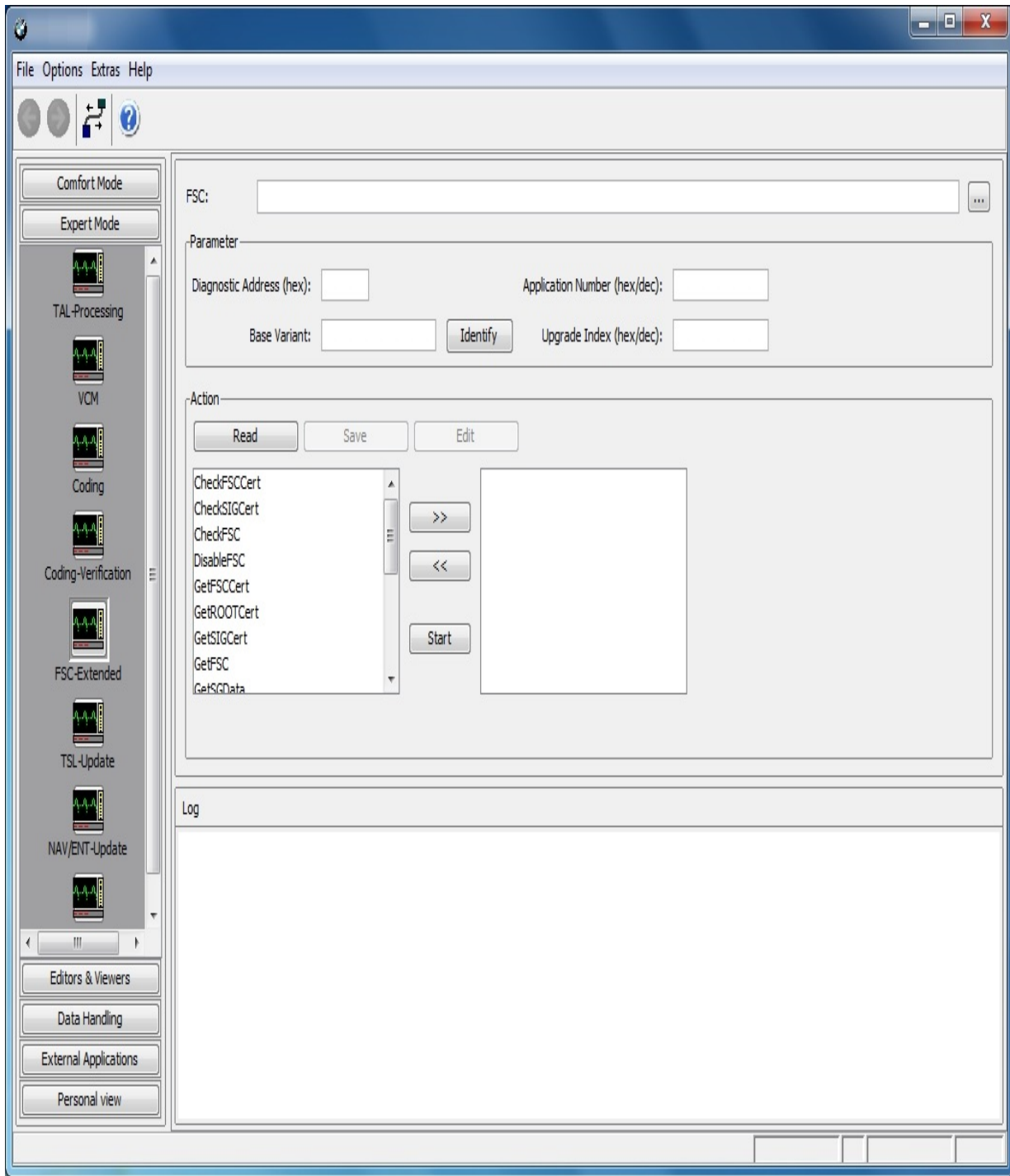
<b>Vehicle Data</b>	
Vehicle Order	File containing a list of vehicle orders. For selecting a single vehicle order, this order needs to be activated by using the context menu item 'Activate FA'. If the list of vehicle orders contains only one element, this element will be activated by default as soon as the file is loaded.
Vehicle Profile	The Vehicle profile will be automatically generated from an activated Vehicle Order and displayed in a tree structure.
Load	Load file with list of vehicle orders from file system.
Edit	Load current Vehicle Order to FA/FP editor
<b>CAF-Verification</b>	
Available CAFs	This window displays the available CAFs in the proper file name format.
CAFs to be verified	These transferred CAFs will be verified when the "Start Verification" button is pressed.
Add (all) >>	Transfer selected CAFs from the left hand side to the right hand side respectively all of them.
<< Remove (all)	Remove items on the right hand side to the left side.
Refresh	Refresh the two CAF windows.
Start verification	Start the process of the CAFs' verification.
<b>Entire Verification</b>	
Radio button	In "Verification by KIS" additionally an

"Verification by SVT" and "Verification by KIS"	SVT is calculated and afterwards verified
"..."-Button	Load SVT from the file sytem for verification by SVT.
Combo box	For selecting the integration step for calculating of SVT.
Start Entire Verification	This button starts "Verification by SVT" or "Verification by KIS" depending on the state of radio button.
Start CVN Calculation	Starts the CVN calculation an writes the result to the file selected by "Settings/CVN file"
Calculate SVT	Calculate SVT from integration step and further data.
<b>Settings</b>	
Trace Directory	Specification of an existing trace path (by combo box) or input of a new path by three-dot-button-dialogue.
NCD, FWL, SVT, FP trace	The discrete types of traces can be selected in any combination. For "Verification by KIS" the SVT trace is strictly recommended (due to SVT creation).
CVN file	Target file for CVN calculation in ObdCvnFaList format.

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

The module FSC Extended in expert mode serves for read out, check and sign of FSC. In addition to that single jobs can be executed in batch mode.

Corresponding to the topic FSC there are other masks in E-Sys. The module [Freischaltcode](#) (FSC) serves for inquiring the FSC State, for writing and (de-)activating of FSC. FSC's can be displayed and edited in [FSC-Editor](#) easily.



## GUI Elements

FSC File	The FSC to be written
Key	The key corresponding to the FSC
Diagnosis Address	of the ECU



Software-ID	the software identifier
Base Variant	Name of ECU variant
Upgrade Index	of FSC
Read FSC	The FSC in the ECU is read.
Edit FSC	Forward FSC to module FSC editor
Check FSC	Check FSC (using public key)
Sign FSC	The current FSC is signed using the specified (private) key.
">>", "<<"	The jobs in the left hand window, selected by the mouse are copied to the right hand window.
Start	The jobs residing in the right hand window are started.
Log	The log window displays the state of the currently processed action.

---

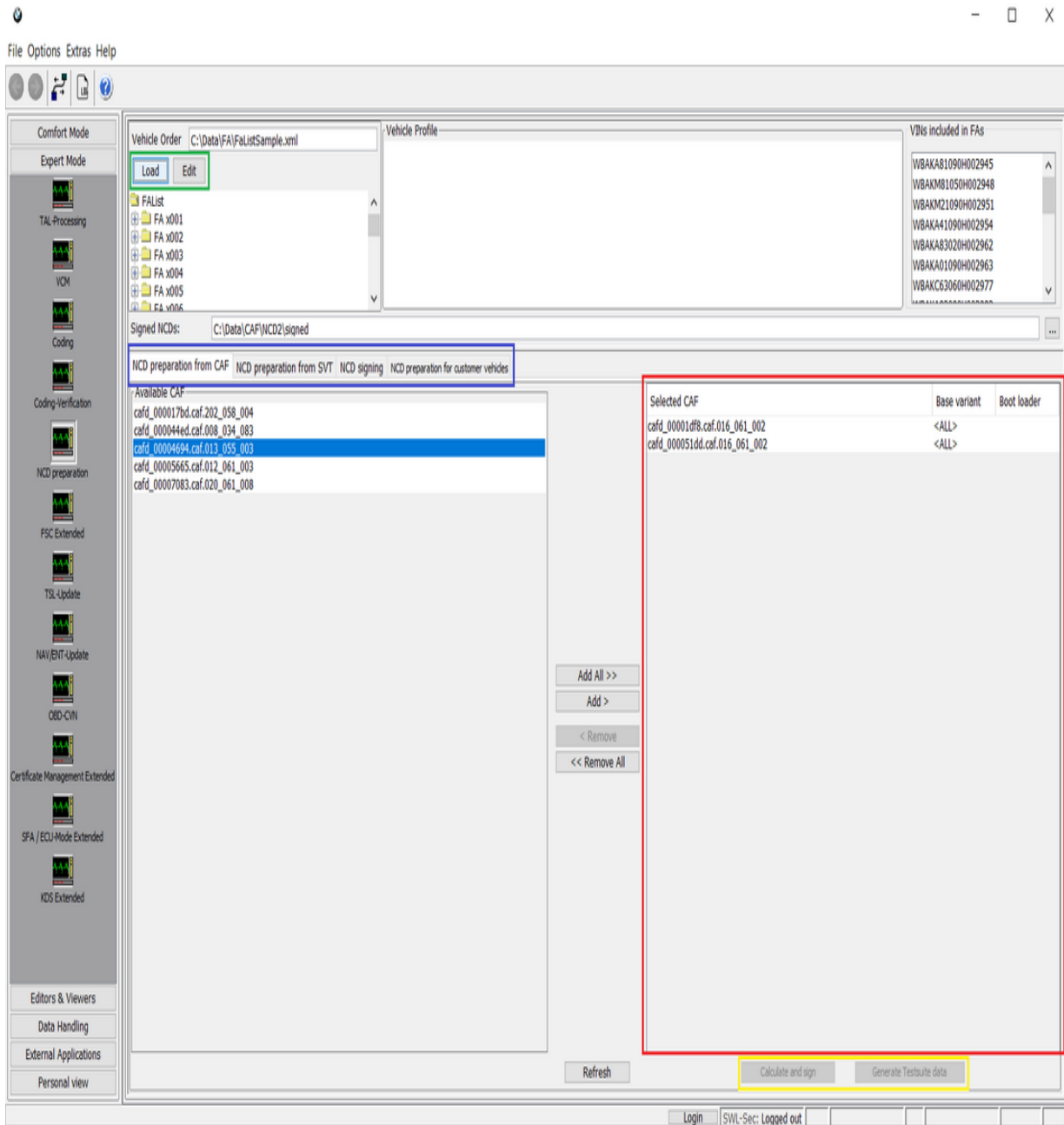
This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

The new coding concept (secure coding) required, that all needed NCDs have to be signed before they can be transferred to the ECU.

Therefore you have now with these new module the possibility to signing the NCDs.

**Attention:**

*If no pdx-container is imported, the user has to perform an container import and restart e-sys afterwards! Otherwise the user is not able to use these module correctly!*

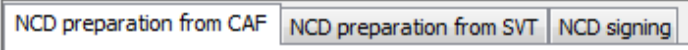


You have the options to sign already local available NCDs (tap "NCD signing") or E-Sys has to calculate the NCDs before they can be signed (tap "preparation from SVT" or "preparation from CAFD").

## **GUI-Elements**

Vehicle-order (FA)  
(green marking)

Load an existing FA. You  
can also edit it, in these

	<p>case you are going to switch the module automatically.  As soon as a FA is loaded, all VIN-entries included in the vehicle order, will appear in the upper right section of the display.  (only if tap "NCD preparation from CAF" or "NCD preparation from SVT" are selected).</p>
<p>Vehicle-profile (FP)</p>	<p>The FP is shown as soon as the FA has been activated.  If more than one FA-entry are selected, no FP will be displayed.</p>
<p>Choose the location for the signed NCDs</p>	<p>If you are going to edit the signedNCD location the option-setup is going to appear. After you changed the location click the Button "OK" and the new location will be visible in these module.</p>
<p>Tap for the three different ways to sign NCDs  (blue marking)</p> 	<p>Depending on the tap the NCDs are going to be calculated and signed</p> <ul style="list-style-type: none"> <li>· "Preperation form SVT": NCDs are going to be calculated and signed</li> <li>· "NCD signing": only perform the signing for local NCDS</li> </ul>

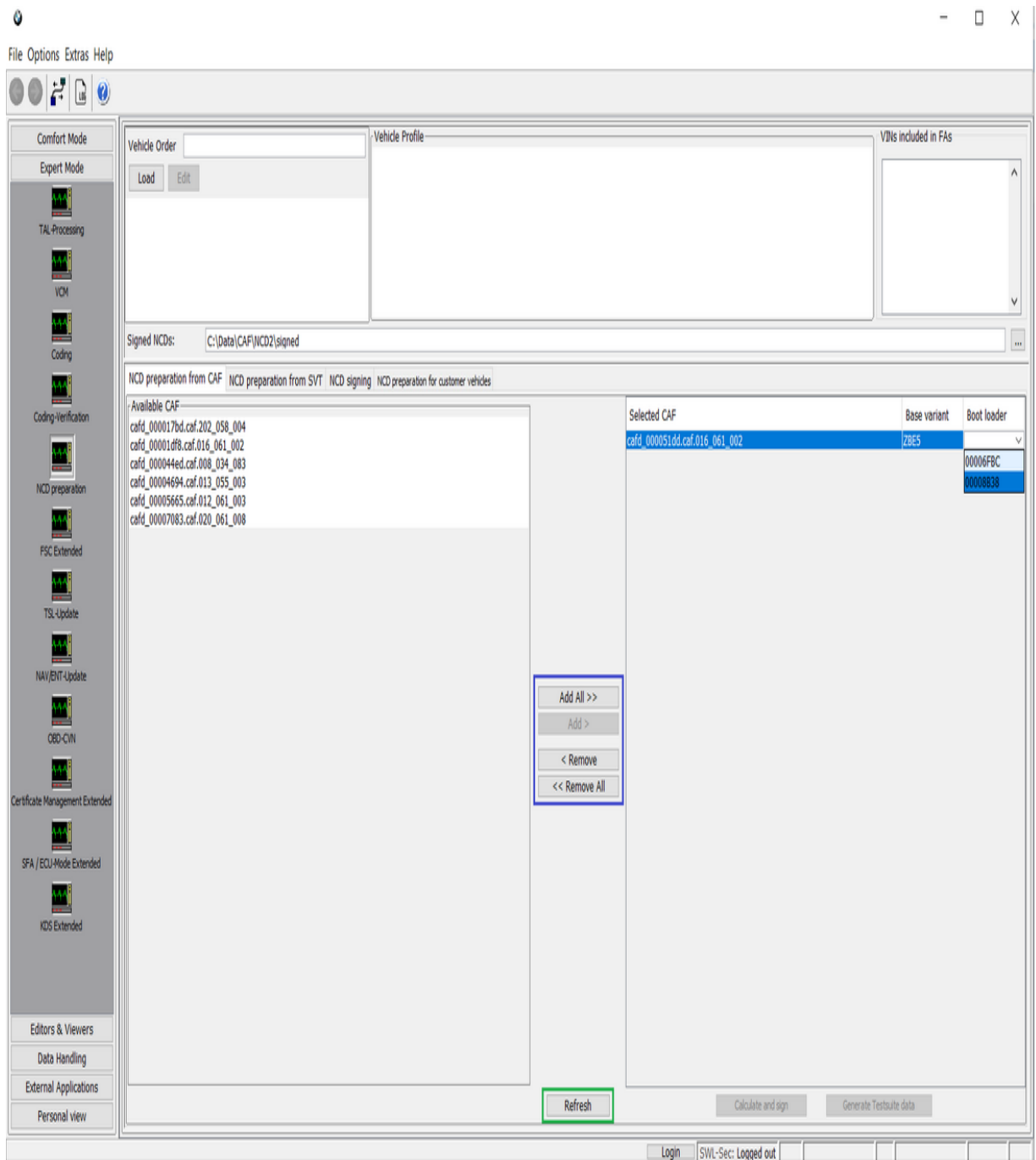
	<ul style="list-style-type: none"> <li>· "Preperation from CAFD": NCDs are going to be calculated and signed</li> </ul> <p>In all three tabs, NCDs are calculated and signed NCDs for shipment in last step.</p>
Backend-Communication ( <a href="#">secure coding</a> )	Call the backend (SWL-Sec2) and set the backend connection options
Selected Files (CAF/NCD) (red marking)	As soon as a CAF or NCD was added, the user can define a bootloader id for each entry.
Calculate and sign (yellow marking)	E-Sys is trying to perform the signing of the NCDs locally (offline). If the local signing is not able the online signing process is automatically performed. Therefore it is mandatory to be logged in at the Backend.
Generate Testsuite Data (yellow marking)	<p>Generation and signing of NCD test sets as preparation for FAT or AmTS coding tests. A ZIP archive is created containing the following data:</p> <p><u>For Coding 2</u></p> <ul style="list-style-type: none"> <li>· FA used</li> </ul>

- NCDs signed with FAT-VIN
- NCDs signed with FA-VIN
- shipping NCDs signed with FA-VIN
- If the CAF contains ASIL functions, an NCD with a modified coding value is calculated for each safety-relevant function and signed with FA-VIN.

### For Coding 3

- FAs and FPs used
- 4 NCD variants that have been calculated and signed differently
- shipping NCDs signed with Default-VIN
- If the CAF contains ASIL functions, an NCD with a modified coding value is calculated for each safety-relevant function and signed with FA-VIN.

## **NCD preparation form CAFD**



The available CAFDs are located at the used psdzdata.

By the action buttons (blue marked) can the CAFDs be added or removed from the list of CAFDs, which are going to be signed.

The refresh button (green marking) is reloading the file location of the calculated CAFDs. It has the side-effect that all CAFDs, which are added and located in these order are been removed automatically.

Following steps have to be fulfilled for activating the "Calculate and sign" button:

- A vehicle-order file with only one entry has to be loaded. This entry will be automatically activated when loading the file.
- As an alternative for loading a single vehicle-order file, one or more vehicle-orders have to be selected in the tree view.
- All added CAFDs have a bootloader-ID.

The bootloader-ID can either be entered manually, in these case the base variant 'All' has to be chosen (default value). Or the specific base variant can be selected in the drop down list, in these case are only the BTLID-ID chose able which are defined for the chosen base variant.

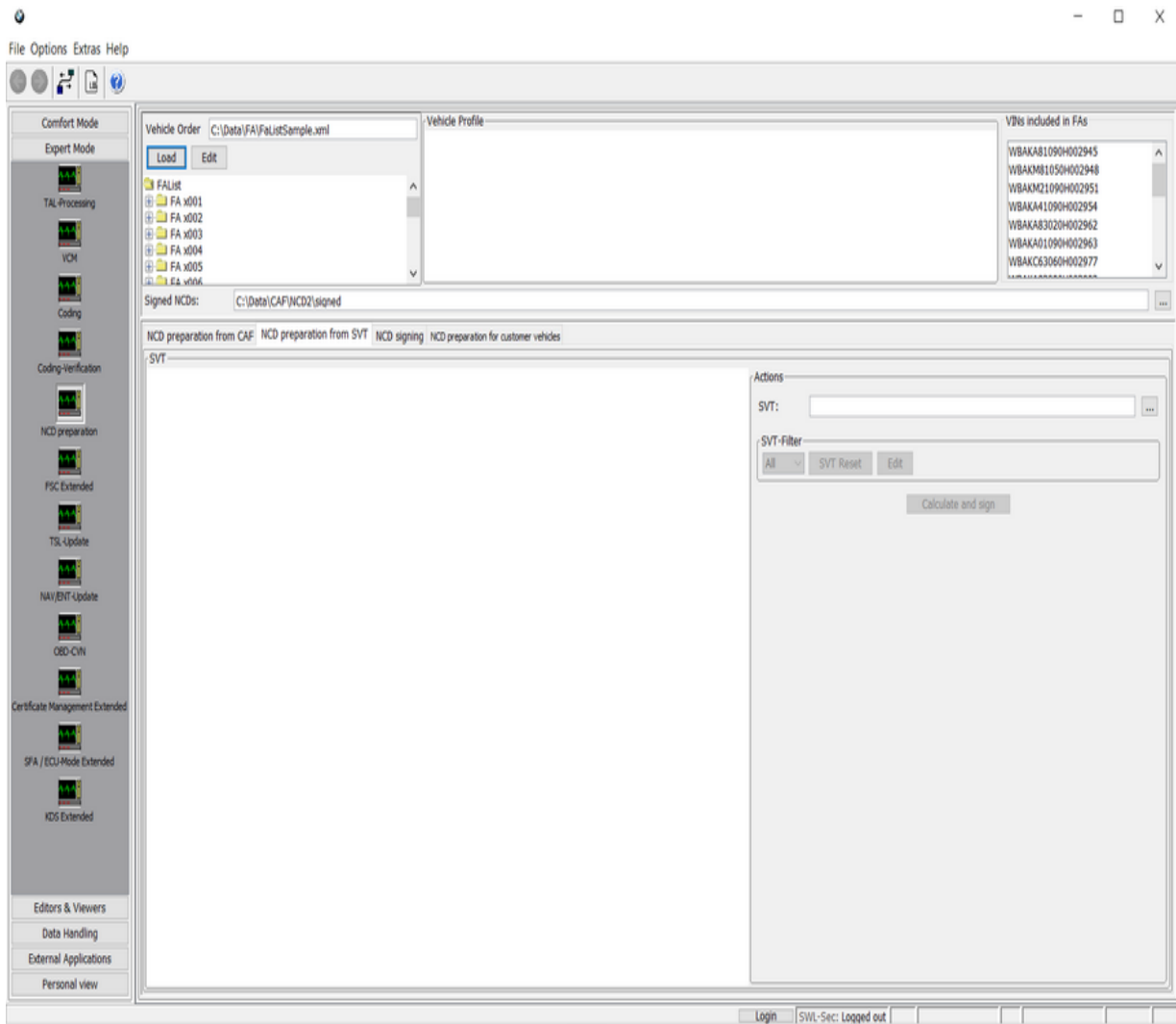
The signing is going to be performed offline/online and saved at the defined location.

Subsequently, NCDs for shipment are calculated and signed and saved as well.

- **NCD preparation form SVT**

-





After a SVT is loaded has the user the options to mark multiple Emus or the root-element.

Following steps have to be fulfilled for activating the "Calculate and sign" button:

- A vehicle-order file with only one entry has to be loaded. This entry will be automatically activated when loading the file.
- As an alternative for loading a single vehicle-order file, one or more vehicle-orders have to be selected in the tree view.
- At least one ECU has been selected.

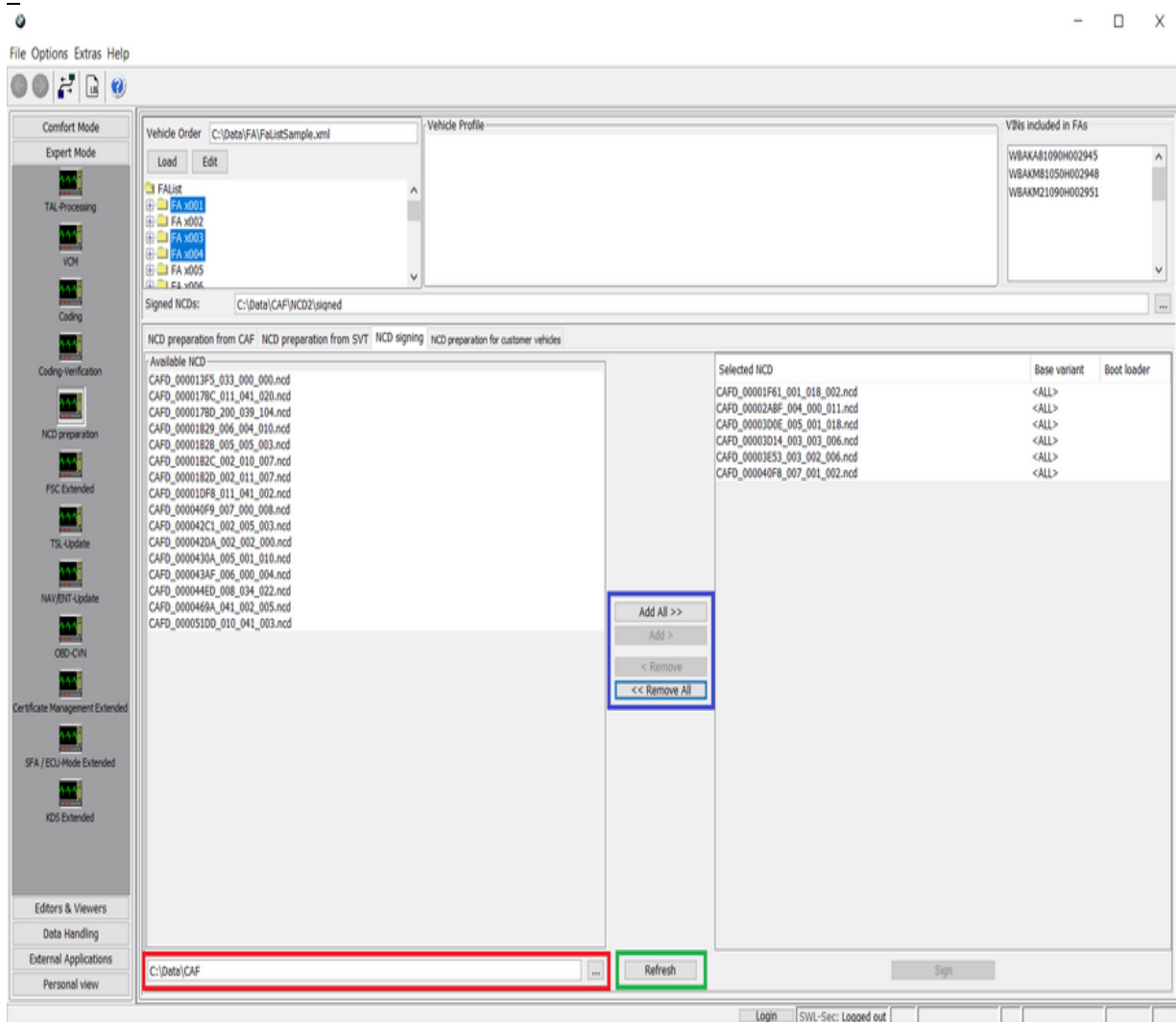
E-Sys is now performing the calculation and signing of the chosen NCDs.

**Attention:**

The NCDs can only be calculated, if the ECU has a valid CAFD (CAFD\_ffffffff\_255\_255\_255 is in these aspect a non valid CAFD).

The CAFD is required to be located at the used psdzdata. Subsequently, NCDs for shipment are calculated and signed.

**NCD signing**



In the red marked row can the user set the file location of the already calculated NCDs (NCDs can be calculated at [Coding-Verification](#)).

By the action buttons (blue marked) can the NCDs be added or removed from the list of NCDs, which are going to be signed.

The refresh button (green marking) is reloading the file location of the calculated NCDs. It has the side-effect that all NCDs, which are added and located in these order are been removed automatically.

The signing button can only be activated, if following preconditions are fulfilled:

- A vehicle-order file with only one entry has to be loaded. This entry will be automatically activated when loading the file.
- As an alternative for loading a single vehicle-order file, one or more vehicle-orders have to be selected in the tree view.
- At least one ECU has been selected.

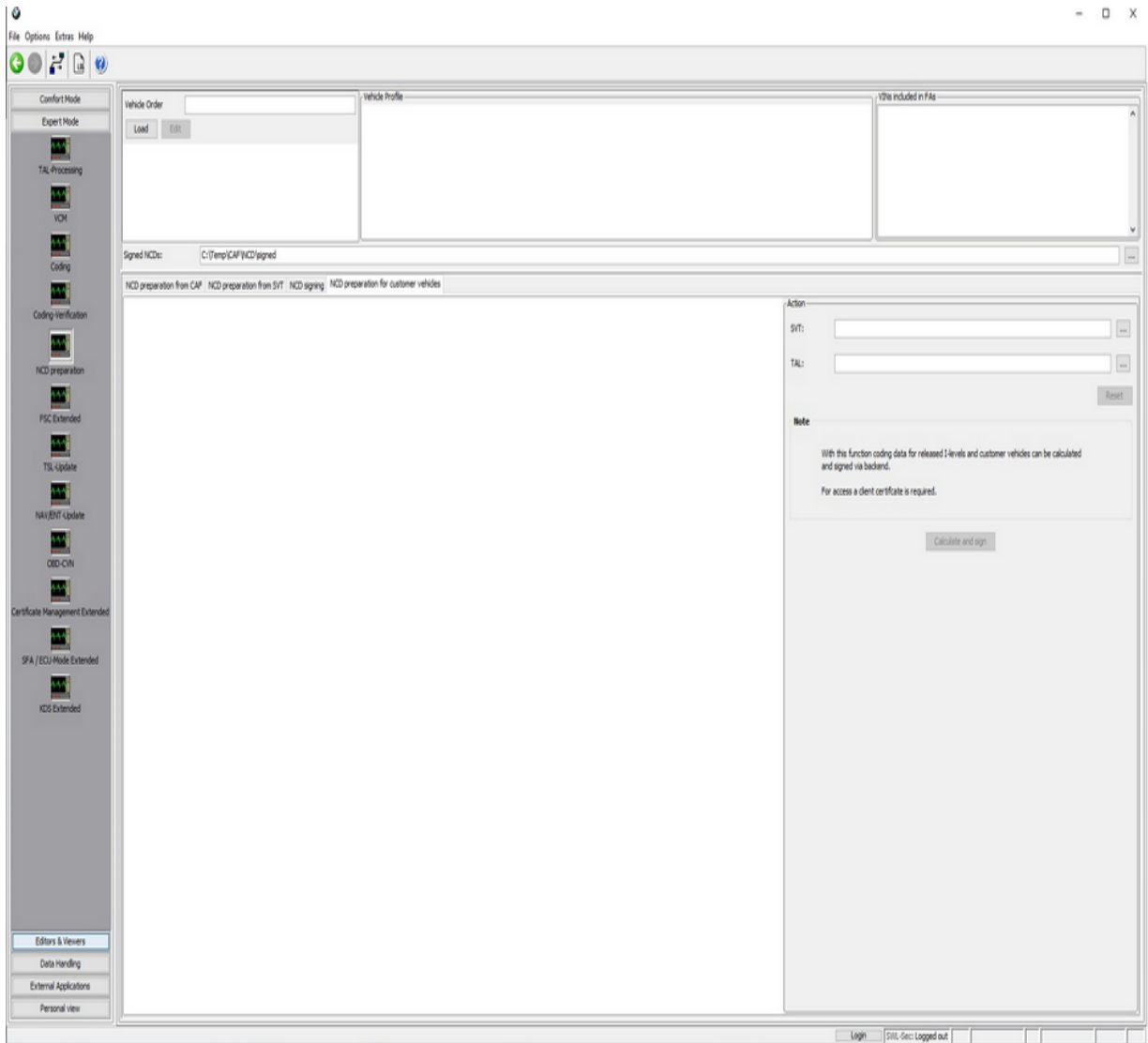
The bootloader-ID can either be entered manually or it can be selected from the list after selecting the basis variant name. The signing is going to be performing (offline/online) depends if all necessary data are local available. Subsequently, NCDs for shipment are calculated and signed.

Hint:

- Currently, NCD-data for Coding 2 and Coding 3 can be signed in this module. NCDs for Coding 1 are not supported.

**NCD preparation for customer vehicles**

-



Following steps have to be fulfilled for activating the "Calculate and sign" button:

- A vehicle-order file with only one entry has to be loaded. This entry will be automatically activated when loading the file.
- As an alternative for loading a single vehicle-order file, one or more vehicle-orders have to be selected in the tree view.
- A SVT or a TAL have to be selected.

E-Sys is now performing the calculation and signing of all NCDs for the CAFS contained in the SVT or the TAL..

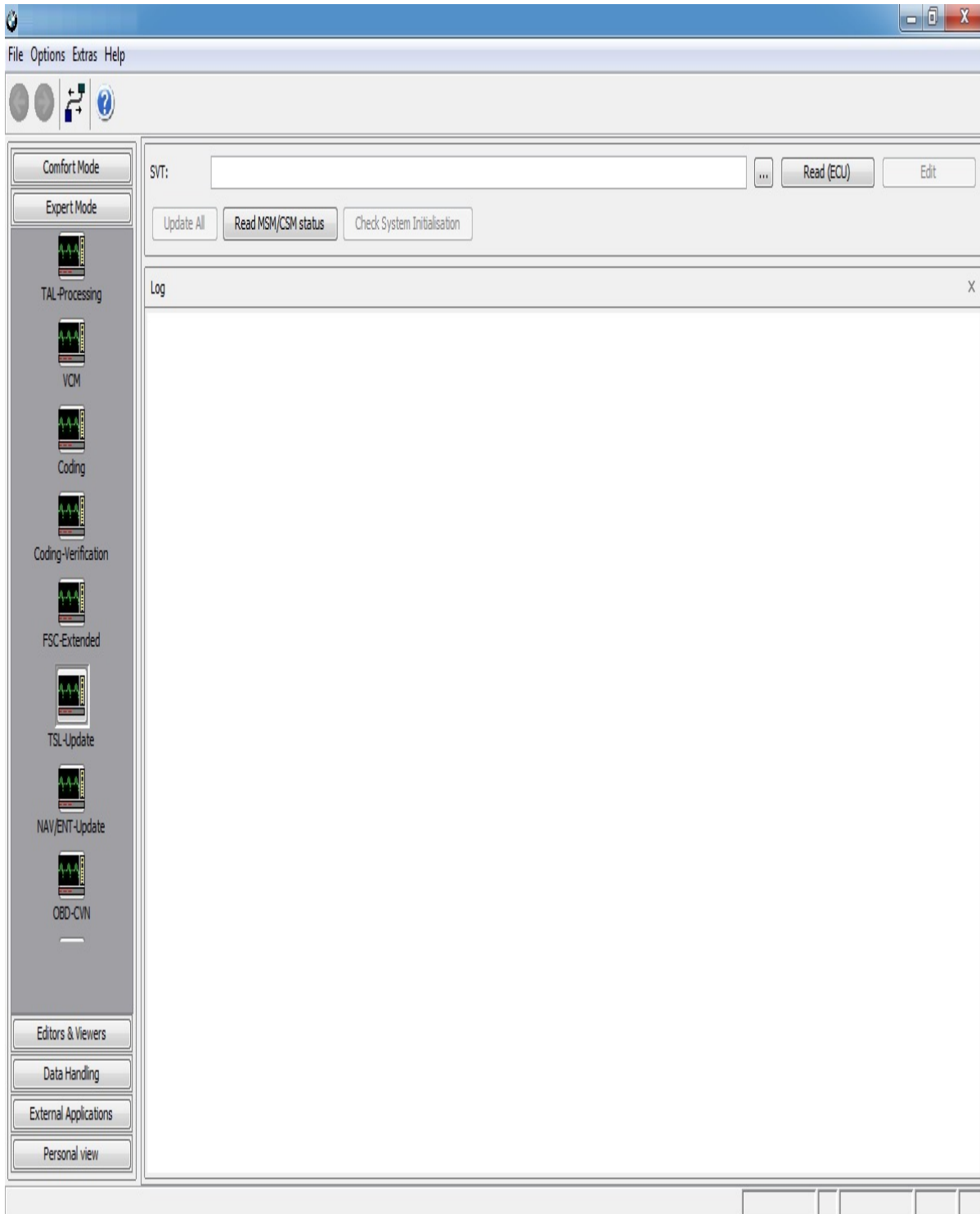
Hint:

- The calculation and signing of the NCDs will be done in the SCB backend
- As a necessary precondition for the access to the SCB backend, a TIS-client Certificate has to be imported first
- The URLs used for connecting to the SCB have to be defined in the esys.properties file by using the property key `scb.backend.url`
- In cases where more than one URLs are available, the complete list of URLs has to be assigned to the property key in a comma separated manner

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

In a vehicle there is a MSM (Master Security Module) and several Emus with CSM (Client Security Module). Putting or updating of transport key into MSM using PSdZ is called transport key update (Transportschlüssel-Update (**TSU**)). This is the job of the present mask.



## **GUI Elements**

SVT	The actual system installation table to be loaded by three-dot-button-dialogue.
-----	---

Read (ECU)	Functional read out of the discrete Emus.
Edit	Change to SVT-Editor and show the opened SVT
TSL-Update	Execute the TSL update.
Read MSM/CSM status	Reads the status and writes the result (or error messages) to the Log view.
Check System Initialization	Execute PSdZ method msm_F25_checkMsmSystemInitialisation.

---

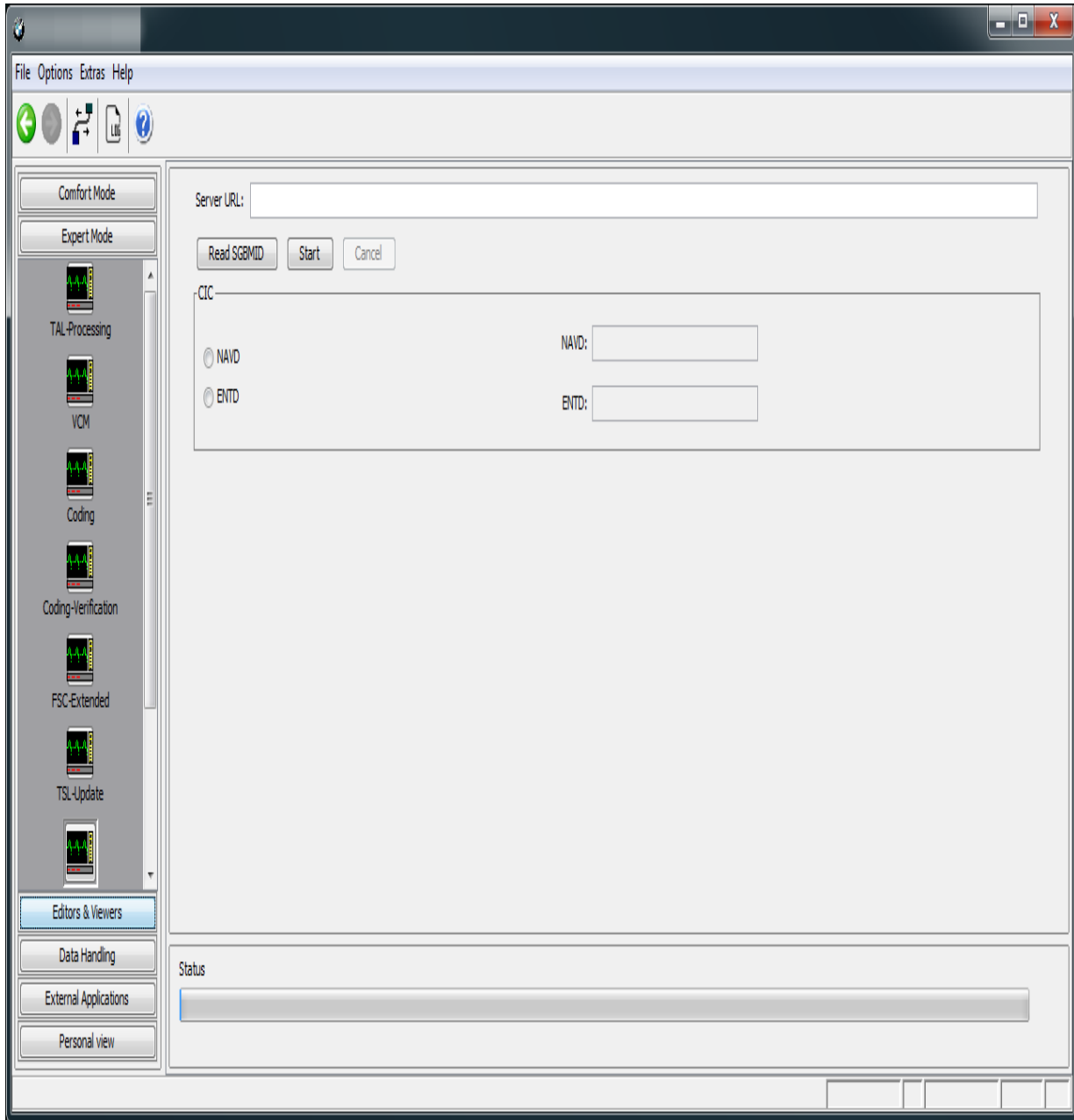
This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.



# NAV/ENT Update

[Top](#) [Previous](#) [Next](#)

Navigation and entertainment data (NAV, ENT) are placed on a hard disk drive (HDD) in the Head unit. These data can be updated using the module NAV/ENT Update.



## Current Data

By pressing "Read SGBMID" you can see the versions of the current navigation and entertainment data. Therefore the current SVT is read from VCM and the

SGBMIDs with process classes NAVD and ENTD are shown.

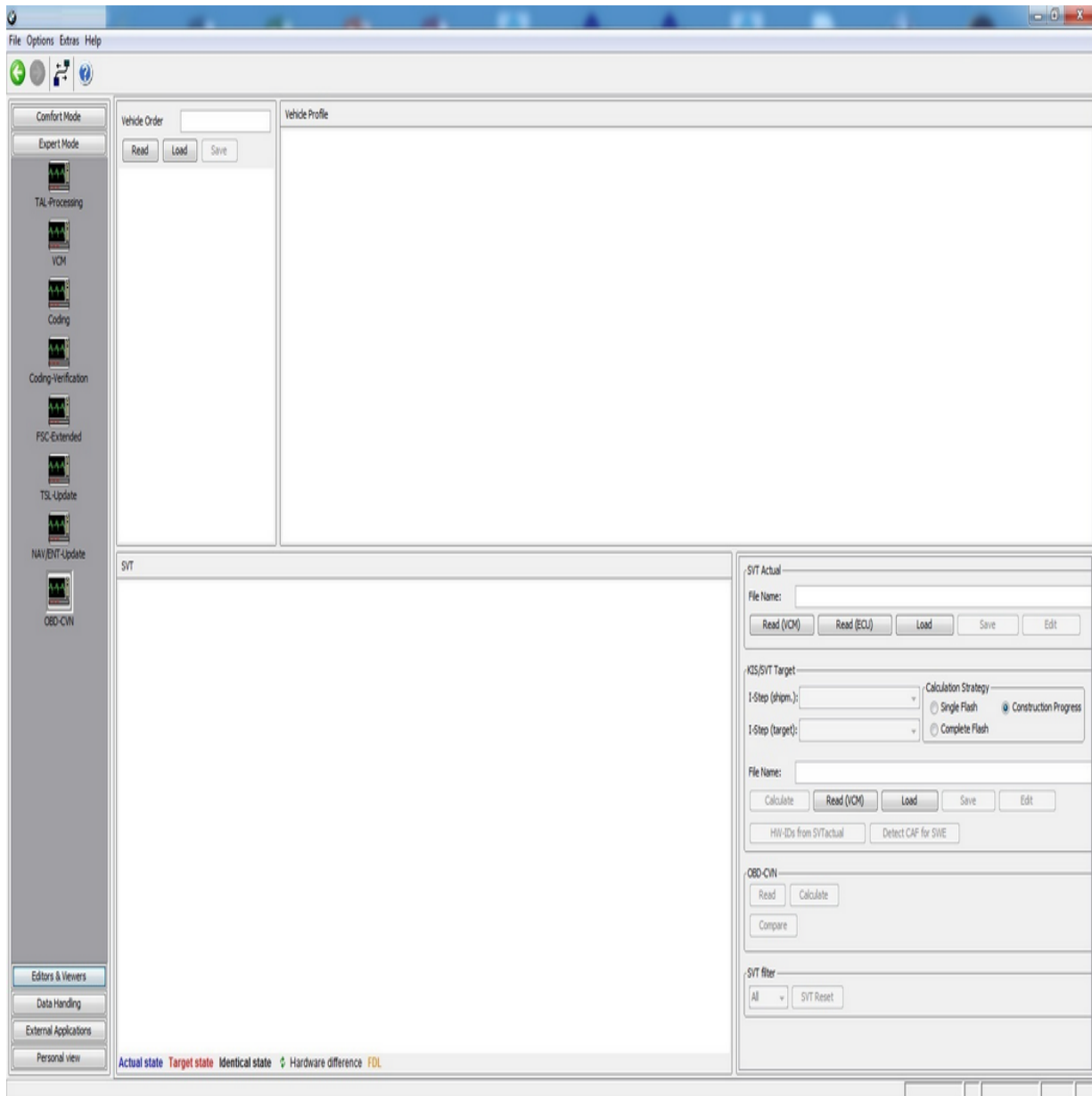
**Update**

After writing the address of the server, where the newest navigation and entertainment data are stored, into the appropriate text field, the update process can be started by pressing the "Start" button. But the data will be updated only, if the appropriate process class is selected. By pressing the "Cancel" button the update process could be aborted.

