

# MCAL Integration Package

## Technical Reference

Basics and workflows

Version 1.05.01

Authors	Andrej Gazvoda, Günther Piehler, Roland Süß, Ingo Wuttke
Status	Released

## Document Information

### History

Author	Date	Version	Remarks
Roland Süß; Ingo Wuttke	2015-02-27	1.00.00	Initial Ideas, usage as Application Note; Porting to Technical Reference template; adding detailed description about 3rd party tools etc.
Günther Piehler	2015-04-24	1.00.01	Review; small changes to increase understandability; Known Issue for missing config items added → released
Andrej Gazvoda; Roland Süß	2015-06-30	1.00.02	7.3 / 7.4 - Added known issues regarding EB tresos™ tool
Günther Piehler	2015-07-17	1.01.00	2.3 - Introduction of Mixed AUTOSAR use case 3 ff - Extend description of MCAL preparation (prerequisites) 4 - hint about recommended workflow added
Günther Piehler	2015-10-26	1.01.01	3.2.2 / 3.2.3 - parameter corrected 4 – added hint for MCAL Integration video
Günther Piehler	2016-01-26	1.02.00	4.2 - added hint for “round trip” ability general - new CI applied
Günther Piehler	2016-07-05	1.03.00	Page 3 - Added further useful documents as reference
Günther Piehler	2016-09-27	1.04.00	1 – completely new Reference to QuickStart document deleted (replaced within this document); Reference to ScreenCase and ReleaseNote added Hints for AUTOSAR3 <-> AUTOSAR4 differentiation added 3.2.2 – non-interactive mode introduced 3.2.3 – reference to Release Notes added for further info

			6 – completely new 7 – issue for “MCAL and SIP storage location” removed 8 – hint to “generate all” added
Roland Süß	2017-11-09	1.05.00	3 - Adapted chapter: Details regarding 3rd party MCAL Integration Helper 4.2 - Adapted Chapter: "Set User Defined" Action 5.3 - Added chapter 5.5 - Added chapter
Roland Süß	2018-04-27	1.05.01	5.3 - Extended chapter

## Reference Documents

No.	Source	Title	Version
[1]	Vector	Product Information MICROSAR Vector SLP4	1.03.02
[2]	Vector	Catalog – Product Information MICROSAR – Chapter MCAL	V1.3 – 2015-02
[3]	Vector	Application Note “AN-ISC-8-1153_ThirdPartyModules.pdf”	Latest (e.g. 1.0)
[4]	Vector	Application Note “AN-ISC-8-1171_Tresos_LicenseHandling.pdf”	Latest (e.g. 1.00.01)
[5]	Vector	Application Note “AN-ISC-8-1180_MCAL-Integration-Variants.pdf”	Latest (e.g. 0.9)
[6]	Vector	Release Note “ReleaseNotes_3rdPartyMCAL_VectorIntegration.pdf”	As provided within your SIP
[7]	Vector	ScreenCast_McalIntegration_Tresos.pdf	As provided within your SIP
[8]	Vector	Technical Reference “TechnicalReference_3rdParty-MCAL-Integration_<controller>.pdf”	As provided within your SIP

**Caution**

We have configured the programs in accordance with your specifications in the questionnaire. Whereas the programs do support other configurations than the one specified in your questionnaire, Vector's release of the programs delivered to your company is expressly restricted to the configuration you have specified in the questionnaire.

## Contents

<b>1</b>	<b>Purpose of the document.....</b>	<b>8</b>
<b>2</b>	<b>Introduction.....</b>	<b>9</b>
2.1	Responsibility.....	9
2.2	Support requests.....	10
2.3	Mix between AUTOSAR specification versions .....	10
<b>3</b>	<b>First Steps.....</b>	<b>11</b>
3.1	Delivery structure .....	11
3.2	Starting up .....	12
3.2.1	MCAL delivered within Vector SIP .....	12
3.2.2	MCAL not contained within Vector SIP .....	12
3.2.3	MCAL Update needed.....	14
<b>4</b>	<b>Workflow .....</b>	<b>15</b>
4.1	Single configuration tool usage .....	15
4.2	Mixed configuration tool usage.....	16
4.3	Split configuration tool usage .....	19
<b>5</b>	<b>Configuration tools.....</b>	<b>20</b>
5.1	Vector DaVinci Configurator .....	20
5.2	EB tresos™ .....	20
5.2.1	Setting up a new Configuration Project .....	20
5.2.2	Project Details.....	21
5.2.3	Selection of components.....	22
5.2.4	Creation of importers and exporters .....	23
5.2.5	Configure the MCAL components .....	24
5.2.6	Generation of the 3rd party MCAL.....	24
5.3	Cooperation Interface between DaVinci Configurator and EB tresos™ .....	24
5.4	Configuration hints for parallel usage of DaVinci Configurator and EB tresos™ .....	25
5.4.1	Vector DaVinci Configurator 5 .....	26
5.4.2	EB tresos™ .....	26
5.5	DaVinci Configurator Services .....	27
5.5.1	Internal Behavior Editor.....	27
<b>6</b>	<b>Restrictions.....</b>	<b>28</b>
6.1	UNC path usage .....	28
<b>7</b>	<b>Known Issues .....</b>	<b>29</b>

7.1	Long path names .....	29
7.2	Missing configuration items within imported configuration .....	29
7.3	Configuration Export with EB tresos™ Version 13.0.0.....	29
7.4	Error messages regarding CommonPublishedInformation with EB tresos™ ....	30
7.5	Problems due to resorting of configuration elements by EB tresos™ Exporter .....	30
<b>8</b>	<b>Frequently Asked Questions .....</b>	<b>31</b>
<b>9</b>	<b>Glossary and Abbreviations .....</b>	<b>33</b>
9.1	Glossary .....	33
9.2	Abbreviations .....	33
<b>10</b>	<b>Contact.....</b>	<b>34</b>

## Illustrations

Figure 4-1	Configuration workflow – Mixed configuration tool usage.....	17
Figure 4-2	Set User Defined .....	18
Figure 5-1	New Configuration Project .....	21
Figure 5-2	Configuration Project Data.....	22
Figure 5-3	Component Configurations .....	23
Figure 5-4	Create an exporter (step 1).....	23
Figure 5-5	Create an exporter (step 2 – AUTOSAR options).....	23
Figure 5-6	Generate Button .....	24
Figure 5-7	Project Settings - Generator User Settings .....	24
Figure 5-8	Tresos Generator Message .....	25
Figure 5-9	Generation Result View .....	25
Figure 5-10	DEM Path using DaVinci Configurator 5 .....	26
Figure 5-11	Settings within DaVinci Configurator 5 Pro.....	26
Figure 5-12	DEM-Path in the Outline window of tresos™ .....	26
Figure 5-13	DEM-Path within tresos™ .....	26
Figure 5-14	Settings for DEM within tresos™ .....	27
Figure 5-15	Internal Behavior Editor .....	27

## Tables

Table 4-1	Guidance for single configuration tool usage .....	16
Table 4-2	Guidance for mixed configuration tool mode .....	18
Table 4-3	Guidance for split configuration tool usage .....	19

## 1 Purpose of the document

This document supports the user by launching the delivery, setting up a project and serving the tooling- and configuration-based interfaces between the Vector MICROSAR BSW and the 3rd party MCAL.

