

# EHANDBOOK-Container-Build

## Key Features User Guide

Version 1.0

1. **Basic**
2. **Simulink/Stateflow**
3. **C-Code**
4. **ASCET-Developer**
5. **ASCET**
6. **ASCET for HMC**
7. **AUTOSAR (Classic)**
8. **How to guides**

# EHANDBOOK-Container-Build

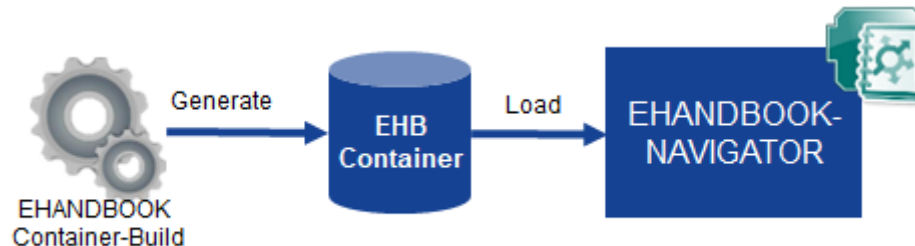
## Basic

Welcome !

**ETAS EHANDBOOK** is an **interactive documentation solution for automotive software**. It makes automotive software **easier and faster to understand** such that developers can **get their jobs done more efficiently**.

EHANDBOOK consist of the three components:

- 1. EHANDBOOK Container-Build:** Tools to generate interactive documentation in EHANDBOOK format
- 2. EHANDBOOK Container:** A file format to aggregate all documentation contents into one file
- 3. EHANDBOOK-NAVIGATOR:** A tool to allow users explore the ECU Software interactively



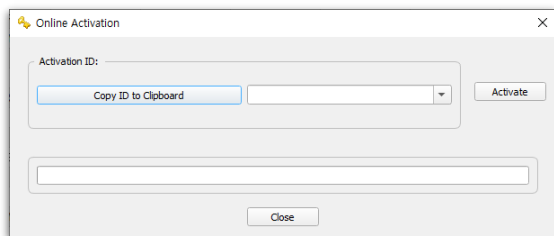
## Installation & License Setup

### Installation

1. Unzip the download package into a directory of your choice.  
As standard directory, it is recommended to install ETAS tools under “C:\ETAS”
2. Go to “C:\ETAS\EHANDBOOK-Container-Build\_<Version>-Windows” and run “License\_Management\_setup.exe” with administrator rights

### License Setup

1. Launch the ETAS License Manager (LiMa) from the Windows start menu
2. Go to the menu Device → Online Activation
3. Paste the Activation ID that you have received from ETAS (e.g., via email)



ETAS License Manager									
File License Tools Device Help									
All									
	Product	Feature	Status	Feature Version	Source	Expiration Date	Remaining Days	License Mode	Days Left
✓	EHandbook-CB	EHB_CB_CON_AUTOSAR_CL...	Installed	1.0	Borrow	31-Dec-2023	Permanent	0	No
✓	EHandbook-CB	EHB_CB_BASE_ASAM	Installed	1.0	Borrow	31-Dec-2023	Permanent	0	No
✓	EHandbook-CB	EHB_CB_BASE_DIR	Installed	1.0	Borrow	31-Dec-2023	Permanent	0	No
✓	EHandbook-CB	EHB_CB_MDL_ASCET	Installed	1.0	Borrow	31-Dec-2023	Permanent	0	No
✓	EHandbook-CB	EHB_CB_MDL_CCODE	Installed	1.0	Borrow	31-Dec-2023	Permanent	0	No
✓	EHandbook-CB	EHB_CB_MDL_SIMULINK	Installed	1.0	Borrow	31-Dec-2023	Permanent	0	No
✓	EHandbook-CB	EHB_CB_PDFGEN	Installed	1.0	Borrow	31-Dec-2023	Permanent	0	No
✓	EHandbook-CB	EHB_CB_UGG_ASCET	Installed	1.0	Borrow	31-Dec-2023	Permanent	0	No
✓	EHandbook-CB	EHB_CB_UGG_CCODE	Installed	1.0	Borrow	31-Dec-2023	Permanent	0	No
✓	EHandbook-CB	EHB_CB_UGG_SIMULINK	Installed	1.0	Borrow	31-Dec-2023	Permanent	0	No
✓	EHandbook-CB-ADDON	EHB_CBTBX_SIMULINK	Installed	1.0	Borrow	31-Dec-2023	Permanent	0	No

# EHANDBOOK-Container-Build

## Simulink/Stateflow

## Directory-based EHB-CB for Simulink

### Setting up the input data

To generate EHB Containers using the **Directory-based** EHANDBOOK Container-Build tools, the input data such as models, interface and label descriptions as well as textual documentation must be arranged in a directory structure. The directory structure must follow a specific convention:

- **A root directory:** This directory is passed to EHB-CB tool-chain as input directory
- **One or more sub-directories:** Each sub-directory represents a separate function
- **Within each sub-directory :** files with model data, interface and label description as well as textual description is placed

```
└── ECU_SW_XYZ
    ├── Function1
    │   ├── function_1_model.slx
    │   ├── interface_and_labels.xlsx
    │   └── textual_description.adoc
    .
    .
    .
    └── Function_n
        ├── function_n_model.slx
        ├── interface_and_labels.xlsx
        └── textual_description.adoc
```

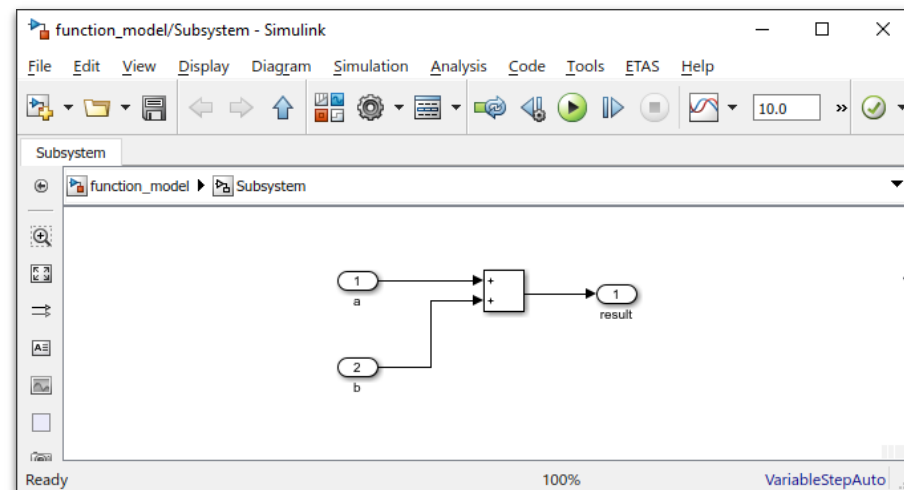
## Directory-based EHB-CB for Simulink

### Simulink / Stateflow

EHANDBOOK Container-Build tools support generating interactive models for ECU functions developed with Simulink & Stateflow.

When using the EHANDBOOK Container-Build tool-Chain, Simulink / Stateflow models can be used as specification models for the algorithmic behavior of individual ASAM MDX-based ECU functions or AUTOSAR (Classic) software components.

Besides this, ETAS has developed a MATLAB toolbox - the [EHANDBOOK Container-Build Toolbox for Simulink](#) - to directly generate EHANDBOOK Containers from Simulink / Stateflow models.





## Directory-based EHB-CB for Simulink

### Input data for label information and interface specification

To document information about the labels used by the ECU function as well as the interface specification in the EHB Container, this has to be described in the form of an **Excel** file which is stored in the same directory as the **Simulink** sources.

The Excel file template comes with two sheets: one for **signals** and one for **calibrations**.

**Signals** are Global variables which **can be measured** with the help of measurement tools such as ETAS INCA. Signals which are read and written by an ECU function are used to exchange data with other ECU functions.

**Calibrations** are Variables for which the values **can be changed / fine-tuned** with the help of calibration tools such as ETAS INCA. Calibrations can be e.g. single scalar values take the form of curves or maps which are interpreted by 1D or 2D interpolation routines.

	1	2	3	4	5	6
1	*	Mark	Xref	Class	Name	Description
2			IN	RAM	a	First summand
3			IN	RAM	b	Second summand
4			OUT	RAM	result	Sum (=result of the addition)
5						

	1	2	3	4	5	6
1	*	Mark	Xref	Class	Name	Description
2			LOC	VALUE	Param	A parameter
3						
4						
5						

## Directory-based EHB-CB for Simulink

### Textual Content

You can add textual documents in **RTF** or **AsciiDoc** format, together with the model file. This content is then transformed into the EHB container. Later it is shown in the function documentation window of the function component in EHB-NAV and describes verbally the functionality of the corresponding model. These documents can contain also images and screenshots of model hierarchies. You can add multiple documents per folder, even in both formats in parallel.

### Referencing to model screenshots

EHANDBOOK Container-Build can automatically generate a screenshots from **Simulink** based interactive models that can be embedded in the textual documentation.