---

This help file was created with an unregistered evaluation copy of Help & Manual.  
© EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

The module OBD-CVN reads CVNs (Calibration Verification Number) from Emus, calculates CVNs over all OBD (On-Board Diagnostics) relevant SWEs.



The SVTs recorded in red letters in the picture above indicate the reference system state of an ECU. The blue ones indicate

the actual state of the ECU. Green entries indicate SVTs with identical reference and actual state.

**GUI Elements**

<b>Vehicle Data</b>	
Vehicle Order	File containing a list of vehicle orders. For selecting a single vehicle order, this order needs to be activated by using the context menu item 'Activate FA'. If the list of vehicle orders contains only one element, this element will be automatically activated as soon as the file is loaded.
Read	Read out the actual vehicle order from the vehicle.
Load	Open a vehicle order from the file system.
Save	Opens a file dialog to save the vehicle order.
Vehicle Profile	The Vehicle profile will be automatically generated from an activated Vehicle Order and displayed in a tree structure.
<b>SVT Actual</b>	
Read (VCM)	read out the vehicle order from the vehicle.
Read (ECU)	read out the vehicle order functionally
Load	Open a vehicle order from the file system.
Save	Opens an file dialog to save the SVT.
Edit	Change to the SVT-Editor and open the SVT
<b>KIS/SVT target</b>	
I-Step (shpim.)	Available shipment integration steps
I-Step (target)	Available target integration steps
Calculation strategy	Possible strategy to calculate the SVT (Single Flash   Complete Flash   Construction Progress).
File Name	The path of the SVT file.
Calculate	Calculates an SVT-Target by integration step

	and vehicle order (and SVT-Actual if exists).
Read (VCM)	Read SVT-Target from VCM
Load, Save	Load/save an SVT-Target from/to file system.
Edit	Edit in the corresponding editor.
HW-IDs from SVTactual	HWEL, HWAP und HWFR elements from SVTactual will be copied to the Emus in the target SVT.
Detect CAF for SWE	<i>not supported yet</i>
<b>Coding</b>	
Read	Read CVNs of selected Emus.
Calculate	Calculate CVNs from the Sub-CVNs of SWEs which are OBD relevant.
Compare	Compare calculated and read CVNs.
<b>Filter</b>	
Filter	filter SVT tree by selected process class
SVT Reset	reset SVT/SVK-Ist/-Soll and delete SVT tree

### **Reading Vehicle Order and VIN**

After reading the vehicle order the VIN will always be read too and put into the vehicle order (FAList/FA/FZAuftrag/Header/Vinlong). The vehicle order and the VIN will be read from the master ECU. If an error occurs they will be read from the backup ECU. In the status bar is shown where the vehicle order and the VIN were actually read from.

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved.

This message will not appear if you compile this help file with the registered version of Help & Manual.

### **ATTENTION**

The functionalities provided in this module which include direct server access require to have several preconditions to be fulfilled:

- The system which runs E-Sys must be in a BMW network segment in which the backend systems (CBB) are available.
- There must be a valid E-Sys client Certificate imported into E-Sys.
- At least one URL of a valid backend system (CBB) must be configured in E-Sys.

A more detailed set of instructions can be found here: [HOWTO](#) (only available within BMW network).

For further questions please contact the support service: [support-security-systemfunktionen@bmw.de](mailto:support-security-systemfunktionen@bmw.de)

### **Introduction:**

As an enabler for new functions and to increase the security of existing functions, individual Certificates are required in the treatment of vehicles. From SP2018, individual Certificates and their binding to the respective vehicle must be installed both in the factory and in the service. In addition, bindings between ECU devices must be distributed. This functionality is provided by this module.

### **Overview Certificate Management Module:**

The Certificate management module has 5 tabs. The individual functions are explained in the following sections:

- 1) Get, write and check data
- 2) Write data individually
- 3) Read data individually
- 4) Calculate otherBindings
- 5) Edit CBB-Transfer data
- 6) SecOC
- 7) IPSec
- 8) MACSec

-

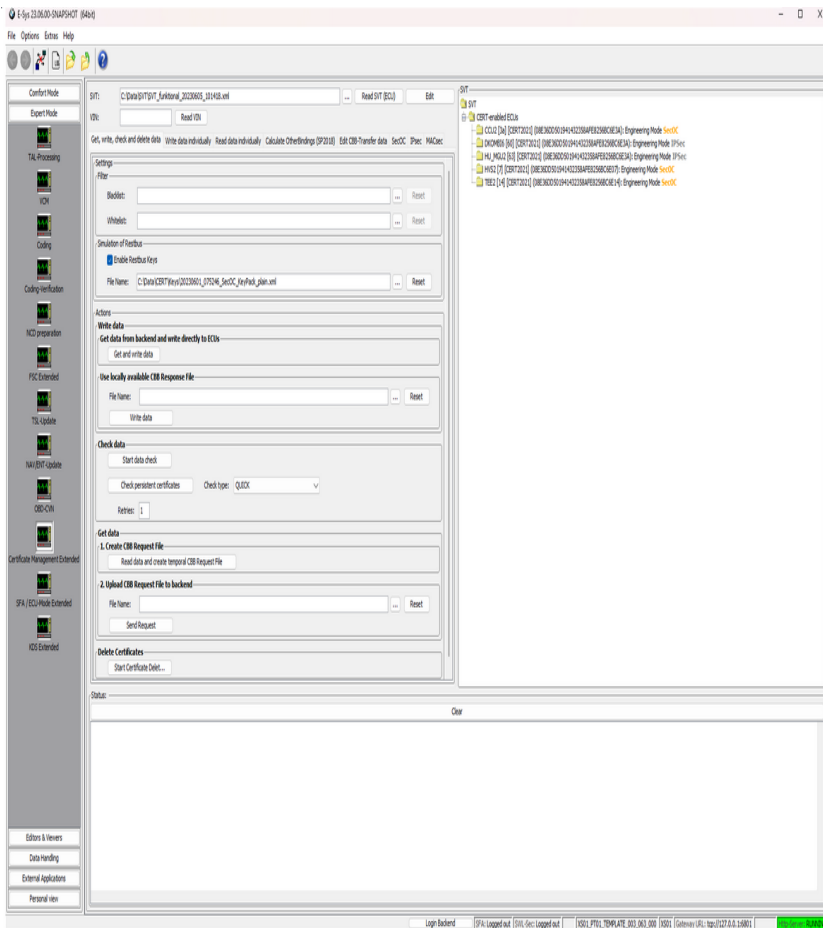
-

#### **1) Get, write, check data and delete data:**

Supports beside the functionality of the [Comfort Mode](#) the possibility of using black- / white-lists.

Format of the blacklist:    0x63,0x61,0x60,... (comma-separated diagnostic addresses in hex format)

File-Type:                    .txt (e.g. blacklist.txt)



It is possible to use either a loaded file as white-list or just click at the Emus you want to treat (in the svt view on the right).

**ATTENTION:** It is not possible to have a loaded filter file and selected Emus in the svt view.

If no filter is selected all Emus will be treated.

### **Simulation of residual bus:**

Checkbox 'Enable residual bus Keys' allows the storage of symmetrical keys for the ECUs in plain text.

With this option selected, the keys will be stored into XML-files.

Name and path for the XML-files can be defined by the user.

### **Start data check:**

Performs the Certificates check of the Emus. The result is logged in a local result-file as well as displayed in the status panel.

### **Check persistent certificates:**

Performs a check of persistently stored certificates and keys in the ECUs, if available. The results are displayed in the status panel.

Two types of checks can be performed:

- Quick: a fast/quick check



- Full: a complete check of certificates, including the certificates signatures

**Create a local CSR-file:**

Generates an [JSON] request-file that can either uploaded in the B2B-portal to get th response or send with E-Sys to the backend (E-Sys only possible in BMW-Intranet).

**Delete certificates:**

Deletes all certificates in the ECUs.

-

**2) Write data individually.**

The user has the possibility to write one memory type in particular.

In order to use the function, a valid XML file containing the type of memory object selected i the table must be specified for the respective ECU in the table.

**ATTENTION:** The data format of the loaded object is xml and not JSON. The response fil can't be used in this slide.

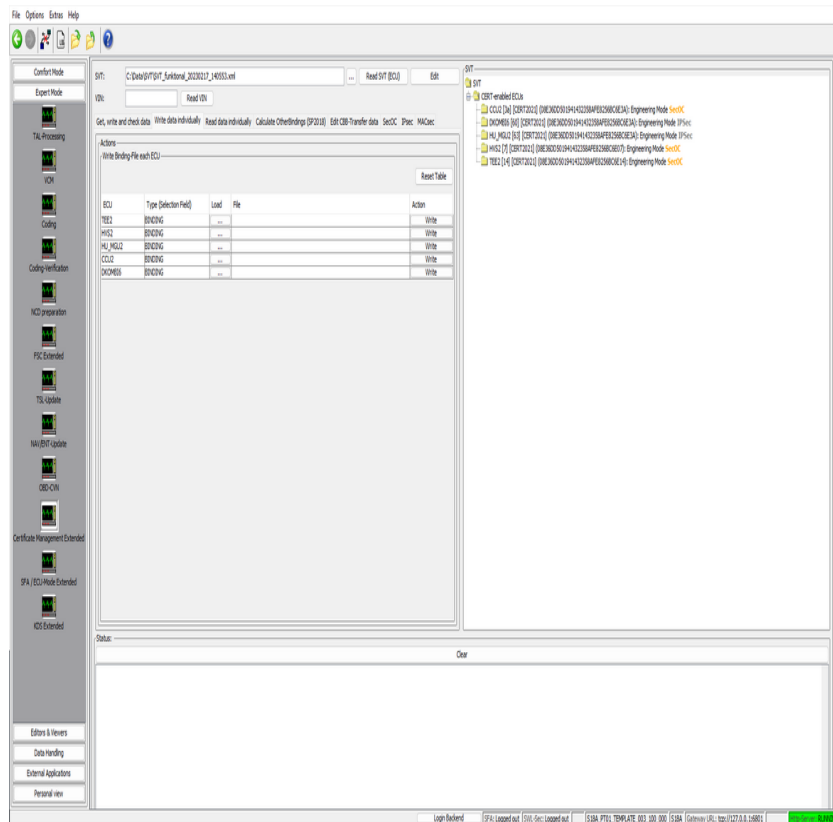
Each ECU can only be written with one memory type (Type1, 2, 5) at once.

In the table on the left side has the user the possibility to

-

**Description of types:**

Certificates / CSRs (Type 1) :	0
Binding (Type 1) :	1
OtherBindings (Type 1) :	2
Certificates / CSRs (Type 2) :	3
Binding (Type 2) :	4
Sec_Oc_Keylist:	5



### **3) Read data individually:**

Reads the selected Certificate type from the configured ECU devices and displays the result in the status window.

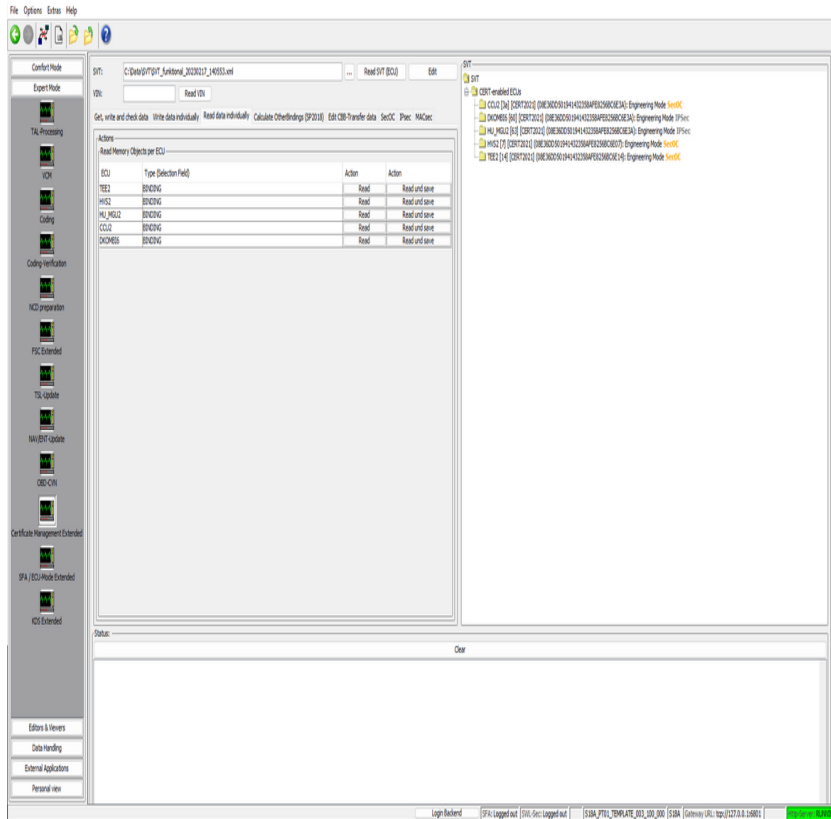
Generates for each ECU to be handled a file in XML format, which contains the entire Certificate container of the respective ECU device. Default storage location is the CERT directory in the E-Sys Data directory.

In the selected storage location, one or more XML files are now dependent on the selected memory object type

For each ECU can only be read out on memory type at the time.

#### Example of stored XMLs:

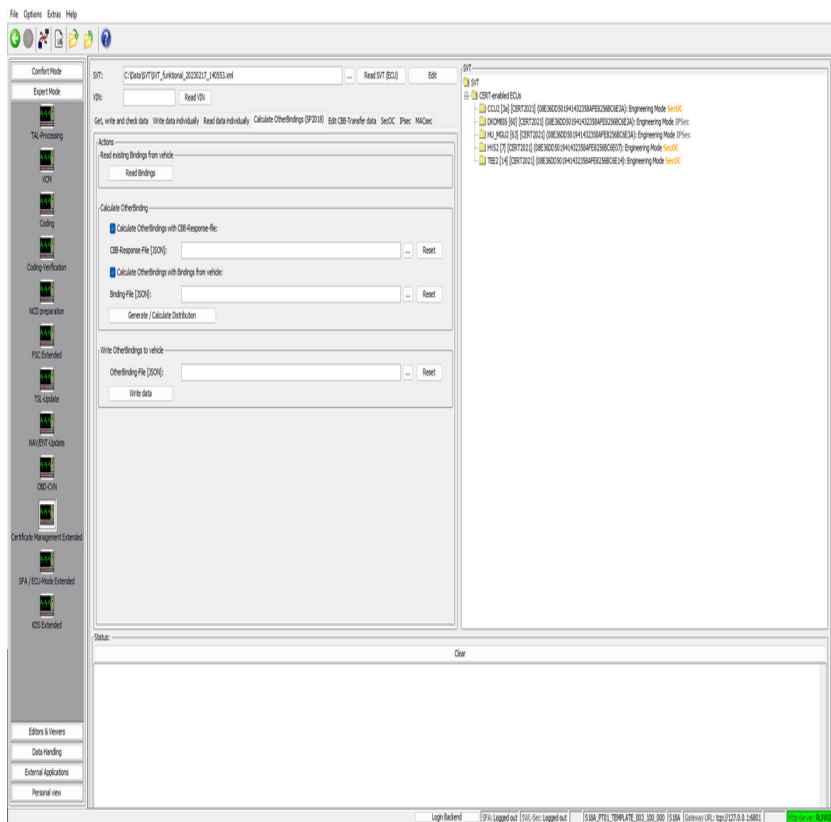
```
" BindingContainer_<Basisvariante>_<DiagAdr[Hex]>_fromVehicle_<Zeitstempel>.xn
"
"
CertificateContainer_<Basisvariante>_<DiagAdr[Hex]>_fromVehicle_<Zeitstempel>.xn
"
```



#### **4) Binding Distribution: (only SP2018 Emus)**

A binding distribution can be created in this tab.

The Binding Distribution contains the information about which Emus are allowed to establish a secure communication with each other.



The "Calculate Distribution" button calculates the OtherBindings, which contain the information which ECU devices may communicate with each other in the vehicle. For this purpose, the user selects a location for the BindingDistribution file. Default storage location is the CERT directory in the E-Sys Data directory. Here, the user has the possibility to influence the calculation of the distribution with the check boxes to the right of the files to be loaded.

Both check boxes active: PSdZ calculates the OtherBindings by using the bindings of the backend with the bindings from the vehicle for calculation.

Checkbox "from backend" active and "from vehicle" inactive: PSdZ does not get any readout binding set from the vehicle to the calculation.

Checkbox "from vehicle" active and "from backend" inactive: PSdZ will only allow the bindings from the vehicle to be included in the calculation.

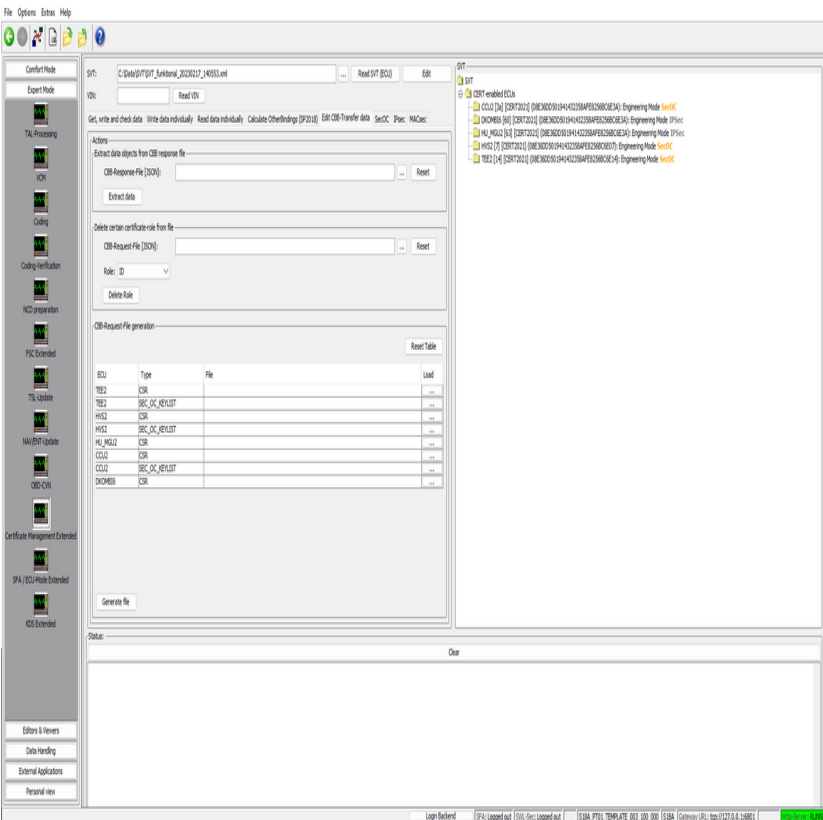
Both check boxes inactive: This combination does not provide a result since no bindings are passed to PSdZ.

**[Note: If BindingContainer is available from the CBB box and the vehicle for the same contr unit, the newly calculated container from the CBB-Box is always used again. The container determined from the vehicle is discarded. ]**

## **5) Edit CBB-Transfer data:**

The user has the possibility:

- extract individual memory objects of the loaded JSON file and store them in a xml file
- deleting individual Certificate roles from a CSR file
- generates the request file with the loaded xml files



## **Extract xml files of a JSON:**

1. Load JSON file
2. Button Extract data

## **Delete role IDs:**

1. Load request file [JSON].
2. Chose role that you want to delete.
3. Button Delete-Role

## **Creating a CSR file:**

1. Load ore read svt.

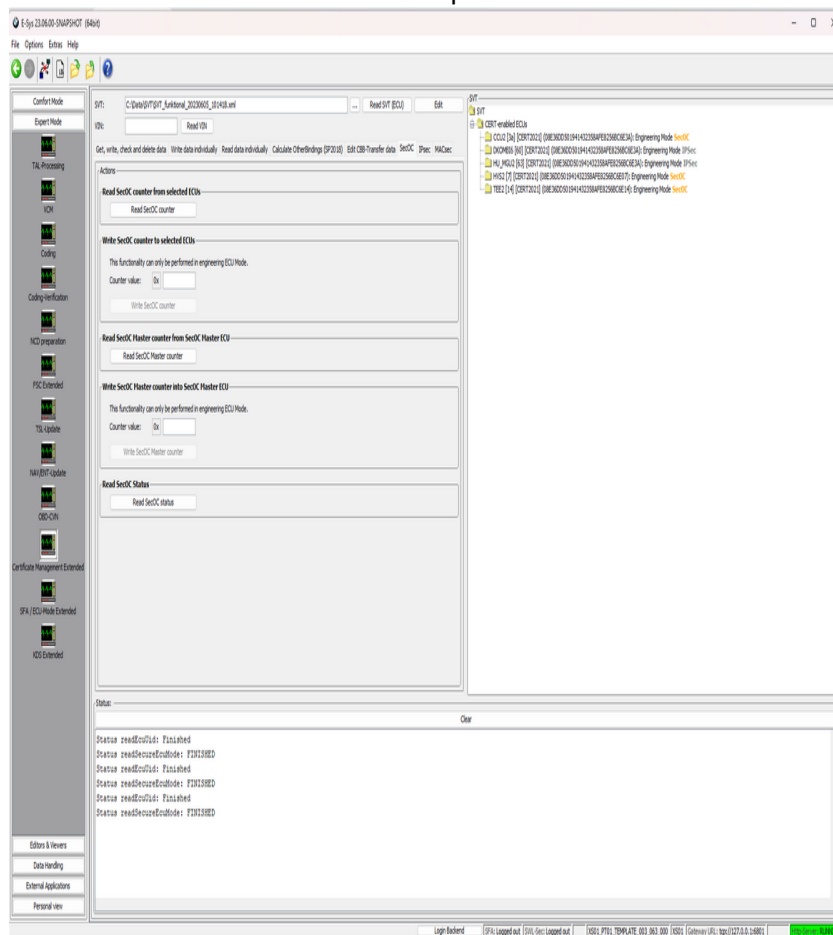
2. Load all the memory object [xml] in the table
5. The "Create file"

## 6) SecOC:

The user has the possibility:

- read out the current SecOC counter value of all selected Emus.
- set a new SecOC counter value to all selected Emus.
- read out the current SecOC Master counter value of the SecOC Master Emu. The SecOC Master Emu will be determined automatically. No selection in SVT-Tree required.
- set a new SecOC Master counter value to the SecOC Master Emu. The SecOC Master Emu will be determined automatically. No selection in SVT-Tree required.
- read the state of the SecOC. This includes the SecOC status, the key pack status, the key pack creation time status, the LCS1 current status, the response status and the ECU unknow status.

The result is shown in the status panel.

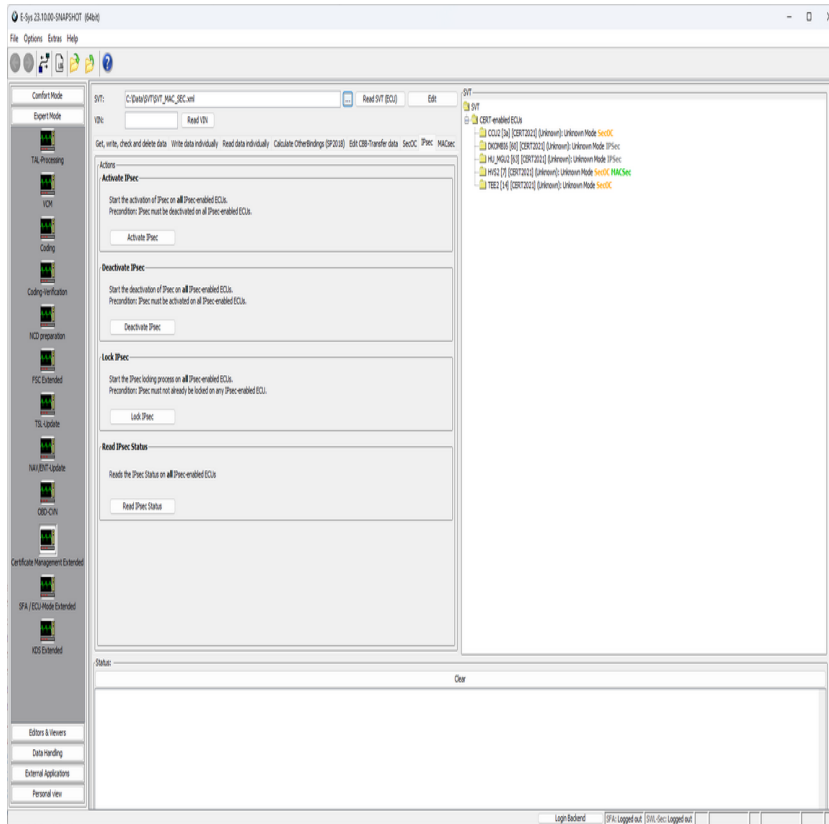


## Notes:

- The actions can only be performed on Emus that support this functionality.
- The Emus must be in engineering ECU Mode (applicable to write operations)
- Text field accepts counter values in HEX in range from 0x0 until 0xFFFFFFFF only

## 7) IPsec:

In this tab, all IPsec-capable ECUs can be activated, deactivated, locked or get the status read.

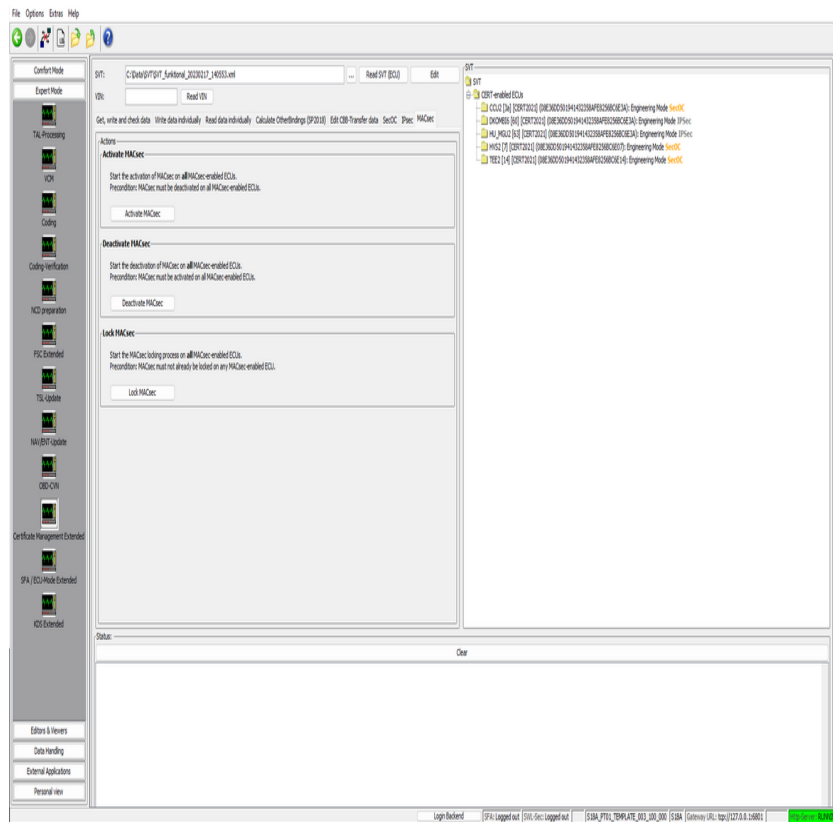


## Notes:

- The following command can only be performed on ECUs that are IPsec-capable.
- For each action, all IPsec-capable ECUs are addressed. This means that the selection in the SVT tree is not considered.

## 8) MACSec:

MACSec of all MACsec-enabled Emus can be activated, deactivated or locked in this tab. MACSec is a security function to provide security on the Ethernet Layer 2. The goal on the Ethernet cable is to protect the integrity of the transferred data from one ECU to another



## Notes:

- The following command can only be performed on Emus that support this functionality.
- For each action, all MACsec-enabled Emus are handled. Selection in the SVT tree is not considered.

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.



### **Introduction:**

The functionalities provided in this module which include direct server access require to have several preconditions to be fulfilled:

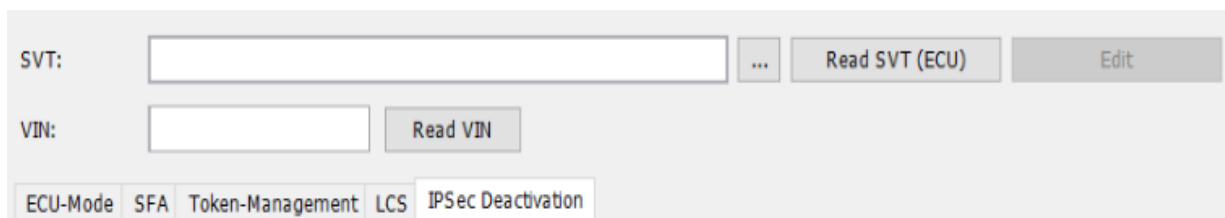
- The system which runs E-Sys must be in a BMW network segment in which the backend systems (SFA-Backend) are available.
- There must be a valid E-Sys client Certificate imported into E-Sys.
- At least one URL of a valid backend system (SFA-Backend) must be configured in E-Sys.

A more detailed set of instructions can be found here:

[HOWTO](#) (only available within BMW network).

[See 'Backend / SFA' if extended rights on SFA-Backend are required](#)

The module "SFA / Secure ECU Mode" empowers the user to write a secure token set, switch ECUs, which supports Secure ECU Modes, between the three states (Engineering, Plant, Field), generates a response-file, delete or clear features, verify tokens with VIN, and read the current SFA-State or ECU-Mode.



The screenshot shows a software interface with the following elements:

- An input field labeled "SVT:" with a text box and a dropdown arrow.
- A button labeled "Read SVT (ECU)" next to the SVT input.
- An "Edit" button to the right of the SVT section.
- An input field labeled "VIN:" with a text box.
- A button labeled "Read VIN" next to the VIN input.
- A breadcrumb trail at the bottom: "ECU-Mode" > "SFA" > "Token-Management" > "LCS" > "IPSec Deactivation".

### **Difference to the Comfort Mode:**

In the extended mode the ECUs which shall be treated can be selected in the tree view. There is single or multiple selection of Emus possible.

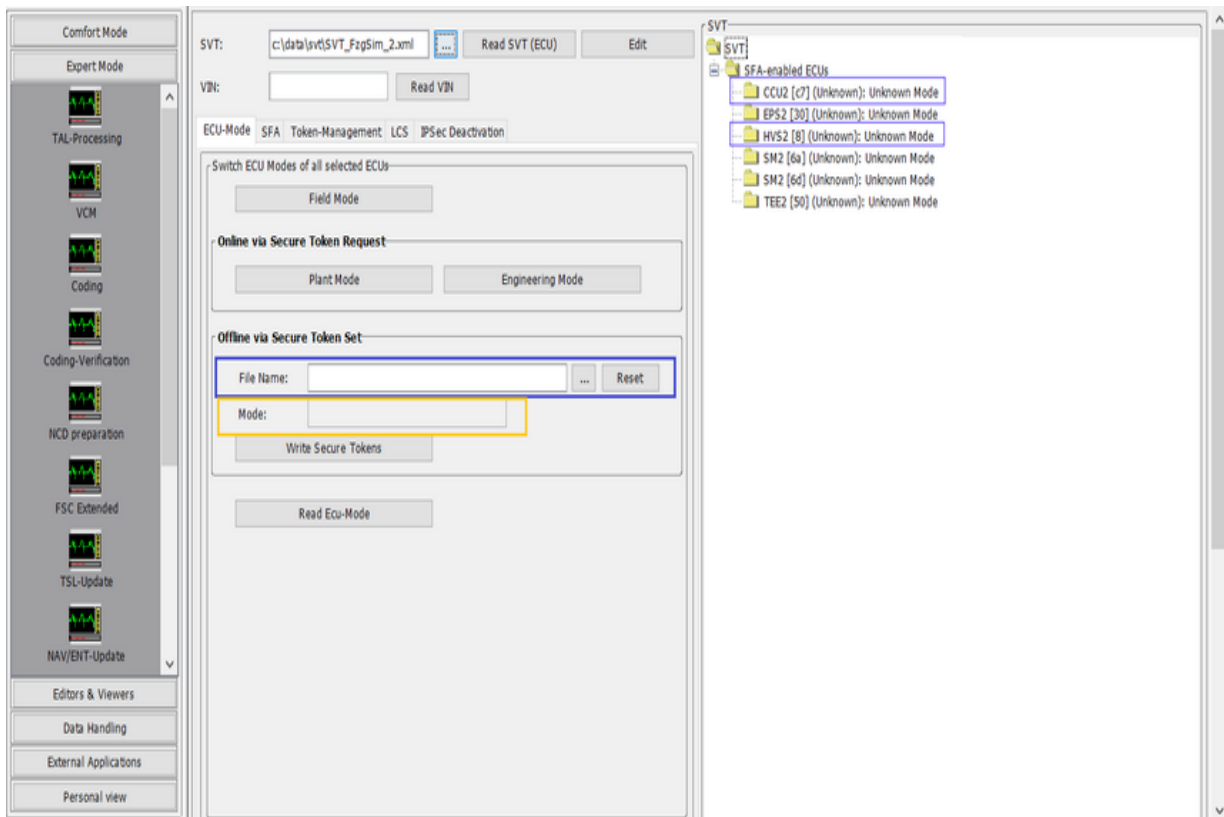
To switch back to a treatment for the entire vehicle, just click on the root element in the tree-view.

### **1) ECU-Mode:**

The Modes Plant and Engineering are only accessible with a valid token and the ECU has to be in den Field-Mode.

To switch back into the Field-Mode is no token necessary.

All ECUs that are contained in the svt are visualized on the right side in the svt-tree-view.



### **Use Cases:**

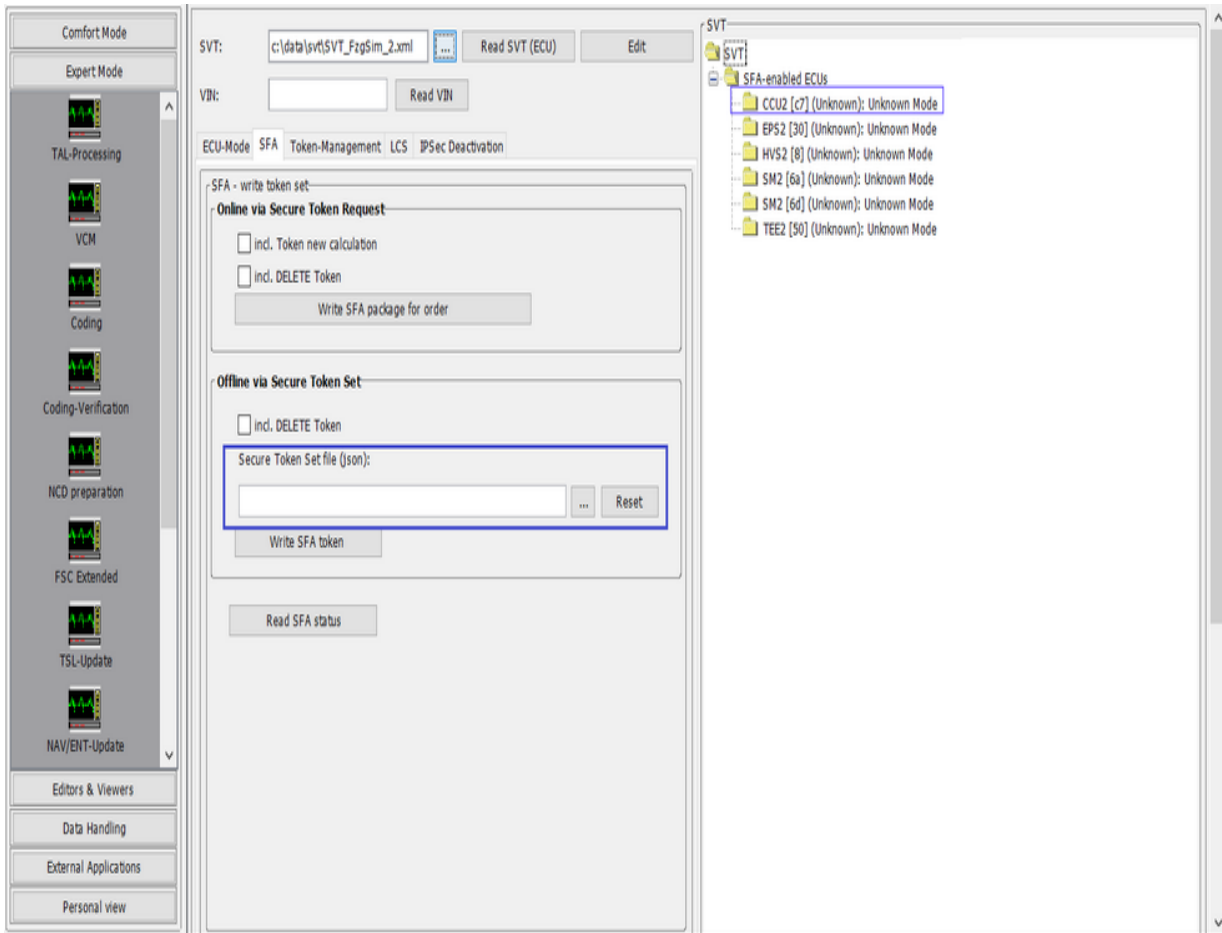
<b>Available only inside</b>		
------------------------------	--	--

<b>BMW network</b>		
	Switch Mode with Token Request: Plant Mode	Generates a Plant-Mode-Token for all Emus, that provides ECU-Mode, sending the request to the Backend and writing the response at the Emus. The Emus are finally in the PlantMode
	Switch Mode with Token Request: Engineering Mode	Generates an Engineering-Mode-Token for all Emus, that provides ECU-Mode, sending the request to the Backend and writing the response at the Emus. The Emus are finally in the EngineeringMode
	Upload Secure Token Set Request File	Sending the request to the backend and store the response at "DATA\SFA"
<b>Available outside BMW network</b>		
	Generate Request file	Generates a request-file for the chosen ECU-Mode for all Emus of the svt.
	Delete Token	Deactivate the chosen ECU-Mode token. ATTENTION: The token can't be used a second time.
	Clear Feature	Delete the chosen ECU-Mode token from all Emus. ATTENTION: The Emus have to be in the Engineering-Mode. After the process the deleted

		token can be used a second time.
	Write Secure Token Set	Write all secure tokens the loaded response file, all Emus that are part of these file are marked in the tree-view (blue marked at the illustration). Depends on the ECU-Mode in the chosen file Engineering Mode, Plant Mode, Mixed (Engineering and Plant tokens are included) or Undef / unknown (beside the ECU-Mode-tokens are also other feature-tokens part of the file) are visible.
	Switch into Field Mode	Sends the job to switch into Field Mode to all Emus.

## **2) SFA:**

SFA is the replacement of SWT from Service-Pack 2021 on.



Use-Case:

<p><b>Available only inside BMW network</b></p>		
	<p>Write SFA Target (package for order)</p>	<p>Generates at the backend a new secure token set. In case of new calculation is active the tokens for the already active features, that should be active based on the VIN, are going also to be generated. If the user wants only the features active, that are calculated to be active by the</p>

		<p>backup-System, "incl. DELETE Token" has be active. These means that it is possible that features which was active before are now deleted. To reactivate these features a new generated token is required.</p>
<b>Available outside BMW network</b>		
	<p>Write a local secure token set</p>	<p>Write the loaded secure token set.  Attention if the inclusion of surplus tokens is active. With these option all features, that are not part of the token set or the target status are going to be deleted. To reactivate these features a new generated token is required. An example of on surplus toke is an active ecu-mode token.  All Emus that are part of the token set are marked in the SVT-treeview. (blue marked in the above illustration)</p>
	<p>Generate SFA-production-status Request</p>	<p>Generates a SFA (secure Token-Request set) for the SFA-production status.</p>
	<p>Read SFA-Status</p>	<p>Reading the SFA-status of the choose Emus or the entire vehicle.</p>

## Color representation of SFA features in the SVT tree view:

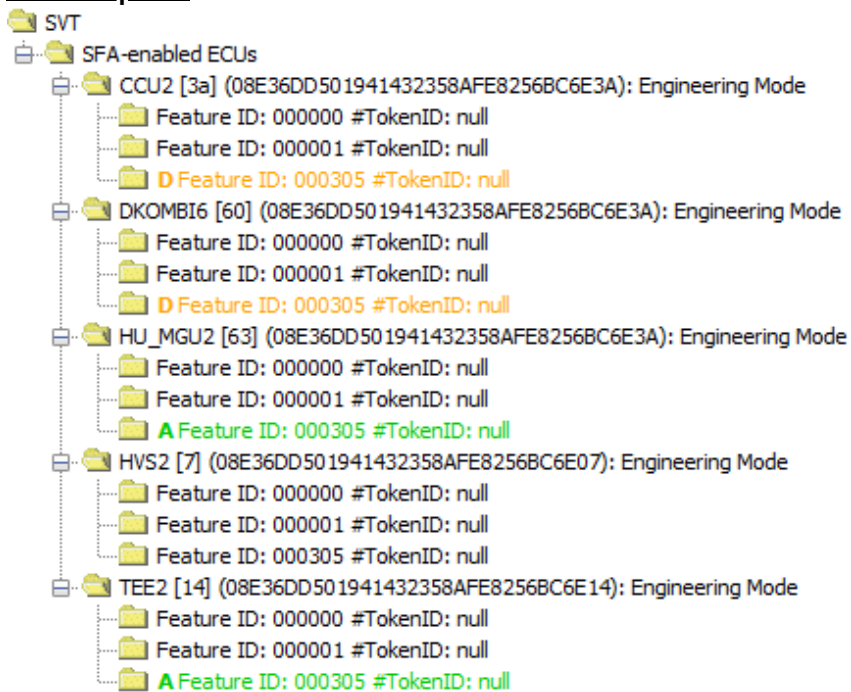
In order to determine the status of all SFA features (activated, deactivated etc.) for each ECU selected in the SVT tree view, the button “Read SFA status” must be activated.

This triggers communication with each selected SFA-capable ECU and its features are read out.

The SVT tree structure is then updated in such a way that each ECU node can be opened up and all available features can be displayed.

Each feature is highlighted in color in the SVT tree view.