The term “tooling” means Vector DaVinci Configurator on the one hand and a 3rd party configuration and generation framework (EB tresos™, KPIT ECU Spectrum™ ...) on the other.



## 2 Introduction

The *MICROSAR MCAL Integration Package* covers the integration of a *3rd party MCAL* package into the Vector MICROSAR BSW stack. As part of this, Vector performs a setup of the MCAL provided by the customer or the semiconductor vendor and supplements it with items needed for the integration with the MICROSAR BSW.

Typically, the semiconductor vendor delivers the MCAL with its own configuration and generation tool chain. Vector provides solutions to deal with the dependencies and interfaces on embedded and on configuration level between the AUTOSAR BSW and the MCAL components (with AUTOSAR4.0.3 these configuration level based interfaces are about 10 parameters).

The following use cases and corresponding workflows are explained and illustrated in detail later on:

1. *Single configuration tool usage*: Usage of the Vector DaVinci Configurator as exclusive configuration tool for both the MICROSAR BSW and the 3rd party MCAL.
2. *Mixed configuration tool usage*: Parallel and synchronized usage of the DaVinci Configurator and a 3rd party MCAL configuration tool.
3. *Split configuration tool usage*: Split configuration and code generation using DaVinci Configurator for the configuration/generation of the MICROSAR BSW and the 3rd party tool for configuration/generation of the MCAL.

In principal, all three use cases are supported and the decision which one is used depends on user requirements as well as on the capabilities of the integrated MCAL. Detailed decision guidance and hints will be given in this document.



### Note

Many items described within this document are AUTOSAR release independent. Wherever necessary we will add a hint when there is a deviation within the workflow or any other topic from AUTOSAR4 to AUTOSAR3.

### 2.1 Responsibility

Vector does not take over any responsibility for the usage and functionality of the integrated 3rd party components. The *MCAL integration* into the MICROSAR SIP does not include a full functional test for the 3rd party components. It remains the responsibility of the customer to ensure freedom of defect and shortcomings as well as functional completeness of the 3rd party components. The MCAL vendors usually provide regular issue lists to inform the users about known, fixed and/or new issues.

## 2.2 Support requests

Vector will support you in all aspects of the *MCAL integration* into your MICROSAR SIP. In case of questions or problems please get in touch with your MICROSAR contact (e.g. as defined in the delivery description that is part of your SIP).

Since Vector is not the vendor of the integrated 3rd party components such support requests shall be directed to the 3rd party software vendor as they can provide you with first-hand information and support.

## 2.3 Mix between AUTOSAR specification versions

Sometimes it is necessary that there is a mixture of components released for different AUTOSAR versions. Most common case of mixture is the use of AUTOSAR 3.x related BSW (because OEMs have fixed their platform SW) with AUTOSAR4.0.x related MCALs. This is based on the fact that there are continuously new HW platforms available through the semiconductor vendors but not all devices will be supported with ASR4- as well as ASR3-based MCALs. In future, there will be also the situation of mixture of AUTOSAR4.x based BSW with AUTOSAR3.x based MCALs (or also AUTOSAR4.x with AUTOSAR4.0.x MCAL to BSW or vice versa). This is due to some OEM decisions to start with AUTOSAR4.x but willing to use some older microcontroller devices.

AUTOSAR3.x and AUTOSAR4.x based SW in general is not directly compatible.

Nevertheless, Vector supports you to be able to mix those components. There are several measures to take and Vector provides a solution for this with the term *MCAL Integration Package (mixed ASR)*. There are some wrappers (on embedded as well as on configuration side) that are part of your SIP.



### Note

Currently Vector supports mixing of AUTOSAR3.x related BSW with MCALs up to AUTOSAR4.0.3 but no higher releases.

## 3 First Steps

During *MCAL integration* at Vector several supplements have been created and integrated into your delivery. A short overview of contributed parts shall be given.

### 3.1 Delivery structure



#### Note

The delivery structure may vary in few elements. The following description is meant to be a general one. Single artefacts may be located slightly different. This depends on evolving our MICROSAR BSW product as well as differentiation between used tooling (with AUTOSAR3 DaVinci Configurator 4 is used which relies on bit different folders as DaVinci Configurator 5 Vector's solution for AUTOSAR4 based product).

Further details, e.g. about delivery structure, can be seen in [5].

The following parts of the delivery (MICROSAR SIP) are relevant for a successful *MCAL integration (based on an AUTOSAR4 BSW SIP)*:

- > BSW\<MCAL\_μC>: Contains the make-files (AUTOSAR make-files) for all 3rd party MCAL components (typically all MCAL components are merged into one single file).
- > BSWMD\<MCAL\_μC>: Contains configuration items for integration into the DaVinci Configurator. After execution of steps described under 3.2 the basic software definition files (VSMD) the 3<sup>rd</sup> party MCAL specific module definition files are located here.
- > DaVinciConfigurator\Generator\<MCAL\_μC>: Contains configuration items for integration into the DaVinci Configurator e.g. XML-preparation files as well as files containing the commands for execution of the 3rd party code generator plus optional convenience features like registration of Init-APIs towards the ECUM component recommended configurations etc.
- > ThirdParty\<MCAL\_μC>\Supply: Contains or might contain (depends on delivered parts according chapter 3.2) 3rd party MCAL and the tooling provided along with. The structure is defined by the MCAL supplier. Typically, the supply folder contains:
  - > MCAL implementation (C- and H files)
  - > BSWMD description files according to AUTOSAR
  - > Tooling for MCAL configuration and generation
  - > MCAL related documentation

**Caution**

Your delivery (SIP) may not finally contain the integrated MCAL because of several reasons (license etc.). The MCAL and its generation tooling, to be used with Vectors SIP, must be installed directly into this above-mentioned folder

`ThirdParty\<MCAL_μC>\Supply` with exactly the default name of your MCALs supplier! Please consider that the Vector SIP contains a GUI-based tool called *3rd party MCAL Integration Helper*. This tool provides some features supporting the user in handling the 3<sup>rd</sup> party MCAL and preparing it for smooth usage together with the Vector Software stack (refer to 3.2).

**Note**

Depending on the deployment of your MCAL supplier (keyword: one package for the whole MCAL, several packages for single functionalities like basic IO, memory, communication) you will have to install one or more packages into the `Supply` folder of your SIP.

Example with MCAL from 3rd party MCAL of Infineon:

```
ThirdParty\<MCAL_μC>\Supply
.\MC-ISAR_AS4XX_AURIX_TC27X_CA_PB_BASE_V100
.\MC-ISAR_AS4XX_AURIX_TC27X_CA_PB_MEM_V100
.\Tresos
```

- > `ThirdParty\<MCAL_μC>\VectorIntegration`: Contains tools that support the user during project setup (refer to 3.2)

## 3.2 Starting up

There are three possible scenarios for your SIP which will be explained in detail.

### 3.2.1 MCAL delivered within Vector SIP

To figure out if this scenario is applied with your SIP, check whether the above-mentioned folders exist. If they do and contain the needed files (e.g. embedded source code, AUTOSAR description files etc.) no preparations are necessary.