## Example:



Note: a character is added to the beginning of the Feature ID string to help color blind users identify the state

## Color marks and characters meaning:

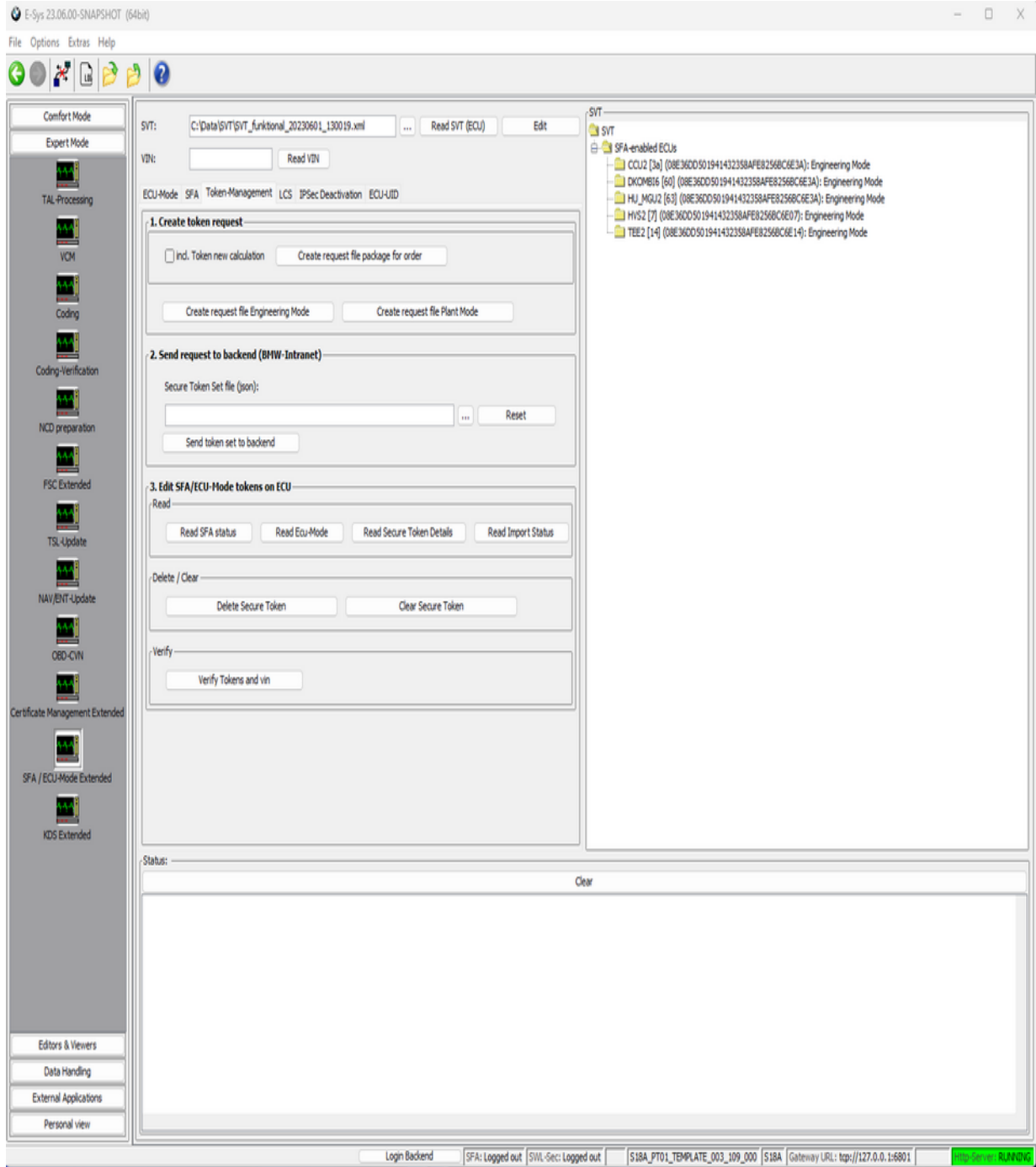
**Green** and character A - the Feature\_ID is **activated**

**Orange** and character D - the Feature\_ID is **deactivated** or **expired**

**Red** and character E - the Feature\_ID is **incorrect**

**Black** and No character - the Feature\_ID is initially **not active**, i.e. no secure token available

### 3) Token-Management:



Use-Case:

**Available**

--	--	--



<b>only inside BMW network</b>		
	Upload Secure Token Set Request File	Sending the request, depends on the request-type (ECU-Mode, package for order, newest package), to the backend and store the response at "DATA\SFA". A "newest package" request has to have "newest" a part of its file-name.
<b>Available outside BMW network</b>		
	Generate ECU-Mode Request file	Generates a request-file for the chosen ECU-Mode for all Emus of the svt.
	Generate SFA-production-status request	Generates a request-file (secure Token-Request set) for the SFA-production-status.
	Delete Token	Deactivate the chosen ECU-Mode token. ATTENTION: The token can't be used a second time.
	Clear Feature	Delete the chosen ECU-Mode token from all Emus. ATTENTION: The Emus have to be in the Engineering-Mode. After the process the deleted token can be used a second time.
	Read Status / Read ECU-Mode	Reading the SFA-/ECU-mode status of the choose Emus or the entire vehicle.

	Read Import Status	Reads the result of the last token import.
	Verify Token and VIN	The system verify if all token are compatible with the VIN of the vehicle.

#### **4) LCS:**

LCS stand for "Locking Configuration Switch". In this tab the user has the option to read and change the value of a specific configuration of the chosen ECU.

The user has either the option to choose the configuration by the drop-down menu or enter the HEX-Value manually.

LCS #0 - #99 are reserved for internal ECU functions. LCS #100 - #255 are reserved for ECU-specific functions.

Currently LCS values are mapped to the following scheme:

<b>LCS Value</b>	<b>LCS Name</b>	<b>Description</b>
LCS#0 (0x00)	SP Switch	LCS #0 is provided for switching Service Packs (SP2018/SP2021)
LCS#1 (0x01)	SecOC Bypass	Activation is required, for example, if a vehicle has mixed ECUs with and without secure onboard communication.
LCS#2 (0x02)	TimeSupreme	System time used in diagnostics.  At Release 23/03, the system time is migrated from "Kombi" ECU to the main ECU ("BCP"). That enables the switchover to

		<p>the AUTOSAR time sync mechanism.</p> <p>The value 2 here corresponds to the switchover to the "BCP".</p>
LCS#3 (0x03)	Integrity Protection OC	<p>The switch activates/deactivates the integrity protection onboard communication.</p> <p>The integrity protection is a protection with high frequency.</p>
LCS#4 (0x04)	IuK Cluster Configuration	<p>If a control unit is part of the IUK (Infotainment and Communication) cluster, it supports LCS#4.</p> <p>The LCS#4 indicates the IuK cluster configuration (whether an IDCevo is present or not).</p>

All other LCS name values (0x05-0xFF) can be entered manually, but are currently not defined. For this reason, the LCS Name is displayed as "Custom" in the drop-down menu.

SVT:  ...

VIN:

ECU-Mode SFA Token-Management **LCS** IPSec Deactivation ECU-UID Basic Security Features

The functionality "Read LCS" and "Set LCS via diagnostic job" can only be performed in engineering- or plant-mode.  
The execution of "Set LCS via token" is also possible in field mode.

**Read LCS**

LCS name:

**Set LCS**

Perform reset after LCS switchover

Set LCS via diagnostic job

LCS name:

LCS value:

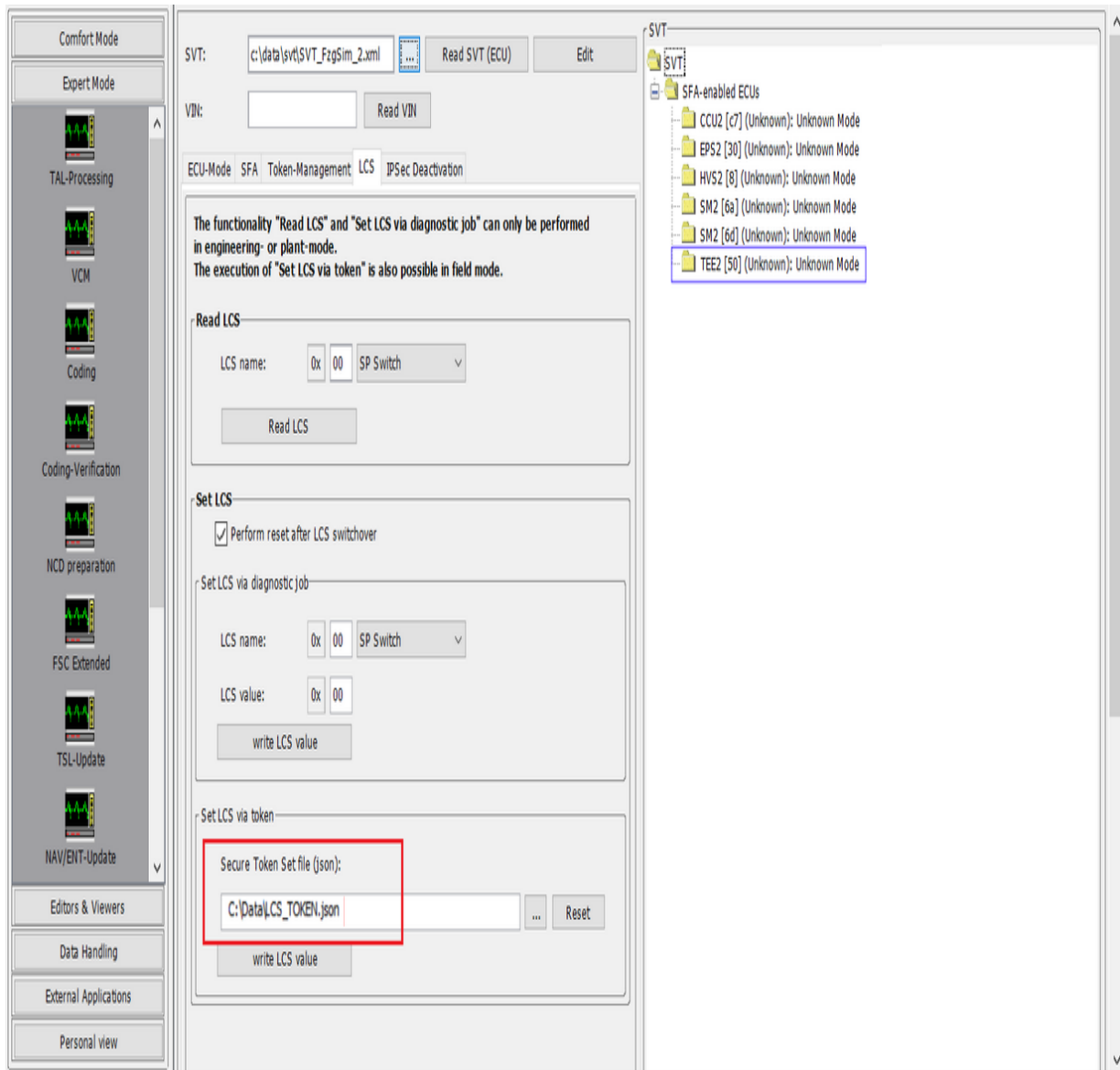
Set LCS via token

Secure Token Set file (json):  ...

Alternatively, a file containing LCS tokens can be used to perform the LCS switch over.

## Notes:

- If the loaded file contains a LCS token that matches an ECU from the loaded SVT, this is marked in the SVT tree structure by a colored text (see figure below).
- After a LCS#01 switch over, an ECU reset is to be performed so that the new LCS#01 value is also implemented by the ECU.

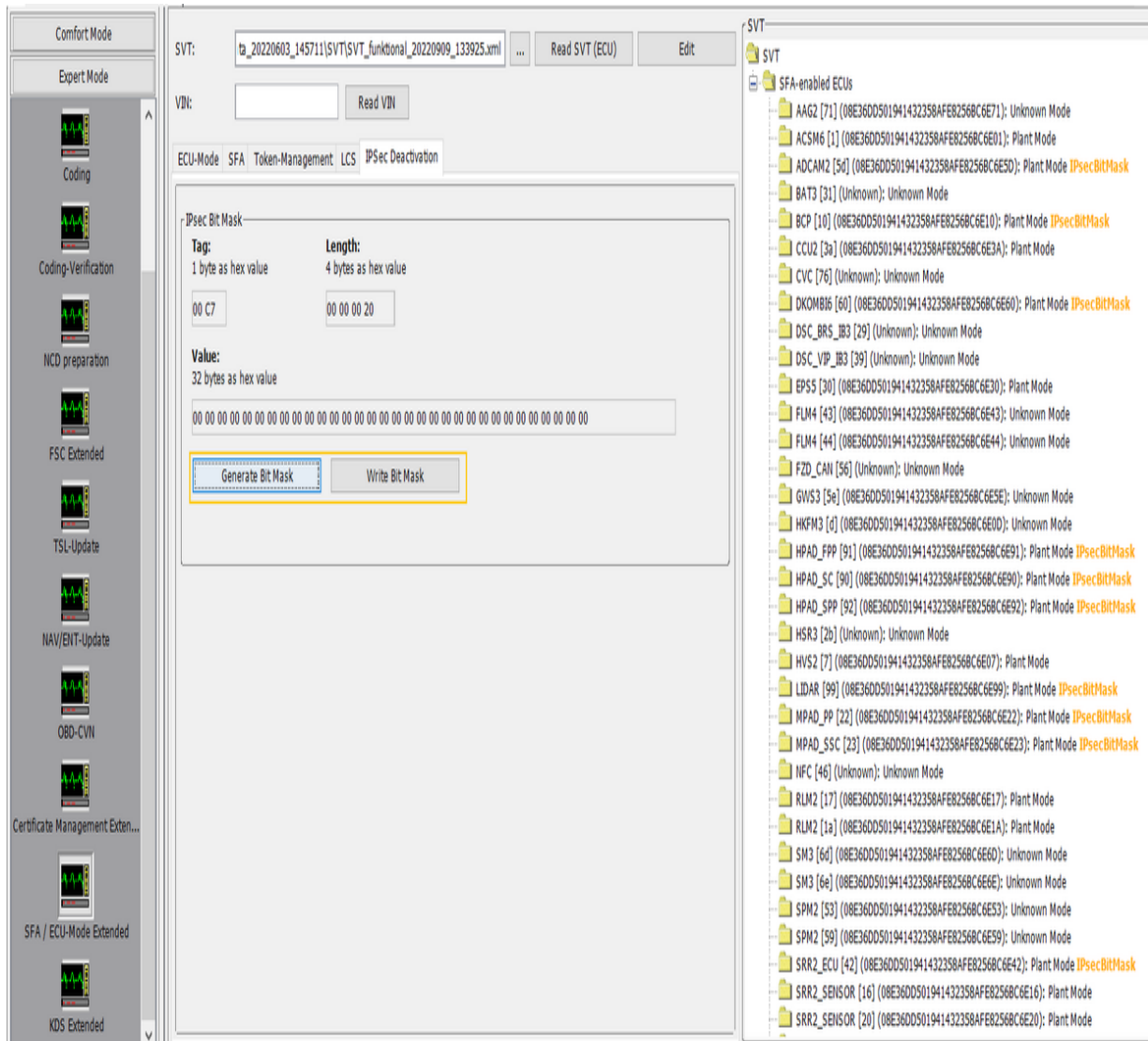


## **5) IPsec Deactivation:**

ECUs with and without IPsec support can be installed on an vehicle. To enable the IPsec-capable ECUs to communicate with the IP ECUs, IPsec can be deactivated for them. To do this, the IPsec ECUs must know their addresses. This is done with the help of a bit mask that can be generated at this point.

To write the bit mask, individual control units can be selected in the tree on the right hand side. If no ECUs are selected, the

bit mask is written to all ECUs visible in the tree that support IPsecBitMask.



### Explanation of the generated bitmask:

IPsec can be excluded for up to 256 addresses. The bitmask exists of 32 8-bit blocks that add up to 256. Each bit position corresponds to a host address (bit 16 = host x.x.x.16). Each 8-bit block (byte) is represented hexadecimally.

Byte	0								1	...	19					31
Bit	255	254	253	252	251	250	249	248	...	...	104	103	102	101	...	...
Bit-Wert	1	0	1	0	0	0	0	1	...	...	0	0	0	1	0	...
Hex-Wert	A1								00	...	08					00

Hint:

Tag and Length are only shown for information. They are used internally to generate the bit mask.

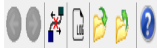
## **6) Basic Security Features:**

In this tab the user can read basic features of a list of ECUs.

The basic features are:

- Tokens
- Certificates
- CSR Keylist
- Sec OC
- LCS
- IPSEC
- EAP
- MACSEC

The result appears in the status panel.



Comfort Mode

Expert Mode

- TAL Processing
- VCM
- Coding
- Coding-Verification
- NCD preparation
- FSC Extended
- TSL Update
- NAI/ENT Update
- OBD-CIN
- Certificate Management Extended
- SFA / ECU Mode Extended
- KDS Extended

Editors & Viewers

Data Handling

External Applications

Personal view

SVT: C:\Data\SVT\SVT\_funktional\_20230605\_104425.xml Read SVT (ECU) Edit

VIN:  Read VIN

ECU-Mode: SFA Token-Management LCS IPsec Deactivation ECU UID Basic Security Features

**Read Basic Security Features**

Read Basic Security Features

SVT

- SVT
- SFA-enabled ECUs
  - AAG2 [71] (08E360D501941432358AF8E2568C6E71): Unknown Mode
  - ACSM6 [1] (08E360D501941432358AF8E2568C6E01): Plant Mode
  - ADCAN2 [5d] (08E360D501941432358AF8E2568C6E3D): Plant Mode IPsecBitMask
  - BAT3 [31] (Unknown): Unknown Mode
  - BCP [10] (08E360D501941432358AF8E2568C6E10): Plant Mode IPsecBitMask
  - CCU2 [3a] (08E360D501941432358AF8E2568C6E3A): Plant Mode
  - CVC [76] (Unknown): Unknown Mode
  - DKOMB6 [60] (08E360D501941432358AF8E2568C6E60): Plant Mode IPsecBitMask
  - DSC\_BRS\_IB3 [29] (Unknown): Unknown Mode
  - DSC\_VIP\_IB3 [38] (Unknown): Unknown Mode
  - EPS5 [30] (08E360D501941432358AF8E2568C6E30): Plant Mode
  - FLM4 [43] (08E360D501941432358AF8E2568C6E43): Unknown Mode
  - FLM4 [44] (08E360D501941432358AF8E2568C6E44): Unknown Mode
  - FZD\_CAN [56] (Unknown): Unknown Mode
  - GVIS3 [5e] (08E360D501941432358AF8E2568C6E5E): Unknown Mode
  - HKFM3 [6] (08E360D501941432358AF8E2568C6E10): Unknown Mode
  - HPAD\_FPP [91] (08E360D501941432358AF8E2568C6E91): Plant Mode IPsecBitMask
  - HPAD\_SC [90] (08E360D501941432358AF8E2568C6E90): Plant Mode IPsecBitMask
  - HPAD\_SPP [92] (08E360D501941432358AF8E2568C6E92): Plant Mode IPsecBitMask
  - HSR3 [2b] (Unknown): Unknown Mode
  - HVS2 [77] (08E360D501941432358AF8E2568C6E07): Plant Mode
  - LIDAR [99] (08E360D501941432358AF8E2568C6E99): Plant Mode IPsecBitMask
  - MPAD\_PP [22] (08E360D501941432358AF8E2568C6E22): Plant Mode IPsecBitMask
  - MPAD\_SSC [23] (08E360D501941432358AF8E2568C6E23): Plant Mode IPsecBitMask
  - NFC [46] (Unknown): Unknown Mode
  - RUM2 [17] (08E360D501941432358AF8E2568C6E17): Plant Mode
  - RUM2 [1a] (08E360D501941432358AF8E2568C6E1A): Plant Mode
  - SM3 [6d] (08E360D501941432358AF8E2568C6E6D): Unknown Mode
  - SM3 [6e] (08E360D501941432358AF8E2568C6E6E): Unknown Mode
  - SPM2 [53] (08E360D501941432358AF8E2568C6E53): Unknown Mode
  - SPM2 [59] (08E360D501941432358AF8E2568C6E59): Unknown Mode
  - SRR2\_ECU [42] (08E360D501941432358AF8E2568C6E42): Plant Mode IPsecBitMask
  - SRR2\_SENSOR [16] (08E360D501941432358AF8E2568C6E16): Plant Mode
  - SRR2\_SENSOR [20] (08E360D501941432358AF8E2568C6E20): Plant Mode

Status: Clear

Login Backend | 
 SFA: Logged out | 
 SVM: Sec: Logged out | 
 F001\_AAG\_01\_EXERCISES\_037\_000\_010 | 
 F001 | 
 Gateway URL: tcp://127.0.0.1:6801 | 
 RUNNING

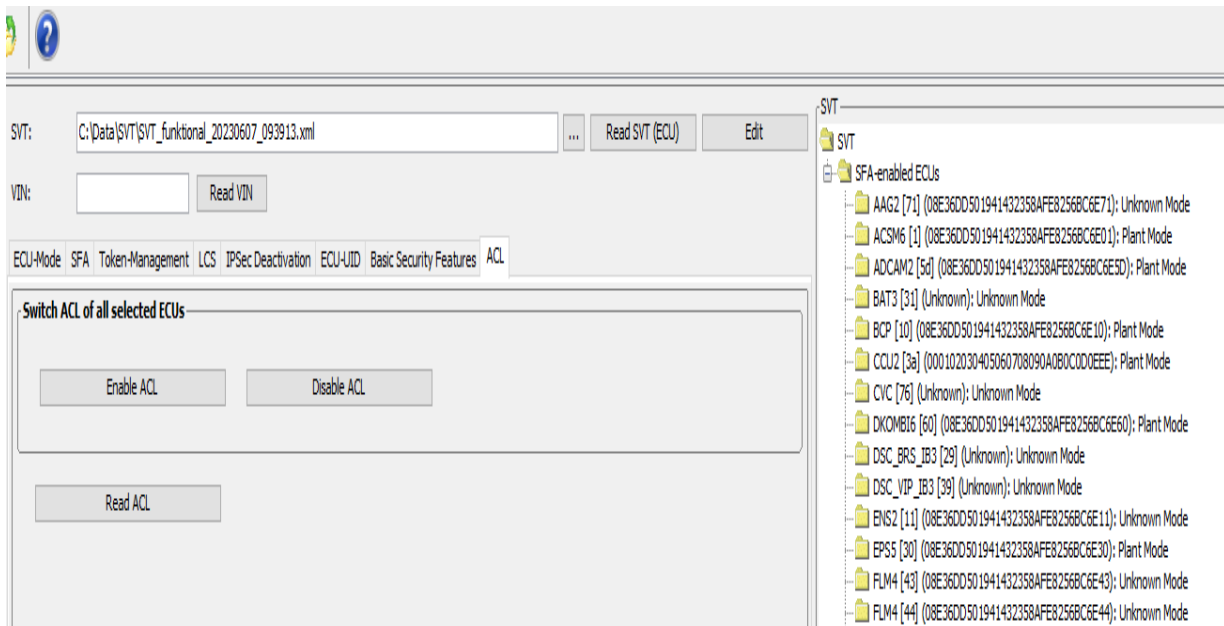


## 7) ACL:

In this tab the user can activate, deactivate or read the ACL (Access Control List) state of a list of ECUs.

Access control enabled ECUs implement an ACL-based filter mechanism to ensure that only allowed SOME/IP communication takes place, according to an ACL whitelist in every ECU.

The ACL actions will be enabled if a valid SVT is selected:



The command will be done to the selected ECUs and the response shown in the status panel.

The response for each ECU comes with the following status:

Enable ACL:

**0x00 OK - the ACL mechanism was activated successfully.**

**0x01 ERROR - the ACL mechanism could not be activated successfully.**

**0x02 OPERATION\_ALREADY\_RUNNING - ACL mechanism already activated.**

**0xFF INVALID.**

Disable ACL:

**0x00 OK - the ACL mechanism was deactivated successfully.**

**0x01 ERROR - the ACL mechanism could not be deactivated successfully.**

**0x02 OPERATION\_ALREADY\_RUNNING - ACL mechanism already deactivated.**

**0xFF INVALID.**

Read ACL:

**0x00 ACTIVATED - ACL mechanism is activated.**

**0x01 DEACTIVATED - ACL mechanism is deactivated.**

**0xFF INVALID.**

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

### **Introduction:**

KDS (anti-theft protection 2.0) based on the secure token sets (see SFA / Secure ECU-Mode).

The functionalities provided in this module which include direct server access require to have several preconditions to be fulfilled:

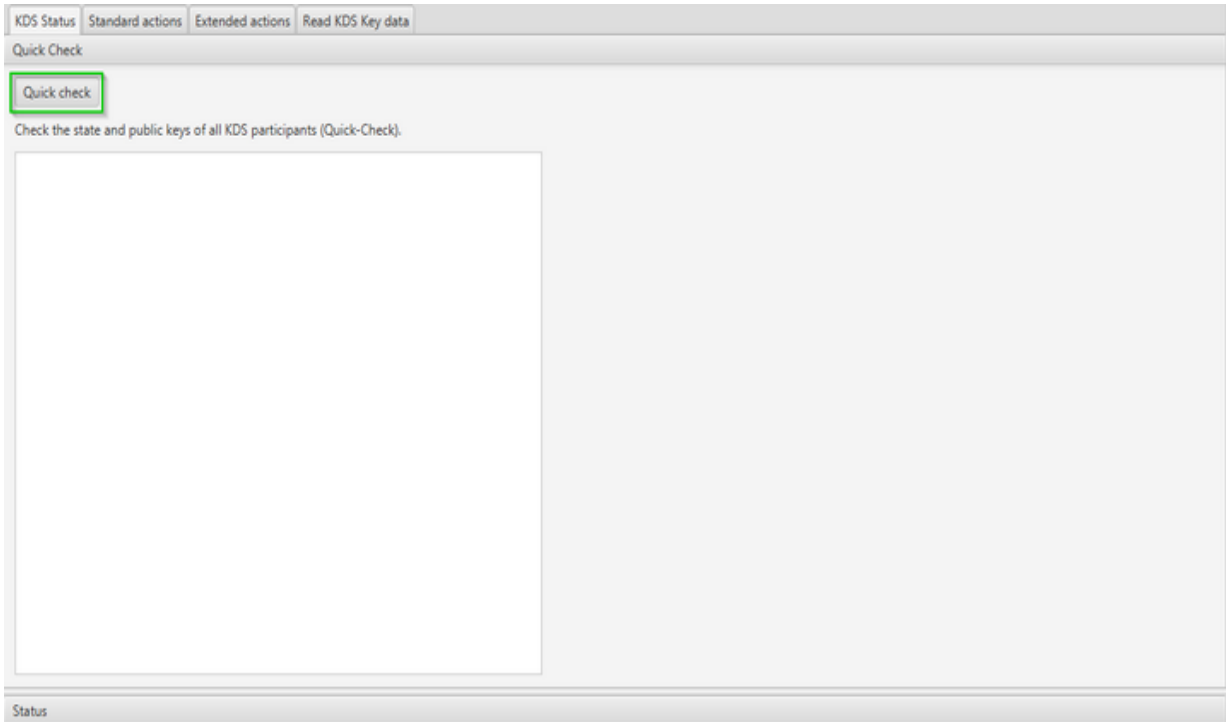
- The system which runs E-Sys must be in a BMW network segment in which the backend systems (SFA-Backend) are available.
- There must be a valid E-Sys client Certificate imported into E-Sys.
- At least one URL of a valid backend system (SFA-Backend) must be configured in E-Sys.

A more detailed set of instructions can be found here:

[HOWTO](#) (only available within BMW network).

### **KDS status:**

-



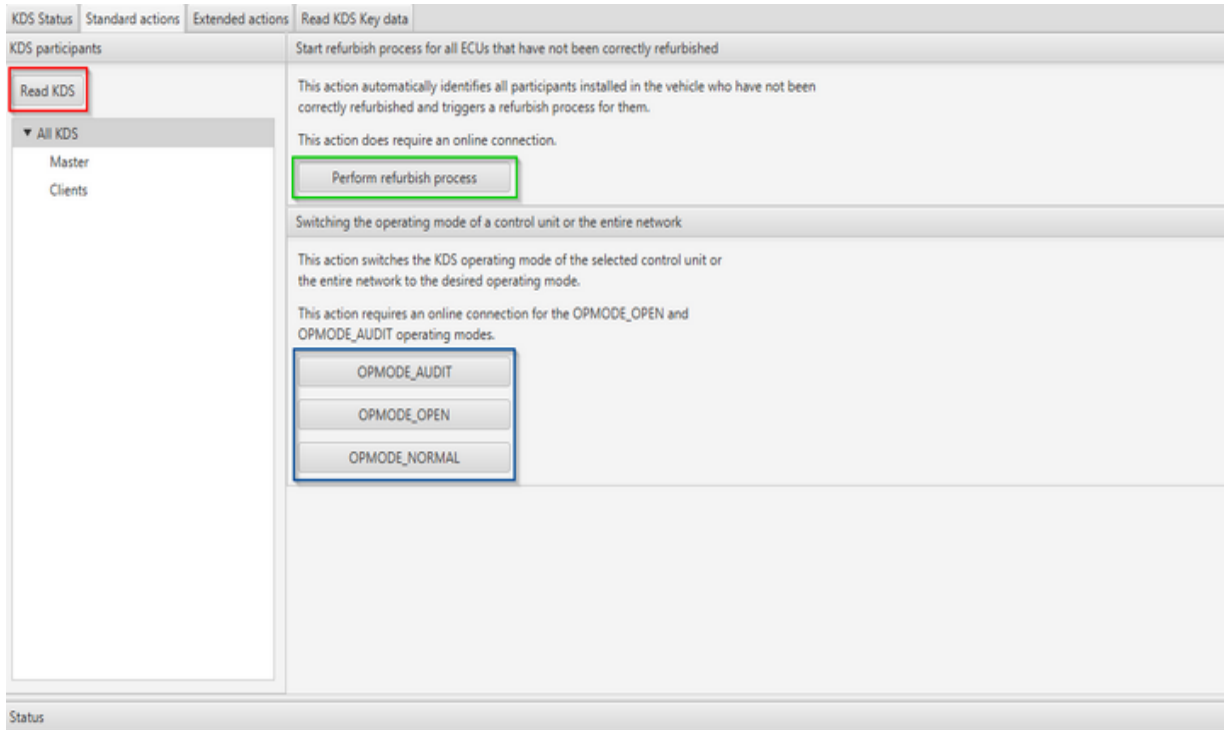
The button "Quick check" (marked green) starts with a KDS system check. The result is visualized in the GUI and has one of the following values:

<b>KDS quick check results</b>
MASTER_OK_CLIENT_OK
MASTER_OK_CLIENT_INVALID
MASTER_INVALID_CLIENT_OK
MASTER_INVALID_CLIENT_INVALID
ERROR_CLIENT_NOT_PAIED

-

-  
**Standard actions:**

-

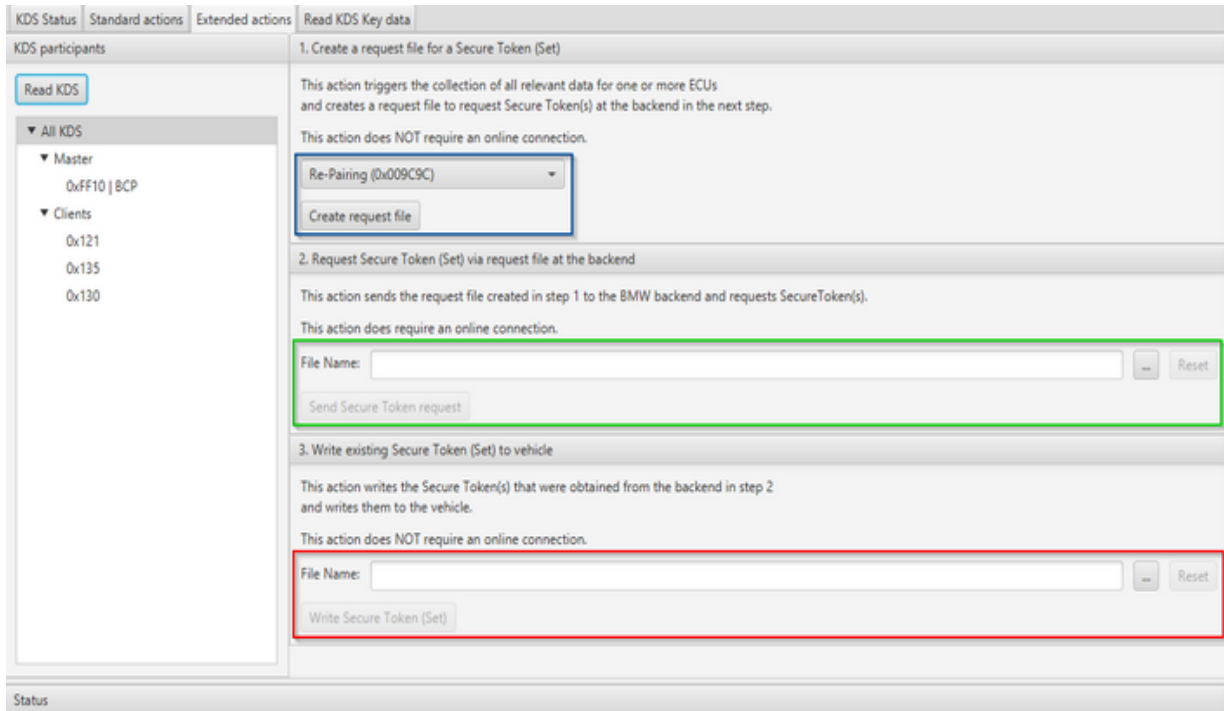


These tab can visualize the KDS system. It is separated in the KDS-master and KDS-clients. All KDS system members are by the KDS-ID.

- Read KDS (marked red): Read the entire KDS system.
- Refurbish process (marked green): **[Only with BMW-Intranet connection possible]** Identifies all participants in the KDS system, that have not been correctly paired and is pairing the entire KDS system.
- Switching operation mode: Change the operation mode for the chose Emus. If no ECU is selected the entire KDS system is going to switch. The switch into the modes AUDIT and OPEN requires a BMW-Intranet connection and the valid client-backend-Certificate (E-Sys [options](#) / authentication)

### **Extended actions:**

—



- Generates a token request file for the chosen Emus or the entire KDS-system. The request-type can be defined by the drop down menu, behind the name of the is the feature-ID displayed. (marked blue)

Type of the secure token request file	Short description
Re-Pairing (0x009C9C)	Generates a token for a re-pairing (like refurbish process).
Remove client pairing (0x00DC9D)	Resets the pairing status of KDS-clients.
Remove master pairing (0x00DCC9)	Resets the component list at the KDS-master for the chosen KDS-clients.
OPMODE_AUDIT (0x00AAFC)	Create a request token for the selected Emus to change into the operation mode AUDIT.
OPMODE_OPEN	Create a request token for

(0x00AAFC)

the selected Emus to change into the operation mode OPEN.

- **[BMW-Intranet necessary]** Sending the loaded request-file to the SFA-Backend and automatically save the response (secure token set). (marked green).
- Write the active KDS secure token set (marked red)

### Read KDS key data:

The screenshot shows a web application interface for reading KDS key data. It features a navigation bar with tabs: 'KDS Status', 'Standard actions', 'Extended actions', and 'Read KDS Key data'. The 'Read KDS Key data' tab is active. Below the navigation bar, there are two buttons: 'Read data' (highlighted with a blue box) and 'Export data' (highlighted with an orange box). The main content area is divided into three sections, each with a table:

- Read KDS Client key data (direkt):** Table with columns 'KDS-ID', 'ECU-UID', 'Publickey (own)', and 'Publickey (paired)'. The table is empty and displays 'No content in table'.
- Read KDS Master key data:** Table with columns 'KDS-ID', 'ECU-UID', and 'Publickey (own)'. The table is empty and displays 'No content in table'.
- KDS component list (Master):** Table with columns 'KDS-ID', 'ECU-UID', and 'Publickey (paired)'. The table is empty and displays 'No content in table'.

At the bottom of the interface, there is a 'Status' section with a 'Clear' button.

- Read data (marked blue): Visualize all KDS-ID with the corresponding publickey.
- Export data (marked orange): The results can be exported as a csv-file. The location where the cvs-file is stored can

be seen in the log-view (marked green); the location is the KDS sub folder in at E-Sys\_Data.

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.



[PDX-Charger](#)

[PDX-Update](#)

[PDX-Delete](#)

[ODX-Checker](#)

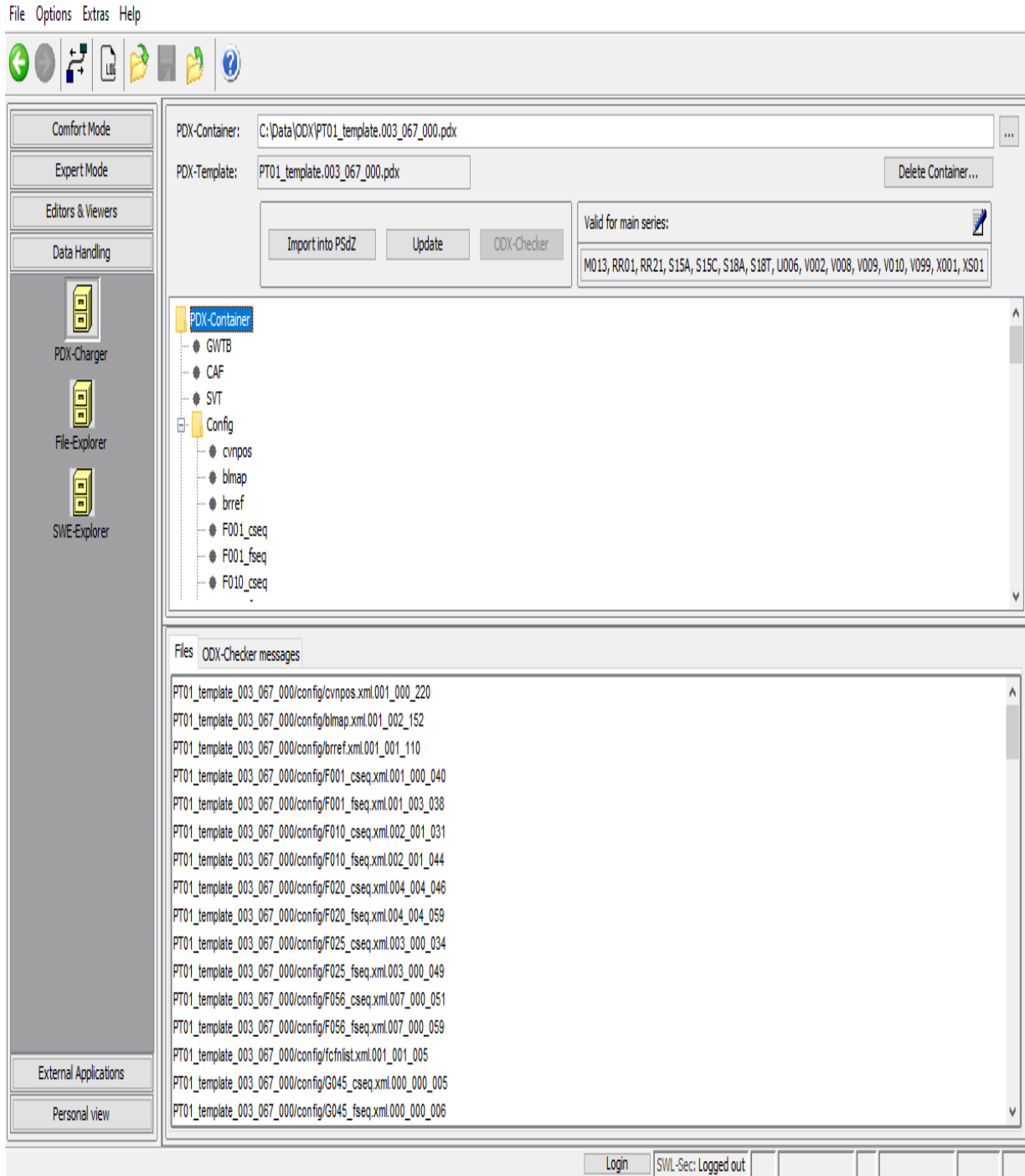
[File-Explorer](#)

[SWE-Explorer](#)

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

To process a vehicle flash or ECU programming the PSdZ (or the more basic runtime system beneath) needs a couple of data files e.g. SWEs, program jobs, programming procedure, vehicle information etc. All those informations are transported in MCD-2D (ODX) containers.



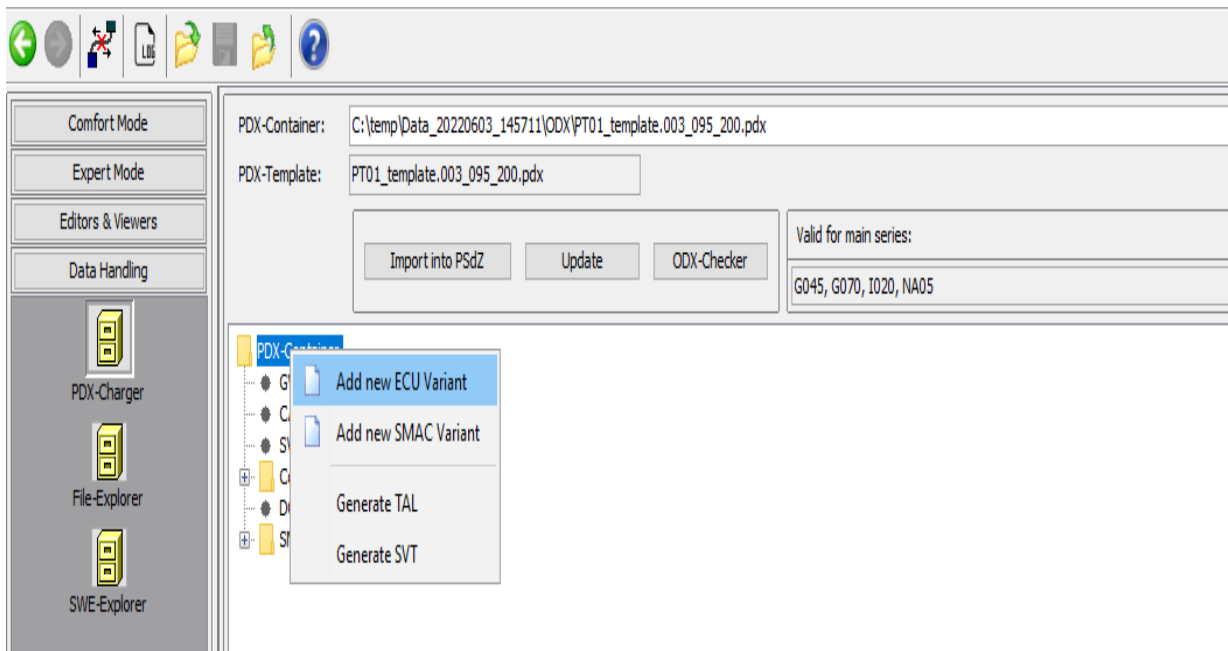
Therefore an existing container is read from disk into E-Sys by the "three-dot"-button.

The displayed design of the container is as follows:

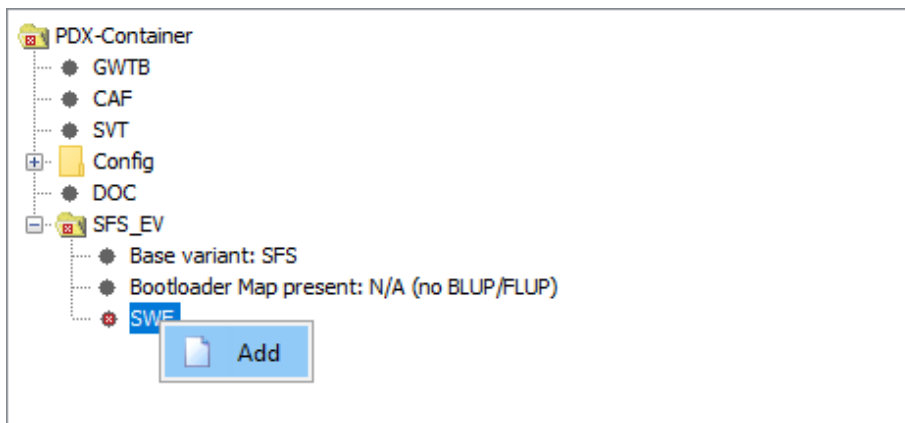
Root element "PDX-	By context menu a new ECU can
--------------------	-------------------------------

Container":	be inserted here
CAF:	A (further) CAF file can be added..
SVT or SWE-Sequent:	A (further) SVT or. SWE sequence can be added.

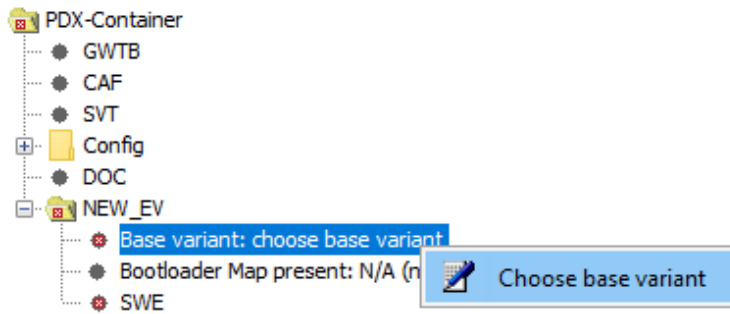
To add new elements, you can basically choose between "ECU" and "SMAC" (Smart Actuator).



After adding a ECU or SMAC it's possible to append one or more SWEs to its substructure.

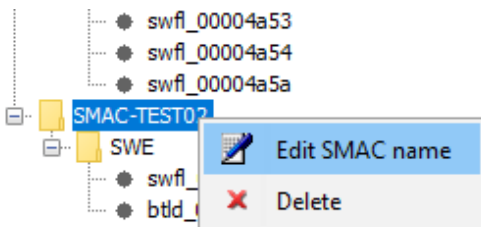


To an ECU exactly one base variant can be assigned. And to every base variant you can specify bootID and diagnosis address. Furthermore it's possible to deactivate a base variant.



The name of the ECU is preset to "NEW\_EV" and can be changed via the context menu (ending "\_EV").

In the case of Smart Actuators (SMAC) there is no base variant and the SMAC name is initially arbitrary. As soon as a BTLD file is added, the name must correspond to a matching EE component name.

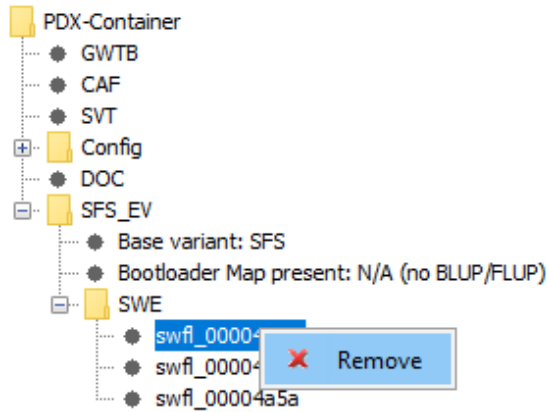


Ecus are displayed with blue letters if of L4 and with black letters if it is an L6 ECU.

The container's file versions are displayed in the lower window. Other than the display in the above window (i.e. swfl\_11111114) the file version gets displayed, too (swfl\_11111114\_001\_001\_002). For SWE files it is possible to

delete single file versions, but only if 2 SWE file versions for the same identifier exist.

Already existing elements of the upper parts can be removed by context menu.



After all modification is done the PDX container can be imported into PSdZ unless there is no connection to an ECU. So in the mask above you have to disconnect before trying to import into PSdZ.

### GUI-Elements

"Import into PSdZ"	Import current container into PSdZ
"..." button	load existing PDX container
"Update" button	PDX update, see correspondent <a href="#">help page</a>
"ODX-Checker"	Starts the ODX-Checker, see correspondent <a href="#">help page</a>
"Delete Container..."	PDX delete, see correspondent <a href="#">help page</a>

### Further Information:

ODX (**O**pen **D**iagnostic **D**ata **E**xchange) is a standardized exchange format based on XML for data and information related to diagnosis. PDX (Packed ODX) is a zipped ODX. The PDX file only contains meta information. The actual data (SWEs) is contained in separate files and is only referenced from the ODX file. In ODX multiple layers are defined which contain the different types of information:

The ODX-Charger performs the task to integrate the particular layers to one ODX-Container. Hereby the XML files for the layers Diag-Layer-Container, Comparam-Spec, Vehicle-Info-Spec and Multiple-Ecu-Job-Spec exist already in a file system or are delivered completed.

The flash layer is assembled by the ODX-Charger from a couple of selected SWEs.

So far the SWEs were included into the ODX-Container as MSR files. Hence since version 2.0.0 of E-Sys also BSW files are supported as SWE. This BSW is a ZIP archive that contains the binary flash data and an XML file containing the header information.

Even if the ODX-Container could consist of a single large file it's recommended for the sake of clearness to leave the distinct layers in it's files and to associate them in the ODX file.

When a PDX Component Container is imported into the PDX-Charger the name will be verified to check if it is valid. The rules vary depending if the name structure is version 3 or older. If it is older the name structure will be the series, that is represented by 4 characters that can be letters or digits. Then it is followed by an underscore and a combination of 1 to 8 characters that can be digits, upper case letters or underscores. Optionally you can two underscores followed by a combination of 1 to 8 characters that can be digits, upper

case letters and underscore. Then this is followed by the version. The version is represented by three numbers of three digits between 0 and 255 that are separated by underscores. It is terminated with a pdx extension. Example:  
 F010\_Example\_\_Ex\_12.001\_020\_223.pdx.

For a PDX Component Container which version is 3 or above, the structure starts by a combination of 1 to 8 characters that can be digits, upper case letters or underscores and then is followed by two underscores and another combination of 1 to 8 characters that can be digits, upper case letters or underscores. This is then followed by two underscores and a combination of 1 to 20 characters that can be digits, upper case letters or underscores. Then it is followed by the version and the pdx extension.

Naming convention:

**<E/E Component>\_\_<freetext>.<version>.pdx**

E/E Component	[A-Z0-9_]{1,8}__[A-Z0-9_]{1,8}	E.g., ZBE__04
Free text	[A-Z0-9_]{1,20}	E.g., F001
Version	\d{3}__\d{3}__\d{3}	E.g., 003_004_000

Example of a valid component container name:

*ZBE\_\_04\_\_F001.003\_004\_000.pdx*

Each version part must be in in the range [0-255]. Any other value will not be allowed. For example, *ZBE\_\_04\_\_F001.003\_004\_256.pdx* is an invalid component container name.

In the case of a container that contains referenced software units, it is important to note that these referenced software



units do not exist as a file in the container and are therefore not imported. If the referenced software units are required, they must be imported via a container in which the software units are available as real files in the container.

---

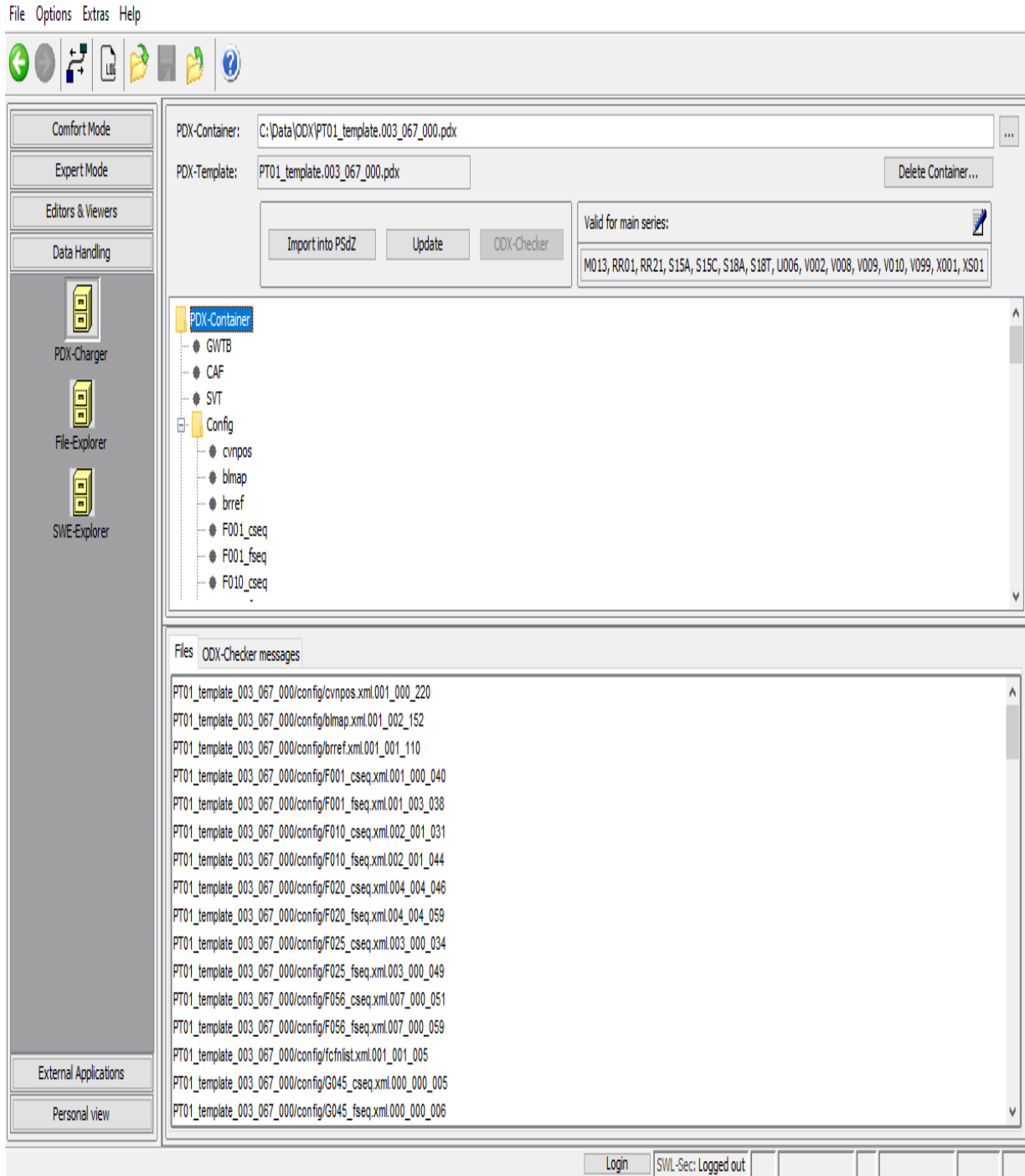
This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

Up to now it was only possible to import a delivered or modified (by E-Sys) PDX container into PSdZ.

In practice on-site it's common to start with a discrete HW/SW configuration and modify and expand it in the course of time of progression (I-Stufe) in the way that when new data is available still there is a wish to preserve the old data.

An amplification of the previous container by the new contents of the new container is desired.

This can be realized now by the update functionality in E-Sys.



During start of mask the update button is still disabled. Primal a container is loaded both button "Import into PSdZ" and "Update" are enabled.

As the first container the 'old' container to be updated is selected.

After pressing the update button a file select box opens and the PDX container (with the new data) used for the update can be specified.

Now E-Sys calculates the set union of the two containers and displays it. Afterwards the modified container can be saved and imported into PSdZ.

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

Starting with E-Sys 3.31, an index file is introduced to allow the deletion of PDX containers. This index file is updated during each import and delete process.

In the dialog 'Delete PDX-Container' the user is offered a selection menu of the PDX containers that have been imported so far.

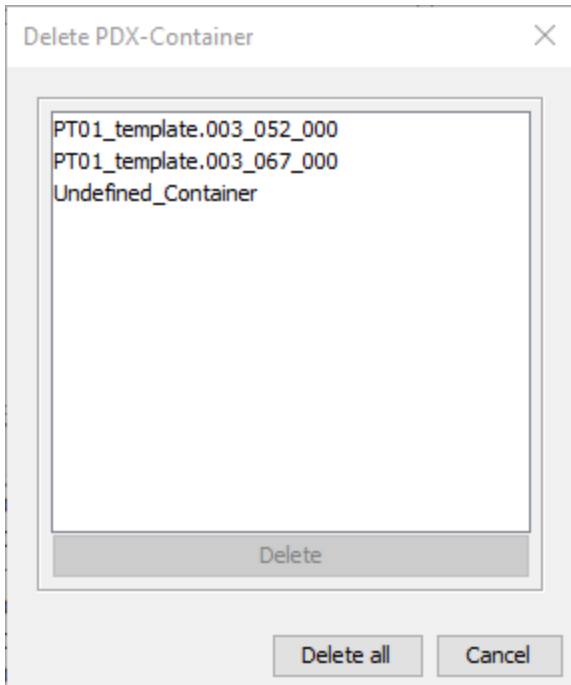
Either individual container files or all containers can be deleted.

The deletion process is limited to entries in the subdirectory "**..\psdzdata\swe**" and "**..\psdzdata\mainseries**".

Only those files that are not needed by any other PDX container are deleted.

To delete individual container contents, the corresponding PDX containers have to be marked in the selection list and the button 'Delete' has to be pressed.

Deleting all containers is done with the 'Delete All' button:



### **Specials:**

- Deletion Container is available only if there is no connection to the ECU.
- In the case of a missing or incorrect index file a new one index file will be created. All existing SWEs files and project names (TargetSelectors) are written in this file as "Undefined\_Container".
- By Import of the same container more times under different project names, all linked project names (Target Selectors) are to deleted by deleting of this container.

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

The PDX container check could be started with the button "ODX checker".

The result is presented in the "ODX checker messages" table.

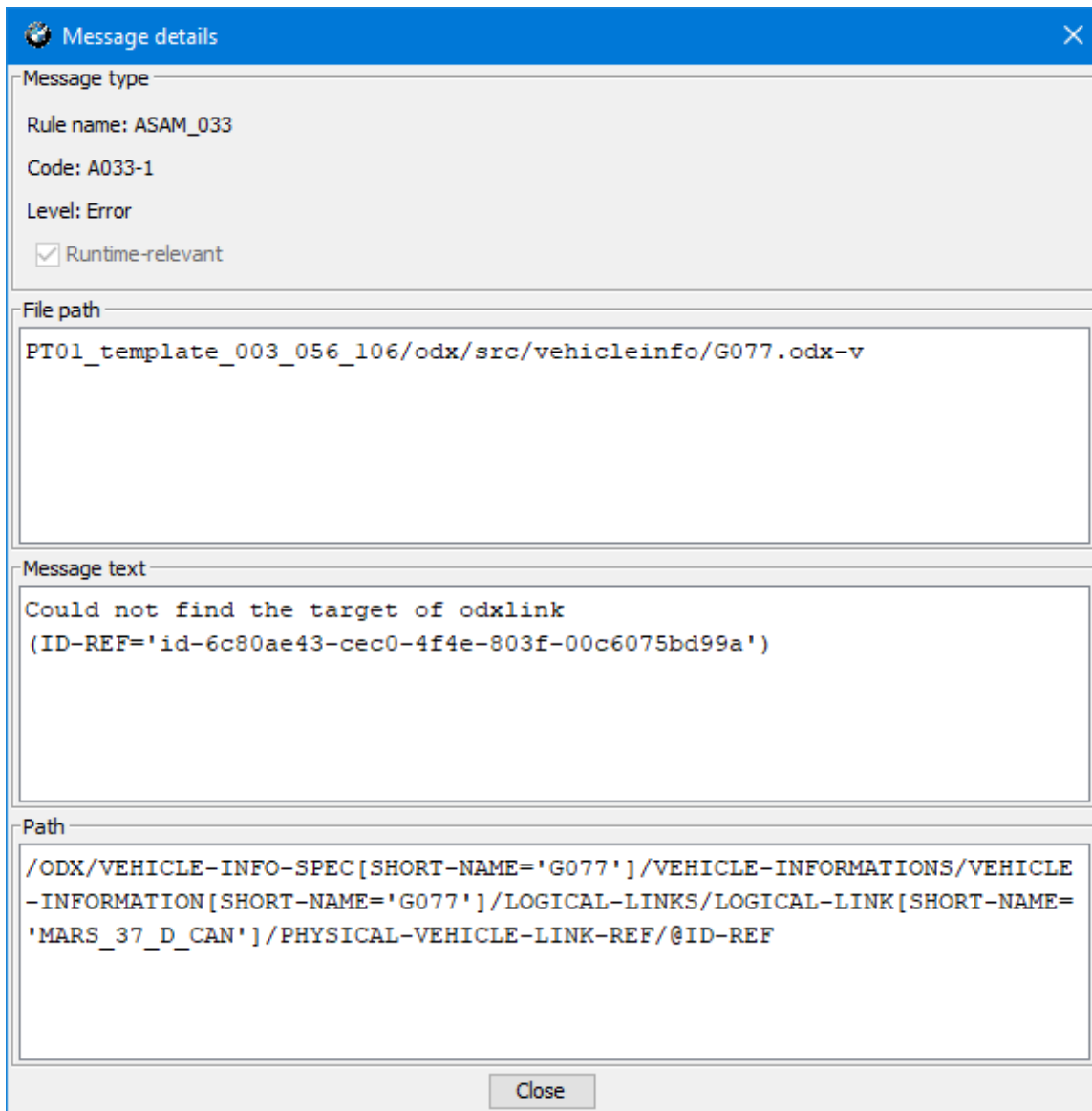
The screenshot shows the ODX-Checker application interface. At the top, there is a menu bar with "File", "Options", "Extras", and "Help". Below the menu bar is a toolbar with various icons. The main window is divided into several sections:

- Left sidebar:** Contains buttons for "Comfort Mode", "Expert Mode", "Editors & Viewers", and "Data Handling". Below these are icons for "PDX-Charger", "File-Explorer", and "SWE-Explorer".
- Top right area:** Contains input fields for "PDX-Container:" (C:\Data\ODX\PT01\_template.003\_056\_106.pdx) and "PDX-Template:" (PT01\_template.003\_056\_106.pdx). There are buttons for "Import into PsdZ", "Update", and "ODX-Checker". A "Delete Container..." button is also present.
- Valid for main series:** A text box containing "M012, M013, RR01, RR21, S15A, S15C, S18A, S18T, U006, V002, V008, V009, V010, V099, X001".
- Tree view:** Shows a hierarchy under "PDX-Container" with sub-items: GWTB, CAF, SVT, Config, and DOC.
- Files ODX-Checker messages:** A tab showing the results of the ODX checker. It displays "2 errors, 0 warnings; 192 rules checked". Below this is a table with the following data:

Rule name	Rule level	File name	Message text
ASAM_033	Error	G077.odx-v	Could not find the target of odxlink (ID-REF=id-6c80ae43-cec0-4f4e-803f-00c6075bd99a)
ASAM_033	Error	G077.odx-v	Could not find the target of odxlink (ID-REF=id-31e995c1-f0d2-406e-a913-0495621ddedd)

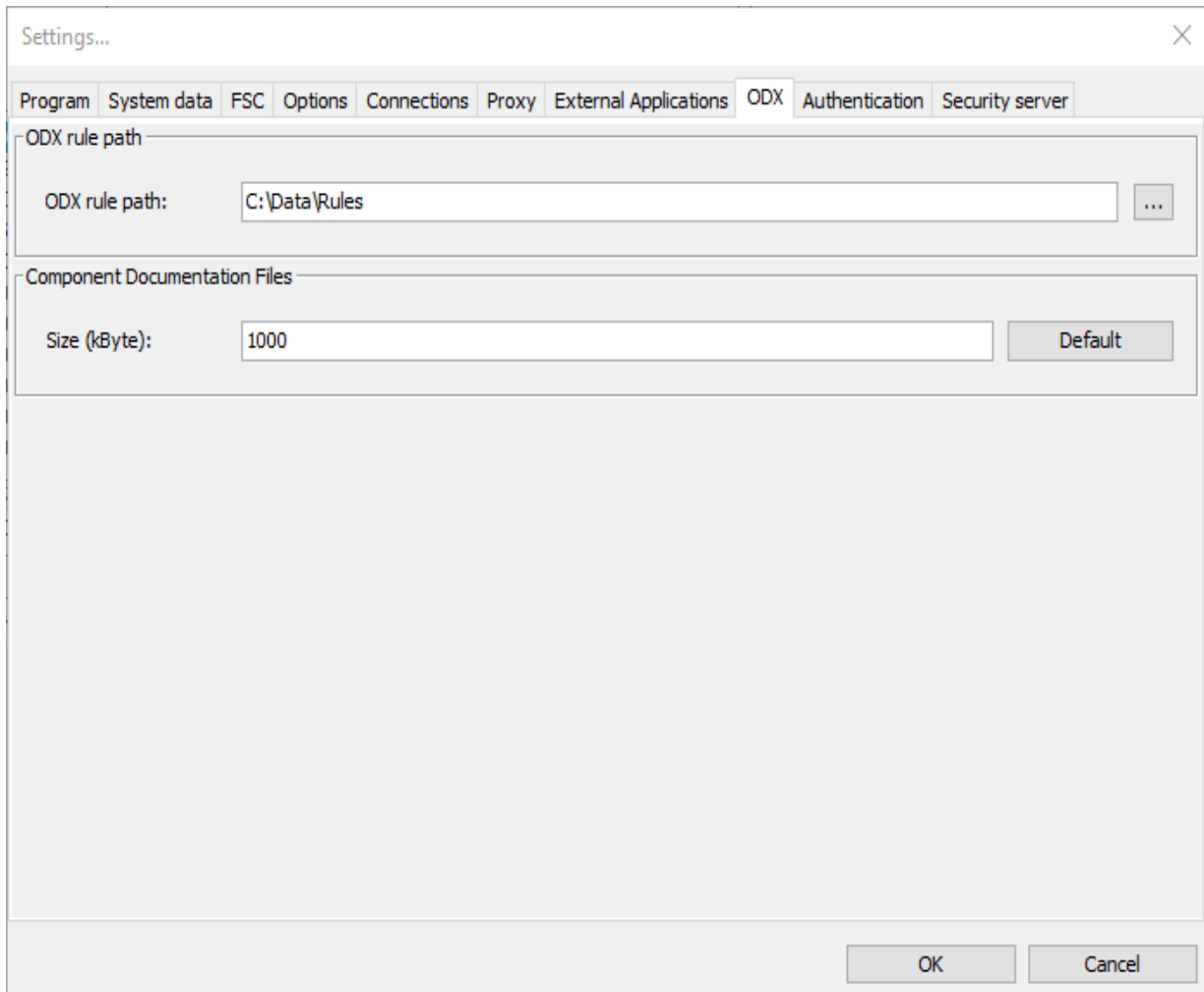
At the bottom of the application, there is a "Login" button and a status bar showing "SWL-Sec: Logged out".

Message details can be open with a double click on a line in the table.



The ODX checker rules could be stored in a directory. The path for this directory could be set in menu "Options / Settings..." in the "ODX" tab:





This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

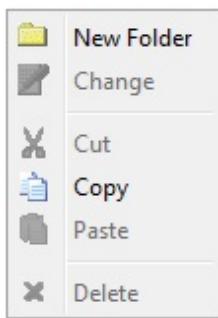
Using the file explorer you can manage the elements of the data directory. The files or directories can be copied, moved, deleted and created newly. In addition to that also an import into PSdZ (in case of ODX files) or passing the file to an external editor is possible.

The screenshot shows a file explorer window with the following components:

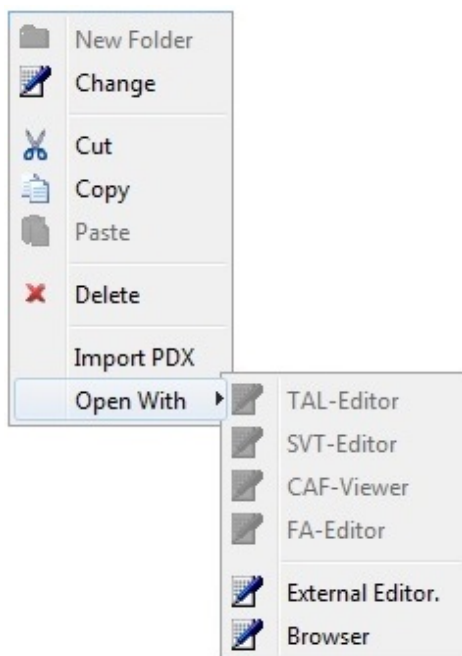
- Menu Bar:** File Edit Options Extras Help
- Toolbar:** Contains icons for back, forward, refresh, print, and help.
- Sidebar:**
  - Comfort Mode
  - Expert Mode
  - Editors & Viewers
  - Data Handling
  - PDX-Charger
  - File-Explorer
  - SWE-Explorer
  - External Applications
  - Personal view
- Directory Tree (Left):**
  - Data\_3.37x
    - CAF
    - CERT
    - Etc
    - ExecutedTAL
    - FA
    - FP
    - FV
    - IDR
    - KDS
    - Logs
    - ODX
    - psdzdata
    - Rules
    - SFA
    - SVT
    - SWE
    - SWT
    - TAL
    - TalFilter
- File List (Right):**

File Name ^	Size	Type	Changed
CAF			01/27/2021 11:24:58 AM
CERT			01/27/2021 11:24:58 AM
Etc			01/27/2021 11:24:15 AM
ExecutedTAL			01/27/2021 11:24:58 AM
FA	4 KB		02/02/2021 12:57:52 PM
FP			01/27/2021 11:24:15 AM
FV			01/27/2021 11:24:58 AM
IDR			01/27/2021 11:24:58 AM
KDS			01/27/2021 11:24:15 AM
Logs	4 KB		02/03/2021 09:18:51 AM
ODX			01/27/2021 11:24:15 AM
psdzdata	4 KB		01/27/2021 11:27:19 AM
Rules	4 KB		01/27/2021 11:24:32 AM
SFA			01/27/2021 11:24:15 AM
SVT			01/27/2021 11:24:15 AM
SWE			01/27/2021 11:24:15 AM
SWT			01/27/2021 11:24:15 AM
TAL			01/27/2021 11:24:15 AM
TalFilter			01/27/2021 11:24:58 AM
- Status Bar:** Login SWL-Sec: Logged out

**Context Menu (left hand side)**



### **Context Menu (right hand side)**



### **Specials:**

- In the right hand display you have many options of editing using the context menu.
- "ImportPDX" only is available in subdirectory ODX, and even then only if there is no connection to the ECU.
- With the item "ImportPDX", it is possible to import single PDX-Container as well as a set of containers simultaneously.
- Using "Open With" menu let you jump directly to the suitable E-Sys editor corresponding to the data

structure. In this editor the desired file is already displayed.

- The other context menu items are to be used in common manner.

---

**This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.**

# SWE-Explorer

[Top](#) [Previous](#) [Next](#)

You can watch the both subdirectories SWE\_signed and SWE\_unsigned using this explorer.

File Options Extras Help

Filename ^	Length (kB)	Changed	Signature	Compression
swfi_00000123_001_002_003.bsw	932	Feb 25, 2021		NOT DEFINED
swfi_00000123_001_002_006.bsw	925	Feb 25, 2021	SIGNED	COMPRESSED
swfi_00000123_002_002_002.bsw	2	Feb 25, 2021		NOT DEFINED
swfi_00000123_003_003_003.bsw	2	Feb 25, 2021		NOT DEFINED
swfi_0000a12f_002_002_002.bsw	2	Feb 25, 2021	SIGNED	COMPRESSED
swfi_10002001_001_002_003.bsw	1	Feb 25, 2021		

Login SWL-Sec: Logged out

## Context Menu

doesn't exist. This is no software bug.

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

## [Applications](#)

---

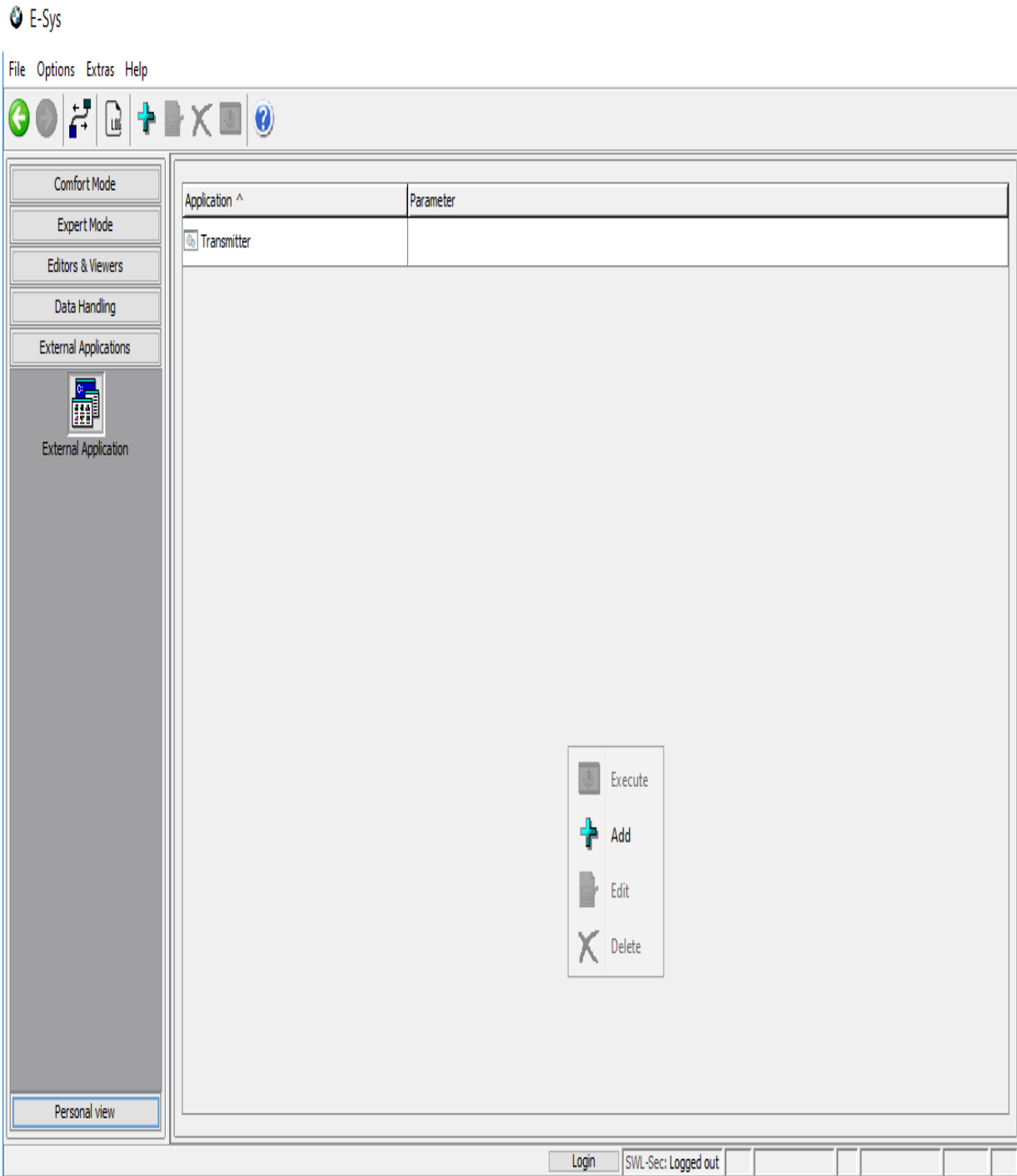
This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.



# Applications

[Top](#) [Previous](#) [Next](#)

In this module you can start any application. Therefore you must add a application by using the add button in the tool bar or in the context menu.



---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

[Motor Bike](#)

[Establish connection](#)

[Batch](#)

---

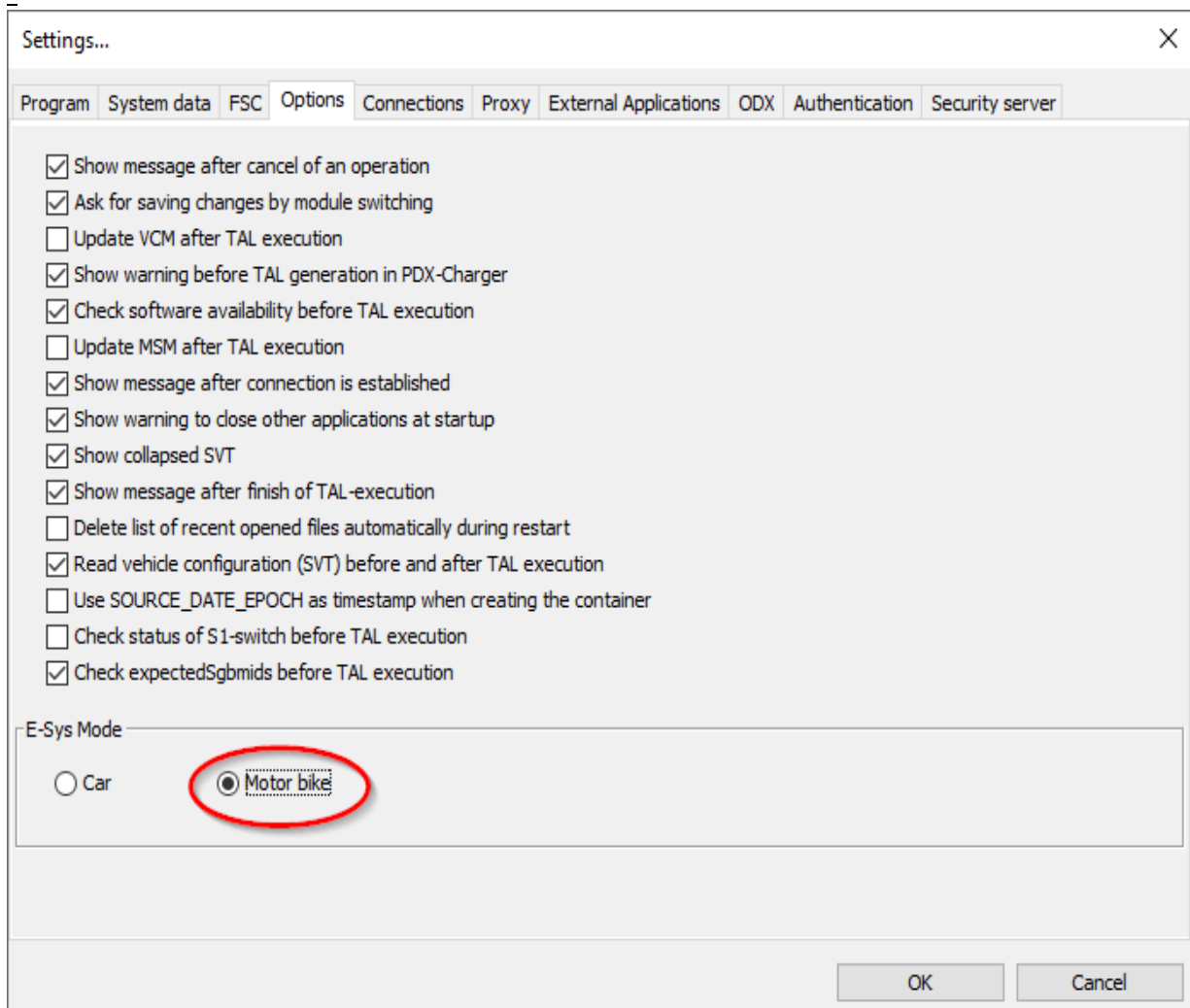
This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

# Motor Bike

[Top](#) [Previous](#) [Next](#)

With E-Sys also motor bikes can be programmed and coded. Therefore E-Sys mode must be changed to "Motor bike". See the menu Options | Settings... | Options.

## Options

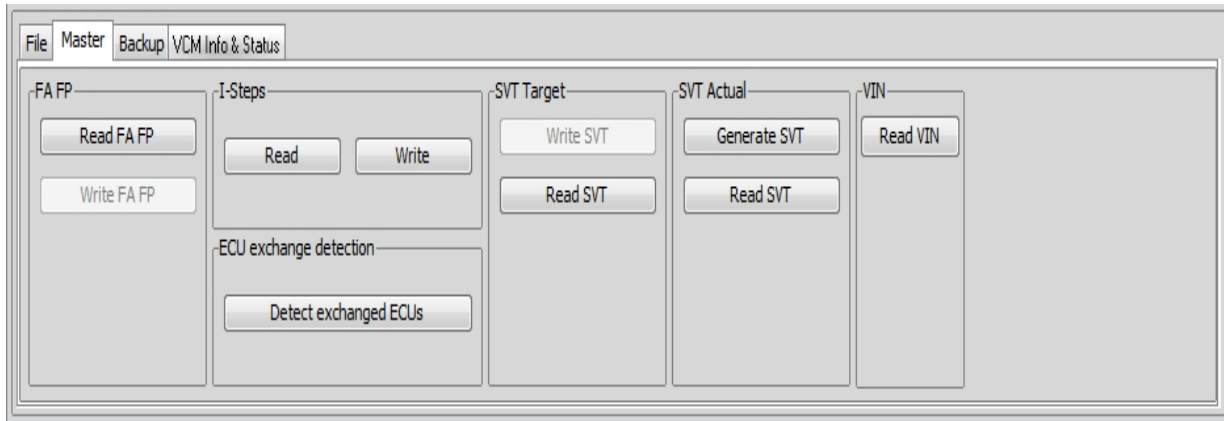


In mode motor bike the option "Update MSM after TAL execution" must be turned off, because of the missing MSM in motor bikes.

## Disabled Actions

Several Emus of a car are not built in motor bikes. Thus some actions from E-Sys mode car are not available in mode motor bike. In the VCM Master tab of the VCM module the following actions are disabled:

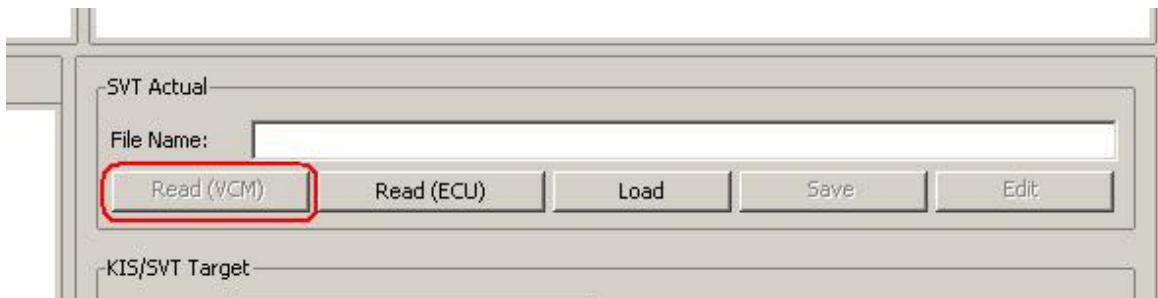
- FA VIN -- Update VIN
- ECU exchange detection -- Detect exchanged Emus
- SVT Target -- Write SVT, Read SVT
- SVT Actual -- Generate SVT, Read SVT



In the TAL-Calculation module and the Coding module the action

- SVT Actual -- Read (VCM)

is disabled.



### **Changed Functionality**

In Modula VCM several actions read, write, or generate a vehicle profile (FP). This functionality is not available for motor bikes and thus will be omitted in the mode motor bike for the following actions:

- VCM Update after TAL execution
- VCM Master: FA VIN -- Read FA VIN
- VCM Master: FA VIN -- Write FA
- VCM Backup: VCM -- Write Data
- VCM Backup: Restore Data

Beside this, the actions will be executed the same as in mode car.

**Currently not used**

TSL-Update is currently not used in mode motor bike.

NAV/ENT Update is currently not used in mode motor bike.

---

This help file was created with an unregistered evaluation copy of Help & Manual.  
© EC Software. All rights reserved. This message will not appear if you compile  
this help file with the registered version of Help & Manual.

## Establish connection

[Top](#) [Previous](#) [Next](#)

Some functionalities of E-Sys need a established connection with the PDX date or the vehicle. Subsequent the process for establishing a connection with the vehicle is described shortly.



In the symbol bar you find the icon for starting/stopping a connection.

After the click on "establish connection" subsequent dialog is displayed:

The screenshot shows the 'Open Connection' dialog box with the following details:

- Target:**
  - Main series: M013
  - Connection type: All
  - TargetSelector list:
    - M012 =M013\_AAG\_01\_EXERCISES\_037\_000\_010, VehideInfo=M013
    - M013 =M013\_AAG\_01\_EXERCISES\_037\_000\_010, VehideInfo=M013\_DIRECT
    - NA05 =M013\_PT01\_TEMPLATE\_003\_063\_000, VehideInfo=M013
    - RR01 =M013\_PT01\_TEMPLATE\_003\_063\_000, VehideInfo=M013\_DIRECT
    - RR21 =M013\_PT01\_TEMPLATE\_003\_093\_006, VehideInfo=M013
    - S15A =M013\_PT01\_TEMPLATE\_003\_093\_006, VehideInfo=M013\_DIRECT
    - S15C =M013\_PT01\_TEMPLATE\_003\_100\_000, VehideInfo=M013
    - S18A =M013\_PT01\_TEMPLATE\_003\_100\_000, VehideInfo=M013\_DIRECT
  - TargetSelector: Project=M013\_PT01\_TEMPLATE\_003\_100\_000, VehideInfo=M013\_DIRECT
- Interface:**
  - Connection via bus: UNKNOWN (dropdown), unknown (dropdown)
  - ECU-URL: tcp://127.0.0.1:6801
  - Connection via gateway URL:  tcp://127.0.0.1:6801
  - Connection via ICOM/D-CAN:  tcp://127.0.0.1:52410
  - Connection via ICOM/Ethernet:  tcp://127.0.0.1:50160
  - Connection via VIN: No ZGW available (dropdown) [Refresh]
  - Number of available vehicles: 0
- Vehicle-specific parameter:**
  - Series, I-step (shipment) [dropdown] [dropdown]
  - Read parameters from VCM
- Buttons: Connect, Cancel

In the window "Target" you find all available projects and vehicle information. These could be filtered by the main series. With the second filter selects either gateway or direct ECU connection.

After the selection of a target selector and vehicle info the connection could be established by clicking on the "Connect" button.

If the connections was established successfully in the status bar you see the selected target.

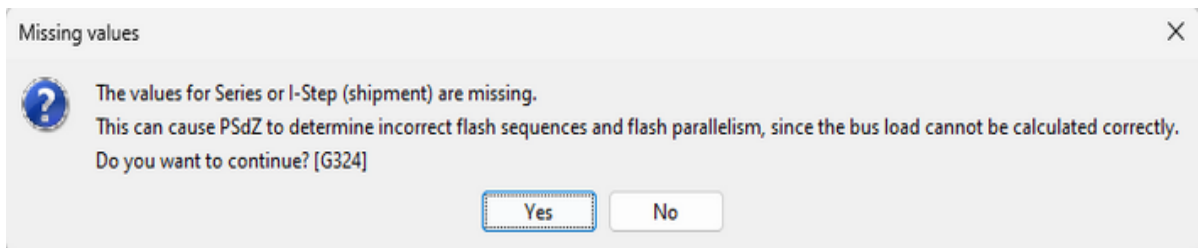
The type of the connection could be chosen in the "Interface" part of the dialog:

- For a connection via bus the bus type and a interface must be selected.
- For a connection via gateway URL the appropriate URL with port must be defined.
- For a connection via ICOM/CAN the URL of the ICOM with the appropriate port must be defined.
- For a connection via ICOM/ETHERNET the URL of the ICOM with the appropriate port must be defined. After an initial connection the port will be calculated from the base port taken from the settings.
- For a connection via VIN a vehicle could be chosen.

In the part "Vehicle-specific parameters" the series and I-step should be set. There are two options:

- Fill the fields: Series and I-step.

If any of these fields are left empty the following warning will be shown. Despite the warning, the user can try to establish the connection.

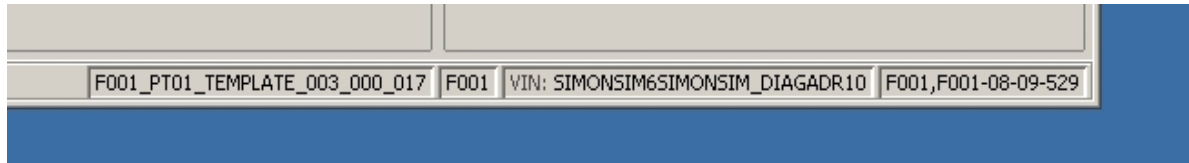


- Read series and I-step from VCM and confirm, or change the



values in the following dialog.