### 3.2.2 MCAL not contained within Vector SIP

With this scenario, you must do some preparations, e.g. install the MCAL etc.

**Caution**

In order to activate the MCAL and to use the workflows described later on some adaptations have to be made on MCAL packages.

To ease this procedure Vector provides a tool to execute these steps automatically. This feature also supports the user during MCAL package updates received from the MCAL manufacturer

**Note**

Please consider that the different semiconductor vendors may provide several packages. Sometimes it might be necessary to perform merge actions by hand before above-mentioned tool can be performed. For specific information please refer to the documents [6] or [8].

**Practical Procedure**

Please start the *3rd party MCAL Integration Helper* directly (*3rdPartyMcalIntegrationHelper.exe*) from the folder `ThirdParty\McalIntegrationHelper\` or alternatively the batch file located here: `ThirdParty\<MCAL_μC>\VectorIntegration\Script_MCAL_Prepare.bat` with the option `--prepare`.

If you are facing problems please refer to chapter 6.

**Note**

For compatibility reasons the `Script_MCAL_Prepare.bat` is taken over, but currently does nothing else than calling a GUI tooling named *3rd party MCAL Integration Helper*.

Using the script instead of the program allows additional actions via other parameters (described when starting the script on the command line).

This tooling asks you for some details like the location the MCAL is currently installed in. It also double checks existence of the MCAL (e.g. for update use case etc.).

The tooling processes the following principal steps:

- > Prepare MCAL configuration tool by copying of plugins or similar steps
- > Delete/Rename MCAL specifics like `Compiler_cfg.h` and `MemMap.h` within include paths (as those files are already contained within the Vector SIP)

- > Copy (maybe rename of) BSWMD files from a location within the 3rd party folder structure for one specific derivative to the location the Vector tooling is expecting it. If needed you will be asked which derivative shall be taken.
- > Adaptions on certain artefacts created and delivered by Vector to integrate the MCAL on several levels (e.g. Makefiles, Tooling-interfaces)



#### Expert Knowledge

The user may have the requirement to process all within an automated test suite, which results in a so called non-interactive mode of the GUI tooling. Non-interactive means that the tooling does not require user inputs whilst showing up the GUI.

This requires that the GUI inputs are handed over via the command line interface.

Further description is included within the `Script_MCAL_Prepare.bat` command line help.

### 3.2.3 MCAL Update needed

Usually within your development phase there will be updates of your 3rd party MCAL. Vector supports you for this use case and does some preparations within the MCAL Integration Package phase.

When you are receiving an update of the MCAL please clean up the MCAL Integration.



#### Practical Procedure

Execute the `Script_MCAL_Prepare.bat` with the option `--undo`.

Install the Update of the MCAL and execute the `Script_MCAL_Prepare.bat` with option `--prepare` again.



#### Caution

Since the update process is quite supplier specific please refer to your provided documents [6] or [8].

## 4 Workflow

As mentioned in the introduction there are three reasonable workflows the user can choose from. Which one leads to the highest benefit-cost ratio depends on user requirements / preference on the one hand and on the level of AUTOSAR conformance of the MCAL on the other hand. In the following all three workflows are introduced and their corresponding prerequisites and advantages/disadvantages are listed for guidance.



### Note

Your SIP contains a document [6] or [8] which describes the recommended / supported workflow.



### Multimedia Link

For the case that semiconductor supplies MCAL with EB tresos studio there is a video available which shows the MCAL integration as well as an exemplary workflow. We highly recommend having a look into this video for better understanding necessary inputs as well as interaction.

Just have a look at [7].

When you are not able to watch this video please get in contact with Vector delivery engineer to provide you an offline-version.



### Note

Since the following chapters only mention the term DaVinci Configurator some words upfront.

You will (have to) use DaVinci Configurator 4 for AUTOSAR3 based MICROSAR product. With AUTOSAR4 based MICROSAR product you will have to use DaVinci Configurator 5.

For some special reasons, you will get DaVinci Configurator 5 even if your delivery is based on MICROSAR3 but this is not driven by topic MCAL but on some others and therefore not explained in further details.

### 4.1 Single configuration tool usage

After execution of the first steps described in 3.2 the user is able to use Vector DaVinci Configurator as the exclusive configuration tool for both the MICROSAR BSW as well as the 3rd party MCAL. The configuration of the 3rd party components is based on the MCAL AUTOSAR component definition files (BSWMD) which are (must be) included in the original semiconductor vendor packages.

You can activate/add the 3rd party components to your configuration in the same way you would activate/add the MICROSAR BSW components.

Vector has integrated the commands for the external code generators so that you will be able to start the MCAL code generation within the Configurator GUI by using this workflow. DaVinci Configurator is used to call the MCAL generator and displays warnings, errors, and other information.

Prerequisite	
AUTOSAR component definition files provided by MCAL manufacturer must be completely AUTOSAR conform and consistent to possibly used proprietary file formats (used internally by the 3rd party tool).	
Advantages	Disadvantages
Only one configuration tool and one configuration file to be used / no redundant data	No usage of convenience features which are possibly provided by MCAL manufacturer but not transferred/modeled within the AUTOSAR component definition files
All dependencies between BSW and MCAL components can be solved easily	

Table 4-1 Guidance for single configuration tool usage

## 4.2 Mixed configuration tool usage

The configuration tooling respectively the internal (proprietary) description files of some 3rd party MCAL contains internal logic which supports the user during configuration steps. Examples are

- > Parameter A is only needed if Parameter B is set to value "X"
- > Parameter C is calculated automatically based on A and B
- > During component instantiation a list of containers is created automatically with reasonable and different values

These mechanisms are not completely specified by the AUTOSAR standard and not always modeled correctly within the AUTOSAR component definition files provided. The fact is that they are not feasible useable outside the original MCAL configuration tool. As a result, and depending on the level of feature convenience of the MCAL tool it is advisable to at least use it for the project setup as it will speed up the process in this phase.

By doing so the user has to deal with two different tools during ECUC setup. But in a later phase it will be easier to use a *one tool solution* as described in 4.1. Thus, the user will have to deal with a tool transition. To ease this procedure Vector has introduced some tool features.

The diagram below shows a workflow to use both the features of the MCAL tool and the convenience of a solution which is based on a single tool.