Information about the established connection are shown on the right hand side in the status bar.



---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

## **Secure Coding**

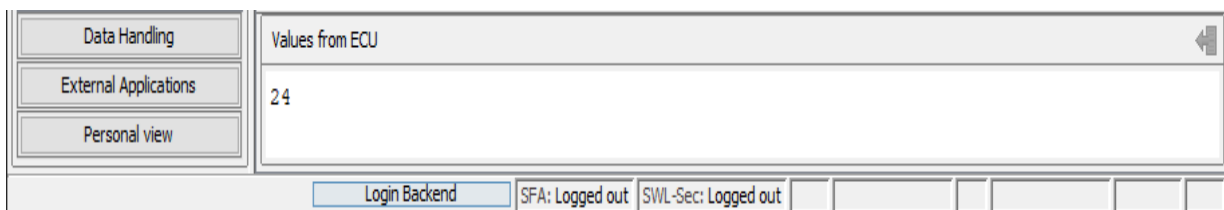
To code an ECU signed NCDs (Netto-Coding-Files) are used. Based of secure aspects the lettering of the NCDs is performed in backend-system (outside of E-Sys). Therefore the user has to be applied at the backend-system as well as to establish a backend connection.

### **Backend-connection for the secure Coding**

To perform a TAL, that includes coding actions a backend-connection (to SWL-Sec2) is needed. If you did not activated the backend-connection already the system is asking you to do it now. Therefore a popup appear (screenshot below) where the username, password and access point (BMW-Intranet or Internet) can be set.

Two authentication methods are supported:

- username/password
- Certificate-based authentication



The button to perform the backend-connection can be find at the status bar .

After the E-Sys user has clicked on the button, a pop-up window is going to appear. You have now two different possible configuration for the backend-system (*BMW-Intranet / Internet*).

Backend Login ✕

SWL-SEC SFA

Authentication method: CERTIFICATE ▾

Access point: Access via BMW-Intranet ▾  
Access via BMW-Intranet  
Access via Internet

User:

Password:

Remember username

Login

Backend Login ✕

SWL-SEC SFA

Authentication method: BASIC ▾  
BASIC  
CERTIFICATE

Access point:

User:

Password:

Remember username

Login

Backend Login

SWL-SEC SFA

Authentication method: BASIC

Access point: Access via BMW-Intranet

User: tester42

Password: ●●●●●●●●●●

Remember username

Login

**If you choose *BMW-Intranet* your computer must be in the BMW IT-network.**

The status of the authentication is indicated in the status bar. In addition to the selected authentication method, the activated role and the validity of the role are displayed. In the case of successful Certificate-based authentication, the quota and the Certificate's usage time are also displayed.

Data Handling Values from ECU

External Applications 24

Personal view

Login Backend SFA: Logged out **SWL-Sec: BASIC** Role: Sign NCD (13.12.2022)

Data Handling Values from ECU

External Applications 24

Personal view

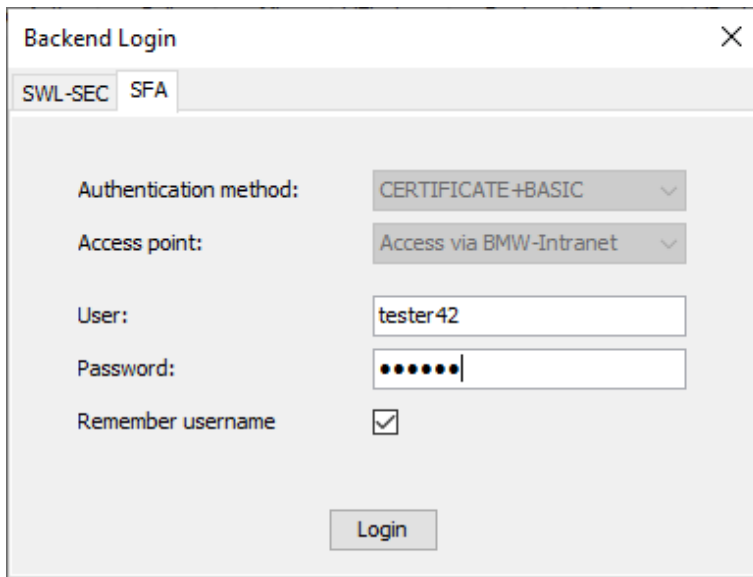
Login Backend SFA: Logged out **SWL-Sec: CERTIFICATE** Role: Sign NCD (13.12.2022)

## SFA Backend

The communication with the SFA Backend is required to write a secure token set, switch ECUs, which supports Secure ECU Modes, between the three states (Engineering, Plant, Field) and generates a response-file. An imported client certificate for CBB/SFA is required for communication. For extended authorization, BASIC login data is needed.

The button to set the BASIC credentials can be find at the status bar .

After the E-Sys user has clicked on the button, a pop-up window is going to appear. You have to switch to the tap "SFA".

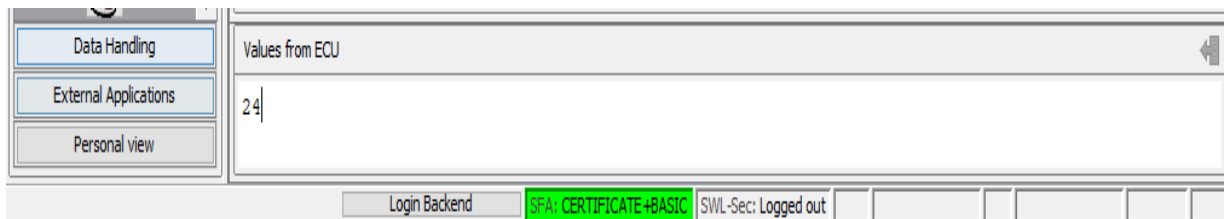


The image shows a 'Backend Login' dialog box with a close button (X) in the top right corner. It has two tabs: 'SWL-SEC' and 'SFA', with 'SFA' selected. The dialog contains the following fields and options:

- Authentication method: A dropdown menu showing 'CERTIFICATE+BASIC'.
- Access point: A dropdown menu showing 'Access via BMW-Intranet'.
- User: A text input field containing 'tester42'.
- Password: A password input field with seven dots.
- Remember username: A checked checkbox.
- Login: A button at the bottom center.

**Your computer must be in the BMW IT-network.**

Whether the Basic credentials are set is displayed in the status bar.



**This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.**

[General information on batch operation](#)

[General overview of commands](#)

[Anflash](#)

[Coding\\_\(NCD, verification, back-end authentication, etc.\)](#)

[CVN / OBD](#)

[Read data](#)

[ECU Modes / SFA \(Secure Feature Activation\)](#)

[Server-Client operation](#)

[FA convert](#)

[FV ExecuteFV](#)

[PDX Containers](#)

[Proxy](#)

[FSC / SWT \(sweeping Technologie\)](#)

[TAL processing](#)

[TAL calculation](#)

[TSL](#)

[VCM \(Vehicle Configuration Management\)](#)

[Certificate Management \(SP18, SP21, etc.\)](#)

[ConnectionParameter](#)

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.



### **General:**

In addition to GUI mode, E-Sys can also be run in batch mode. For this purpose, the file E-Sys.bat with various parameters is started in the E-Sys installation directory (default: C:\EC\Apps\ESG\E-Sys). Depending on the specified parameter, an appropriate action is performed.

In addition, 2 operating modes are available to the user in batch modes.

One is client batch mode and other is client-server batch mode.

### **What is client operation?**

In client mode, the following steps are always followed during TAL processing in batch mode:

1. Start E-Sys batch instance
2. Establish connection to vehicle
3. Execute TAL
4. Close connection to vehicle
5. Close E-Sys-Batch Instance

If you want to execute multiple TALs with the same vehicle connection, opening and closing the vehicle connection and starting and stopping an E-Sys batch instance means unnecessary overhead.

On the production-line, fast coding times are required, so a client-server-based batch mode has been introduced.

Thus, it is possible to start an E-Sys server instance, which is controlled with the help of a client.

In this way, individual E-Sys functions can be offered in batch.

### **How client-server batch mode works:**

An E-Sys batch server is started, waiting for client requests. To check if an E-Sys batch server is running, the `-check` command is offered (see [Server Client Operation](#)).

A client can be started by specifying the commands with the batch parameter `-server` in another command line window (see [overview of commands](#)).

The client connects to a running E-Sys batch server, transfers the parameters specified on the command line to the server, and waits for a response.

The server performs the appropriate action and reports the result to the client. The client then exits with the appropriate return value.

The E-Sys batch server runs until it exits.

### **Example of a server-client operation:**

1. Start E-Sys batch instance
2. Establish connection to vehicle
3. Read SVT
4. Close connection to vehicle
5. Import PDX Container
6. Establish connection to vehicle
7. Calculate TAL
8. Execute TAL
9. Close connection to vehicle
10. Read SVT

## 11. Close E-Sys-Batch Instance

### **Outputs to the console**

When an action is written to the console, messages are output to the client console.

### **Logging**

The log behavior can be set in the E-Sys.properties.

Default behavior: (log.file.client = false or not present in E-Sys.properties)

The client does not create a log file. This is not necessary because the client is only used to start an action on the server.

This means that the complete business logic runs on the server and is also logged there.

Log-File Splitting in Server Client Mode: (log.file.client = true in E-Sys.properties)

This setting creates a new log file for each client request for the server and client.

The file name shows which client request is involved.

Example:

```
E-Sys_20160203_150924_client_openconnection.log  
E-Sys_20160203_150924_server_openconnection.log  
E-Sys_20160203_151145_client_closeconnection.log  
E-Sys_20160203_151145_server_closeconnection.log
```

## **Client-Server Connection**

A TCP connection is established between the client and the server via port 30303.

If the port is not available, another port can be configured in the E-Sys.properties file (BatchServer.port) (see `-prop <E-Sys.properties>`).

Connections between client and server go through "localhost", i.e. the client and server must always be started on the same machine.

## **Relative Paths**

Batch can use relative paths in addition with absolute paths.

This feature is configured in the E-Sys properties files via the line `Path.Batch=`

Only the properties file needs to be defined using absolute path.

Example:

```
Path.Batch=C:\Users\User\Documents\
```

## **Restrictions**

Only one E-Sys instance can run on a computer at a time. This means that either the GUI mode, the E-Sys client or the E-Sys batch server must be started.

Only one client can be connected to the server at a time.

### **Note: TAL processing in client-server batch mode**

While the client that started the TAL processing is still waiting for the result, another client can request a cancellation of the TAL processing.

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

## Legend:

- [ ].....optional, can be specified, but does not have to
- ( ).....Selection, must be specified
- | ..... "or", is used in a selection

## [Anflash:](#)

E-Sys.bat -anflash <config file> [-ignoreBATHAF]  
E-Sys.bat -server -anflash <config file> [-ignoreBATHAF]

## [Coding \(NCD, verification, back-end authentication, etc.\):](#)

E-Sys.bat -server -authenticationCoding -connection <bmw\_intranet|internet> (-user <username>|-useSwlSecCertificate) [-dialog]  
E-Sys.bat -server -backendCheckSigning -vin <vin file> -sgbmno <SGBM file>  
~~E-Sys.bat -fwl2ncd <config file> **(Veraltet, bitte fwl2Ncd verwenden)**~~  
~~E-Sys.bat -server -fwl2ncd <config file> **(Veraltet, bitte fwl2Ncd verwenden)**~~  
E-Sys.bat -fwl2Ncd <config file>  
E-Sys.bat -server -fwl2Ncd <config file>  
~~E-Sys.bat -generatencd <FA file> [-caf <CAF name>] -trace <trace directory> **(Veraltet, bitte generateNcd verwenden)**~~  
~~E-Sys.bat -server -generatencd <FA file> [-caf <CAF name>] -trace <trace directory> **(Veraltet, bitte generateNcd verwenden)**~~  
E-Sys.bat -generateNcd <FA file> [-caf <CAF name>] -trace <trace directory>  
E-Sys.bat -server -generateNcd <FA file> [-caf <CAF name>] -trace <trace directory>  
~~E-Sys.bat -ncdsign <config file> **(Veraltet, bitte signNcd verwenden)**~~  
~~E-Sys.bat -server -ncdsign <config file> **(Veraltet, bitte signNcd verwenden)**~~  
E-Sys.bat -signNcd <config file>  
E-Sys.bat -server -signNcd <config file>  
~~E-Sys.bat -readncd <SVT file> -connection <config file> -out <NCD directory> **(Veraltet, bitte readNcd verwenden)**~~  
~~E-Sys.bat -server -readncd <SVT file> -connection <config file> -out <NCD directory> **(Veraltet, bitte readNcd verwenden)**~~  
E-Sys.bat -readNcd <SVT file> -connection <config file> -out <NCD directory> [-notReadVin]

E-Sys.bat -server -readNcd <SVT file> -connection <config file> -out <NCD directory> [-notReadVin]  
E-Sys.bat -importSwlSecCertificate <P12 file> [-dialog]  
E-Sys.bat -server -importSwlSecCertificate <P12 file> [-dialog]  
E-Sys.bat -generateTestNcds <config file>  
E-Sys.bat -server -generateTestNcds <config file>  
E-Sys.bat -createCustomerNcd <config file>  
E-Sys.bat -server -createCustomerNcd <config file>

## CVN / OBD:

E-Sys.bat -comparecvn -connection <config file> [-sg <ecu list>] (-svt <SVT file>|-svtvcmist|-svtvcmsoll|-svtecu) -trace <trace directory> [-fa <FA file>] [-xmlreport]  
E-Sys.bat -server -comparecvn -connection <config file> [-sg <ecu list>] (-svt <SVT file>|-svtvcmist|-svtvcmsoll|-svtecu) -trace <trace directory> [-fa <FA file>] [-xmlreport]  
E-Sys.bat -cvncalculation <FA file> -out <OBD-CVN-FA-List file> [-caf <CAF name>]  
E-Sys.bat -server -cvncalculation <FA file> -out <OBD-CVN-FA-List file> [-caf <CAF name>]

## Read data:

E-Sys.bat -getbrv <series>  
E-Sys.bat -server -getbrv <series>  
E-Sys.bat -help  
E-Sys.bat -server -help  
E-Sys.bat -readfa -connection <config file> -out <FA directory>  
E-Sys.bat -server -readfa -connection <config file> -out <FA directory>  
E-Sys.bat -readsvt -connection <config file> -out <SVT directory>  
E-Sys.bat -server -readsvt -connection <config file> -out <SVT directory>  
E-Sys.bat -version  
E-Sys.bat -server -version

## ECU Modes / SFA / LCS:

E-Sys.bat -server -authenticationSfa -user <username> [-dialog]  
E-Sys.bat -server -removeAuthenticationSfa

E-Sys.bat -activateAcl -connection <config file> -svt <SVT file> [-blacklist <file>] [-whitelist <file>]  
E-Sys.bat -server -activateAcl -connection <config file> -svt <SVT file> [-blacklist <file>] [-whitelist <file>]

E-Sys.bat -activateSecOC -connection <config file> -svt <SVT file> [-noreset] [-blacklist <file>] [-whitelist <file>]  
E-Sys.bat -server -activateSecOC -connection <config file> -svt <SVT file> [-noreset] [-blacklist <file>] [-whitelist <file>]  
E-Sys.bat -clearSecureToken -connection <config file> -featureid <FeatureID> (-diagaddress <diagnosis address>|-svt <SVT file>)  
E-Sys.bat -server -clearSecureToken -connection <config file> -featureid <FeatureID> (-diagaddress <diagnosis address>|-svt <SVT file>)  
E-Sys.bat -createRequestEcuMode -connection <config file> (-plant|engineering) [-vin <VIN17>] [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]  
E-Sys.bat -server -createRequestEcuMode -connection <config file> (-plant|engineering) [-vin <VIN17>] [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]  
E-Sys.bat -createRequestNewestPackage -vin <VIN17> [-whitelist <file>] [-rebuild]  
E-Sys.bat -server -createRequestNewestPackage -vin <VIN17> [-whitelist <file>] [-rebuild]  
E-Sys.bat -createRequestPackageForOrder -vin <VIN17> [-rebuild]  
E-Sys.bat -server -createRequestPackageForOrder -vin <VIN17> [-rebuild]  
E-Sys.bat -deactivateAcl -connection <config file> -svt <SVT file> [-blacklist <file>] [-whitelist <file>]  
E-Sys.bat -server -deactivateAcl -connection <config file> -svt <SVT file> [-blacklist <file>] [-whitelist <file>]  
E-Sys.bat -deactivateSecOC -connection <config file> -svt <SVT file> [-noreset] [-blacklist <file>] [-whitelist <file>]  
E-Sys.bat -server -deactivateSecOC -connection <config file> -svt <SVT file> [-noreset] [-blacklist <file>] [-whitelist <file>]  
E-Sys.bat -deleteSecureToken -connection <config file> -featureid <FeatureID> -diagaddress <diagnosis address>  
E-Sys.bat -server -deleteSecureToken -connection <config file> -featureid <FeatureID> -diagaddress <diagnosis address>  
E-Sys.bat -discoverAllFeatureStatus -connection <config file> [-svt <SVT file>]  
E-Sys.bat -server -discoverAllFeatureStatus -connection <config file> [-svt <SVT file>]  
E-Sys.bat -getResponseFromRequest [-out <response file>] -request <request file> -svt <SVT file>  
E-Sys.bat -server -getResponseFromRequest [-out <response file>] -request <request file> -svt <SVT file>  
E-Sys.bat -readShortStatus -connection <config file> -featureid <FeatureID> -diagaddress <diagnosis address>  
E-Sys.bat -server -readShortStatus -connection <config file> -featureid <FeatureID> -diagaddress <diagnosis address>  
E-Sys.bat -readSoftwareVersion -connection <config file> -diagaddress <diagnosis address>  
E-Sys.bat -server -readSoftwareVersion -connection <config file> -diagaddress <diagnosis address>



E-Sys.bat -switchEcuMode -connection <config file> -tokenset <SecureToken file> [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]  
E-Sys.bat -server -switchEcuMode -connection <config file> -tokenset <SecureToken file> [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]  
E-Sys.bat -switchEcuModeField -connection <config file> [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]  
E-Sys.bat -server -switchEcuModeField -connection <config file> [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]  
E-Sys.bat -switchEcuModeOnline -connection <config file> (-plant|-engineering) [-vin <VIN17>] [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]  
E-Sys.bat -server -switchEcuModeOnline -connection <config file> (-plant|-engineering) [-vin <VIN17>] [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]  
E-Sys.bat -verifySecureTokens -connection <config file> [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]  
E-Sys.bat -server -verifySecureTokens -connection <config file> [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]  
E-Sys.bat -verifySecureTokensVehicle -connection <config file> [-svt <SVT file>]  
E-Sys.bat -server -verifySecureTokensVehicle -connection <config file> [-svt <SVT file>]  
E-Sys.bat -readSecureTokenImportStatus -connection <config file> [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]  
E-Sys.bat -server -readSecureTokenImportStatus -connection <config file> [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]  
E-Sys.bat -writeNewestPackageOnline -connection <config file> [-disableDelete] [-vin <VIN17>] [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>] [-rebuild]  
E-Sys.bat -server -writeNewestPackageOnline -connection <config file> [-disableDelete] [-vin <VIN17>] [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>] [-rebuild]  
E-Sys.bat -writePackageForOrderOnline -connection <config file> [-disableDelete] [-vin <VIN17>] [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>] [-rebuild]  
E-Sys.bat -server -writePackageForOrderOnline -connection <config file> [-disableDelete] [-vin <VIN17>] [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>] [-rebuild]  
E-Sys.bat -writeSecureTokens -connection <config file> -tokenset <SecureToken file> [-svt <SVT file>] [-disableDelete] [-blacklist <file>] [-whitelist <file>]  
E-Sys.bat -server -writeSecureTokens -connection <config file> -tokenset <SecureToken file> [-svt <SVT file>] [-disableDelete] [-blacklist <file>] [-whitelist <file>]  
E-Sys.bat -writelpsecBitmask -connection <config file> -svt <SVT file> [-blacklist <file>] [-whitelist <file>]  
E-Sys.bat -server -writelpsecBitmask -connection <config file> -svt <SVT file> [-blacklist <file>] [-whitelist <file>]

E-Sys.bat -readAcl -connection <config file> -svt <SVT file> [-blacklist <file>] [-whitelist <file>]

E-Sys.bat -server -readAcl -connection <config file> -svt <SVT file> [-blacklist <file>] [-whitelist <file>]

E-Sys.bat -readLcs -connection <config file> [-lcsName <LCS name>] [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]

E-Sys.bat -server -readLcs -connection <config file> [-lcsName <LCS name>] [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]

E-Sys.bat -setLcs -connection <config file> -lcsName <LCS name> -lcsValue <LCS value> [-noreset] [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]

E-Sys.bat -server -setLcs -connection <config file> -lcsName <LCS name> -lcsValue <LCS value> [-noreset] [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]

E-Sys.bat -writeLcsToken -connection <config file> -tokenset <SecureToken file> [-noreset] [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]

E-Sys.bat -server -writeLcsToken -connection <config file> -tokenset <SecureToken file> [-noreset] [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]

## Server-Client operation:

E-Sys.bat -server -check

E-Sys.bat -server -stop

E-Sys.bat -server -openconnection <config file>

E-Sys.bat -server -closeconnection

E-Sys.bat -server -aborttallexecution

E-Sys.bat -startserver

E-Sys.bat -startserver [-prop <property file>] [-setProxy <config file>] [-authenticationCoding ( -user <Benutzername> | -useSwlSecCertificate ) -connection <bmw\_intranet|internet>] [-dialog]

## FA convert:

E-Sys.bat -convertorderxml <FO directory> -out <FA directory>

E-Sys.bat -server -convertorderxml <FO directory> -out <FA directory>

## FV ExecuteFV:

E-Sys.bat -executeFv <config file> [-ignoreBATHAF]

E-Sys.bat -server -executeFv <config file> [-ignoreBATHAF]

## PDX Containers:

E-Sys.bat -mergepdxcontainer <PDX container definition file> -out <PDX container file>  
E-Sys.bat -server -mergepdxcontainer <PDX container definition file> -out <PDX container file>  
E-Sys.bat -pdxcontainer <PDX container definition file> -out <PDX container file>  
E-Sys.bat -server -pdxcontainer <PDX container definition file> -out <PDX container file>  
E-Sys.bat -pdximport <PDX container file> -project <project name>  
E-Sys.bat -server -pdximport <PDX container file> -project <project name>  
E-Sys.bat -pdxupdate <PDX container file> [-out <PDX container file>] -template <PDX template file>  
E-Sys.bat -server -pdxupdate <PDX container file> [-out <PDX container file>] -template <PDX template file>

## Proxy:

E-Sys.bat -readStatusProxy  
E-Sys.bat -server -readStatusProxy  
E-Sys.bat -server -removeProxy  
E-Sys.bat -server -setProxy <config file> [-dialog]

## SWT:

E-Sys.bat -swtaction <config file>  
E-Sys.bat -server -swtaction <config file>  
E-Sys.bat -swtstatus -project <project name> -vehicleinfo <vehicle info> [-diagaddress <diagnosis address> -basevariant <basevariant>]  
E-Sys.bat -server -swtstatus -project <project name> -vehicleinfo <vehicle info> [-diagaddress <diagnosis address> -basevariant <basevariant>]  
E-Sys.bat -swtstatus -connection <connection config file> [-diagaddress <diagnosis address> -basevariant <basevariant>]  
E-Sys.bat -server -swtstatus -connection <connection config file> [-diagaddress <diagnosis address> -basevariant <basevariant>]

## TAL processing:

E-Sys.bat -talexecution <config file> [-ignoreBATHAF]  
E-Sys.bat -server -talexecution <config file> [-ignoreBATHAF]

## TAL calculation:

E-Sys.bat            -talcalculation <config file>  
E-Sys.bat -server -talcalculation <config file>

## TSL:

E-Sys.bat            -tslstatus -connection <config file>  
E-Sys.bat -server -tslstatus -connection <config file>  
E-Sys.bat            -updatetsl <SVT filename> -connection <config file>  
E-Sys.bat -server -updatetsl <SVT filename> -connection <config file>

## Vehicle Configuration Management (VCM):

E-Sys.bat            -readVcmBackup <FA|ISTUFEN> -connection <config file> -  
out <target directory>  
E-Sys.bat -server -readVcmBackup <FA|ISTUFEN> [-connection <config file>]  
-out <target directory>  
E-Sys.bat            -readVcmMaster <FA|FP|ISTUFEN|SVTSOLL> -connection  
<config file> -out <target directory>  
E-Sys.bat -server -readVcmMaster <FA|FP|ISTUFEN|SVTSOLL> [-connection  
<config file>] -out <target directory>  
E-Sys.bat            -readVinFromBackup -connection <config file>  
E-Sys.bat -server -readVinFromBackup [-connection <config file>]  
E-Sys.bat            -readVinFromMaster -connection <config file>  
E-Sys.bat -server -readVinFromMaster [-connection <config file>]  
E-Sys.bat            -writeVcmBackup <FA|ISTUFEN> -connection <config file> -in  
<source file>  
E-Sys.bat -server -writeVcmBackup <FA|ISTUFEN> [-connection <config file>]  
-in <source file>  
E-Sys.bat            -writeVcmMaster <FA|ISTUFEN|SVTSOLL> -connection <config  
file> -in <source file>  
E-Sys.bat -server -writeVcmMaster <FA|ISTUFEN|SVTSOLL> [-connection  
<config file>] -in <source file>  
E-Sys.bat            -readVcmVersionsinfoMaster -connection <config file>  
E-Sys.bat -server -readVcmVersionsinfoMaster [-connection <config file>]

## Certificate Management:

E-Sys.bat            -activateIPsec -connection <config file>  
E-Sys.bat -server -activateIPsec -connection <config file>  
E-Sys.bat            -deactivateIPsec -connection <config file>  
E-Sys.bat -server -deactivateIPsec -connection <config file>

E-Sys.bat -lockIPsec -connection <config file>  
 E-Sys.bat -server -lockIPsec -connection <config file>  
 E-Sys.bat -readIPsecStatus -connection <config file>  
 E-Sys.bat -server -readIPsecStatus -connection <config file>  
 E-Sys.bat -activateMACsec -connection <config file>  
 E-Sys.bat -server -activateMACsec -connection <config file>  
 E-Sys.bat -certexecution <config file>  
 E-Sys.bat -server -certexecution <config file>  
 E-Sys.bat -checkCERT -connection <config file> -retries <value> [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]  
 E-Sys.bat -server -checkCERT -connection <config file> -retries <value> [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]  
 E-Sys.bat -checkPersistentCert [-full] -connection <config file> -retries <value> [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]  
 E-Sys.bat -server -checkPersistentCert [-full] -connection <config file> -retries <value> [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]  
 E-Sys.bat -deactivateMACsec -connection <config file>  
 E-Sys.bat -server -deactivateMACsec -connection <config file>  
 E-Sys.bat -generateCSR -connection <config file> -out <target file> [-secOckKeys] [-vin <VIN17>] [-blacklist <file>] [-whitelist <file>]  
 E-Sys.bat -server -generateCSR -connection <config file> -out <target file> [-secOckKeys] [-vin <VIN17>] [-blacklist <file>] [-whitelist <file>]  
 E-Sys.bat -getCbbResponseFromRequest <cbb request file> [-out <target directory>]  
 E-Sys.bat -server -getCbbResponseFromRequest <cbb request file> [-out <target directory>]  
 E-Sys.bat -lockMACsec -connection <config file>  
 E-Sys.bat -server -lockMACsec -connection <config file>  
 E-Sys.bat -readCERT -connection <config file> -type <certificate type> [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]  
 E-Sys.bat -server -readCERT -connection <config file> -type <certificate type> [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]  
 E-Sys.bat -setSecOCCounter -connection <config file> -value <hex value> [-blacklist <file>] [-whitelist <file>]  
 E-Sys.bat -server -setSecOCCounter -connection <config file> -value <hex value> [-blacklist <file>] [-whitelist <file>]  
 E-Sys.bat -writeBindings -connection <config file> -in <CBB response file> [-secOckKeys] [-svt <SVT file>] [-secOckKeysPath <SecOckKeyPack file>]  
 E-Sys.bat -server -writeBindings -connection <config file> -in <CBB response file> [-secOckKeys] [-svt <SVT file>] [-secOckKeysPath <SecOckKeyPack file>]  
 E-Sys.bat -addTLSCertificate -certificateFile <certificate file> -keyFile <key file>  
 E-Sys.bat -getTLSCertificates  
 E-Sys.bat -deleteExpiredTLSCertificate

Others:

E-Sys.bat      [-prop <property file>]

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

## General:

The Anflash module offers the possibility to perform a complete flash cycle fully automated. The exact structure of the required configuration files is described in the Anflash manual.

[ ].....optional, can be specified, but does not have to  
( ).....Selection, must be specified  
| ..... "or", is used in a selection

## Batch-command Anflash

### Command in client batch mode:

E-Sys.bat -anflash <config file> [-ignoreBATHAF]

### Command in client-server batch mode:

E-Sys.bat -server -anflash <config file> [-ignoreBATHAF]

### Declaration:

-anflash <config file>	Absolute path to Anflash-Configuration-file (siehe Anflash-Handlungsanweisung).
-ignoreBATHAF	Optional parameter. Ignores the S1-switch.

### Example:

E-Sys.bat -anflash C:\data\executeAnflash.config -ignoreBATHAF

### Information:

- Information about the batch command -anflash is to be obtained from the "Anflash Manual".
- If a BAT/HAF test reveals that switch S1 is open, the

further execution of the command depends on parameter `-ignoreBATHAF`.

`-ignoreBATHAF` has been set:

The flash sequence continues. However, the user must be aware that the execution can lead to interruptions. In this case, the TAL-processing cannot be ended with value 0 (OK), but at best with value 4 (with warning).

`-ignoreBATHAF` not set:

The execution of the command `-talexecution` is canceled.

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 2 - The action was performed with errors.
  - 3 - Unable to connect to the server.
  - 4 - The action was performed without errors but with warnings.
  - 5 - The action was not processed.
  - 6 - The action was aborted by the user.

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.



**General:**

For NCDs, the following commands are offered in the batch.

- [ ].....optional, can be specified, but does not have to be
- ( ).....Selection, must be specified
- | ..... "or", is used in a selection

**Backend authentication for Coding**

**Log in to the Coding-Backend (necessary to sign the NCDs)**

Command in client batch mode:

---only applicable in server-client operation---

Command in client-server batch mode:

E-Sys.bat -server -authenticationCoding -connection <bmw\_intranet|internet> (-user <username>|-useSwlSecCertificate

Declaration:

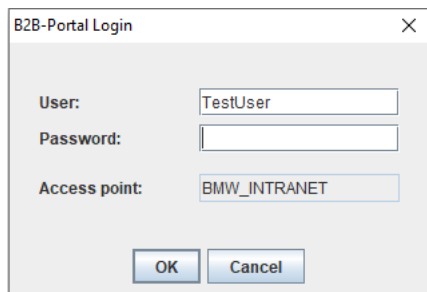
-authenticationCoding	command.
-connection <bmw_intranet internet>	Login via bmw_intranet (inside of BMW-network) or internet (out network).
-user <username>	Backend-Login via username (example: max.mustermann)
-useSwlSecCertificate	Backend-Login via Certificate.
-dialog	Backend-Login with GUI-dialog.

Example:

E-Sys.bat -server -authenticationCoding -user Max.Mustermann -connection intranet  
 E-Sys.bat -server -authenticationCoding -user Max.Mustermann -connection bmw\_intranet -dialog  
 E-Sys.bat -server -authenticationCoding -useSwlSecCertificate -connection bmw\_intranet

Information:

- The -connection parameter determines whether the back-end system, which is addressed for NCDs, should be reached via the BMW intranet or over the Internet.
- Either the parameter -user <username> for Basic Authentication or -useSwlSecCertificate for Certificate authentication can be specified.
- In a basic authentication, the password must be entered manually in the command line of in a GUI dialog.
- If the switch -dialog is specified, a GUI dialog is displayed in which login data must be entered. The Login dialog looks like this:



- For Certificate-based authentication, the required client Certificate must first be imported using the "importSwlSecCertificate" batch command or in GUI mode.
- Whether logging in to the back-end system was successful can be seen in the command line output of the E-Sys server.  
Basic authentication displays both the authorization and the expiration date of the authorization. In the case of successful Certificate-based authentication, the quota and the Certificate's validity are also displayed.
- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 3 - Unable to connect to the server.

### **Importing PKCS-12 Containers with Client Certificate for SWL Security System (SWL)**

This command can be used to import client Certificates that are required to communicate with the SWL (NCD) system.

With a successfully imported client Certificate, Certificate-based authentication to the SWL S is possible.

#### **Command in client batch mode:**

E-Sys.bat -importSwlSecCertificate <P12 file> [-dialog]

#### **Command in client-server batch mode:**

E-Sys.bat -server -importSwlSecCertificate <P12 file> [-dialog]

#### **Declaration:**

-importSwlSecCertificate <P12 file>	Absolute path to PKCS12 container with client Certificate.
-dialog	Backend-Login with GUI-dialog.

#### **Example:**

E-Sys.bat -importSwlSecCertificate C:\CERT\000\_USR\_Max.Mustermann@partner.bmw.de.p12  
E-Sys.bat -importSwlSecCertificate C:\CERT\000\_USR\_Max.Mustermann@partner.bmw.de.p12 -dialog

#### **Information:**

- If the switch -dialog is defined, a GUI dialog is displayed in which authentication data must be entered.
- The command is supported in both client- and server-client-based batch operation.
- The required Certificate for SWL Security System must be obtained separately via the B2B interface.
- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 3 - Unable to connect to the server.

### **Authorization check for NCD signing in the backend**

This command checks whether the user is authorized to sign at the backend for the VINs or for the B2B interface.

#### **Command in client batch mode:**

---only applicable in server-client operation---

### Command in client-server batch mode:

E-Sys.bat -server -backendCheckSigning -vin <vin file> -sgbmno <SGBM file>

### Declaration:

-backendCheckSigning	command.
-vin <vin file>	Absolute path to VIN-file.
-sgbmno <SGBM file>	Absolute path to SGBMNo-file.

### Example:

E-Sys.bat -server -backendCheckSigning -vin C:\NCD\winsToCheck.txt -sgbmno C:\NCD\sgbmnrToCheck.txt

### Information:

- The command is only supported in server-client-mode and only with a BASIC authentication. The user must first login in to the backend using the "-authenticationCoding" command. A '\ ' may not be used in the configuration file. Instead, use '/' or '\\ ' for path information.

- The format of the configuration files to pass is defined as follows:

Example for a VIN-file:

```
# VIN17,VIN17,VIN17
TESTVIN1234567890,VINTEST1234567890,1234567890TEST,TEST1234567890VIN
```

Example for a SGBMNo-file:

```
# SGBMNo,SGBMNo,SGBMNo
00001234,00021000,00004321
```

- The following codes are returned:  
0 - The action was successful.  
1 - An exception occurred so the action could not be completed.  
3 - Unable to connect to the server.

## **Working with NCD**

### **Creating NCD Files from FWL Files**

NCD files are created from a list of FWL-files to be specified as a list in a configuration file. The location for the generated NCD files is also set in this configuration file.

In addition, it contains the specification of a VIN, which is required to calculate the CPS (Code) of the NCD files.

### Command in client batch mode:

E-Sys.bat ~~server~~ -fwl2ncd <config file> **(deprecated)**  
E-Sys.bat -fwl2Ncd <config file>

### Command in client-server batch mode:

E-Sys.bat ~~server~~ -fwl2ncd <config file> **(deprecated)**  
E-Sys.bat -server -fwl2Ncd <config file>

### Declaration:

-fwl2Ncd <config file>	Absolute path to configuration file.
------------------------	--------------------------------------

<config file>...Absolute path to configuration file

### Example:

E-Sys.bat -fwl2Ncd C:\conf\fwl.properties

**Information:**

The <config file> must contain the following entries:

```
# Absolute path of an FA file. E.g. FA = C:/Data/fa.xml
FA = <FA file>

# e.g. NCD_DIR = C:/NCD
NCD_DIR = <NCD destination folder>

# e.g. FWL_LIST = C:/FWL/flw_1.fwl;C:/FWL/flw_2.fwl;
FWL_LIST = <path to fw1>; <path to fw2>; ....
```

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 2 - The action was performed with errors.
  - 3 - Unable to connect to the server.
  - 4 - The action was performed without errors but with warnings.

**Reading of Netto Coding Data**

The Netto Coding Data of the ECUs is read out for all CAFs present in the SVT and stored as I In addition, the VIN is determined and used to determine the file name.

**Command in client batch mode:**

```
E-Sys.bat readncd <SVT> connection <config file> out <NCD directory> (deprecated)
E-Sys.bat -readNcd <SVT file> -connection <config file> -out <NCD directory> [-notReadVin]
```

**Command in client-server batch mode:**

```
E-Sys.bat server readncd <SVT> connection <config file> out <NCD directory> (deprecated)
E-Sys.bat -server -readNcd <SVT file> -connection <config file> -out <NCD directory> [-notReadVin]
```

**Declaration:**

-readNcd <SVT file>	Absolute path to SVT-file.
-connection <config file>	Absolute path to configuration file with the connection param <a href="#">Parameter</a> ).
-out <NCD directory>	Absolute path to output-directory.
-notReadVin	VIN is not read from the Master ECU if used

**Example:**

```
E-Sys.bat -readNcd C:\Data\SVT -connection C:\conf\connection.properties -out C:\Data\CAF
```

**Information:**

- The file name of the FWL file is formed according to the following pattern: "<VIN>\_<CAF>".
  - VIN.....the last 7 digits of the VIN from the VIN-Master
  - CAF.....SGBM-number of the CAF used
  - BV.....Name of the basic variant
- The file name of the NCD-file is formed according to the following pattern: "<SGBMID>.caf\_<SV>\_<PV>.ncd"
  - SGBMID.....SGBM-ID of the CAF used
  - MV.....Main version of the CAF used
  - SV.....Sub version of the CAF used
  - PV.....Patch version of the CAF used

- The following return codes are returned:
  - 0 - The processing was completed successfully.
  - 1 - Errors have occurred.
  - 3 - Unable to connect to the server.

### **Signing of Netto Coding Data**

This command signs a list of NCD-files for the corresponding boot loaders..  
 The NCDs must be complete (incl. CPS). Validation or modification of the NCD-files is no long signature.

#### Command in client batch mode:

```
E-Sys.bat ncdsign <config file> (deprecated)
E-Sys.bat -signNcd <config file>
```

#### Command in client-server batch mode:

```
E-Sys.bat server ncdsign <config file> (deprecated)
E-Sys.bat -server -signNcd <config file>
```

#### Declaration:

-signNcd <config file>	Absolute path to configuration file.
------------------------	--------------------------------------

#### Example:

```
E-Sys.bat -server -signNcd C:\NCD\ncdSign.config
```

#### Information:

- In client-mode, the command is supported only in combination with Certificate-based-auth
- In server-client-mode, the user must first log in to the backend using the "-authenticationCoding
- Required parameters are FA and NCD\_LIST\_1.
- Optional parameters are BACKEND\_SIGNATURE, SHIPMENT\_NCD\_GENERATION, SIGNED\_NCD\_DIR, and SHIPMENT\_NCD\_GENERATION, the information from the E-Sys.properties-file is read or default settings are used.
- Depending on the BACKEND\_SIGNATURE-setting (ALLOW - allow, MUST\_NOT - prohibit, FORCE - enforce) signed in the backend or local.
- The SHIPMENT\_NCD\_GENERATION-setting controls the calculation and signing of NCDs for delivery. NCDs for delivery status are then calculated and signed.
- If multiple NCD lists are defined, the NCD\_LIST\_x-key must be unique.
- If multiple NCD lists are defined, the bootloader-number <BTLD-Nummer-x> must be unique.
- A '\' may not be used in the configuration file. Instead, use '/' or '\\' for path information.

The format of the configuration file to pass is defined as follows:

```
# Absolute path of an FA file. E.g. FA = C:/Data/fa.xml
FA = <FA file>

# Format MUST_NOT | FORCE | ALLOW
BACKEND_SIGNATURE = <MUST_NOT | FORCE | ALLOW>

# Generate shipment NCDs. DEFAULT = on
# Format: on | off
SHIPMENT_NCD_GENERATION = <on|off>
```

```

# z.B. SIGNED_NCD_DIR = C:/NCD/signed
SIGNED_NCD_DIR = <Destination directory for signed NCD>

# z.B. SHIPMENT_NCD_DIR = C:/NCD/shipment
SHIPMENT_NCD_DIR = <Destination directory for signed NCD>

# Format <BTLD-Number>;<NCD-Pfad> , z.B. "00001234;C:/NCD/CAFD_0000DDDD_026_000_002.ncd;C:/NCD/CAFD_0
NCD_LIST_1 = <BTLD-Nummer-1>;<NCD-1>;<NCD-2>;<NCD-3>;<NCD-4>

# Optional parameter
# If specified, only client-batch-mode is considered
# Default value = off
USE_SWL_SEC_CERTIFICATE = <on|off>

# Optional parameter.
# Considered only in client-batch-mode
# If the parameter is owithted SWL_SEC_CONNECTION, the value from Esys.properties is used.
SWL_SEC_CONNECTION = <bmw_intranet|internet>

```

### Example for a configuration file:

```

# Absolute path of an FA file. E.g. FA = C:/Data/fa.xml
FA = <FA file>

# Format MUST_NOT | FORCE | ALLOW
BACKEND_SIGNATURE = ALLOW

# Generate shipment NCDs. DEFAULT = on
# Format: on | off
SHIPMENT_NCD_GENERATION = off

# z.B. SIGNED_NCD_DIR = C:/NCDs/signed
SIGNED_NCD_DIR = C:/NCD/signed

# z.B. SHIPMENT_NCD_DIR = C:/NCD/shipment
SHIPMENT_NCD_DIR = C:/NCD/shipment

# Format <BTLD-Nummer>;<NCD path>
NCD_LIST_1 = 00001ffd;C:/NCD/CAFD_0000AAAA_006_011_017.ncd;C:/NCD/CAFD_0000BBBB_006_011_008.ncd;
NCD_LIST_2 = 00001c44;C:/NCD/CAFD_0000CCCC_026_000_000.ncd;C:/NCD/CAFD_0000DDDD_026_000_002.ncd;

```

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 2 - The action was performed with errors.
  - 3 - Unable to connect to the server.
  - 4 - The action was performed without errors but with warnings.

### **Calculate Netto Coding Data (NCD)**

Similar to the GUI module coding verification, a verification can be performed in batch mode vehicle order (FA) and thus Netto Coding Data (NCD) can be calculated.

During verification, NCD- and FP-traces as well as a verification report are generated and stored in a directory.

Verification is performed on all FAs at the FA-list and all imported CAFs.

The optional parameter [-caf] allows verification to be restricted to individual (imported) CAFs. If multiple CAFs are specified, the names are separated by commas (",").

### **Command in client batch mode:**

```

E-Sys.bat generateNcd <FA file> [-caf <CAF name>] -trace <trace directory> (deprecated)
E-Sys.bat -generateNcd <FA file> [-caf <CAF name>] -trace <trace directory>

```

### Command in client-server batch mode:

E-Sys.bat ~~server~~ generateNcd <FA file> [-caf <CAF name>] -trace <trace directory> **(deprecated)**  
E-Sys.bat -server -generateNcd <FA file> [-caf <CAF name>] -trace <trace directory>

### Declaration:

-generateNcd <FA file>	Absolute path of an FA- or FA-list-file.
-caf <CAF name>	Name of the CAF to be verified.
-trace <trace directory>	Absolute path of the directory to which the data to be gener.

### Example:

E-Sys.bat -generateNcd C:\Data\FA\F020.xml -trace C:\Data\Trace -caf cafd\_0000815.caf.001\_001\_003

### Information:

- No check will be performed to verify that the trace directory is empty. This means that existing directory can be overwritten without warning.
- The following codes are returned:
  - 0 - All NCDs and FPs were generated without errors.
  - 1 - Errors occurred, so not all NCDs or FPs could be generated.
  - 2 - The action was performed with errors.
  - 3 - Unable to connect to the server.
  - 4 - The action was performed without errors but with warnings.

### **Calculate NCD-Testdata**

NCD test data will be created and signed based on a list of CAF files, which have to be specific file.

In addition to the CAFs, a vehicle order (file name + absolute path) must be specified in the It is also mandatory to provide a BTLID SGBM number, as this is required for signing the NCD

Optionally, the storage location for the generated NCD files can also be specified in the conf

### Command in client batch mode:

E-Sys.bat -generateTestNcds <config file>

### Command in client-server batch mode:

E-Sys.bat -server -generateTestNcds <config file>

### Declaration:

-generateTestNcds <configuration file>	Absolute path to configuration file.
--	--------------------------------------

### Example:

E-Sys.bat -generateTestNcds C:\conf\MyConfigForTestdata.cfg

### Information:

- **In client mode, the command is only supported in combination with Certificate-based authentication.**
- **In client-server mode, the user must first log in with a separate command to the**
- If the BACKEND\_SIGNATURE parameter is omitted, the value from *Esys.properties* is used
- If the parameter NCD\_TEST\_DATA\_DIR is omitted, then the E-Sys data directory is read from

and from this a path is constructed to the folder in which the target folder "NCD test data"

- Generation of test NCDs for Coding 2:
- FA file must contain exactly 1 FA
- Generation of test NCDs for Coding 3:
- FA file must contain exactly 2 FAs
- FAs must include Test-VINs (11. char of Long-VIN is 'H')
- FAs must not be identical, i.e. they must differ in at least one of the following points: colour time criteria, SA entries, E entries, K entries

The format of the configuration file is defined as follows:

```
# e.g. FA = C:/FA/fa_I020.xml
FA = <Vehicle order with absolute path>

# e.g. BTLD = 00001234
BTLD = <BTLD SGBM-Number>

# e.g. CAF_LIST = "cafd_0000DDDD.caf.026_000_002;cafd_0000BBBB.caf.006_011_008"
CAF_LIST = <CAF_file_name_1>;<CAF_file_name_2>;<CAF_file_name_3>;..

# Optional parameter
# e.g. NCD_TEST_DATA_DIR = C:/Testdata
NCD_TEST_DATA_DIR = <target folder for the ZIP-file containing the NCD-Testdata>

# Optional parameter
# Format MUST_NOT | FORCE | ALLOW
BACKEND_SIGNATURE = <MUST_NOT | FORCE | ALLOW>

# Optional parameter. Will be only considered in the Client Batch-Mode.
# If the parameter SWL_SEC_CONNECTION is missing, the value from Esys.properties will be used instead.
SWL_SEC_CONNECTION = <bmw_intranet|internet>
```

At the end of the execution a generation report will be created. After the NCD-Calculation placed in parallel to the generated ZIP-Archive.

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 3 - Unable to connect to the server.

### **Netto Coding Data (NCD) for customer vehicles**

Similar to the GUI module 'NCD-Preparation', it is also possible to request NCDs in the SCB b mode.

For this process, a list of vehicle orders (FA\_LIST) is required, which must contain at least one. In addition to the FA\_LIST, either SVT or TAL must be defined.

The generated NCD files are stored in a directory after the command has been executed. This can also be defined in the configuration file if required (SIGNED\_NCD\_DIR).

If the specification of an alternative storage location is missing, then the default directory of Esys.properties is used.

This command can be used only in the BMW environment, because there is a communication Coding Backend (SCB).

Command in client batch mode:



E-Sys.bat -createCustomerNcd <config file>

### Command in client-server batch mode:

E-Sys.bat -server -createCustomerNcd <config file>

### Declaration:

-createCustomerNcd <configuration file>	Absolute path to configuration file.
---	--------------------------------------

### Example:

E-Sys.bat -createCustomerNcd C:\conf\configForSCB.cfg

### Information:

- The calculation and signing of the NCDs will be done in the SCB backend
- As a necessary precondition for the access to the SCB backend, a TIS-client Certificate has to be installed on the client
- The URLs used for connecting to the SCB have to be defined in the esys.properties file by the key scb.backend.url
- Required parameters are FA\_LIST und either SVT or TAL.
- The configuration file can contain either an SVT or a TAL, but not both.
- Optional parameter is SIGNED\_NCD\_DIR. If omitted, the information from the E-Sys.properties-file settings are used.

The format of the configuration file is defined as follows:

```
# e.g. FA_LIST = C:/Data/FA/fa_1.xml;C:/Data/FA/fa_2.xml;  
FA_LIST = <Absolute path FA1>;<Absolute path FA2>;<Absolute path FA3>
```

```
# e.g. SVT = C:/Data/SVT/svt.xml  
SVT = <Absolute path SVT>
```

```
# e.g. TAL = C:/Data/SVT/tal.xml  
TAL = <Absolute path TAL>
```

```
# Optional parameter.  
# e.g. SIGNED_NCD_DIR = C:/Data/NCD  
SIGNED_NCD_DIR = <target folder for the NCD data>
```

- The following codes are returned:
  - 0 - All NCDs were generated without errors.
  - 1 - Errors occurred, so not all NCDs could be generated.
  - 3 - Unable to connect to the server.

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software reserved. This message will not appear if you compile this help file with the registered version.

**General:**

CVNs (Calibration Verification Number) are calculated via the OBD-relevant (On-Board Diagnostics) SWEn for a control unit and compared with the CVNs that are read from the ECUs. The ECUs and SWEs required for the calculation are read from an SVT document.

The CVN values and the comparison results are written to a report file as text or in XML format (-xmlreport).

The SVT document can be read from the ECU, from the VCM, or from a file to be specified.

[ ].....optional, can be specified, but does not have to be

( ).....Selection, must be specified

| ..... "or", is used in a selection

**Calculate CVN**

The following command can be used to calculate CVN values.

**Command in client batch mode:**

```
E-Sys.bat      -cvncalculation <FA file> -out <OBD-CVN-FA-List file> [-caf
<CAF name>]
```

**Command in client-server batch mode:**

```
E-Sys.bat -server -cvncalculation <FA file> -out <OBD-CVN-FA-List file> [-caf
<CAF name>]
```

**Declaration:**

-cvncalculation <FA file>	Absolute path of an FA- or FA-list-file.
-out <OBD-CVN-FA-List file>	Absolute path of the output-file.
-caf <CAF name>	Names of the CAFs to use.

## Example:

```
E-Sys.bat -cvncalculation  
C:\Data_3_39_1_09f71b0_64bit_JDK11\FA\FaListSample.xml -out  
C:\Data_3_39_1_09f71b0_64bit_JDK11\Etc\CVN.xml -caf  
cafd_000000ae.caf.001_002_003
```

## Information

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 3 - Unable to connect to the server.

## **Compare CVN**

### Command in client batch mode:

```
E-Sys.bat -comparecvn -connection <config file> [-sg <ecu list>] (-svt  
<SVT file>|-svtvcmist|-svtvcmsoll|-svtecu) -trace <trace directory> [-fa <FA  
file>] [-xmlreport]
```

### Command in client-server batch mode:

```
E-Sys.bat -server -comparecvn -connection <config file> [-sg <ecu list>] (-  
svt <SVT file>|-svtvcmist|-svtvcmsoll|-svtecu) -trace <trace directory> [-fa  
<FA file>] [-xmlreport]
```

## Declaration:

-comparecvn	command.
-connection <config file>	Absolute path to configuration file with the connection parameter (see <a href="#">Connection Parameter</a> ).
-sg <ecu list>	ECU-filters. If the parameter is specified, only ECUs from the list are processed.

	The ECUs are separated by commas in the list.
-svt <SVT file>	Absolute path to SVT-file.
-svtvcmist	The SVTist is read from the VCM.
-svtvcmsoll	The SVTsoll is read from the VCM.
-svtecu	The SVT is read from the ECU.
-trace <trace directory>	Absolute path to the directory for the report-file.
-fa <FA file>	Absolute path to the FA-List-file. If the parameter "-fa" is not specified, the FA is read out of the vehicle.
-xmlreport	Erstellen einer Report-file im XML-Format.

### Example:

```
E-Sys.bat -comparecvn -trace C:\Data\trace -svtecu -fa C:\Data\FA\FaList.xml -sg
ZBE,ZGW -connection C:\conf\connection.properties
```

### Information

- The following codes are returned:
  - 0 - The comparison was completed successfully.
  - 1 - The comparison could not be performed due to errors. The errors were written to the report file.
  - 3 - Unable to connect to the server.

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

### General:

[ ].....optional, can be specified, but does not have to be  
( ).....Selection, must be specified  
| ..... "or", is used in a selection

### read out the E-Sys-Version

With this command, the E-Sys-Version will be read out.

#### Command in client batch mode:

E-Sys.bat -version

#### Command in client-server batch mode:

E-Sys.bat -server -version

#### Declaration:

-version	command.
----------	----------

#### Example:

E-Sys.bat -version

E-Sys.bat -server -version

#### Information:

- The E-Sys-Version is output to stdout.

### read BRV:

From the development-series, the name of the series-network is determined and output on the console.

#### Command in client batch mode:

E-Sys.bat -getbrv <series>

#### Command in client-server batch mode:

E-Sys.bat -server -getbrv <series>

### Declaration:

-getbrv <series>	Name of the development Series / Derivative / Serie.
------------------	--

### Example:

E-Sys.bat -getbrv F003  
E-Sys.bat -getbrv F44  
E-Sys.bat -getbrv RR25

### Information:

- The following codes are returned: 0 - The development series has been correctly determined.  
1 - Errors have occurred.  
3 - Unable to connect to the server.

### **read FA:**

The vehicle-order is read out of the vehicle and stored in a file. If only one directory is specified, the VIN is also determined and used to determine the file-name. Otherwise, the specified file will be used. If it already exists, it will be overwritten.

### Command in client batch mode:

E-Sys.bat -readfa -connection <config file> -out <FA directory>

### Command in client-server batch mode:

E-Sys.bat -server -readfa -connection <config file> -out <FA directory>

### Declaration:

-readfa	command.
-connection <config file>	Absolute path to configuration file with the connection parameter (see <a href="#">Connection Parameter</a> ).
-out <FA directory>	Absolute path for the output directory with or without file-specification.

### Example:

E-Sys.bat -readfa -out C:\Data\FA -connection C:\conf\connection.properties  
 E-Sys.bat -readfa -out C:\Data\FA\FA\_123.xml -connection C:\conf\connection.properties

**Information:**

- If no file-name is specified, it is determined after the pattern "FA\_<VIN>\_<Timestamp>.xml".  
 VIN..... the last 7 digits of the VIN from the VIN-Master  
 Timestamp..... Timestamp according to the pattern ddmmyyyhhmm
- The following codes are returned:  
 0 - The processing was completed successfully.  
 1 - Errors have occurred.  
 3 - Unable to connect to the server.

**read SVT:**

The SVT is read by the vehicle and stored in a file.  
 If only one directory is specified, the VIN is also determined and used to determine the file-name.  
 Otherwise, the specified file will be used. If it already exists, it will be overwritten.

**Command in client batch mode:**

E-Sys.bat -readsvt -connection <config file> -out <SVT directory>

**Command in client-server batch mode:**

E-Sys.bat -server -readsvt -connection <config file> -out <SVT directory>

**Declaration:**

-readsvt	command.
-connection <config file>	Absolute path to configuration file with the connection parameter (see <a href="#">Connection Parameter</a> ).
-out <SVT directory>	Absolute path for the output directory with or without file specification.

**Example:**

E-Sys.bat -readsvt -out C:\Data\SVT -connection C:\conf\connection.properties  
 E-Sys.bat -readsvt -out C:\Data\SVT\SVT\_123.xml -connection C:\conf\connection.properties

Information:

- If no file name is specified, it is determined after the pattern "SVT\_<VIN>\_<Timestamp>.xml".  
Vin..... the last 7 digits of the VIN from the VIN-Master  
Timestamp..... Time stamp according to the pattern yyymmddhhmm
  
- The following codes are returned:
  - 0 - The processing was completed successfully.
  - 1 - Errors have occurred.
  - 3 - Unable to connect to the server.

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.



# ECU-Modes / SFA (Secure Feature Activation) / LCS (Locking Configuration Switch)

[Top](#)  
[Previous](#)  
[Next](#)

## General:

For Secure ECU-Modes / Secure Feature Activation, the following commands are offered in Batch.

[ ].....optional, can be specified, but does not have to be  
( ).....Selection, must be specified  
| ..... "or", is used in a selection

## BASIC Backend authentication for SFA (CERTIFICATE+BASICS)

### Set optional BASIC credentials for SFA (extended rights)

#### Command in client batch mode:

---only applicable in server-client operation---

#### Command in client-server batch mode:

E-Sys.bat -server -authenticationSfa -user <username> [-dialog]

#### Declaration:

-authenticationSfa	command.
-user <username>	Backend-Login via username (example: max.mustermann).
-dialog	Backend-Login with GUI-dialog.

#### Example:

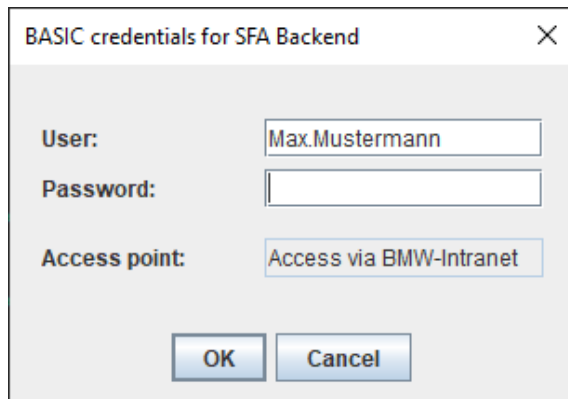
E-Sys.bat -server -authenticationSfa -user Max.Mustermann

E-Sys.bat -server -authenticationSfa -user Max.Mustermann -dialog

#### Information:

- **The client certificate that is still required must have already been imported.**

- **The validity of the specified credentials is not checked. This is only done with the respective backend request.**
- In a basic authentication, the password must be entered manually in the command line of the **E-Sys server** or in a GUI dialog.
- If the switch `-dialog` is specified, a GUI dialog is displayed in which login data must be entered. The Login dialog looks like this:



- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 2 - The action was performed with errors.
  - 3 - Unable to connect to the server.

## **Remove optional BASIC credentials for SFA**

### Command in client batch mode:

---only applicable in server-client operation---

### Command in client-server batch mode:

E-Sys.bat `-server -removeAuthenticationSfa`

### Declaration:

<code>-authenticationSfa</code>	command.
---------------------------------	----------

## Example:

E-Sys.bat -server -removeAuthenticationSfa

## Information:

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 3 - Unable to connect to the server.

## **Switch ECU-Modes / write tokens (with backend-connection)**

### **Switching to plant- / engineering-mode**

#### Command in client batch mode:

E-Sys.bat -switchEcuModeOnline -connection <config file> (-plant|-engineering) [-vin <VIN17>] [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]

#### Command in client-server batch mode:

E-Sys.bat -server -switchEcuModeOnline -connection <config file> (-plant|-engineering) [-vin <VIN17>] [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]

## Declaration:

-switchEcuModeOnline	command.
-connection <config file>	Absolute path to configuration file with the connection parameter (see <a href="#">Connection Parameter</a> ).
-plant	Argument.
-engineering	Argument.
-vin <VIN17>	17-digit VIN.
-svt <SVT file>	Absolute path to The SVT-file.
-blacklist <file>	Absolute path to Blacklist-file.
-whitelist <file>	Absolute path to Whitelist-file.

## Example:

```
E-Sys.bat -switchEcuModeOnline -connection C:\conf\connection.properties -plant -vin  
WBA11CF070H123456 -blacklist C:\Daten\blacklist.txt  
E-Sys.bat -switchEcuModeOnline -connection C:\conf\connection.properties -  
engineering -svt C:\SVT\SVT_I20.xml -whitelist C:\Daten\whitelist.txt
```

## Information:

- The accessibility of the BMW-backend is assumed.
- In order to be able to use the required services of the BMW-backend, a corresponding configuration of the client and server including authentication in the [options](#) is required.
- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 3 - Unable to connect to the server.
  - 4 - The action was performed without errors but with warnings.

## **Write the latest secure-tokens**

### Command in client batch mode:

```
E-Sys.bat -writeNewestPackageOnline -connection <config file> [-  
disableDelete] [-vin <VIN17>] [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]  
[-rebuild]
```

### Command in client-server batch mode:

```
E-Sys.bat -server -writeNewestPackageOnline -connection <config file> [-  
disableDelete] [-vin <VIN17>] [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]  
[-rebuild]
```

## Declaration:

-writeNewestPackageOnline	command.
-connection <config file>	Absolute path to configuration file with the connection parameter (see <a href="#">Connection Parameter</a> ).

-disableDelete	Do not delete Secure Features.
-vin <VIN17>	17-digit VIN.
-svt <SVT file>	Absolute path to SVT-file.
-blacklist <file>	Absolute path to Blacklist-file.
-whitelist <file>	Absolute path to Whitelist-file.
-rebuild	Recalculation of tokens.

### Example:

E-Sys.bat -writeNewestPackageOnline -connection C:\conf\connection.properties -disableDelete -vin WBA11CF070H123456 -svt -svt C:\SVT\SVT\_I20.xml -blacklist C:\Daten\blacklist.txt

E-Sys.bat -writeNewestPackageOnline -connection C:\conf\connection.properties -vin WBA11CF070H123456 -svt <SVT file> -whitelist C:\Daten\whitelist.txt -rebuild

### Information:

- The accessibility of the BMW-backend is assumed.
- In order to be able to use the required services of the BMW-backend, a corresponding configuration of the client and server including authentication in the [options](#) is required.
- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 2 - The action was performed with errors.
  - 3 - Unable to connect to the server.
  - 4 - The action was performed without errors but with warnings.

## **Write secure-tokens for factory-state**

### Command in client batch mode:

E-Sys.bat -writePackageForOrderOnline -connection <config file> [-disableDelete] [-vin <VIN17>] [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>] [-rebuild]

### Command in client-server batch mode:

E-Sys.bat -server -writePackageForOrderOnline -connection <config file> [-disableDelete] [-vin <VIN17>] [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>] [-rebuild]

### Declaration:

-writePackageForOrderOnline	command.
-connection <config file>	Absolute path to configuration file with the connection parameter (see <a href="#">Connection Parameter</a> ).
-disableDelete	Do not delete Secure Features.
-vin <VIN17>	17-digit VIN.
-svt <SVT file>	Absolute path to SVT-file.
-blacklist <file>	Absolute path to Blacklist-file.
-whitelist <file>	Absolute path to Whitelist-file.
-rebuild	Recalculation of tokens.

### Example:

E-Sys.bat -writePackageForOrderOnline -connection C:\conf\connection.properties -disableDelete -vin WBA11CF070H123456 -svt -svt C:\SVT\SVT\_I20.xml -blacklist C:\Daten\blacklist.txt

E-Sys.bat -writePackageForOrderOnline -connection C:\conf\connection.properties -vin WBA11CF070H123456 -svt <SVT file> -whitelist C:\Daten\whitelist.txt -rebuild

### Information:

- The accessibility of the BMW-backend is assumed.
- In order to be able to use the required services of the BMW-backend, a corresponding configuration of the client and server including authentication in the [options](#) is required.
- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 2 - The action was performed with errors.
  - 3 - Unable to connect to the server.
  - 4 - The action was performed without errors but with warnings.

## Create request-file (without backend-connection)

### Create request-file for switching to plant- / engineering-mode

#### Command in client batch mode:

E-Sys.bat -createRequestEcuMode -connection <config file> (-plant|-engineering) [-vin <VIN17>] [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]

#### Command in client-server batch mode:

E-Sys.bat -server -createRequestEcuMode -connection <config file> (-plant|-engineering) [-vin <VIN17>] [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]

#### Declaration:

-createRequestEcuMode	command.
-connection <config file>	Absolute path to configuration file with the connection parameter (see <a href="#">Connection Parameter</a> ).
-plant	Argument.
-engineering	Argument.
-vin <VIN17>	17-digit VIN.
-svt <SVT file>	Absolute path to SVT-file.
-blacklist <file>	Absolute path to Blacklist-file.
-whitelist <file>	Absolute path to Whitelist-file.

#### Example:

E-Sys.bat -createRequestEcuMode -connection C:\conf\connection.properties -plant -vin WBA11CF070H123456 -blacklist C:\Daten\blacklist.txt  
E-Sys.bat -createRequestEcuMode -connection C:\conf\connection.properties -engineering -svt C:\SVT\SVT\_I20.xml -whitelist C:\Daten\whitelist.txt

#### Information:

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 2 - The action was performed with errors.

- 3 - Unable to connect to the server.
- 4 - The action was performed without errors but with warnings.

**Create a request-file for the latest secure-tokens**

Command in client batch mode:

E-Sys.bat -createRequestNewestPackage -vin <VIN17> [-whitelist <file>] [-rebuild]

Command in client-server batch mode:

E-Sys.bat -server -createRequestNewestPackage -vin <VIN17> [-whitelist <file>] [-rebuild]

Declaration:

-createRequestNewestPackage	command.
-vin <VIN17>	17-digit VIN.
-whitelist <file>	Absolute path to Whitelist-file.
-rebuild	Recalculation of tokens.

Example:

E-Sys.bat -createRequestNewestPackage -vin WBA11CF070H123456 -blacklist C:\Daten\blacklist.txt  
 E-Sys.bat -createRequestNewestPackage -vin WBA11CF070H123456 -whitelist C:\Daten\whitelist.txt -rebuild

Information:

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 2 - The action was performed with errors.
  - 3 - Unable to connect to the server.

**Create a request-file for factory-state secure-tokens**



### Command in client batch mode:

E-Sys.bat            -createRequestPackageForOrder -vin <VIN17> [-rebuild]

### Command in client-server batch mode:

E-Sys.bat -server -createRequestPackageForOrder -vin <VIN17> [-rebuild]

### Declaration:

-createRequestPackageForOrder	command.
-vin <VIN17>	17-digit VIN.
-rebuild	Recalculation of tokens.

### Example:

E-Sys.bat -createRequestPackageForOrder -vin WBA11CF070H123456 -rebuild

### Information:

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 2 - The action was performed with errors.
  - 3 - Unable to connect to the server.

## **Request-token-set-file from a request-file in the BMW backend**

### **Request-token-set-file for ECU-Mode or SFA from a request-file in the BMW backend**

### Command in client batch mode:

E-Sys.bat            -getResponseFromRequest [-out <response file>] -request <request file> -svt <SVT file>

### Command in client-server batch mode:

E-Sys.bat -server -getResponseFromRequest [-out <response file>] -request <request file> -svt <SVT file>

### Declaration:

-getResponseFromRequest	command.
-out <response file>	Absolute path for the output-directory with file-specification (response-file [JSON]).
-request <request file>	Absolute path to request-file [JSON].
-svt <SVT file>	Absolute path to SVT-file.

### Example:

```
E-Sys.bat -getResponseFromRequest -request  
C:\Daten\WBA11CF070H123456_Plant_Mode_Request_20200528_131607[JSON].txt -  
svt C:\SVT\SVT_I20.xml  
E-Sys.bat -getResponseFromRequest -out <response file> -request  
C:\Daten\WBA11CF070H123456_Plant_Mode_Request_20200528_131607[JSON].txt -  
svt C:\SVT\SVT_I20.xml
```

### Information:

- The accessibility of the BMW-backend is assumed.
- In order to be able to use the required services of the BMW-backend, a corresponding configuration of the client and server including authentication in the [options](#) is required.
- All four request-file-types can be handled (EcuMode, PackageForOrder, NewestPackageForVehicle, NewestPackageForEcu).
- NewestPackage-request-files must contain "Newest" or they will be interpreted as PackageForOrder-request.
  
- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 2 - The action was performed with errors.
  - 3 - Unable to connect to the server.

# Write tokens from existing token-sets

## Write ECU-mode-token from existing token-set

### Command in client batch mode:

E-Sys.bat -switchEcuMode -connection <config file> -tokenset <SecureToken file> [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]

### Command in client-server batch mode:

E-Sys.bat -server -switchEcuMode -connection <config file> -tokenset <SecureToken file> [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]

### Declaration:

-switchEcuMode	command.
-connection <config file>	Absolute path to configuration file with the connection parameter (see <a href="#">Connection Parameter</a> ).
-tokenset <SecureToken file>	Absolute path to ECU-mode-token-set (response-file [JSON]).
-svt <SVT file>	Absolute path to SVT-file.
-blacklist <file>	Absolute path to Blacklist-file.
-whitelist <file>	Absolute path to Whitelist-file.

### Example:

E-Sys.bat -switchEcuMode -connection C:\conf\connection.properties -tokenset C:\SecureTokenRequests\SecureTokenResponse\_ADCAM2\_ACSM6\_Engineering\_Mode.txt -blacklist C:\Daten\blacklist.txt

E-Sys.bat -switchEcuMode -connection C:\conf\connection.properties -tokenset C:\SecureTokenRequests\SecureTokenResponse\_ADCAM2\_ACSM6\_Engineering\_Mode.txt -svt C:\SVT\SVT\_I20.xml -whitelist C:\Daten\whitelist.txt

### Information:

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 3 - Unable to connect to the server.
  - 4 - The action was performed without errors but with warnings.

## Write secure-tokens from existing token-set

### Command in client batch mode:

```
E-Sys.bat -writeSecureTokens -connection <config file> -tokenset  
<SecureToken file> [-svt <SVT file>] [-disableDelete] [-blacklist <file>] [-whitelist  
<file>]
```

### Command in client-server batch mode:

```
E-Sys.bat -server -writeSecureTokens -connection <config file> -tokenset  
<SecureToken file> [-svt <SVT file>] [-disableDelete] [-blacklist <file>] [-whitelist  
<file>]
```

### Declaration:

-writeSecureTokens	command.
-connection <config file>	Absolute path to configuration file with the connection parameter (see <a href="#">Connection Parameter</a> ).
-tokenset <SecureToken file>	Absolute path to secure-token-set (response-file [JSON]).
-svt <SVT file>	Absolute path to SVT-file.
-disableDelete	Do not delete secure-features.
-blacklist <file>	Absolute path to Blacklist-file.
-whitelist <file>	Absolute path to Whitelist-file.

### Example:

```
E-Sys.bat -writeSecureTokens -connection C:\conf\connection.properties -tokenset  
C:\Daten\SecureTokenResponse_ADCAM2_ACSM6_Engineering_Mode.txt -svt  
C:\SVT\SVT_I20.xml -disableDelete  
E-Sys.bat -writeSecureTokens -connection C:\conf\connection.properties -tokenset  
C:\Daten\SecureTokenResponse_ADCAM2_ACSM6_Engineering_Mode.txt -blacklist  
C:\Daten\blacklist.txt  
E-Sys.bat -writeSecureTokens -connection C:\conf\connection.properties -tokenset  
C:\Daten\SecureTokenResponse_ADCAM2_ACSM6_Engineering_Mode.txt -whitelist  
C:\Daten\whitelist.txt
```

### Information:

- The following codes are returned:

- 0 - The action was successful.
- 1 - An exception occurred so the action could not be completed.
- 2 - The action was performed with errors.
- 3 - Unable to connect to the server.
- 4 - The action was performed without errors but with warnings.

## Switch ECUs to field-mode

### Switch ECUs to field-mode

#### Command in client batch mode:

E-Sys.bat -switchEcuModeField -connection <config file> [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]

#### Command in client-server batch mode:

E-Sys.bat -server -switchEcuModeField -connection <config file> [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]

#### Declaration:

-switchEcuModeField	command.
-connection <config file>	Absolute path to configuration file with the connection parameter (see <a href="#">Connection Parameter</a> ).
-svt <SVT file>	Absolute path to SVT-file.
-blacklist <file>	Absolute path to Blacklist-file.
-whitelist <file>	Absolute path to Whitelist-file.

#### Example:

E-Sys.bat -switchEcuModeField -connection C:\conf\connection.properties -svt C:\SVT\SVT\_I20.xml -blacklist C:\Daten\blacklist.txt  
 E-Sys.bat -switchEcuModeField -connection C:\conf\connection.properties -whitelist C:\Daten\whitelist.txt

#### Information:

- The following codes are returned:

- 0 - The action was successful.
- 1 - An exception occurred so the action could not be completed.
- 3 - Unable to connect to the server.
- 4 - The action was performed without errors but with warnings.

## Clean / delete secure-tokens from ECUs

### Clean up secure-tokens from ECUs (CLEAR\_FEATURE --> secure-tokens can be reused)

#### Command in client batch mode:

E-Sys.bat        -clearSecureToken -connection <config file> -featureid <FeatureID>  
 (-diagaddress <diagnosis address>|-svt <SVT file>)

#### Command in client-server batch mode:

E-Sys.bat -server -clearSecureToken -connection <config file> -featureid  
 <FeatureID> (-diagaddress <diagnosis address>|-svt <SVT file>)

#### Declaration:

-clearSecureToken	command.
-connection <config file>	Absolute path to configuration file with the connection parameter (see <a href="#">Connection Parameter</a> ).
-featureid <FeatureID>	FeatureID in Hex, e.g. 0x23.
-diagaddress <diagnosis address>	Diagnostic address of the ECU in hex.
-svt <SVT file>	Absolute path to SVT-file.

#### Example:

---

#### Information:

- All ECUs must be in engineering mode.

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 2 - The action was performed with errors.
  - 3 - Unable to connect to the server.
  - 4 - The action was performed without errors but with warnings.

### **Delete secure-tokens from ECUs (DELETE --> secure-tokens cannot be reused)**

#### Command in client batch mode:

```
E-Sys.bat      -deleteSecureToken -connection <config file> -featureid <FeatureID>
               -diagaddress <diagnosis address>
```

#### Command in client-server batch mode:

```
E-Sys.bat -server -deleteSecureToken -connection <config file> -featureid
<FeatureID> -diagaddress <diagnosis address>
```

#### Declaration:

-deleteSecureToken	command.
-connection <config file>	Absolute path to configuration file with the connection parameter (see <a href="#">Connection Parameter</a> ).
-featureid <FeatureID>	FeatureID in Hex, e.g. 0xa23456.
-diagaddress <diagnosis address>	Diagnostic address of the ECU in hex.

#### Example:

```
E-Sys.bat -deleteSecureToken -connection C:\conf\connection.properties -featureid
0xa23456 -diagaddress 0x5d
```

#### Information:

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.

- 2 - The action was performed with errors.
- 3 - Unable to connect to the server.
- 4 - The action was performed without errors but with warnings.

## Query SFA status

### Status query of all ECUs per FeatureID

#### Command in client batch mode:

E-Sys.bat        -discoverAllFeatureStatus -connection <config file> [-svt <SVT file>]

#### Command in client-server batch mode:

E-Sys.bat -server -discoverAllFeatureStatus -connection <config file> [-svt <SVT file>]

#### Declaration:

-discoverAllFeatureStatus	command.
-connection <config file>	Absolute path to configuration file with the connection parameter (see <a href="#">Connection Parameter</a> ).
-svt <SVT file>	Absolute path to SVT-file.

#### Example:

E-Sys.bat -discoverAllFeatureStatus -connection C:\conf\connection.properties  
E-Sys.bat -discoverAllFeatureStatus -connection C:\conf\connection.properties -svt C:\SVT\SVT\_I20.xml

#### Information:

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 2 - The action was performed with errors.
  - 3 - Unable to connect to the server.
  - 4 - The action was performed without errors but with warnings.



## **Status query of a featureID of a ECU**

### Command in client batch mode:

E-Sys.bat            -readShortStatus -connection <config file> -featureid <FeatureID> -diagaddress <diagnosis address>

### Command in client-server batch mode:

E-Sys.bat -server -readShortStatus -connection <config file> -featureid <FeatureID> -diagaddress <diagnosis address>

### Declaration:

-readShortStatus	command.
-connection <config file>	Absolute path to configuration file with the connection parameter (see <a href="#">Connection Parameter</a> ).
-featureid <FeatureID>	FeatureID in Hex, e.g. 0xa23456.
-diagaddress <diagnosis address>	Diagnostic address of the ECU in hex.

### Example:

E-Sys.bat -readShortStatus -connection C:\conf\connection.properties -featureid 0xa23456 -diagaddress 0x5d

### Information:

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 2 - The action was performed with errors.
  - 3 - Unable to connect to the server.
  - 4 - The action was performed without errors but with warnings.

## **Check the validity of the secure-tokens (for a list of ECUs)**

### Command in client batch mode:

E-Sys.bat -verifySecureTokens -connection <config file> [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]

### Command in client-server batch mode:

E-Sys.bat -server -verifySecureTokens -connection <config file> [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]

### Declaration:

-verifySecureTokens	command.
-connection <config file>	Absolute path to configuration file with the connection parameter (see <a href="#">Connection Parameter</a> ).
-svt <SVT file>	Absolute path to SVT-file.
-blacklist <file>	Absolute path to Blacklist-file.
-whitelist <file>	Absolute path to Whitelist-file.

### Example:

E-Sys.bat -verifySecureTokens -connection C:\conf\connection.properties -blacklist C:\Daten\blacklist.txt

E-Sys.bat -verifySecureTokens -connection C:\conf\connection.properties -svt C:\SVT\SVT\_I20.xml -whitelist C:\Daten\whitelist.txt

### Information:

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 2 - The action was performed with errors.
  - 3 - Unable to connect to the server.
  - 4 - The action was performed without errors but with warnings.

## **Check the validity of the secure-tokens (for all ECUs of a vehicle)**

### Command in client batch mode:

E-Sys.bat -verifySecureTokensVehicle -connection <config file> [-svt <SVT file>]

### Command in client-server batch mode:

E-Sys.bat -server -verifySecureTokensVehicle -connection <config file> [-svt <SVT file>]

### Declaration:

-verifySecureTokensVehicle	command.
-connection <config file>	Absolute path to configuration file with the connection parameter (see <a href="#">Connection Parameter</a> ).
-svt <SVT file>	Absolute path to SVT-file.

### Example:

E-Sys.bat -verifySecureTokensVehicle -connection C:\conf\connection.properties  
E-Sys.bat -verifySecureTokensVehicle -connection C:\conf\connection.properties -svt C:\SVT\SVT\_I20.xml

### Information:

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 2 - The action was performed with errors.
  - 3 - Unable to connect to the server.
  - 4 - The action was performed without errors but with warnings.

## **Read the result of the last token import**

### Command in client batch mode:

E-Sys.bat -readSecureTokenImportStatus -connection <config file> [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]

### Command in client-server batch mode:

E-Sys.bat -server -readSecureTokenImportStatus -connection <config file> [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]

-

### Declaration:

-readSecureTokenImportStatus	command.
-connection <config file>	Absolute path to configuration file with the connection parameter (see <a href="#">Connection Parameter</a> ).
-svt <SVT file>	Absolute path to SVT-file.
-blacklist <file>	Absolute path to Blacklist-file.
-whitelist <file>	Absolute path to Whitelist-file.

### Example:

E-Sys.bat -readSecureTokenImportStatus -connection C:\conf\connection.properties -blacklist C:\Data\blacklist.txt

E-Sys.bat -readSecureTokenImportStatus -connection C:\conf\connection.properties -svt C:\Data\svt.xml -whitelist C:\Data\whitelist.txt

### Information:

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 2 - The action was performed with errors.
  - 3 - Unable to connect to the server.
  - 4 - The action was performed without errors but with warnings.

## **Read SFA-software-version from ECU**

### Command in client batch mode:

E-Sys.bat -readSoftwareVersion -connection <config file> -diagaddress <diagnosis address>

### Command in client-server batch mode:

E-Sys.bat -server -readSoftwareVersion -connection <config file> -diagaddress <diagnosis address>

### Declaration:

-readSoftwareVersion	command.
-connection <config file>	Absolute path to configuration file with

	the connection parameter (see <a href="#">Connection Parameter</a> ).
-diagaddress <diagnosis address>	Diagnostic address of the ECU in hex.

**Example:**

E-Sys.bat -readSoftwareVersion -connection C:\conf\connection.properties -diagaddress 0x5d

**Information:**

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 2 - The action was performed with errors.
  - 3 - Unable to connect to the server.
  - 4 - The action was performed without errors but with warnings.

## Read or switch over Locking Configuration Switch (LCS)

### Read LCS values

Command in client batch mode:

E-Sys.bat -readLcs -connection <config file> [-lcsName <LCS name>] [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]

Command in client-server batch mode:

E-Sys.bat -server -readLcs -connection <config file> [-lcsName <LCS name>] [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]

**Declaration:**

-readLcs	command.
-connection <config file>	Absolute path to configuration file with the connection parameter (see <a href="#">Connection Parameter</a> ).

-lcsName <LCS name>	LCS-function in hex (e.g.: 0x01).
-svt <SVT file>	Absolute path to SVT-file.
-blacklist <file>	Absolute path to Blacklist-file.
-whitelist <file>	Absolute path to Whitelist-file.

**Example:**

```
E-Sys.bat -readLcs -connection C:\conf\connection.properties -whitelist
C:\Daten\whitelist.txt
E-Sys.bat -readLcs -connection C:\conf\connection.properties -lcsName 0x01 -svt
C:\SVT\SVT_I20.xml
E-Sys.bat -readLcs -connection C:\conf\connection.properties -lcsName 0x01 -svt
C:\SVT\SVT_I20.xml -blacklist C:\Daten\blacklist.txt
```

**Information:**

[see LCS section](#)

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 2 - The action was performed with errors.
  - 3 - Unable to connect to the server.

**Set a new LCS value via diagnostic job**

**Command in client batch mode:**

```
E-Sys.bat -setLcs -connection <config file> -lcsName <LCS name> -lcsValue
<LCS value> [-noreset] [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]
```

**Command in client-server batch mode:**

```
E-Sys.bat -server -setLcs -connection <config file> -lcsName <LCS name> -lcsValue
<LCS value> [-noreset] [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]
```

**Declaration:**

-setLcs	command.
-connection <config file>	Absolute path to configuration file with the connection parameter (see <a href="#">Connection Parameter</a> ).

-lcsName <LCS name>	LCS-function in hex (e.g.: 0x01).
-lcsValue <LCS value>	LCS-function-value in hex (e.g.: 0x01).
-noreset	Do not perform an ECU reset after switching an LCS.
-svt <SVT file>	Absolute path to SVT-file.
-blacklist <file>	Absolute path to Blacklist-file.
-whitelist <file>	Absolute path to Whitelist-file.

### Example:

```
E-Sys.bat -setLcs -connection C:\conf\connection.properties -lcsName 0x01 -lcsValue 0x01 -noreset -svt C:\SVT\SVT_I20.xml -blacklist C:\Daten\blacklist.txt
E-Sys.bat -setLcs -connection C:\conf\connection.properties -lcsName 0x01 -lcsValue 0x01 -noreset -whitelist C:\Daten\whitelist.txt
E-Sys.bat -setLcs -connection C:\conf\connection.properties -lcsName 0x01 -lcsValue 0x01 -svt C:\SVT\SVT_I20.xml
```

### Information:

[see LCS section](#)

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 2 - The action was performed with errors.
  - 3 - Unable to connect to the server.

## **Set a new LCS value via secure token**

### Command in client batch mode:

```
E-Sys.bat -writeLcsToken -connection <config file> -tokenset <SecureToken file> [-noreset] [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]
```

### Command in client-server batch mode:

```
E-Sys.bat -server -writeLcsToken -connection <config file> -tokenset <SecureToken file> [-noreset] [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]
```

### Declaration:

-writeLcsToken	Command.
----------------	----------

-connection <config file>	Absolute path to configuration file with the connection parameter (see <a href="#">Connection Parameter.</a> )
-tokenset <SecureToken file>	Absolute path to secure-token-set (response-file [JSON]).
-noreset	Do not perform an ECU reset after switching an LCS.
-svt <SVT file>	Absolute path to SVT-file.
-blacklist <file>	Absolute path to Blacklist-file.
-whitelist <file>	Absolute path to Whitelist-file.

### Example:

```
E-Sys.bat -writeLcsToken -connection C:\conf\connection.properties -tokenset
C:\Daten\token-3730977-000101-000102030405060708090A0B0C0D0EEE.json -
noreset -whitelist C:\Daten\whitelist.txt
E-Sys.bat -writeLcsToken -connection C:\conf\connection.properties -tokenset
C:\Daten\token-3730977-000101-000102030405060708090A0B0C0D0EEE.json -svt
C:\SVT\SVT_I20.xml
E-Sys.bat -writeLcsToken -connection C:\conf\connection.properties -tokenset
C:\Daten\token-3730977-000101-000102030405060708090A0B0C0D0EEE.json -
noreset -svt C:\SVT\SVT_I20.xml -blacklist C:\Daten\blacklist.txt
```

### Information:

[see LCS section](#)

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 2 - The action was performed with errors.
  - 3 - Unable to connect to the server.

## **IPsec Deactivation**

### **Generate and write Bit Mask**

Command in client batch mode:



E-Sys.bat -writelpsecBitmask -connection <config file> -svt <SVT file> [-blacklist <file>] [-whitelist <file>]

### Command in client-server batch mode:

E-Sys.bat -server -writelpsecBitmask -connection <config file> -svt <SVT file> [-blacklist <file>] [-whitelist <file>]

### Declaration:

-writelpsecBitmask	Command.
-connection <config file>	Absolute path to configuration file with the connection parameter (see <a href="#">Connection Parameter.</a> )
-svt <SVT file>	Absolute path to SVT-file.
-blacklist <file>	Absolute path to Blacklist-file.
-whitelist <file>	Absolute path to Whitelist-file.

### Example:

E-Sys.bat -writelpsecBitmask -connection C:\conf\connection.properties -svt C:\SVT\SVT\_I20.xml -blacklist C:\Daten\blacklist.txt -whitelist C:\Daten\whitelist.txt  
E-Sys.bat -writelpsecBitmask -connection C:\conf\connection.properties -svt C:\SVT\SVT\_I20.xml

### Information:

[see IPsec section](#)

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 2 - The action was performed with errors.
  - 3 - Unable to connect to the server.
  - 4 - The action was performed without errors but with warnings.

## **Activate/Deactivate SecOC**

The SecOC Activate and Deactivate commands update the Security Onboard Communication (SecOC) for a specified group of Electronic Control Units (ECUs). Moreover, it provides the capability to update and reset all ECUs following the update.

## **Activate SecOC**

Action to comfortably enable the SecOC Lcs and update the ServicePack Lcs according to each ECU Service Pack (SP) and the SP of the SecOC Master Boardnet.

### Command in client batch mode:

```
E-Sys.bat -activateSecOC -connection <config file> -svt <SVT file> [-noreset] [-blacklist <file>] [-whitelist <file>]
```

### Command in client-server batch mode:

```
E-Sys.bat -server -activateSecOC -connection <config file> -svt <SVT file> [-noreset] [-blacklist <file>] [-whitelist <file>]
```

### Declaration:

-activateSecOC	Command.
-connection <config file>	Absolute path to configuration file with the connection parameter (see <a href="#">Connection Parameter.</a> )
-svt <SVT file>	Absolute path to SVT-file.
-noreset	Do not perform an ECU reset after switching an LCS.
-blacklist <file>	Absolute path to Blacklist-file.
-whitelist <file>	Absolute path to Whitelist-file.