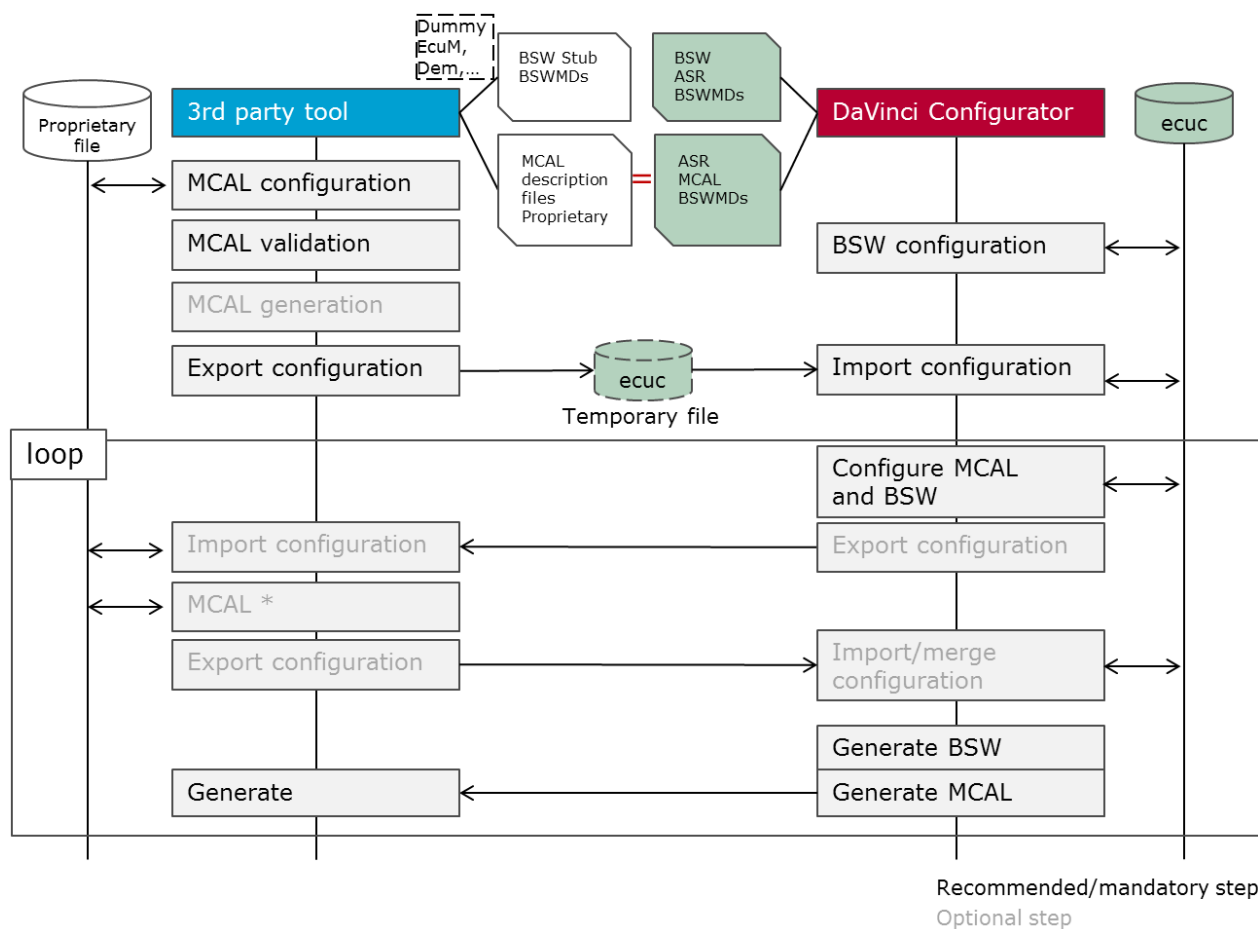


Figure 4-1 Configuration workflow – Mixed configuration tool usage

Prerequisite	
AUTOSAR component definition files provided by MCAL manufacturer must be completely AUTOSAR conform and as consistent as possible to used proprietary files	
Advantages	Disadvantages
Usage of convenience features of both tools	Additional tool knowledge is needed
Dependencies between BSW and MCAL components can be solved by data synchronization (export/import)	Overhead due to export/import workflow
Switch to single configuration tool and configuration file possible	

Table 4-2 Guidance for mixed configuration tool mode



### Practical Procedure (startup)

1. Create a DaVinci project
2. Activate BSW components needed, but no MCAL components
3. Create a project within the 3rd party configuration tool (see 5.2)
4. Configure all MCAL components needed within the 3rd party configuration tool
5. Export all MCAL components from the 3rd party tool to an AUTOSAR conform ECUC file
6. Import this file in DaVinci Configurator



### Note

The 3rd party tool might use a proprietary description format that deviates from the AUTOSAR BSWMD format. This might lead to inconsistencies between the formats and result in warnings and/or errors after import of the configuration in DaVinci Configurator (e.g. due to different container multiplicities). Those errors might be handled via DaVinci Configurator feature "Set User Defined" in the configuration GUI.

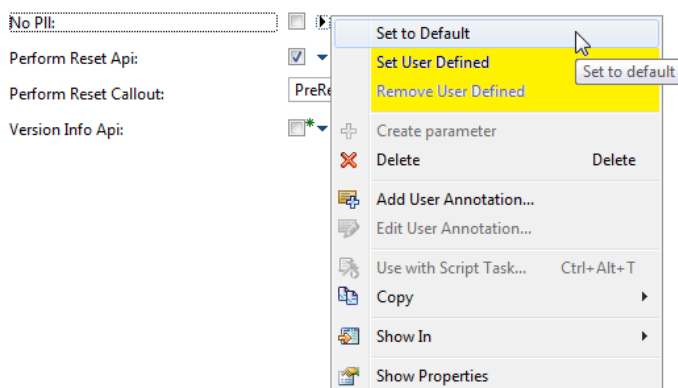


Figure 4-2 Set User Defined

**Note**

As marked in Figure 4-1 above, the loop - sometimes mentioned as round trip - is generally supported. Precondition for a successful round trip is DaVinci Configurator 5.10 and above (this implies that with AUTOSAR3 based BSW and DaVinci Configurator 4 this scenario is not fully supported).

Round trip workflow has been evaluated with several semiconductor vendor's MCALs and is working in principal.

Based on the semiconductors AUTOSAR compatibility there may raise some warnings or sometimes even single parameters may be reconfigured, based on insufficient importer functionality.

### 4.3 Split configuration tool usage

This workflow is required if the 3rd party configuration tool is not completely AUTOSAR compliant as it e.g. violates the AUTOSAR standard parameter definition or does not support a lossless ECUC roundtrip with DaVinci Configurator or other AUTOSAR tools.

As the number of interfaces respectively the number of exchanged parameters between the AUTOSAR BSW and the MCAL components is comparatively small it may also be a solution to use both tools permanently in parallel. The implication is that the code generation will be triggered from the tool in which the configuration has been made.

In order to integrate both parts on embedded side there is still the need to share some parameters between both configuration tools. This can either be achieved by manual replication (using configuration stubs, e.g. DEM stub of 3rd party MCAL, MCU stub of Vector) or by using the import export mechanisms introduced in 4.2.

Prerequisite	
AUTOSAR component definition files provided by MCAL manufacturer must be available but must be matching to proprietary formats only partly (at interfaces between BSW and MCAL).	
Advantages	Disadvantages
Relatively tolerant workflow in case that AUTOSAR component definition files and/or 3rd party tools are not completely AUTOSAR conform	Dependencies between BSW and MCAL components must be solved manually
	Additional tool knowledge is needed
	Overhead due to 2 tool workflow and redundant configuration data → manual synchronization needed

Table 4-3 Guidance for split configuration tool usage

## 5 Configuration tools

In this chapter, some hints for the configuration tools included in your package are given.



### Note

Most common tooling used by 3rd party MCAL supplier is EB tresos™ therefore the usage is explained extensively.

### 5.1 Vector DaVinci Configurator

Use DaVinci Configurator to configure your MICROSAR BSW using a DPA project. Please refer to the startup manual included in the delivery. In addition, the DaVinci Configurator provides user support via the help menu.

### 5.2 EB tresos™



### Reference

Additional and useful information can be found here [7].

With questions about tooling license (applicable until EB tresos 16) additional information can be found there [4].

The program itself is located in the folder beneath: `ThirdParty\<MCAL_μC>\Supply\`

To start the tool the user must execute `tresos_gui.exe` which can be found usually beneath `Tresos\bin`. Just to note, some 3rd party MCAL suppliers also provide 64bit versions of EB tresos™.

A useful documentation when you are struggling with a task can be found within the tresos folder structure by searching for a document named “*Studio\_documentation\_users\_guide*”.

#### 5.2.1 Setting up a new Configuration Project

- > To create a new project the Project Wizard must be started by selecting File → New → Configuration Project.
- > The project name as well as the workspace location plus the relevant AUTOSAR Release Version can now be chosen.

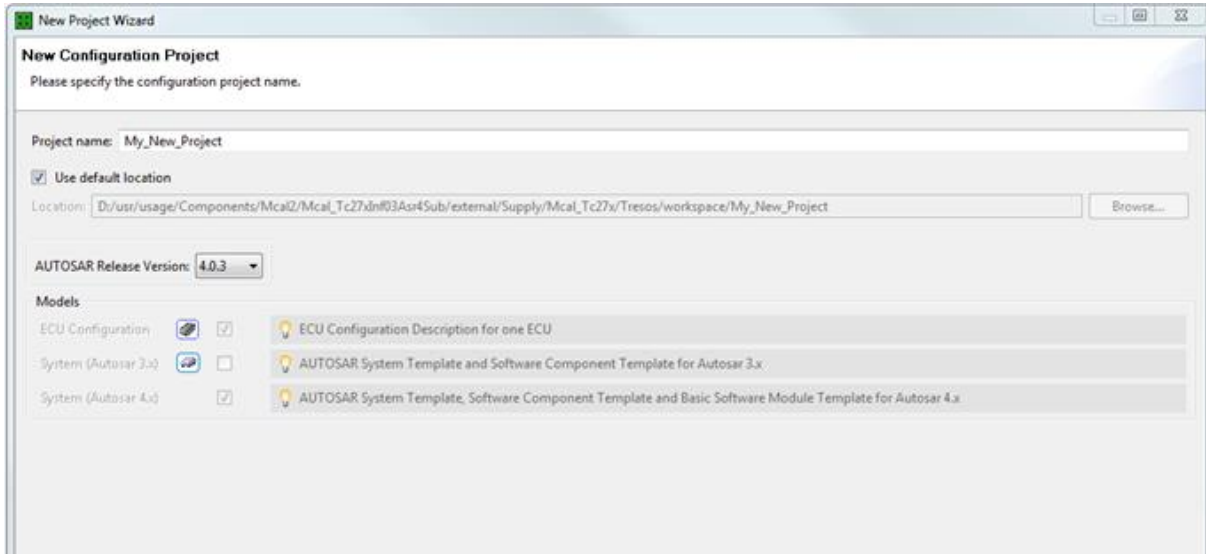


Figure 5-1 New Configuration Project

- > Hint: The workspace location can be chosen freely the path for code generation output can be set in the next step.

### 5.2.2 Project Details

In the next step, the wizard requests the following data:

- > ECU ID: fill in a symbolic name
- > Target: Select your microcontroller target
- > Generation Path: please choose the settings according to the settings within the DaVinci Configurator to keep consistency and avoid problems regarding make process
- > The remaining choices can be selected such as in the figure below
- > We recommend activating the check box to 'Automatically add the minimum number of child elements in lists'



Figure 5-2 Configuration Project Data

- > Hint: Several project settings can be changed after creation of the project by editing the file `preferences.xdm` located within the created workspace folder; e.g. adapt the generation path by changing the default value 'output' to a relative or absolute path (`<d:var name="GenerationPath" value="output"/>`)

### 5.2.3 Selection of components

Now choose the components, which should be added to the project. In general, every MCAL component should be selected. In that case, use STRG + A to select all and click on the tagged button.

Attention: Some additional "dummy" components (stubs) might be necessary in order to configure the MCAL components e.g. Base, Resource, Dem or EcuM. Those components are only used to provide parameters and references needed by the MCAL components such as *Dem Events* and *EcuM run modes*, please refer to 5.3.

The window should now be similar to this:

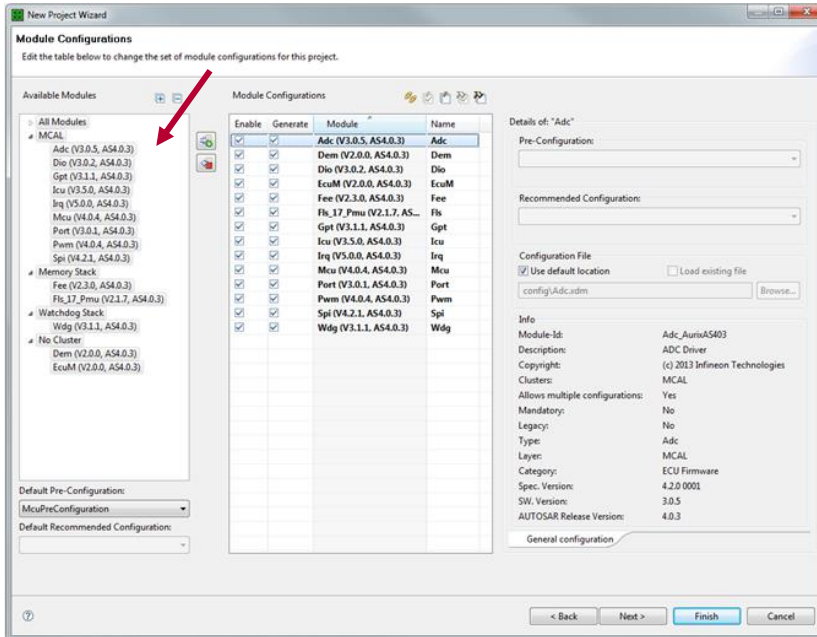


Figure 5-3 Component Configurations

## 5.2.4 Creation of importers and exporters

Now the user has the possibility to create importer and/or exporter to create a configuration file in an AUTOSAR standardized format out of the proprietary tresos™ configuration files. This functionality is needed to enable the workflows described in chapter 4.2 and 0.