-

## **Deactivate SecOC**

Action to comfortably disable the SecOC Lcs and update the ServicePack Lcs according to each ECU Service Pack (SP) and the SP of the SecOC Master Boardnet.

### Command in client batch mode:

E-Sys.bat -deactivateSecOC -connection <config file> -svt <SVT file> [-noreset] [-blacklist <file>] [-whitelist <file>]

### Command in client-server batch mode:

E-Sys.bat -server -deactivateSecOC -connection <config file> -svt <SVT file> [-noreset] [-blacklist <file>] [-whitelist <file>]

### Declaration:

-deactivateSecOC	Command.
-connection <config file>	Absolute path to configuration file with the connection parameter (see <a href="#">Connection Parameter.</a> )
-svt <SVT file>	Absolute path to SVT-file.
-noreset	Do not perform an ECU reset after switching an LCS.
-blacklist <file>	Absolute path to Blacklist-file.
-whitelist <file>	Absolute path to Whitelist-file.

## **Activate/Deactivate/Read ACL**

SP25 introduced ACL (access control list). The ACL allows defined SOME/IP communication in a form of a whitelist. ACL protection can be started or stopped and the current ACL status can be read out. The control units are delivered with an inactive ACL.

### **Activate ACL**

Action to activate the ACL in an ECU or list of ECUs.

### Command in client batch mode:

E-Sys.bat -activateAcl -connection <config file> -svt <SVT file> [-blacklist <file>] [-whitelist <file>]

### Command in client-server batch mode:

E-Sys.bat -server -activateAcl -connection <config file> -svt <SVT file> [-blacklist <file>] [-whitelist <file>]

### Declaration:

-activateAcl	Command.
-connection <config file>	Absolute path to configuration file with the connection parameter (see <a href="#">Connection Parameter.</a> )
-svt <SVT file>	Absolute path to SVT-file.
-blacklist <file>	Absolute path to Blacklist-file. (Optional)
-whitelist <file>	Absolute path to Whitelist-file. (Optional)

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 2 - The action was performed with errors.

### **Deactivate ACL**

Action to deactivate the ACL in an ECU or list of ECUs.

#### Command in client batch mode:

E-Sys.bat -deactivateAcl -connection <config file> -svt <SVT file> [-blacklist <file>] [-whitelist <file>]

#### Command in client-server batch mode:

E-Sys.bat -server -deactivateAcl -connection <config file> -svt <SVT file> [-blacklist <file>] [-whitelist <file>]

### Declaration:

-deactivateAcl	Command.
-connection <config file>	Absolute path to configuration file with the connection parameter (see <a href="#">Connection Parameter.</a> )
-svt <SVT file>	Absolute path to SVT-file.
-blacklist <file>	

	Absolute path to Blacklist-file. (Optional)
-whitelist <file>	Absolute path to Whitelist-file. (Optional)

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 2 - The action was performed with errors.

## **Read ACL**

Action to read the status of ACL in an ECU or list of ECUs.

### Command in client batch mode:

E-Sys.bat        -readAcl -connection <config file> -svt <SVT file> [-blacklist <file>]  
 [-whitelist <file>]

### Command in client-server batch mode:

E-Sys.bat -server -readAcl -connection <config file> -svt <SVT file> [-blacklist  
 <file>] [-whitelist <file>]

### Declaration:

-readAcl	Command.
-connection <config file>	Absolute path to configuration file with the connection parameter (see <a href="#">Connection Parameter.</a> )
-svt <SVT file>	Absolute path to SVT-file.
-blacklist <file>	Absolute path to Blacklist-file. (Optional)
-whitelist <file>	Absolute path to Whitelist-file. (Optional)

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 2 - The action was performed with errors.

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

## General:

- [ ].....optional, can be specified, but does not have to be
- ( ).....Selection, must be specified
- | ..... "or", is used in a selection

## Start E-Sys-Batch-Server

Command to start an E-Sys-Batch-Server.

### Command in client batch mode:

E-Sys.bat -startserver

or

E-Sys.bat -startserver [-prop <property file>]  
                          [-setProxy <configuration file> ]  
                          [-authenticationCoding ( -user <username> | -  
useSwlSecCertificate ) -connection <bmw\_intranet|internet>]  
                          [-authenticationSfa -userForSfa <username>]  
                          [-dialog]

### Command in client-server batch mode:

---Only applicable in client-operation---

### Declaration:

-startserver	command um den Server zu starten.
-prop <property file>	Absolute path to the E-Sys-property-file to be used.
- <b>setProxy</b> <config file>	Absolute path to Proxy-Configuration-file.
- <b>authenticationCoding</b>	Argument.
-user <username>	Secure Coding: Backend-Login via username (BASIC) (example: max.mustermann).
-useSwlSecCertificate	





- If the parameter `-authenticationSfa` is set, the E-Sys-server the additional credentials for SFA-Backend are set. The client certificate that is still required must have already been imported. The validity of the specified credentials is not checked. This is only done with the respective backend request.
- If the switch `-dialog` is included, a GUI-dialog is displayed in which login data must be entered.
- It makes no sense to specify the property-file (`-prop <property file>`) in connection with a client-request to the server (`-server`) because the property-file is not used by the server. The server uses the property-file loaded at server-startup.
- The following codes are returned:
  - 0 - The action was successful.
  - 1 - Errors occurred so that the action could not be completed.
  - 2 - The action was performed with errors.
  - 3 - Unable to connect to the server.

## **Stop E-Sys-Batch-Server**

Command to exit the E-Sys-Batch-Servers.

### Command in client batch mode:

-----Only applicable in server-client-operation---

### Command in client-server batch mode:

E-Sys.bat `-server -stop`

### Declaration:

<code>-server</code>	command for Server-Client-operation.
<code>-stop</code>	command to stop the Server.

Example:

E-Sys.bat -server -stop

Information:

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - Errors occurred so that the action could not be completed.
  - 2 - The action was performed with errors.
  - 3 - Unable to connect to the server.

**Check if an E-Sys-Batch-Server is active**

Command to check if an E-Sys-Batch-Server has been started.

Command in client batch mode:

---Only applicable in server-client-operation---

Command in client-server batch mode:

E-Sys.bat -server -check

Declaration:

-server	command for Server-Client-operation.
-check	command to check the Server-Client-operation.

Example:

E-Sys.bat -server -check

Information:

- The following codes are returned:
  - 0 - The action was successful.

- 1 - Errors occurred so that the action could not be completed.
- 2 - The action was performed with errors.
- 3 - Unable to connect to the server.

## **Establishing a vehicle-connection in server-client-operation**

If you need a vehicle-connection for an action, it must be established with `-openconnection`.

### Command in client batch mode:

---Only applicable in server-client-operation---

### Command in client-server batch mode:

E-Sys.bat `-server -openconnection <config file>`

### Declaration:

<code>-server</code>	command for Server-Client-operation.
<code>-openconnection &lt;config file&gt;</code>	Absolute path to configuration file with the connection parameter (see <a href="#">Connection Parameter</a> ).

### Example:

E-Sys.bat `-server -openconnection C:\config\Client-Server_Server.config`

### Information:

- Not all actions require vehicle-connection-parameters. If any are specified, they will not lead to an error, but will be ignored.
- The following codes are returned:
  - 0 - The action was successful.
  - 1 - Errors occurred so that the action could not be

completed.

2 - The action was performed with errors.

3 - Unable to connect to the server.

## **Disconnect vehicle-connection in server-client-operation**

### Command in client batch mode:

---Only applicable in server-client-operation---

### Command in client-server batch mode:

E-Sys.bat -server -closeconnection

### Declaration:

-server	command for Server-Client-operation.
-closeconnection	command to close the vehicle-connection.

### Example:

E-Sys.bat -server -closeconnection

### Information:

- The following return codes are returned:
  - 0 - The action was successful.
  - 1 - Errors occurred so that the action could not be completed.
  - 2 - The action was performed with errors.
  - 3 - Unable to connect to the server.

## **Cancel ongoing TAL-processing**

Command to cancel a running TAL-processing

## Command in client batch mode:

---Only applicable in server-client-operation---

## Command in client-server batch mode:

E-Sys.bat -server -aborttalexecution

## Declaration:

-server	command für Server-Client-Betrieb.
-aborttalexecution	command to cancel a running TAL-processing.

## Example:

E-Sys.bat -server -aborttalexecution

## Information:

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - Errors occurred so that the action could not be completed.
  - 2 - The action was performed with errors.
  - 3 - Unable to connect to the server.

## **Expiration example for a server-client-operation**

```
E-Sys.bat -startserver
E-Sys.bat -server -check
E-Sys.bat -server -authenticationCoding -user <username> -connection
<Backend-Connection> -setProxy <configuration-file> -dialog
E-Sys.bat -server -openconnection <configuration-file>
E-Sys.bat -server -talexecution <configuration-file>
...
E-Sys.bat -server -talexecution <configuration-file>
E-Sys.bat -server -closeconnection
E-Sys.bat -server -stop
```

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

## General:

It converts OrderXML-files (vehicle order) from a specified path to FA (vehicle-order-files). The file-names of the OrderXML-files must correspond to pattern A<xxxxxxx>.xml (x = letter or digit).

[ ].....optional, can be specified, but does not have to be

( ).....Selection, must be specified

| ..... "or", is used in a selection

## Convert OrderXML to FA

### Command in client batch mode:

E-Sys.bat -convertorderxml <FO directory> -out <FA directory>

### Command in client-server batch mode:

E-Sys.bat -server -convertorderxml <FO directory> -out <FA directory>

### Declaration:

-convertorderxml <FO directory>	Absolute path to the directory with the FO files.
-out <FA directory>	Absolute path to the directory for the storage of the FA files.

### Example:

E-Sys.bat -convertorderxml c:\Data\FO -out C:\Data\FA

### Information:

- For each OrderXML-file, an FA-file is created with the file-name pattern "<OrderXML>\_FA\_<Timestamp>.xml".

OrderXML..... Name of OrderXML  
Timestamp..... Time stamp according to the pattern ddmmyyyyhhmm

- The following codes are returned:
  - 0 - The processing was completed successfully.
  - 1 - Errors have occurred, so no verification was possible.
  - 2 - Errors have occurred while processing individual files.  
It is possible that individual files have been converted correctly.
  - 3 - Unable to connect to the server.

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.



## General:

- [ ].....optional, can be specified, but does not have to be
- ( ).....Selection, must be specified
- | ..... "or", is used in a selection

## Batch-command for executeFv

### Command in client batch mode:

E-Sys.bat -executeFv <config file> [-ignoreBATHAF]

### Command in client-server batch mode:

E-Sys.bat -server -executeFv <config file> [-ignoreBATHAF]

### Declaration:

-executeFv <config file>	Absolute path to ecexuteFv-Configuration-file.
-ignoreBATHAF	Optional parameter. Ignores the S1-switch.

### Example:

E-Sys.bat -executeFv C:\data\executeFv.config -ignoreBATHAF

### Information:

- Information about the batch-command -executeFv can be obtained from the "executeFv manual".
- If a BAT/HAF test reveals that switch S1 is open, the further execution of the command depends on parameter -ignoreBATHAF.  
-ignoreBATHAF has been set:  
The flash sequence continues. However, the user must be aware that the execution can lead to interruptions. In

this case, the TAL-processing cannot be ended with value 0 (OK), but at best with value 4 (with warning).

-ignoreBATHAF not set:

The execution of the command -talexecution is canceled.

- An eight hexadecimal character code is returned with the format:

[Overall] [Not defined] [ErrorLcs] [ErrorIPSec]  
[ErrorEcuModes] [ErrorCert] [ErrorTal] [ErrorVcm]

Possible values:

- 0: Success
- 1: Error
- 2: Warning
- 3: not processed
- F: not defined

Example: 1F021333

"Overall" ..... "Error"  
"ErrorLcs" ..... "Success"  
"ErrorIPSec" ..... "Warning"  
"ErrorEcuModes" ... "Error"  
"ErrorCert" ..... "not processed"  
"ErrorTal" ..... "not processed"  
"ErrorVcm" ..... "not processed"

**General:**

[ ].....optional, can be specified, but does not have to be  
( ).....Selection, must be specified  
| ..... "or", is used in a selection

**PDX-Import.**

This command can be used to import containers (KC, IC or PDX template) into the program

Command in client batch mode:

E-Sys.bat -pdximport <PDX container file> -project <project name>

Command in client-server batch mode:

E-Sys.bat -server -pdximport <PDX container file> -project <project name>

Declaration:

-pdximport <PDX container file>	Absolute path of the container to be imported.
-project <project name>	Name of the project to import a container into.

Example:

E-Sys.bat -pdximport C:\Data\ODX\TEST\_03\_I020.001\_023\_015.pdx -project TEST\_03\_I020\_001\_023\_015

Important additional information for partial KCs:

If you want to import a constructed container (maybe by yourself) you have to keep in mind, that the referenced software is imported into psdzdata. **The referenced SWE(s) are only a reference and do not exist as actual data in the Container, so it is important to import the actual data.**

In case the referenced software unit(s) are going to be needed, they have to be imported by another container, in which the unit(s) exist as a real data.

Information:

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - Errors occurred so that the action could not be completed.
  - 2 - The action was performed with errors.
  - 3 - Unable to connect to the server.
  - 4 - The action was performed without errors but with warnings.

**PDX-Update**

This command can be used to update containers with a PDX-template.

Command in client batch mode:

E-Sys.bat -pdxupdate <PDX container file> [-out <PDX container file>] -template <PDX template file>

Command in client-server batch mode:

E-Sys.bat -server -pdxupdate <PDX container file> [-out <PDX container file>] -template <PDX template file>

Declaration:

-pdxupdate <PDX container file>	Absolute path of the container whose template is to be updated.
-out <PDX container file>	Absolute path of the container to be created. If the -out parameter is not specified, the container will be created in the same directory as the original container.

	the container is saved with the same name after the update.
-template <PDX template file>	Absolute path of the PDX-template, which the container should be updated with.

**Example:**

```
E-Sys.bat -pdxupdate C:\Data\ODX\TEST_03_I020.001_023_015.pdx -out C:\Data\ODX\TEST_03_I020.001_023_020.pdx
C:\Data\ODX\PT01_template.003_040_000.pdx
```

**Information:**

- The following codes are returned:
  - 0 - A PDX container was created without errors.
  - 1 - Errors occurred so that no PDX container could be created.
  - 2 - A PDX container with errors was created.
  - 3 - Unable to connect to the server.

**Create PDX-component-containers**

This command can be used to create component-containers.

Partial PDX-component-container:

It is also possible to create a partial PDX-component-container. Here, a md5 hash value is used to assign software units (btld, flsl, swfk, swfl) to be assigned to this PDX-component-container. These software-units have already been successfully delivered to SWL-Cockpit with a previous PDX container and are already available there.

Further information can be found in the instructions E-Sys\_Handlungsanweisung\_partieller\_KC

**Command in client batch mode:**

```
E-Sys.bat -pdxcontainer <PDX container definition file> -out <PDX container file>
```

**Command in client-server batch mode:**

```
E-Sys.bat -server -pdxcontainer <PDX container definition file> -out <PDX container file>
```

**Declaration:**

-pdxcontainer <PDX container definition file>	Absolute path of the definition-file.
-out <PDX container file>	Absolute path of the PDX-container to be generated.

**Example:**

```
E-Sys.bat -pdxcontainer C:\Data\pdx_def.xml -out C:\Data\ODX\TEST_03_I020.001_023_020.pdx
```

**Information:**

- The PDX-container-definition-file describes the contents of the component-container. If the corresponding PDX-container is created and stored below the specified path.
- SWE files (BTLD, SWFL, SWFK, FLSL) of a specific ECU can be defined by reference only (see SWEReferenceList). Such SWE are not part of the created container and are not present in the container. The software units, that are listed under the reference (SWEReferenceList) are not real existing software units. They will be at the container.

- A simple definition-file has the following content:

```
<PDXContainerDefinition xmlns="http://bmw.com/2010/esys.data.pdxcontainerdefinition">
  <PDXTemplate>C:/Data/ODX/F001_template.002_006_009.pdx</PDXTemplate>
  <BRVList>
    <BRV>F001</BRV>
  </BRVList>
</PDXContainerDefinition>
```

```

    <BRV>F010</BRV>
  </BRVList>
  <ECUVariantList>
    <ECUVariant name="ECU1">
      <BaseVariant>EVALBOARD</BaseVariant>
      <SWEList> <!--List of software unites, that are part of the container (existing at the container)-->
        <File>C:/Data/SWE/btld_01020304_002_000_000.bsw</File>
      </SWEList>
      <SWEReferenceList> <!--List of software unites that are no part of the container (the hash value reference only to an at swl-cockpit ;
        <TE checksum="735208badabcd1e447c217f30aa621b0" descTableStartAddress ="1">btld_01234567_004_005_006</TE>
        <TE checksum="d8462a1b22db2c88b29bc6dc989b2004" descTableStartAddress ="13FE44">swfl_01234567_007_005_001</
      </SWEReferenceList>
    </ECUVariant>
  </ECUVariantList>
</PDXContainerDefinition>

```

- The XML-file is defined by the PDXContainerDefinition.xsd schema (see <E-System-Installationsverzeichnis>/help). The schema defines all elements and describes them in comments that can be used in the definition file.
- As in the command-line, all path information of files and directories must be absolute in the definition file.
- If errors occur, an error-file (named errors... txt) will be created. It is placed in the directory where the container is stored and contains the error messages of the errors that occurred.
- The following codes are returned:
  - 0 - A PDX container was created without errors.
  - 1 - Errors occurred so that no PDX container could be created. An error file has been created.
  - 2 - A PDX container with errors was created. An error file has been created.
  - 3 - Unable to connect to the server.

### Creation of containers using Smart Actuators (SMAC)

Containers can consist of ECUs, SMACs or a combination of these. For usage of SMACs, the container definition XML file makes use of the SMACVariant XML tag (under SMACVariantList). SMACs define variants. The name of a SMAC must be according to the EE component name which is used in its SWEList.

*See full example:*

```

<PDXContainerDefinition xmlns="http://bmw.com/2010/esys.data.pdxcontainerdefinition">
  <PDXTemplate>C:\Data\PT01_template.003_100_000.pdx</PDXTemplate>
  <BRVList>
    <BRV>S15C</BRV>
    <BRV>S18A</BRV>
  </BRVList>
  <ECUVariantList>
    <ECUVariant name="NEU_EV">
      <BaseVariant>BDC_GW3</BaseVariant>
      <SWEList>
        <File>C:\data\SWE\btld_00004139.bsw.008_041_040</File>
      </SWEList>
    </ECUVariant>
  </ECUVariantList>
  <SMACVariantList>
    <SMACVariant name="SMAC-TEST01">
      <SWEList>
        <File>C:\Data\SWE\btld_0000ac0b.bsw.011_000_001</File>
      </SWEList>
    </SMACVariant>
  </SMACVariantList>
</PDXContainerDefinition>

```

### Notice:

The command to create a container with SMACs remains as for a KC with ECUs. Only the container definition file must be adapted.

## **Merge multiple PDX-Container**

This command can be used to package multiple component-containers into one component-

### **Command in client batch mode:**

```
E-Sys.bat -mergepdxcontainer <PDX container definition file> -out <PDX container file>
```

### **Command in client-server batch mode:**

```
E-Sys.bat -server -mergepdxcontainer <PDX container definition file> -out <PDX container file>
```

### **Declaration:**

-mergepdxcontainer <PDX container definition file>	Absolute path of the definition-file.
-out <PDX container file>	Absolute path of the PDX-container to be generated.

### **Example:**

```
E-Sys.bat -mergepdxcontainer C:\Data\pdx_merge.xml -out C:\Data\ODX\TEST_03_1020.001_023_020.pdx
```

### **Information:**

- The PDX-container-definition-file describes the contents of the component-container. The file path, the template to use for the component-container (element <PDXTemplate>), and the path to merge PDX-containers (element <ContainerList>). The file is read and a corresponding PDX-container is stored below the path specified on the command-line
- A simple definition-file has the following content:

```
<PDXContainerDefinitionMerge xmlns="http://bmw.com/2010/esys.data.pdxcontainerdefinition">
  <PDXTemplate>C:/Data/ODX/F001_template.002_006_009.pdx</PDXTemplate>
  <ContainerList>
    <File>C:/Data/ODX/F001_EKPM_03.001.000.001.pdx</File>
    <File>C:/Data/ODX/F001_DME_MSD85.001.001.002.pdx</File>
    <File>C:/Data/ODX/F001_EGS_GKEB23.002.008.003.pdx</File>
  </ContainerList>
</PDXContainerDefinitionMerge>
```
- The XML-file is defined by the PDXContainerDefinition.xsd schema (see <E-Sys-Installationsverzeichnis>/help). The schema defines all elements and contains additional comments which can be used in the definition file.
- As in the command-line, all path information of files and directories must be absolute in the command-line.
- If errors occur, an error-file (named errors... txt) will be created. It is placed in the directory where the container is stored and contains the error messages of the errors that occurred.
- The following codes are returned:
  - 0 - A PDX container was created without errors.
  - 1 - Errors occurred so that no PDX container could be created. An error file has been created.
  - 2 - A PDX container with errors was created. An error file has been created.
  - 3 - Unable to connect to the server.
- Not supported:
  - ECU-Variant-Patterns
  - Documentation files of component containers
  - Merging containers with the same ECU variants

## **PDX component containers name convention**

Component containers should be created according to the following name convention:  
**<E/E Component>\_<freetext>.<version>.pdx**

E/E Component	[A-Z0-9_]{1,8}__[A-Z0-9_]{1,8}	E.g., ZBE__04
Free text	[A-Z0-9_]{1,20}	E.g., F001
Version	\d{3}_\d{3}_\d{3}	E.g., 003_004_000

Example of a valid component container name: *ZBE\_\_04\_\_F001.003\_004\_000.pdx*

Each version part must be in in the range [0-255]. Any other value will not be allowed. For e: *ZBE\_\_04\_\_F001.003\_004\_256.pdx* is an invalid component container name.

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This r appear if you compile this help file with the registered version of Help & Manual.

## General:

If an active proxy has been defined in the properties (proxy.useProxy=**true**), it is taken into account in both client- and server-client-based E-Sys-batch calls, **even if you did not use a -setProxy in the server-client-batch.**

If the proxy-server requires authentication (proxy.authRequired=**true** in the Properties), you are prompted to enter the login and password for the proxy. Only Basic-authentication is currently supported as an authentication method on the proxy server.

- [ ].....optional, can be specified, but does not have to be
- ( ).....Selection, must be specified
- | ..... "or", is used in a selection

## Read Proxy-Status

The current Proxy-Status is output to stdout.

### Command in client batch mode:

E-Sys.bat        -readStatusProxy

### Command in client-server batch mode:

E-Sys.bat -server -readStatusProxy

### Declaration:

-readStatusProxy	command.
------------------	----------

### Example:

E-Sys.bat -readStatusProxy



## Information:

- Here is an example edition:

A proxy is used:  
https://web-proxy.de:8080  
No proxy-authentication required  
Do not use proxy on the following URLs:  
https://bmw.de  
https://bmwgroup.com

- The following codes are returned:  
0 - The action was successful.  
1 - An exception occurred so the action could not be completed.  
3 - Unable to connect to the server.

## **Setting a Proxy**

The command provides the required proxy information (proxy-host, proxy-port, proxy-authentication, username, password, URL-blacklist) E-Sys.

An active proxy is considered for all backend calls.

### Command in client batch mode:

---Only applicable in server-client operation---

### Command in client-server batch mode:

E-Sys.bat -server -setProxy <configuration file> [-dialog]

### Declaration:

-setProxy <config file>	Absolute path to the configuration-file with the proxy-server-parameters.
-dialog	Backend-Login with GUI-dialog.

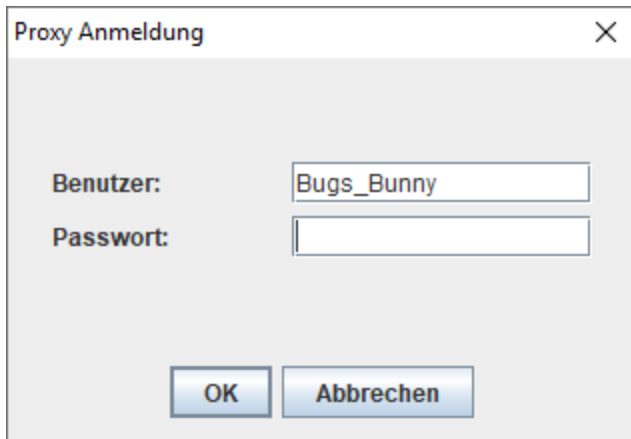
### Example:

E-Sys.bat -server -setProxy c:\data\proxyConfig.config

E-System.bat -server -setProxy c:\data\proxyConfig.config -dialog

### Information:

- If the switch -dialog is defined, a GUI-dialog is displayed in which authentication-data must be entered.
- The authentication-dialog looks like this:



- The format of the configuration-file to pass is defined as follows:

```
# Proxy-URL e.g.: https://web-proxy.de
PROXY_HOST = <URL>

# Proxy-Port z.B. 8080
PROXY_PORT = <Port>

# Proxy authentication required
# DEFAULT = off
PROXY_AUTH = <on|off>

# PROXY_USERNAME is mandatory parameter if PROXY_AUTH = on
PROXY_USERNAME = <Proxy ID>

# PROXY_EXCLUDE_LIST is an optional parameter, but if it is used, the file
must be in the file path
# A list of the backend-URLs for which an active proxy should not be used.
PROXY_EXCLUDE_LIST = <file-path to the list>
```

- **The command is only supported in server-client-mode.**

- Necessary parameters are PROXY\_HOST and PROXY\_PORT.
- Optional parameters are PROXY\_AUTH, PROXY\_USERNAME, and PROXY\_EXCLUDE\_LIST. If omitted, the information from the E-Sys.properties-file is read or default settings are used.
- The set proxy-settings are stored in the properties (except PROXY\_USERNAME).
- The value of the PROXY\_USERNAME is used only for the course of the E-Sys-session. There is no Proxy-ID storage.
- If PROXY\_AUTH = set on, the password for proxy must be entered manually in the command-line of the E-Sys-server.
- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 3 - Unable to connect to the server.

### **Turn off Proxy**

Do not use a set Proxy anymore.

#### Command in client batch mode:

---Only applicable in server-client operation---

#### Command in client-server batch mode:

E-Sys.bat -server -removeProxy

#### Declaration:

-removeProxy	command.
--------------	----------

#### Example:

E-Sys.bat -server -removeProxy

Information:

- **The command is only supported in server-client-mode.**
- In the Config-file, the use of Proxy is disabled, all other proxy settings remain.
  
- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 3 - Unable to connect to the server.

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

## General:

The FSC comfort functions are also available in batch mode.

[ ].....optional, can be specified, but does not have to be

( ).....Selection, must be specified

| ..... "or", is used in a selection

## **SWT-Status - Without using a configuration file with the connection parameter**

This command determines the SWT status.

### Command in client batch mode:

```
E-Sys.bat -swtstatus -project <project name> -vehicleinfo <vehicle info> [-diagaddress <diagnosis address> -basevariant <basevariant>]
```

### Command in client-server batch mode:

```
E-Sys.bat -server -swtstatus -project <project name> -vehicleinfo <vehicle info> [-diagaddress <diagnosis address> -basevariant <basevariant>]
```

### Declaration:

-swtstatus	command.
-project <project name>	Project name for the connection.
-vehicleinfo <vehicle info>	Vehicle for connection.
-diagaddress <diagnosis address>	The diagnostic-address of the ECU (optional parameter).
-basevariant <basevariant>	The basevariant of the ECU (optional parameter).

### Example:

```
E-Sys.bat -swtstatus -project I020_TEST_03_I020_001_023_015 -vehicleinfo I020 -diagaddress 0x33 -basevariant TEST
```

E-Sys.bat -swtstatus -project I020\_TEST\_\_03\_\_I020\_001\_023\_015 -vehicleinfo I020

### Information:

- The following codes are returned: 0 - The action was successful.  
1 - An exception occurred so the action could not be completed.  
2 - The action was performed with errors.  
3 - Unable to connect to the server.

## **SWT-Status - Using a configuration file with the connection parameter**

This command determines the SWT status.

### Command in client batch mode:

E-Sys.bat -swtstatus -connection <config file> [-diagaddress <diagnosis address> -basevariant <basevariant>]

### Command in client-server batch mode:

E-Sys.bat -server -swtstatus -connection <config file> [-diagaddress <diagnosis address> -basevariant <basevariant>]

### Declaration:

-swtstatus	command.
-connection <config file>	Absolute path to configuration file with the connection parameter (see <a href="#">Connection Parameter</a> ).
-diagaddress <diagnosis address>	The diagnostic-address of the ECU (optional parameter).
-basevariant <basevariant>	The basevariant of the ECU (optional parameter).

## Example:

```
E-Sys.bat -swtstatus -connection C:\conf\connection.properties -diagaddress 0x33 -basevariant TEST
```

```
E-Sys.bat -swtstatus -connection C:\conf\connection.properties
```

## Information:

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 2 - The action was performed with errors.
  - 3 - Unable to connect to the server.

-  
-  
-

## **SWT-Action**

This command can be used to perform an SWT action.

## Command in client batch mode:

```
E-Sys.bat -swtaction <config file>
```

## Command in client-server batch mode:

```
E-Sys.bat -server -swtaction <config file>
```

## Declaration:

-swtaction <config file>	Absolute path to configuration file.
--------------------------	--------------------------------------

## Example:

```
E-Sys.bat -swtaction C:\conf\connection.properties
```

## Information:

- The concrete SWT-action is defined in the configuration-file together with the [Connection Parameter](#) and the following parameters.

FA = <FA-Filename>

VIN = <VIN-String>

# write\_vin|store|update|upgrade|deactivate

SWT\_ACTION = <SWT Aktion>

BASE\_VARIANT = <Basevariant>

DIAG\_ADDRESS = <Diagnoseaddress (hex)>

APPLICATION\_NUMBER = <Applikationsnumber>

UPGRADE\_INDEX = <Upgrade-Index>

FSC = <FSC-Filename>

- No concurrent indication of FA and VIN is allowed.
- The FSC is only necessary for the store and upgrade actions.
- For the write\_vin action, the VIN parameter must be defined.
- The write\_vin action does not take into account the parameters APPLICATION\_NUMBER and UPGRADE\_INDEX.
- The store|update|upgrade|deactivate actions must always have the parameters BASE\_VARIANT, DIAG\_ADDRESS, APPLICATION\_NUMBER, and UPGRADE\_INDEX defined.
- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 3 - Unable to connect to the server.

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.



**General:**

With this batch-mode you can program and encode ECUs (SWE and CAF).  
 Before you can start a TAL-processing in the batch, you must first import the appropriate cor  
 You can import PDX containers in GUI-mode (see [PDX-Charger](#)) or batch-mode (see [PDX-Con](#)

- [ ].....optional, can be specified, but does not have to be
- ( ).....Selection, must be specified
- | ..... "or", is used in a selection

**Note:** [Programming protection, mirror protocol and their impact on TAL processing](#)

**TAL-processing**

This command starts a TAL-processing.

Command in client batch mode:

E-Sys.bat -talexecution <config file> [-ignoreBATHAF]

Command in client-server batch mode:

E-Sys.bat -server -talexecution <config file> [-ignoreBATHAF]

Declaration:

-talexecution <config file>	Absolute path to configuration file.
-ignoreBATHAF	Optional parameter. Ignores the S1-switch.

Example:

E-Sys.bat -talexecution C:\conf\connection.properties -ignoreBATHAF

Information:

- If a BAT/HAF test reveals that switch S1 is open, the further execution of the command depends on the parameter `-ignoreBATHAF`.

-ignoreBATHAF has been set:

The flash sequence continues. However, the user must be aware that the execution can lead to a failure. In the worst case, the TAL-processing cannot be ended with value 0 (OK), but at best with value 4 (with error).

-ignoreBATHAF not set:

The execution of the command `-talexecution` is canceled.

- In the configuration file, the following parameters can be defined in addition to the [Connection](#)

```
TAL = <TAL-Filename>
FA = <FA-Filename>
VIN = <VIN-String>

SVT = <SVT-Filename>

# fa|shipment|ncd (Default: fa)
CODING_TYPE = <Coding-Type>

# optional path indication where the NCDs are located
# Format <NCD-path 1>;<NCD-path 2>... , e.g. "C:/NCD/00001234_0000dddd.caf.026_000_002.ncd;C:/NCD/00012345
# Only considered for coding type "ncd". Caution: only one ECU may be handled during NCD encoding.
NCD_LIST = <NCD-path 1>;

# Perform FA-coding locally
# on|off OR true|false
```

USE\_LOCAL\_NCD = on  
# Optional path where the NCDs are located, attention: in this directory a subfolder with the VIN as the name is search  
# If omitted, the NCD directory is used, which is specified in the settings. (stored in properties)  
# Only considered for coding type "fa" und "shipment".  
PATH\_NCD = C:/signedNCD/  
  
# on|off  
PARALLEL\_PROGRAMMING = off  
  
# on|off  
CHECK\_PROGRAMMING\_COUNTER = on  
  
REPEATS\_ON\_ERROR = <number of repetitions in case of failure>  
  
# off|all|progress|transaction|error|tal|aep  
EVENT= <Event-Filter for console output>  
  
# on|off  
FILL\_INSTALLED\_ECU\_LIST = on  
  
# merge|overwrite  
FILL\_INSTALLED\_ECU\_LIST\_MODE = <Mode, how to fill the InstalledECUList>  
  
# on|off  
MODE\_SWITCH\_PROGRAMMING = <Mode-Switch during programming on/off>  
# List of ECUs (diagnostic addresses in HEX), to be switched to programming mode.  
# Without the parameter, all available ECUs are switched.  
# E.g.: 10,63  
ECUS\_TO\_SWITCH\_TO\_PROGRAMMING\_MODE = <Ecu List>  
  
# Absolute path to TAL-Filter-file.  
TAL\_FILTER = <TAL-Filter-Filename>  
  
# Absolute path to a backup-directory with Individual-data.  
IDR\_BACKUP\_PATH = <IDR-Backup-Directory>  
  
# on|off (If the parameter is on, the mileage is read out before the TAL-processing and written in fingerprint.  
READ\_MILEAGE\_FROM\_GWSZ = off  
  
# Mileage for Fingerprint  
# If a valid value is specified, this will be written in Fingerprint, independent of the mileage being read out.  
# Default value '0' is used for Fingerprint, if neither MILEAGE nor READ\_MILEAGE\_FROM\_GWSZ was indicated  
MILEAGE = <0-655330>  
  
# URL of HDD-Update Servers  
HDD\_UPDATE\_SERVER\_URL = <URL for HDD-Update>  
  
# on|off (Response on event during TAL-processing enable/disable)  
ROE\_DEACTIVATION = off  
  
# on|off (prevent Http-transmission-protocol during TAL-processing)  
PREVENT\_HTTP\_TRANSMISSION = off  
  
# on|off (perform Optimized Bootloader flash on ECUs that support this functionality.  
OPTIMIZABLE\_BOOTLOADER\_FLASH = off  
  
# on|off (check expectedSgbmids)  
EXPECTED\_SGBMID\_VALIDATION = on  
  
# Optional Parameter.  
# If specified, only client-batch-mode is considered  
# Default value = off  
USE\_SWL\_SEC\_CERTIFICATE = <on|off>  
  
# Optional Parameter. Considered only in client-batch-mode.  
# If the parameter is omitted SWL\_SEC\_CONNECTION, the value from Esys.properties is used.  
SWL\_SEC\_CONNECTION = <bmw\_intranet|internet>  
  
# Optional Parameter.  
# Specify whether all SWEn for SMAC should always be transmitted.  
# If deactivated, only required SWEn are transferred. SWEn that are already present and valid on the smart actuator a  
FORCE\_SWE\_TRANSMISSION\_FOR\_SMAC\_SWES = <on|off>

```

# Optional Parameter.
# List of ECUs (diagnostic addresses in HEX) for which no UDS fallback may be performed.
# Alternatively, 'all' can be specified to disable the UDS fallback for all ECUs.
# The enumerated diagnostic addresses are only separated by a comma. No space may be inserted between the enum
# Without the parameter, the UDS flash is executed in case of an HTTP error.
# E.g.: 10,63
ECUS_TO_PREVENT_UDS_FALLBACK = < Diagnosis addresses in HEX> | all >

# Optional Parameter.
# UseCase Programming Protection BASIC (Field Mode)
# If not set and a token is required for programming protection basic, the token is automatically requested in the SFA t
PROGRAMMING_PROTECTION_BASIC_TOKEN = <absolute path to signed token (json)>

# Optional Parameter.
# UseCase Programming Protection PLUS (Plant Mode)
PROGRAMMING_PROTECTION_PLUS_TOKEN = <absolute path to signed token file (xml)>

# Optional Parameter.
# UseCase Mirror-Protocol
# Possible values: WITHOUT_AUTH, SIGNED_TOKEN, PLANT_TOKEN, SFA_TOKEN
# Default value = WITHOUT_AUTH
AUTH_METHOD_MIRROR_PROTOCOL = <WITHOUT_AUTH | SIGNED_TOKEN | PLANT_TOKEN | SFA_TOKEN>

# Optional Parameter.
# Mirror-Protocol UseCases: Authentication methods SIGNED_TOKEN, PLANT_TOKEN, SFA_TOKEN
# Mandatory for authentication methods SIGNED_TOKEN, PLANT_TOKEN, SFA_TOKEN
UNSIGNED_TOKEN_FOR_MIRROR_PROTOCOL = <absolute path to unsigned token file>

# Optional Parameter.
# Mirror-Protocol UseCase: Authentication method SIGNED_TOKEN
# Mandatory for authentication method SIGNED_TOKEN
SIGNED_TOKEN_FOR_MIRROR_PROTOCOL = <absolute path to signed token file>

# Optional Parameter.
# Mirror-Protocol UseCase: Authentication method SFA-TOKEN offline
# If the parameter is not set, the token is requested in SFA backend (online BMW-Intranet)
SFA_TOKEN_FOR_MIRROR_PROTOCOL = <absolute path to SFA token file>

# Optional Parameter.
# Mirror-Protocol UseCase: Authentication method PLANT-TOKEN offline
# If the parameter is not set, the token is requested in SFA backend (online BMW-Intranet)
PLANT_TOKEN_FOR_MIRROR_PROTOCOL = <absolute path to PLANT-Mode token file>

```

- Because of "safe coding", a connection to the SWL-Sec2 must be established before execution elements.  
This also means that the computer must have two NICs. Since E-Sys potentially has two di at the same time (SWL-Sec2 backend and vehicle).
- Communication with the SFA backend is required for processing a TAL with mirror protocol relevant transactions.
- **A TAL with coding elements can only be run in E-Sys without server-client operation**
  - a [client Certificate for SWL Security System](#) is imported and the parameter USE\_SWL\_SEC\_C
  - OR the required test-keys are available locally
  - OR the parameter USE\_LOCAL\_NCD is active (NCDs are signed locally)
  - OR at CODING\_TYPE=ncd (NCDs are signed locally)
- Necessary parameters are TAL, PROJECT, VEHICLEINFO and FA or VIN.
- The configuration file can contain either an FA or a VIN, but not both. If an FA is specified, i
- An FA is mandatory to handle a TAL with coding-actions (coding-TAs). Applies to CODING\_TYPE
- CODING\_TYPE = ncd allows you to encode an SG with an already existing NCD file. The op necessary.
- With CODING\_TYPE = ncd, only one ECU may be encoded during a TAL-processing.

- If the configuration-file contains the parameter HDD\_UPDATE\_SERVER\_URL, an **HDD-upda** a **TAL-processing** is performed.
  - For the switches PARALLEL\_PROGRAMMING, CHECK\_PROGRAMMING\_COUNTER and FILL\_INSTALLED\_ECU\_LIST th be defined.
  - The default value for the parameters is the GUI values (from the Esys.properties file).
  - A '\' may not be used in the configuration file. Instead, use '/' or '\\ for path information.
  - The path-information for the parameters TAL, FA and TAL\_FILTER can be both, absolute and rel: information, the reference point is the location of the config-file.
  - All other E-Sys settings are read from the Esys.Properties file.
  - The history of the TAL-processing is stored in the E-Sys log file.
- The following codes are returned:
- 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 2 - The action was performed with errors.
  - 3 - Unable to connect to the server.
  - 4 - The action was performed without errors but with warnings.

### **Check Programming Protection**

This command can be used to output which components are subject to authorization for the

-

#### Command in client batch mode:

E-Sys.bat -isTokenForSecureFlashRequired -connection <config> -tal <path>

#### Command in client-server batch mode:

E-Sys.bat -server -isTokenForSecureFlashRequired -connection <config> -tal <path>

#### Information:

-connection <config file>	Absolute path to configuration file.
-tal	Absolute path to TAL

#### Example:

E-Sys.bat -isTokenForSecureFlashRequired -connection C:\conf\connection.properties -tal C:\Data\TAL.xml

-

- The following codes are returned:
- 0 - The action was successful.
- 1 - An exception occurred so the action could not be completed.
- 3 - Unable to connect to the server.

### **Generate Unsigned Token**

A signed Programming Token is required to execute a TAL with Mirror-Deploy actions. Therefore programming Tokens needs to be created.

This command can be used to generate an Unsigned Programming Token file.

#### Command in client batch mode:

E-Sys.bat -unsignedTokenCreation <config file path>

#### Command in client-server batch mode:

E-Sys.bat -server -unsignedTokenCreation <config file path>

**Information:**

---

-unsignedTokenCreation <config file path>	Absolute path to configuration file.
---	--------------------------------------

---

**Example:**

E-Sys.bat -unsignedTokenCreation C:\conf\configFile

**Information:**

- In the configuration file, the following parameters (beside others) need to be defined in ad [Parameter:](#)

TAL = <TAL-Filename>

VIN = <VIN-String>

SVT\_SOLL = <SVT\_Target-Filename>

UNSIGNED\_TOKEN\_FOR\_MIRROR\_PROTOCOL = <absolute path to unsigned token file>

- If successful the command will create a Unsigned Programming Token File at the specified

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This r compile this help file with the registered version of Help & Manual.

## General:

A TAL is calculated from an SVT-actual- and an SVT-target-file.

[ ].....optional, can be specified, but does not have to be

( ).....Selection, must be specified

| ..... "or", is used in a selection

## TAL-calculation

This command calculates a TAL

### Command in client batch mode:

E-Sys.bat -talcalculation <config file>

### Command in client-server batch mode:

E-Sys.bat -server -talcalculation <config file>

### Declaration:

-talcalculation <config file>	Absolute path to configuration file.
-------------------------------	--------------------------------------

### Example:

E-Sys.bat -talcalculation C:\conf\connection.properties

### Information:

- In the configuration-file, the following parameters can be defined in addition to the [Connection Parameter](#):

# Absolute path of the SVT-Actual file  
SVT\_IST = <SVT-Actual-file>

# Absolute path of the SVT-Target file  
SVT\_SOLL = <SVT-Target-file>

```
# Absolute path of the generated TAL file
GENERATED_TAL = <TAL-filename>

# Absolute path to a TAL-Filter file.
TAL_FILTER = <TAL-Filter-filename>

# normal | idrBackup | idrRestore
TAL_TYPE = <TAL-Type>

# Absolute path to a Backup-Directory with Individual-data.
IDR_BACKUP_PATH = <IDR-Backup-Directory>

# on | off
USE_MIRROR_PROTOKOLL = <mirror-protocol-usage>
```

- Required parameters are SVT-Actual, SVT-Target, and GENERATED\_TAL.
- In standalone mode, additional connection parameters are required (see [Connection Parameter](#))
- In server-client-mode, the server must have a PSdZ-connection open.
- The default value for the parameters is the GUI values (from the Esys.properties file).
- A '\' may not be used in the configuration file. Instead, use '/' or '\\' for path information.
- If the parameter TAL\_TYPE has the value idrRestore, the parameter IDR\_BACKUP\_PATH must also be defined.
- By default the value defined in the property TalGeneration.param.useMirror (from the Esys.properties file) sets the USE\_MIRROR\_PROTOKOLL. However, if the parameter USE\_MIRROR\_PROTOKOLL is defined, it will override the value previously defined. If none of these properties are defined, the default value will be false.
- The following codes are returned: 0 - The action was

successful.

1 - An exception occurred so the action could not be completed.

3 - Unable to connect to the server.

4 - The action was performed without errors but with warnings.

### **Example TAL Filter:**

```
<?xml version="1.0" encoding="UTF-8"?>
<talfilter>
  <allEcu>
    <blUpdate>empty</blUpdate>
    <cdDeploy>empty</cdDeploy>
    <fscDeploy>empty</fscDeploy>
    <hwDeinstall>empty</hwDeinstall>
    <hwInstall>empty</hwInstall>
    <idBackup>empty</idBackup>
    <idDelete>empty</idDelete>
    <idRestore>empty</idRestore>
    <swDeploy>empty</swDeploy>
    <ibaDeploy>empty</ibaDeploy>
    <hddUpdate>empty</hddUpdate>
    <gatewayTableDeploy>empty</gatewayTableDeploy>
  </allEcu>
  <ecuFilter>
    <!-- decimal diagnostic address -->
    <diagAdr>10</diagAdr>
    <setting>
      <blUpdate>mustBeTreated</blUpdate>
      <cdDeploy>mustNotBeTreated</cdDeploy>
      <fscDeploy>mustNotBeTreated</fscDeploy>
      <hwDeinstall>allowedToBeTreated</hwDeinstall>
      <hwInstall>mustNotBeTreated</hwInstall>
      <idBackup>mustNotBeTreated</idBackup>
      <idDelete>mustNotBeTreated</idDelete>
      <idRestore>mustNotBeTreated</idRestore>
      <swDeploy>mustNotBeTreated</swDeploy>
      <ibaDeploy>mustNotBeTreated</ibaDeploy>
      <hddUpdate>mustNotBeTreated</hddUpdate>
    </setting>
  </ecuFilter>
</gatewayTableDeploy>mustNotBeTreated</gatewayTableDeploy>
</setting>
</tالfilter>
```



</tafilter>

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

## General:

2 TSL functions are supported in Batch-Mode.

[ ].....optional, can be specified, but does not have to be

( ).....Selection, must be specified

| ..... "or", is used in a selection

## Read TSL-Status

The TSL-Status is output to stdout

### Command in client batch mode:

E-Sys.bat -tslstatus -connection <config file>

### Command in client-server batch mode:

E-Sys.bat -server -tslstatus -connection <config file>

### Declaration:

-tslstatus	command.
-connection <config file>	Absolute path to configuration file with the connection parameter (see <a href="#">Connection Parameter</a> ).

### Example:

E-Sys.bat -tslstatus -connection C:\conf\connection.properties

### Information:

- The following codes are returned: 0 - The action was successful.
- 1 - An exception occurred so the action could not be completed.
- 3 - Unable to connect to the server.

## **TSL-Update**

### Command in client batch mode:

E-Sys.bat        -updatetsl <SVT filename> -connection <config file>

### Command in client-server batch mode:

E-Sys.bat -server -updatetsl <SVT filename> -connection <config file>

### Declaration:

-updatetsl <SVT filename>	Absolute path to The SVT-file.
-connection <config file>	Absolute path to configuration file with the connection parameter (see <a href="#">Connection Parameter</a> ).

### Example:

E-Sys.bat -updatetsl C:\data\svt.xml -connection C:\conf\connection.properties

### Information:

The following codes are returned:

- 0 - The processing was completed successfully.
- 1 - TSL update could not be started
- 2 - TSL update with errors
- 3 - Unable to connect to the server.

## General:

The following data can be read and written from the VCM (VCM-Master or VCM-Backup): FA, I

- [ ].....optional, can be specified, but does not have to be
- ( ).....Selection, must be specified
- | ..... "or", is used in a selection

## Write VCM-Master (FA, I-Step or SVT-Target)

### Command in client batch mode:

```
E-Sys.bat -writeVcmMaster <FA|ISTUFEN|SVTSOLL> -connection <config file> -in <source file>
```

### Command in client-server batch mode:

```
E-Sys.bat -server -writeVcmMaster <FA|ISTUFEN|SVTSOLL> [-connection <config file>] -in <source file>
```

### Declaration:

-writeVcmMaster <FA ISTUFEN SVTSOLL>	Parameters that can take the value FA , ISTUFEN or SVT
-connection <config file>	Absolute path to configuration file with the connecti
-in <source file>	Absolute path for "FA-file", "text file with I-step" or "

### Example:

```
E-Sys.bat -writeVcmMaster FA -connection C:\conf\connection.properties -in C:\Data\FA\FA_Y120BX.xml
```

```
E-Sys.bat -writeVcmMaster ISTUFEN -in C:\Data\istufen_Y120BX.txt
```

```
E-Sys.bat -writeVcmMaster SVTSOLL -connection C:\conf\connection.properties -in C:\Data\SVT\SVT_SOLL_Y120BX.xml
```

### Information:

- The I-Steps are to be passed in the "Text-file with I-Steps" in the following format:
  - <Current I-Step>
  - <Last I-Step>
  - <Auslieferungs-I-Step>
- The I-Steps are specified in the usual format, e.g. S15A-15-07-500
- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 3 - Unable to connect to the server.
  - 4 - The action was performed without errors but with warnings.

## Write VCM-Backup (FA or I-Step)

### Command in client batch mode:

```
E-Sys.bat -writeVcmBackup <FA|ISTUFEN> -connection <config file> -in <source file>
```

### Command in client-server batch mode:

```
E-Sys.bat -server -writeVcmBackup <FA|ISTUFEN> [-connection <config file>] -in <source file>
```

### Declaration:

-writeVcmBackup <FA ISTUFEN>	Parameters that can take the value FA or ISTUFEN.
-connection <config file>	Absolute path to configuration file with the connecti
-in <source file>	

Absolute path for "FA file" or "text file with I-Steps".

**Example:**

E-Sys.bat -writeVcmBackup FA -connection C:\conf\connection.properties -in C:\Data\FA\FA\_Y120BX.xml  
E-Sys.bat -writeVcmBackup ISTUFEN -connection C:\conf\connection.properties -in C:\Data\istufen\_Y120BX.txt

**Information:**

- The I-Steps are to be passed in the "Text-file with I-Steps" in the following format:  
    <Current I-Step>  
    <Last I-Step>  
    <Auslieferungs-I-Step>
- The I-Steps are specified in the usual format, e.g. S15A-15-07-500
- The following codes are returned:  
    0 - The action was successful.  
    1 - An exception occurred so the action could not be completed.  
    3 - Unable to connect to the server.

**Read VCM-Master (FA, FP, I-Steps or SVT-Target)**

**Command in client batch mode:**

E-Sys.bat -readVcmMaster <FA|FP|ISTUFEN|SVTSOLL> -connection <config file> -out <target directory>

**Command in client-server batch mode:**

E-Sys.bat -server -readVcmMaster <FA|FP|ISTUFEN|SVTSOLL> [-connection <config file>] -out <target directory>

**Declaration:**

-readVcmMaster <FA FP ISTUFEN SVTSOLL>	Parameters that can take the value FA, FP , ISTUFEN or
-connection <config file>	Absolute path to configuration file with the connecti
-out <target directory>	Absolute path for FA, FP, I-Steps- or SVT-Target-file w

**Example:**

E-Sys.bat -readVcmMaster FA -connection C:\conf\connection.properties -out C:\Data\FA  
E-Sys.bat -readVcmMaster FA -connection C:\conf\connection.properties -out C:\Data\FA\_123.xml  
or  
E-Sys.bat -readVcmMaster FP -connection C:\conf\connection.properties -out C:\Data\FP  
E-Sys.bat -readVcmMaster FP -connection C:\conf\connection.properties -out C:\Data\FP\FP\_123.xml  
or  
E-Sys.bat -readVcmMaster ISTUFEN -connection C:\conf\connection.properties -out C:\Data  
E-Sys.bat -readVcmMaster ISTUFEN -connection C:\conf\connection.properties -out C:\Data\IS\_123.txt  
or  
E-Sys.bat -readVcmMaster SVTSOLL -connection C:\conf\connection.properties -out C:\Data\SVT -connection  
E-Sys.bat -readVcmMaster SVTSOLL -connection C:\conf\connection.properties -out C:\Data\SVT\SVTSOLL\_123.xml

**Information:**

- FA.....If no file name is specified, the selected FA with the following file name: F
  - FP.....If no file name is specified, the selected FP with the following file name: F
  - I-Stufe.....If no file name is specified, the destination directory contains the i-levels rea
  - SVT-Target.....If no file name is specified, the selected SVTSOLL with the following file name directory.
- The following codes are returned:

- 0 - The action was successful.
- 1 - An exception occurred so the action could not be completed.
- 3 - Unable to connect to the server.

**Read VCM-Backup (FA or I-Step)**

Command in client batch mode:

E-Sys.bat -readVcmBackup <FA|ISTUFEN> -connection <config file> -out <target directory>

Command in client-server batch mode:

E-Sys.bat -server -readVcmBackup <FA|ISTUFEN> [-connection <config file>] -out <target directory>

Declaration:

-readVcmBackup <FA ISTUFEN>.....	Parameters that can take the value FA or ISTUFEN.
-connection <config file>.....	Absolute path to configuration file with the connect
-out <target directory>.....	Absolute path for FA or I-Step file with or without fil

Example:

E-Sys.bat -readVcmBackup FA -connection C:\conf\connection.properties -out C:\Data\FA  
 E-Sys.bat -readVcmBackup FA -connection C:\conf\connection.properties -out C:\Data\FA\FA\_123.xml  
 or  
 E-Sys.bat -readVcmBackup ISTUFEN -connection C:\conf\connection.properties -out C:\Data  
 E-Sys.bat -readVcmBackup ISTUFEN -connection C:\conf\connection.properties -out C:\Data\IS\_123.txt

Information:

- FA. ....If no file name is specified, the selected FA with the following file name: FA\_
- I-Stufe. ....If no file name is specified, the destination directory contains the read I levels
- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 3 - Unable to connect to the server.

**Read VIN from VCM-Master**

Command in client batch mode:

E-Sys.bat -readVinFromMaster -connection <config file>

Command in client-server batch mode:

E-Sys.bat -server -readVinFromMaster [-connection <config file>]

Declaration:

-readVinFromMaster	command.
-connection <config file>	Absolute path to configuration file with the connecti

Example:

E-Sys.bat -readVinFromMaster -connection C:\conf\connection.properties

Information:

- VIN is output to stdout
- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 3 - Unable to connect to the server.

### **Read VIN from VCM-Backup**

#### Command in client batch mode:

E-Sys.bat -readVinFromBackup -connection <config file>

#### Command in client-server batch mode:

E-Sys.bat -server -readVinFromBackup [-connection <config file>]

#### Declaration:

-readVinFromBackup	command.
-connection <config file>	Absolute path to configuration file with the connecti

#### Example:

E-Sys.bat -readVinFromBackup -connection C:\conf\connection.properties

#### Information:

- VIN is output to stdout
- The following codes are returned:
  - 0 - The processing was completed successfully.
  - 1 - Errors have occurred.
  - 3 - Unable to connect to the server.

### **Read Versionsinfos from VCM-Master**

#### Command in client batch mode:

E-Sys.bat -readVcmVersionsinfoMaster -connection <config file>

#### Command in client-server batch mode:

E-Sys.bat -server -readVcmVersionsinfoMaster [-connection <config file>]

#### Declaration:

-readVcmVersionsinfoMaster	command.
-connection <config file>	Absolute path to configuration file with the connecti

#### Example:

E-Sys.bat -readVcmVersionsinfoMaster -connection C:\conf\connection.properties

#### Information:

- VCM Versions are output to stdout
  - Version VCM Spec

Version FA  
Version FP  
Version VPC  
Version I-Stufen-Triple  
Version SVT-Soll

- The following codes are returned:
  - 0 - The processing was completed successfully.
  - 1 - Errors have occurred.
  - 3 - Unable to connect to the server.

---

**This help file was created with an unregistered evaluation copy of Help & will not appear if you compile this help file with the registered version of**



**General:**

For Certificate Management, batch offers the following commands.

- [ ].....optional, can be specified, but does not have to be
- ( ).....Selection, must be specified
- | ..... "or", is used in a selection

**Read Certificate**

With the following batch command, it is possible to read from the ECUs according to the con provided.

Attention!

Not all ECUs support reading of all container types.

Command in client batch mode:

E-Sys.bat -readCERT -connection <config file> -type <Certificate type> [-svt <SVT file>] [-blacklist <file>] [-whit

Command in client-server batch mode:

E-Sys.bat -server -readCERT -connection <config file> -type <Certificate type> [-svt <SVT file>] [-blacklist <file>] [-w

Declaration:

-readCERT	command.
-connection <config file>	Absolute path to configuration file with the connection parameter ( <a href="#">Connection Parameter</a> ).
-type <Certificate type>	Type of Certificate to read: CERTIFICATE, BINDING , ONLINE_CERTIFICATES, SEC_OC_KEYLIST.
-svt <SVT file>	Absolute path to SVT-file.
-blacklist <file>	Absolute path to Blacklist-file.
-whitelist <file>	Absolute path to Whitelist-file.

Example:

```
E-Sys.bat -readCERT -connection C:\conf\connection.properties -type CERTIFICATE -blacklist C:\Daten\blacklist.txt
E-Sys.bat -readCERT -connection C:\conf\connection.properties -type BINDING -svt C:\SVT\SVT_I20.xml -whitelist C:\Daten\w
E-Sys.bat -readCERT -connection C:\conf\connection.properties -type ONLINE_CERTIFICATES_ECU -svt C:\SVT\SVT_I20.xml -l
C:\Daten\blacklist.txt -whitelist C:\Daten\whitelist.txt
E-Sys.bat -readCERT -connection C:\conf\connection.properties -type SEC_OC_KEYLIST
```

Information:

- The following codes are returned: 0 - The action was successful.
- 1 - An exception occurred so the action could not be completed.
- 2 - The action was performed with errors.
- 3 - Unable to connect to the server.
- 4 - The action was performed without errors but with warnings.

**Generating a Certificate request**

This command generates a request file to the CBB.

This can then be passed on to the corresponding process (web interface, mail).

Command in client batch mode:

E-Sys.bat -generateCSR -connection <config file> -out <target file> [-noSecOKeys] [-vin <VIN17>] [-blacklist <file>]

**Command in client-server batch mode:**

E-Sys.bat -server -generateCSR -connection <config file> -out <target file> [-noSecOKeys] [-vin <VIN17>] [-blacklist <file>]

**Declaration:**

-generateCSR	command.
-connection <config file>	Absolute path to configuration file with the connection parameter ( <a href="#">Connection Parameter</a> ).
-out <target file>	Absolute path to the target file.
-noSecOKeys	Disable Restbus Keys. Do not add a virtual ECU to the CBB request.
-vin <VIN17>	17-digit VIN.
-blacklist <file>	Absolute path to Blacklist-file.
-whitelist <file>	Absolute path to Whitelist-file.

**Example:**

E-Sys.bat -generateCSR -connection C:\conf\connection.properties -out C:\Data\CERT\requestCBB[JSON].txt

**Information:**

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 2 - The action was performed with errors.
  - 3 - Unable to connect to the server.
  - 4 - The action was performed without errors but with warnings.

**Send a Certificate request to the CBB and save the response (online)**

This command sends a request file to the CBB. The obtained result is stored in a CBB response file. This command can only be used in the BMW environment, as Certificates are obtained from servers.

**Command in client batch mode:**

E-Sys.bat -getCbbResponseFromRequest <cbb request file> [-out <target directory>]

**Command in client-server batch mode:**

E-Sys.bat -server -getCbbResponseFromRequest <cbb request file> [-out <target directory>]

**Declaration:**

-getCbbResponseFromRequest <cbb request file>	Absolute path to CBB-Request-file.
-out <target directory>	Absolute path to the target file. If omitted, the default path is the current directory. CBB response files.

**Example:**

E-Sys.bat -getCbbResponseFromRequest C:\Daten\WBAAE810X0H123456\_CBBRequest.txt -out C:\cert\cert.txt

**Information:**

- The following codes are returned:
  - 0 - The action was successful.

- 1 - An exception occurred so the action could not be completed.
- 2 - The action was performed with errors.
- 3 - Unable to connect to the server.

### **Writing the bindings from file**

This command writes all bindings, Certificates, and bindings from the specified response file to the connected vehicle.

The "Other Bindings" are also generated and also written into the vehicle.

The `-secOckKeys` option also displays a file for residual bus simulation.

### **Command in client batch mode:**

```
E-Sys.bat -writeBindings -connection <config file> -in <CBB response file> [-noSecOckKeys] [-svt <SVT file>] [-secOckKeyPack file>]
```

### **Command in client-server batch mode:**

```
E-Sys.bat -server -writeBindings -connection <config file> -in <CBB response file> [-noSecOckKeys] [-svt <SVT file>] [-secOckKeyPack file>]
```

### **Declaration:**

<code>-writeBindings</code>	command.
<code>-connection &lt;config file&gt;</code>	Absolute path to configuration file with the connection param <a href="#">Connection Parameter</a> ).
<code>-in &lt;CBB response file&gt;</code>	Absolute path to file.
<code>-noSecOckKeys</code>	Disable Restbus Keys. Do not add a virtual ECU to the CBB rec
<code>-svt &lt;SVT file&gt;</code>	Absolute path to SVT-file.
<code>-secOckKeysPath &lt;SecOckKeyPack file&gt;</code>	Absolute path to the target file with symmetric keys. Only cor without the <code>secOckKeys</code> parameter.

### **Example:**

```
E-Sys.bat -writeBindings -connection C:\conf\connection.properties -in C:\Daten\WBAAE810X0H123456_CBBResponse_20210331_125817[JSON].txt" -secOckKeysPath C:\Data\CERT\Keys\20210102_131515_SecOC_KeyPack_plain.xml
```

### **Information:**

- If the parameter `-secOckKeysPath <SecOckKeyPack target file>` is omitted, the default path is used
- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 2 - The action was performed with errors.
  - 3 - Unable to connect to the server.
  - 4 - The action was performed without errors but with warnings.

### **Checking the Certificates on the Certificate-enabled ECUs**

This command can be used to initiate a Certificate check in the ECU.

After the internal inspection is complete, E-Sys waits for the longest returned inspection time. After that, the results are requested and written to result files.

The status is read out in parallel for several ECUs, taking into account the topology.

### Command in client batch mode:

E-Sys.bat -checkCERT -connection <config file> -retries <value> [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]

### Command in client-server batch mode:

E-Sys.bat -server -checkCERT -connection <config file> -retries <value> [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]

### Declaration:

-checkCERT	command.
-connection <config file>	Absolute path to configuration file with the connection parameter ( <a href="#">Connection Parameter</a> ).
-retries <value>	Number of performed scans (for 1 to a maximum of 10). example
-svt <SVT file>	Absolute path to SVT-file.
-blacklist <file>	Absolute path to Blacklist-file.
-whitelist <file>	Absolute path to Whitelist-file.

### Example:

E-Sys.bat -checkCERT -connection C:\conf\connection.properties -retries 3

E-Sys.bat -checkCERT -connection C:\conf\connection.properties -retries 3 -blacklist C:\Daten\blacklist.txt

E-Sys.bat -checkCERT -connection C:\conf\connection.properties -retries 3 -blacklist C:\Daten\blacklist.txt -whitelist C:\Daten\whitelist.txt

### Information:

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 2 - The action was performed with errors.
  - 3 - Unable to connect to the server.
  - 4 - The action was performed without errors but with warnings.

### **Checking the persistent certificates stored in the certificate-enabled ECUs**

This command performs a check of persistently stored certificates and keys, if available.

The results are displayed in the console and written to a file.

Two types of checks can be performed:

- Quick: a fast/quick check
- Full: a complete check of certificates, including the certificate signatures

By default, a quick check is performed. To perform a complete check, the option "-full" should be used.

### Command in client batch mode:

E-Sys.bat -checkPersistentCert [-full] -connection <config file> [-retries <value>] [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]

### Command in client-server batch mode:

E-Sys.bat -server -checkPersistentCert [-full] -connection <config file> [-retries <value>] [-svt <SVT file>] [-blacklist <file>] [-whitelist <file>]

### Declaration:

-checkPersistentCert	command.
-full	To perform a complete check of certificates.
-connection <config file>	Absolute path to configuration file with the connection parameter ( <a href="#">Connection Parameter</a> ).
-retries <value>	Number of scans to perform (range: 1 to 10). If not specified

	value (i.e., 1) will be used.
-svt <SVT file>	Absolute path to the SVT file.
-blacklist <file>	Absolute path to the blacklist file.
-whitelist <file>	Absolute path to the whitelist file.

-  
**Example:**

```
E-Sys.bat -checkPersistentCert -connection C:\conf\connection.properties
E-Sys.bat -checkPersistentCert -full -connection C:\conf\connection.properties
E-Sys.bat -checkPersistentCert -full -connection C:\conf\connection.properties -retries 3 -blacklist C:\Data\blacklist.txt
E-Sys.bat -checkPersistentCert -connection C:\conf\connection.properties -retries 3 -blacklist C:\Data\blacklist.txt -whitelist
```

**Information:**

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 2 - The action was performed with errors.
  - 3 - Unable to connect to the server.
  - 4 - The action was performed without errors but with warnings.

**Automated Certificate execution with configuration-file**

The following command performs an automated Certificate execution. In addition, a file can residual bus simulation.

This command can only be used in the BMW environment, as Certificates are obtained from servers to transfer them to the connected vehicle.

**Command in client batch mode:**

```
E-Sys.bat -certexecution <config file>
```

**Command in client-server batch mode:**

```
E-Sys.bat -server -certexecution <config file>
```

**Declaration:**

-certexecution <config file>	Absolute path to configuration file.
------------------------------	--------------------------------------

**Example:**

```
E-Sys.bat -certexecution C:\Cert\CertExecution.config
```

**Information:**

- The specific certificate action is defined in the configuration file along with the [Connection](#) and the following parameters:

Parameter	Example value	Mandatory parameters?	Comments
SVT	C:/Data/SVT/SVT-Target.xml	yes	SVT Target
VIN	WBA12345671234567	yes	VIN17
CERT_BLACKLIST	C:/Data/etc/blacklist.txt	no	Indicates ECUs that handled
CERT_WHITELIST	C:/Data/etc/whitelist.txt	no	Specifies ECUs to be handled exclusively
CERT_SERVER_URL_LIST	https://www.maxmustermann.de/; <optional-Port-No.>/<Server-path>;	yes	List of Backend-Servers

	https://www.maxmustermann.com); <optional-Port-No.>/<Server-path>;		
CERT_RETRIES	5	no	Number of connecti to BMW-Backend; default: 1; maximu
CERT_TIME_BETWEEN_RETRIES	30000	no	Duration connectio BMW-Backend: default: 20000; ma. 60000
NO_SECOC_KEYS	off	no	Ability residual bus Possible values: .on
SECOC_KEYS_PATH	C:\SecOC_KeyPack_plain.xml	no	Absolute path incl the target file w keys. Only considered for If owithted, the d used. Example: C:-Data-CERT-Keys- 20210102_131515_SecOC
PROJECT	S18A_...	yes	Project name in psc directory
VEHICLEINFO	S18A	yes	Baureihenverbund
CONNECTION	vin	yes	valid values: bus   gateway_url   icom, icom_ethernet   vin
GATEWAY_VIN	WBA12345671234567_DIAGADR10	ja, if CONNECTION=vin	valid values: <VIN17>_DIAGADR<Gate   AUTODETECT

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 2 - The action was performed with errors.
  - 3 - Unable to connect to the server.
  - 4 - The action was performed without errors but with warnings.

### **Setting the Freshness Value of a specific SecOC ECU**

This command can be used to write a new SecOC Counter value to the ECU. The affected ECU Engineering-Mode.

Either the black/white mechanism may be used as a parameter (blacklist and/or whitelist) or diag. If neither diagaddress nor blacklist/whitelist is passed, the new SecOc Counter value is set for all SecOC ECUs.

#### **Command in client batch mode:**

```
E-Sys.bat -setSecOCCounter -connection <config file> -value <hex value> { [-diagaddress <diagnosis address>] <file> } [-whitelist <file>] }
```

#### **Command in client-server batch mode:**

```
E-Sys.bat -server -setSecOCCounter -connection <config file> -value <hex value> { [-diagaddress <diagnosis address>] <file> } [-whitelist <file>] }
```

### Declaration:

-setSecOCCounter	command.
-connection <config file>	Absolute path to configuration file with the connection param <a href="#">Connection Parameter</a> ).
-value <hex value>	Counter-Value as HEX in range of 0x0 to 0xFFFFFFFF.
-diagaddress <diagnosis address>	The diagnostic address of the ECU.
-blacklist <file>	Absolute path to Blacklist-file.
-whitelist <file>	Absolute path to Whitelist-file.

### Example:

```
E-Sys.bat -setSecOCCounter -connection C:\conf\connection.properties -value 0x01 -diagaddress 0x33
E-Sys.bat -setSecOCCounter -connection C:\conf\connection.properties -value 0x01 -blacklist C:\Daten\blacklist.txt -whitelis
C:\Daten\whitelist.txt
```

### Information:

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 3 - Unable to connect to the server.
  - 4 - The action was performed without errors but with warnings.

### **Getting the Freshness Value of a specific SecOC ECU**

This command can be used to get the current SecOC Counter value from an ECU.  
Either the black/white mechanism may be used as a parameter (blacklist and/or whitelist) or diag  
If neither diagaddress nor blacklist/whitelist is passed, the new SecOc Counter value is set for all Se  
ECUs.

### Command in client batch mode:

```
E-Sys.bat -getSecOCCounter -connection <config file> -value <hex value> { [-diagaddress <diagnosis address>]
<file>] [-whitelist <file>] } }
```

### Command in client-server batch mode:

```
E-Sys.bat -server -getSecOCCounter -connection <config file> -value <hex value> { [-diagaddress <diagnosis address>]
<file>] [-whitelist <file>] } }
```

### Declaration:

-getSecOCCounter	command.
-connection <config file>	Absolute path to configuration file with the connection param <a href="#">Connection Parameter</a> ).
-diagaddress <diagnosis address>	The diagnostic address of the ECU.
-blacklist <file>	Absolute path to Blacklist-file.
-whitelist <file>	Absolute path to Whitelist-file.

### Example:

```
E-Sys.bat -getSecOCCounter -connection C:\conf\connection.properties -diagaddress 0x33
E-Sys.bat -getSecOCCounter -connection C:\conf\connection.properties -blacklist C:\Daten\blacklist.txt -whitelist C:\Daten\w
```

### Information:

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.

- 3 - Unable to connect to the server.
- 4 - The action was performed without errors but with warnings.

### **Setting the Freshness Value of the SecOC Master ECU**

This command can be used to set the SecOC Master Counter value of the SecOC Master ECU. The SecOC Master ECU will be determined automatically. The affected ECU must be in Engine mode.

#### Command in client batch mode:

```
E-Sys.bat -setSecOCMasterCounter -connection <config file> -value <hex value>
```

#### Command in client-server batch mode:

```
E-Sys.bat -server -setSecOCMasterCounter -connection <config file> -value <hex value>
```

#### Declaration:

-setSecOCMasterCounter	command.
-connection <config file>	Absolute path to configuration file with the connection parameter ( <a href="#">Connection Parameter</a> ).
-value	SecOC Master Counter-Value as HEX in range of 0x0 to 0xFF

#### Example:

```
E-Sys.bat -setSecOCMasterCounter -connection C:\conf\connection.properties -value 0x01
```

#### Information:

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 3 - Unable to connect to the server.
  - 4 - The action was performed without errors but with warnings.

### **Getting the Freshness Value of the SecOC Master ECU**

This command can be used to get the SecOC Master Counter value from the SecOC Master ECU. The SecOC Master ECU will be determined automatically.

#### Command in client batch mode:

```
E-Sys.bat -getSecOCMasterCounter -connection <config file>
```

#### Command in client-server batch mode:

```
E-Sys.bat -server -getSecOCMasterCounter -connection <config file>
```

#### Declaration:

-getSecOCMasterCounter	command.
-connection <config file>	Absolute path to configuration file with the connection parameter ( <a href="#">Connection Parameter</a> ).

#### Example:

```
E-Sys.bat -getSecOCMasterCounter -connection C:\conf\connection.properties
```

#### Information:

- The following codes are returned:
  - 0 - The action was successful.



- 1 - An exception occurred so the action could not be completed.
- 3 - Unable to connect to the server.
- 4 - The action was performed without errors but with warnings.

### **Activation of IPsec on all IPsec-enabled ECUs**

This command enables IPsec to be enabled for all ECUs that support this functionality

#### Command in client batch mode:

E-Sys.bat -activateIPsec -connection <config file>

#### Command in client-server batch mode:

E-Sys.bat -server -activateIPsec -connection <config file>

#### Declaration:

-activateIPsec	command.
-connection <config file>	Absolute path to configuration file with the connection parameter ( <a href="#">Connection Parameter</a> ).

#### Example:

E-Sys.bat -activateIPsec -connection C:\conf\connection.properties

#### Information:

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 2 - The action was performed with errors.
  - 3 - Unable to connect to the server.
  - 4 - The action was performed without errors but with warnings.

### **Deactivate IPsec on all IPsec-enabled ECUs**

This command deactivates IPsec for all ECUs that support this functionality

#### Command in client batch mode:

E-Sys.bat -deactivateIPsec -connection <config file>

#### Command in client-server batch mode:

E-Sys.bat -server -deactivateIPsec -connection <config file>

#### Declaration:

-deactivateIPsec	command.
-connection <config file>	Absolute path to configuration file with the connection parameter ( <a href="#">Connection Parameter</a> ).

#### Example:

E-Sys.bat -deactivateIPsec -connection C:\conf\connection.properties

#### Information:

- The following codes are returned:

- 0 - The action was successful.
- 1 - An exception occurred so the action could not be completed.
- 2 - The action was performed with errors.
- 3 - Unable to connect to the server.
- 4 - The action was performed without errors but with warnings.

### **Lock IPsec on all IPsec-enabled ECUs**

This command can be used to lock IPsec to all ECUs that support this functionality

#### Command in client batch mode:

E-Sys.bat        -lockIPsec -connection <config file>

#### Command in client-server batch mode:

E-Sys.bat -server -lockIPsec -connection <config file>

#### Declaration:

-lockIPsec	command.
-connection <config file>	Absolute path to configuration file with the connection parameter ( <a href="#">Connection Parameter</a> ).

#### Example:

E-Sys.bat -lockIPsec -connection C:\conf\connection.properties

#### Information:

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 2 - The action was performed with errors.
  - 3 - Unable to connect to the server.
  - 4 - The action was performed without errors but with warnings.

### **Read the status of all IPsec-enabled ECUs**

This command reads the status of all IPsec-capable ECUs.

#### Command in client batch mode:

E-Sys.bat        -readIPsecStatus -connection <config file>

#### Command in client-server batch mode:

E-Sys.bat -server -readIPsecStatus -connection <config file>

#### Declaration:

-readIPsecStatus	command.
-connection <config file>	Absolute path to configuration file with the connection parameter ( <a href="#">Connection Parameter</a> ).

#### Example:

E-Sys.bat -readIPsecStatus -connection C:\conf\connection.properties

#### Information:

- The following codes are returned:

- 0 - The action was successful.
- 1 - An exception occurred so the action could not be completed.
- 2 - The action was performed with errors.
- 3 - Unable to connect to the server.
- 4 - The action was performed without errors but with warnings.

### **Activation of MACsec on all MACsec-enabled ECUs**

This command enables MACsec to be enabled for all ECUs that support this functionality

#### Command in client batch mode:

E-Sys.bat -activateMACsec -connection <config file>

#### Command in client-server batch mode:

E-Sys.bat -server -activateMACsec -connection <config file>

#### Declaration:

-activateMACsec	command.
-connection <config file>	Absolute path to configuration file with the connection param <a href="#">Connection Parameter</a> ).

#### Example:

E-Sys.bat -activateMACsec -connection C:\conf\connection.properties

#### Information:

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 3 - Unable to connect to the server.
  - 4 - The action was performed without errors but with warnings.

### **Deactivate MACsec on all MACsec-enabled ECUs**

This command deactivates MACsec for all ECUs that support this functionality

#### Command in client batch mode:

E-Sys.bat -deactivateMACsec -connection <config file>

#### Command in client-server batch mode:

E-Sys.bat -server -deactivateMACsec -connection <config file>

#### Declaration:

-deactivateMACsec	command.
-connection <config file>	Absolute path to configuration file with the connection param <a href="#">Connection Parameter</a> ).

#### Example:

E-Sys.bat -deactivateMACsec -connection C:\conf\connection.properties

#### Information:

- The following codes are returned:
  - 0 - The action was successful.

- 1 - An exception occurred so the action could not be completed.
- 3 - Unable to connect to the server.
- 4 - The action was performed without errors but with warnings.

### **Lock MACsec on all MACsec-enabled ECUs**

This command can be used to lock MACsec to all ECUs that support this functionality

#### Command in client batch mode:

E-Sys.bat -lockMACsec -connection <config file>

#### Command in client-server batch mode:

E-Sys.bat -server -lockMACsec -connection <config file>

#### Declaration:

-lockMACsec	command.
-connection <config file>	Absolute path to configuration file with the connection parameter ( <a href="#">Connection Parameter</a> ).

#### Example:

E-Sys.bat -lockMACsec -connection C:\conf\connection.properties

#### Information:

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 3 - Unable to connect to the server.
  - 4 - The action was performed without errors but with warnings.

### **Add TLS certificate to the E-Sys TLS Key store**

This command is used to add TLS certificates and the respective key to the E-Sys Key store

#### Command in client batch mode:

E-Sys.bat -addTLSCertificate -certificateFile <certificate file> -keyFile <key file>

#### Declaration:

-addTLSCertificate	command.
-certificateFile <config file>	Absolute path to certificate file. The certificate file name needs to contain the VIN.
-keyFile <key file>	Absolute path to the file containing the private key. The file name needs to contain the vehicle VIN and be the same VIN as the certificate.

#### Example:

E-Sys.bat -addTLSCertificate -certificateFile C:\cert\Service29-external\_WBA0000100G08B325\_20230817084756.crt -keyFile external\_WBA0000100G08B325\_20230817084756.key

#### Information:

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 4 - The action was performed without errors but with warnings.

### **Get TLS certificates from the E-Sys TLS Key store**

This command is used to print all TLS certificates' VIN and expiration date to the console.

#### Command in client batch mode:

E-Sys.bat      -getTLSCertificates

#### Declaration:

-getTLSCertificates	command.
---------------------	----------

-

#### Example:

E-Sys.bat -getTLSCertificates

#### Information:

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 4 - The action was performed without errors but with warnings.

### **Delete Expired TLS certificates from the E-Sys TLS Key store**

This command is used to delete all expired certificates from the TLS Key Store

#### Command in client batch mode:

E-Sys.bat      -deleteExpiredTLSCertificate

#### Declaration:

-deleteExpiredTLSCertificate	command.
------------------------------	----------

-

#### Example:

E-Sys.bat -deleteExpiredTLSCertificate

#### Information:

- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 4 - The action was performed without errors but with warnings.

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This r appear if you compile this help file with the registered version of Help & Manual.

## General:

In order to establish a connection, the following parameters can be defined in a batch configuration file:

```
PROJECT = <project name>
VEHICLEINFO = <vehicle info>

# bus|gateway_url|icom_d_can|icom_ethernet|vin
CONNECTION = <connection type>
```

The parameter `Connection` is optional. Without this parameter the information's are read from file `E-Sys.properties`. If the parameter is defined, the batch configuration file has to consist the following parameter depending on the connection type:

### Connection type `bus`:

```
# z.B. BODY_CAN, FLEXRAY, SYSTEMBUS_ETHERNET, A_CAN, D_CAN, ...
BUS_NAME = <bus name>

# VECTOR_DIRECT | VECTOR_DIRECT_FLEXRAY | PASS_THRU |
MEDIA_CONVERTER
INTERFACE = <interface>

# If BUS_NAME=SYSTEMBUS_ETHERNET and
INTERFACE=MEDIA_CONVERTER, it is necessary to set the parameter URL
# Format <protocoll>://<IP address>:<port>
URL = <URL matching the format above>
```

### Connection type `gateway_url`, `icom_d_can`, `icom_ethernet`:

```
# Format <protocoll>://<IP address>:<port>
URL = <URL matching the format above>
```

### Connection type `vin`:

```
# Format <VIN 17-digit>_DIAGADR<gateway diagnostic address
hex>|AUTODETECT
GATEWAY_VIN = <VIN matching the format above>
```

If AUTODETECT is used, E-Sys is creating a list with available VINs with the following behaviour:

List = 1 VIN: this VIN is used for connection

List = 0 VINs: "No available vehicles were found"

List > 1VINs: "More than one available vehicle found: VIN1, VIN2, ..."

The following vehicle-specific parameters should be defined.

# on|off

READ\_VEHICLE\_CONNECTION\_PARAMETER

# used if READ\_VEHICLE\_CONNECTION\_PARAMETER = off

VEHICLE\_CONNECTION\_PARAMETER\_SERIES = <series>

# used if READ\_VEHICLE\_CONNECTION\_PARAMETER = off

VEHICLE\_CONNECTION\_PARAMETER\_ISTEP\_SHIPMENT = <l-step>

For the value `on` of parameter

`READ_VEHICLE_CONNECTION_PARAMETER`, the series and l-step will be read from VCM.

Otherwise the parameters `VEHICLE_CONNECTION_PARAMETER_SERIES` and `VEHICLE_CONNECTION_PARAMETER_ISTEP_SHIPMENT` will be read. If any of these parameters are left undefined, a warning message will be written to the log file. Despite the warning, the connection will attempt to be established.

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

### General:

For some Connections settings, batch offers the following commands.

[ ].....optional, can be specified, but does not have to be

( ).....Selection, must be specified

| ..... "or", is used in a selection

-

### **Print the Maximum Http Throughput**

With this command, the current Maximum Http Bandwidth Throughput in bytes per second that is allowed to the Http server will be printed on the console.

#### Command in client batch mode:

E-Sys.bat        -printHttpBandwidthLimit

#### Command in client-server batch mode:

E-Sys.bat -server -printHttpBandwidthLimit

#### Example:

E-Sys.bat -printHttpBandwidthLimit

#### Information:

- The value is shown as bytes per second or UNLIMITED.
- The following codes are returned: 0 - The action was successful.  
1 - An exception occurred so the action could not be completed.



- 3 - Unable to connect to the server.
- 4 - The action was performed without errors but with warnings.

### **Set the Maximum Http Throughput:**

With this command, it is possible to set the current Maximum Http Bandwidth Throughput in Mbit per second. The input field can either be empty (corresponding to : Unlimited) or a value above 0,008192 Mbit/s (1024 Byte/s). Floating point numbers are allowed.

#### Command in client batch mode:

E-Sys.bat -setHttpBandwidthLimit <http bandwidth max throughput value (mbps)>

#### Command in client-server batch mode:

E-Sys.bat -server -setHttpBandwidthLimit <http bandwidth max throughput value (mbps)>

#### Declaration:

<pre>-setHttpBandwidthLimit &lt;http bandwidth max throughput value (mbps)&gt;</pre>	<p>Value in Mbit per second of the wanted Maximum Http Bandwidth Throughput. Values must be above 0,008192 Mbit/s (1024 Byte/s). If value is empty then it will be set to Unlimited. Floating point numbers are allowed</p>
--	---

#### Example:

```
E-Sys.bat -setHttpBandwidthLimit 25
E-Sys.bat -setHttpBandwidthLimit 0,008192
E-Sys.bat -setHttpBandwidthLimit
```

### Information:

- If the values are set below the minimum (0,008192 Mbit/s), an error message will appear.
- In order to choose an Unlimited Maximum Http Throughput, the parameter must be empty.
  
- The following codes are returned:
  - 0 - The action was successful.
  - 1 - An exception occurred so the action could not be completed.
  - 3 - Unable to connect to the server.
  - 4 - The action was performed without errors but with warnings.

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

## Abbreviations

[Top](#) [Previous](#) [Next](#)

ASAM	Association for Standardization of Automation and Measuring Systems
AVAKON	Automatic Configuration of Variants
BAF	Boot Sector Exchange Format
BT	Description Table
CAF	Coding Application File
CBD	Coding Description File
COS	Central Operating Services
CVN	Calibration Verification Number
DAF	Data Exchange Format
ECCO-NF	Electric / Electronic Change- & Configuration-Management-Successor
E-Sys	New programming system for department E
FA	Vehicle Order
FFS	Flash File System
FP	Vehicle Programming
FP-Prozess	Base System Process of Vehicle Programming
FSC	Activation Code
FWL	Function value list
FZM	Vehicle State Management
GUI	Graphical User Interface
HO	Dealership Network
IEX	Integration Test Vehicle (new: EBG Development Component)
Intel-Hex	Data Format Intel-Hex
KIS	Compatibility and Information

	System
KMM	Configuration Management Module
LabCar	Labor Vehicle
LAN	Local Area Network
LBT	Logical Block Table
MCD	Measurement, Calibration and Diagnosis
MCD-2D (ODX)	An ASAM standard, that comprises the exchange format of ECU data and ECU descriptions in conjunction with diagnosis tasks.
MMI	Man Machine Interface
NCD	Netto coding values (Nettocodierdaten)
OBD	On-Board Diagnostics
OffBPS	Offboard Programming System
OnBPS	Onboard Programming System
PAF	Program Exchange Format
PDM	Product Data Management
PDX	Packed ODX
PE	Programming Unity
PEP	Product Manufacturing Process
PIA	Personalization, Individualization, Adoption
PPP	Product-Process-Prototype (new: BBG Actuation Component)
PSdZ	Programming System of the Future
PV	Patch Version
SA	Special Equipment
SALAPA	Special Equipment, Country Equipment, Package
SG	ECU (Electronic Control Unit)
SG-ID	ECU ID

SGBM	Control Unit Description Model
SLdZ	Software Logistic of the Future
SPA	Software Package Administration
SPS	Software Package Storage
SREC	Data Format Motorola SREC
SVK	System Installation Identifier
SVT (Ist)	Actual System Installation Table
SVT (Soll)	Reference System Installation Table
SW	Software (process class)
SWE	Software Unity
SWT	Sweeping Technologies
TAL	Transaction List
TI	Technical Integration
UV	Sub Version
VCM	Vehicle Configuration Management
VIN	Vehicle Identification Number
VS	Sale and Service
WLAN	Wireless Local Area Network
XML	eXtensible Markup Language
XSD	XML Scheme Definition

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

## Link to usefull documentation

[Top](#) [Previous](#) [Next](#)

Collection of links to more help documents in GIS:

[1. User information on handling of inconsistent PDX containers in E-Sys.](#)

[2. Information on a programming problem via Ethernet with Windows 7 in Zeroconf network environments.](#)

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.

To perform a positive flash/coding-action with the e/e-component WAVE-11 it is necessary to set a new parameter at the registry.

Keep in mind, that **local admin rights** are needed to make changes in the registry.

For the WAVE-11 it is necessary to add the parameter TcpTimeWaitDelay with the value of 30.

The change can be executed automatically with the batch-file "registry-wave11.bat" located in the lib folder (E-Sys\lib\registry-wave11.bat)

---

This help file was created with an unregistered evaluation copy of Help & Manual. © EC Software. All rights reserved. This message will not appear if you compile this help file with the registered version of Help & Manual.