Please Click on + to add an Im-/Exporter and use the following settings:

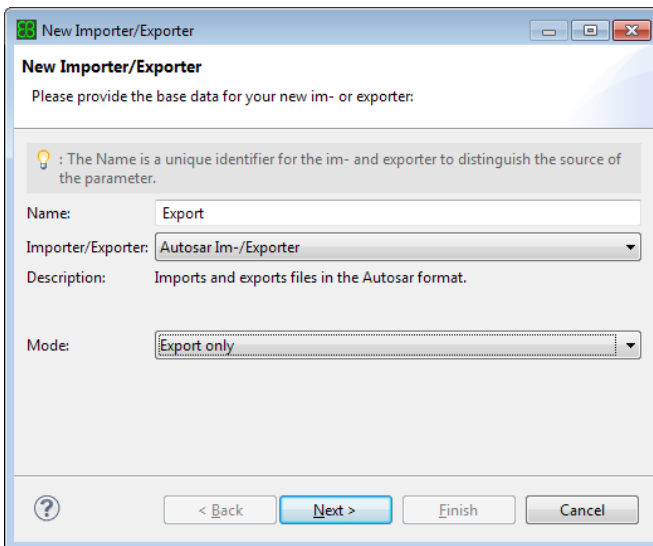


Figure 5-4 Create an exporter (step 1)

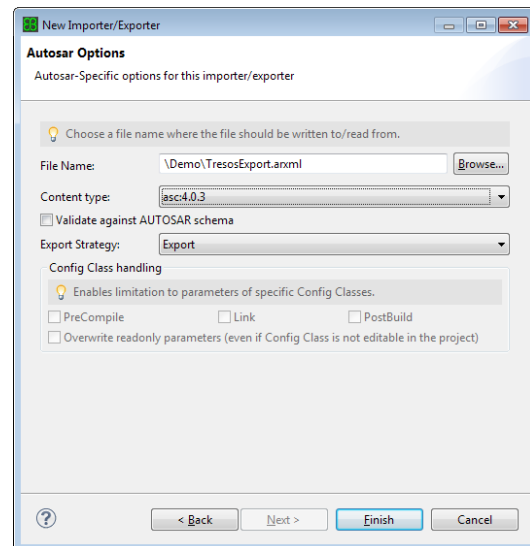


Figure 5-5 Create an exporter (step 2 – AUTOSAR options)

The importer can be configured in a similar manner.

If you don't create the im-/exporters during setup you can do this later by using the context menu on the corresponding project.

### 5.2.5 Configure the MCAL components

Now the wizard is finished and the MCAL itself can be configured.

### 5.2.6 Generation of the 3rd party MCAL

If everything is configured, click on the generation button. Tresos will now generate your project (note: there will be a validation phase triggered before to ensure that configuration is formal correct).



Figure 5-6 Generate Button

## 5.3 Cooperation Interface between DaVinci Configurator and EB tresos™

With the Vector MCAL integration package the SIP is prepared for cooperation between DaVinci Configurator and EB tresos™ according the recommended workflow (refer to 4.2). Therefore there is no necessity for additional actions.

Nevertheless, some settings that are relevant for EB tresos™ can be configured in the DaVinci Configurator 5 Project Settings editor for Generator User Settings. Depending on the tool version it looks like follows:

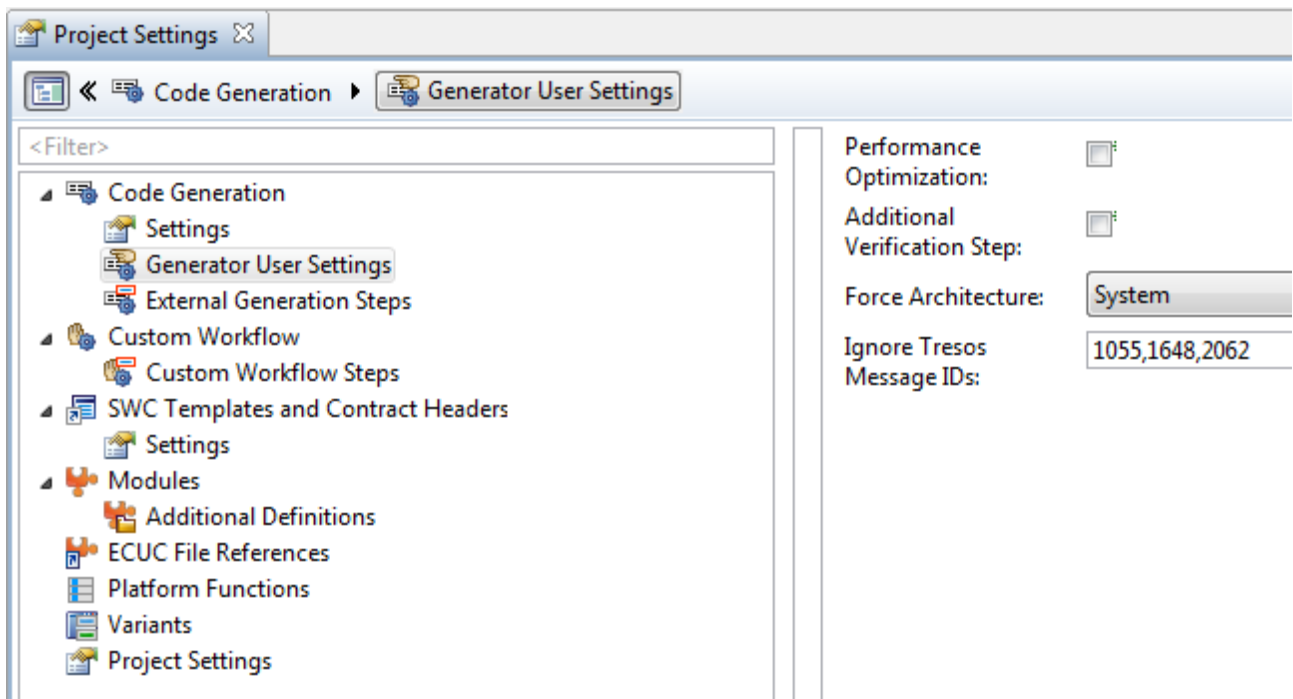


Figure 5-7 Project Settings - Generator User Settings

The following configuration possibilities influence the behavior of the external generation tool:

► **Performance Optimization:**

This switch reduces the time needed for the generation process. This can be reached



by calling the external generation tool only once for all selected MCAL modules.  
Please consider that there are some restrictions when using the optimization switch:

- > The validation results for the single generated modules cannot be displayed separately, but arranged in one container named *Tresos...*:

Tresos01055	Tresos generator message (2 messages)
Tresos01055	Removed node "DemDebounceCounterStorage": The schema does not define a parameter with name "DemDebounceCounterStorage" in container "ASPath/TS_T2D22M1011R0/Dem/DemConfigSet/DemEventParameter/DemEventClass/DemDebounceAlgorithmClass/DemDebounceCounterBased".
Tresos01055	Removed node "IMPLEMENTATION_CONFIG_VARIANT": The schema does not define a parameter with name "IMPLEMENTATION_CONFIG_VARIANT"
Tresos01072	Tresos generator message (1 message)
Tresos01648	Tresos generator message (111 messages)

Figure 5-8 Tresos Generator Message

- > The Generation Result View does no longer show the single generated files, but only the output folder:

Generator	Output
Base Services	
I/O	
Adc	
include [File is up-to-date]	
output [File is up-to-date]	
src [File is up-to-date]	
Dio	
include [File is up-to-date]	
output [File is up-to-date]	
src [File is up-to-date]	
Icu	
Port	
Pwm	
Spi	
Memory	
Spi	
Mode Management	

Figure 5-9 Generation Result View

- ▶ **Additional Verification Step:**  
This switch activates a verification call of the external generation tool before generating the modules. Please consider that this leads to a longer time needed for the MCAL generation process.
- ▶ **Force Architecture:**  
These drop-down entries set the execution architecture of EB tresos™ (32 bit or 64 bit) manually. By default, the system property is evaluated. Further valid values are 'x86' (means 32 bit) and 'x64' (means 64 bit).
- ▶ **Ignore Tresos Message IDs:**  
This setting disables printing of any messages with a specific message ID. Several IDs can be passed comma separated. This is meant to suppress the (sometimes) huge amount of not relevant warnings during the generation process. Please use this setting carefully.

## 5.4 Configuration hints for parallel usage of DaVinci Configurator and EB tresos™

One important thing to consider is that the configurations for Dem and EcuM must be consistent between DaVinci Configurator and tresos as these are interface components provided by Vector that also have impact on the 3rd party software. Affected configuration entities are Dem/DemConfigSet/DemEventParameter and EcuM/EcuMConfiguration/EcuMCommonConfiguration/EcuMWakeupSource.

The following chapters handle the configuration direction from upper layer to lower layer like mentioned above, but please are aware that the direction might also be inverse, i.e. for configuration of *Dio Ports*, *Mcu ConfigSets*, *Mcu ClockReferences* or *Fee Blocks*. If you use the workflows described in 0 and 4.2 this is done by export and import features.

### 5.4.1 Vector DaVinci Configurator 5

You can find the configuration of e.g. DemEventParameter in the Basic Editor.

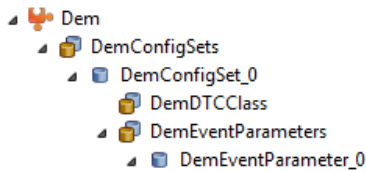


Figure 5-10 DEM Path using DaVinci Configurator 5

Short Name:	<input type="text" value="DemEventParameter_0"/>	▼
DTC Class Ref:	<input type="text" value=""/>	[...] ▼
Event Id:	<input type="text" value="65535"/>	dec ▼
Event Kind:	<input type="text" value="DEM_EVENT_KIND_BSW"/>	▼
Extended Data Class Ref:	<input type="text" value=""/>	[...] ▼
Freeze Frame Class Ref:	<input type="text" value=""/>	[...] ▼
Freeze Frame Rec Num Class Ref:	<input type="text" value="DemFreezeFrameRecNumClass"/>	[...] ▼
Max Number Freeze Frame Records:	<input type="text" value="1"/>	dec ▼

Figure 5-11 Settings within DaVinci Configurator 5 Pro

### 5.4.2 EB tresos™

- > One fast way to change the configuration for DEM in tresos™ is with the Outline window.
- > To do so you, double-click on DEM. Now, if you have Outline opened, you will see the folder structure of DEM.
- > Navigate to DemEventParameter\_0, select the same Event ID and Event Kind as in the DaVinci Configurator Pro.

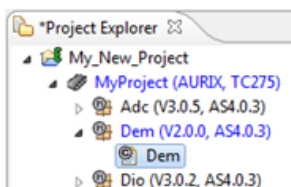


Figure 5-13 DEM-Path within tresos™

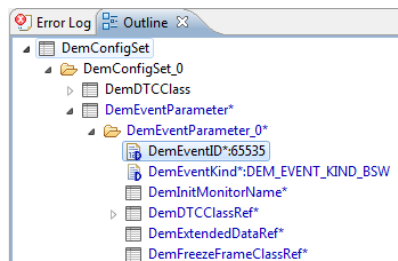


Figure 5-12 DEM-Path in the Outline window of tresos™

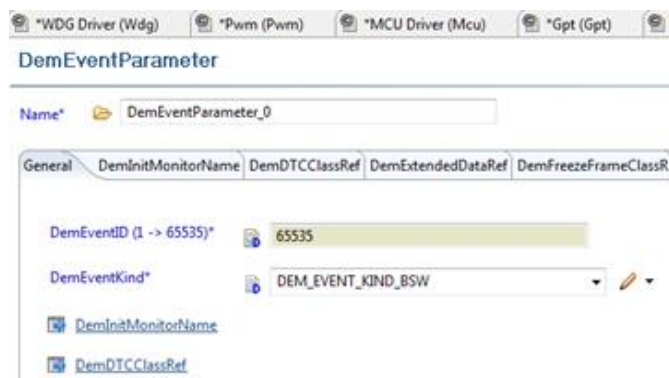


Figure 5-14 Settings for DEM within tresos™

## 5.5 DaVinci Configurator Services

### 5.5.1 Internal Behavior Editor

With the MCAL integration static values for the Internal Behavior (like Critical Sections and Main function time events) are provided. To enable the user changing and / or adding those values, a service in the User Interface exists:

Runtime System --> Module Internal Behavior

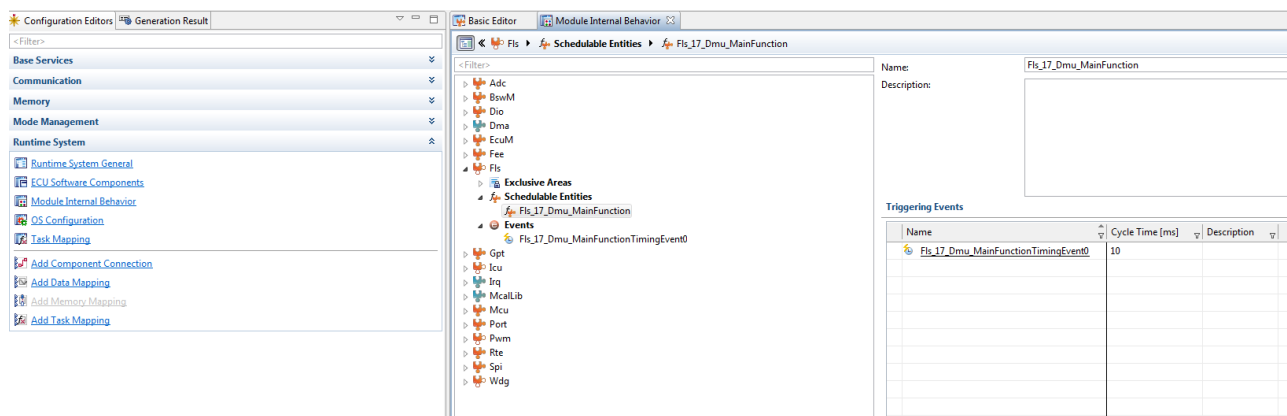


Figure 5-15 Internal Behavior Editor

## 6 Restrictions

### 6.1 UNC path usage

*3rd party MCAL Integration Helper* tool does not support calling it via UNC path. There is no workaround etc. available.

## 7 Known Issues

This chapter describes known issues which are related to MCAL integration package but are not fixable by Vector without support of 3rd parties. Vector is working on solutions in coordination with the other involved parties.

### 7.1 Long path names



#### Caution

The 3rd party deliverables are currently stored / copied into the Vector delivery structure and may use very long path names. The Vector delivery structure itself usually resides some folders below the root directory. It may happen that some Microsoft Windows™ tools are not able to access those files anymore due to resulting path names which are too long.

Currently Vector is working on a solution for this problem.

As long as there is no final fix we recommend installing the Vector SIP as well as the 3rd party MCAL as near to the root-directory as possible (e.g. D:\<ECU-Acronym> like

D:\BCM)

### 7.2 Missing configuration items within imported configuration

Sometimes it happens that 3rd party tools do not export all configuration items / information into AUTOSAR standard format of ECUC.

Currently there is no workaround available for this. The user will be informed within the import process into DaVinci Configurator. After importing there must be manual editing of the configuration to add all of those “missed information” (Note: mainly these are not used by 3rd party tooling but relevant for consistent configuration with components BSWMDs).



#### Note

Due to the fact that 3rd party supply must be AUTOSAR conform please ask your MCAL vendor to provide an update of the corresponding artefacts.

### 7.3 Configuration Export with EB tresos™ Version 13.0.0

After you have exported the EB tresos™ configuration, you have to close the project before you proceed the export the next time. If not, the export will be faulty and will cause problems during import into DaVinci Configurator.

## 7.4 Error messages regarding CommonPublishedInformation with EB tresos™

After setting up an MCAL project with EB tresos™ error messages might occur for each parameter of container CommonPublishedInformation like follows:

```
Invalid value for node "/AUTOSAR/TOP-LEVEL-  
PACKAGES/Wdg/ELEMENTS/Wdg/CommonPublishedInformation/VendorId  
": Value "" is no number
```

Closing EB tresos™ tool once and opening again resolves the problem. This issue affects all MCAL modules.

## 7.5 Problems due to resorting of configuration elements by EB tresos™ Exporter

In some cases, the Tresos EB tresos™ Exporter resorts configuration elements when converting the configuration created by EB tresos™ GUI into AUTOSAR EcuC format. Trying to generate then might lead to several problems like error messages from the external tool or code mismatch in the generated dynamic sources. This behavior has already been observed for different NXP PowerPC MCALs (Flash Sector List) and Infineon AURIX MCALs (Interrupt defininitons).

If this problem occurs, the only reliable workaround is to create a configuration in EB tresos™ and use the EB tool for generation of the affected module.

## 8 Frequently Asked Questions

**Q:** My delivery is a mixed version of BSW regarding AUTOSAR3 and MCAL regarding AUTOSAR4, what do I have to care for?

**A:** When you ordered a SIP from Vector with “MCAL Integration package (mixed ASR)” we already have cared about this and also provide you some embedded software wrappers (e.g. for Watchdog driver). The “integration process” at Vectors side has done all necessary topics like extensions of makefiles etc. that you as the user do not have to care about it.

**Q:** I have got a DaVinci Configurator 5 but my delivery is AUTOSAR3 based, are there any consequences or tasks I have to care about?

**A:** In regard of MCAL integration for AUTOSAR3 based products the DaVinci Configurator 5 does not care! As a hint: There may be some AUTOSAR4-based components which use DaVinci Configurator 5 framework or definitely have to be configured using DaVinci Configurator 5 but this will be explained in detail with the respective components. But as mentioned before, this is not relevant for MCAL configuration / generation purpose.

**Q:** The MCAL will be generated component by component and takes a while. Is there any option to increase or speed up this generation?

**A:** Based on EB tresos as generation tooling there is the possibility to call the generation with an option to generate all MCAL components contained in the ECUC file. The user therefore has to add a so called *external generation step* (applies to DaVinci Configurator 5 naming but is also possible with DaVinci Configurator 4). For details about how to create an *external generation step* please use DaVinci tooling help.



### Example

External generation step needs a command line parameter as follows:

```
tresos_cmd.bat -Dtarget=PA -Dderivate=MPC574XG legacy generate
D:\BCM\XSL_Output.arxml@asc:4.0.3 -o D:\BCM\Gendata -n Adc -n
Dio -n EcuM -n Fls -n Gpt -n Icu -n Mcl -n Mcu -n Port -n Pwm -n
Spi -n Wdg -n Resource -n Dem -g Adc_TS_T2D35M10I1R0 -g
Dio_TS_T2D35M10I1R0 -g Fls_TS_T2D35M10I1R0 -g
Gpt_TS_T2D35M10I1R0 -g Icu_TS_T2D35M10I1R0 -g
Mcl_TS_T2D35M10I1R0 -g Mcu_TS_T2D35M10I1R0 -g
Port_TS_T2D35M10I1R0 -g Pwm_TS_T2D35M10I1R0 -g
Spi_TS_T2D35M10I1R0 -g Wdg_TS_T2D35M10I1R0 -g
Base_TS_T2D35M10I1R0
```

Please refer to EB tresos studio documentation about all needed details (e.g. derivative, component list etc.). Please watch out that there may be a “generate all” option. Please do not use this because then also Dem, EcuM or other components (maybe other COM components) will also be generated. This will definitely create conflicts for the build process later on!

**Caution**

Based on EB tresos internal behavior and plug-in management the ECUC which is handed over for generation reasons has to comply from naming perspective to the registered plugins.

This results in the fact that ECUC file must not contain any other component then registered (e.g. NvM, OS etc.) and shortname / AR-package name has to fit to the registered one (Dem, EcuM).

Vector does this during its integration of the MCAL via XSL-transformation file but only on component level and not for “generate all” purpose. So, the user would have to create such a XSL-transformation file by its own if this use case (generate all) should be applied.



## 9 Glossary and Abbreviations

### 9.1 Glossary

Term	Description
3rd party components / MCAL	BSW components that have been provided by a company other than Vector. Vector may have integrated the software within the SIP but does not take over any responsibility with regard to the functionality of these components.
3rd party MCAL Integration Helper	Windows based tooling providing a GUI to support the user to integrate a 3 <sup>rd</sup> party MCAL into the Vector SIP.
DaVinci Configurator	Vectors configuration and generation tool of the MICROSAR components

### 9.2 Abbreviations

Abbreviation	Description
AUTOSAR	Automotive Open System Architecture
BSW	Basic Software
BSWMD	Basic Software Module Description; AUTOSAR standard file
DEM	Diagnostic Event Manager; AUTOSAR BSW component
DPA	DaVinci Project Assistant
ECU	Electronic Control Unit
ECUC	ECU Configuration; configuration of all BSW as well as MCAL components (Note: AUTOSAR also supports ECUC splits containing just a subset of components with its configuration)
ECUM	ECU State Manager; AUTOSAR BSW component
(G)UI	(graphical) User Interface
MCAL	Microcontroller Abstraction layer
MICROSAR	Vectors solution / brand for AUTOSAR BSW
SIP	Software Integration Package (as provided by Vector); MICROSAR delivery containing customers' selection of MICROSAR BSW components including all parts of <i>MCAL Integration</i> .
StMD	Standard Module Definition also known as BSWMD
VSMD	Vendor Specific Module Definition (within AUTOSAR methodology it is derived from StMD)

## 10 Contact

Visit our website for more information on

- > News
- > Products
- > Demo software
- > Support
- > Training data
- > Addresses

**[www.vector.com](http://www.vector.com)**