To generate a **screenshot**, you have to create a model reference to the **Simulink** model and the **subsystem** you want to depict. This can be achieved with the EHANDBOOK model reference directive:

```
ehbmodelref:<simulink-model-file>/<subsystem-path>[ ]
```

# EHANDBOOK-Container-Build

## C-Code

## Directory-based EHB-CB for C-Code

### Setting up the input data

To generate EHB Containers using the **Directory-based** EHANDBOOK Container-Build tools, the input data such as models, interface and label descriptions as well as textual documentation must be arranged in a directory structure. The directory structure must follow a specific convention:

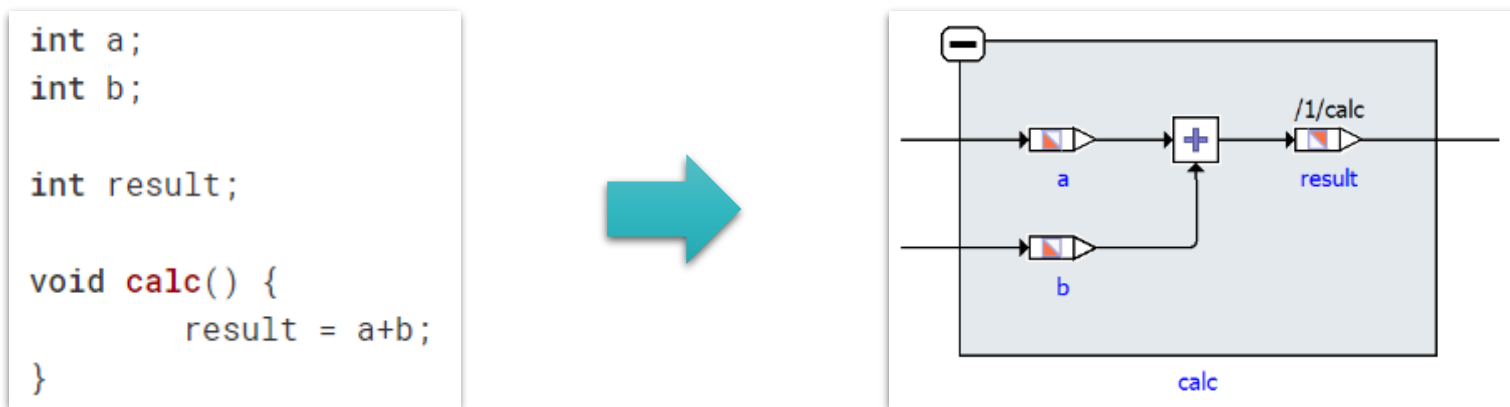
- **A root directory:** This directory is passed to EHB-CB tool-chain as input directory
- **One or more sub-directories:** Each sub-directory represents a separate function
- **Within each sub-directory :** files with model data, interface and label description as well as textual description is placed

```
└── ECU_SW_XYZ
    ├── Function1
    │   ├── function_1_implementation.c
    │   ├── interface_and_labels.xlsx
    │   └── textual_description.adoc
    ├── .
    ├── .
    ├── .
    └── Function_n
        ├── function_n_implementation.c
        ├── interface_and_labels.xlsx
        └── textual_description.adoc
```

## Directory-based EHB-CB for C-Code

### C Code

EHANDBOOK Container-Build tools support generating interactive models for ECU functions implemented in ANSI **C-Code**. Through this, an interactive, graphical visualization of the data and control flow structure of a C-code implementation can automatically be derived. Such **auto-generated interactive models** can be used by automotive engineers as a basis for understanding algorithms, tracing signal flows, etc. - even if there is no graphical model available at the source.



## Directory-based EHB-CB for C-Code

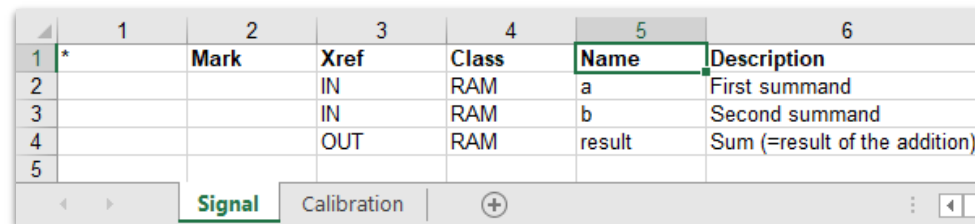
### Input data for label information and interface specification

To document information about the labels used by the ECU function as well as the interface specification in the EHB Container, this has to be described in the form of an **Excel** file which is stored in the same directory as the **C-code** sources.

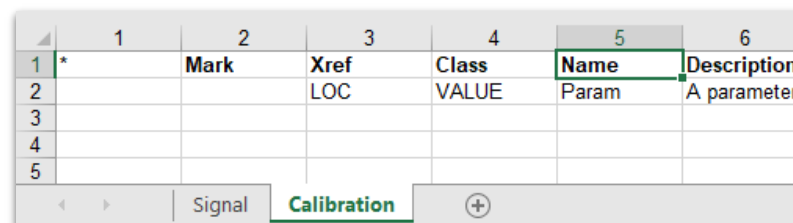
The Excel file template comes with two sheets: one for **signals** and one for **calibrations**.

**Signals** are Global variables which **can be measured** with the help of measurement tools such as ETAS INCA. Signals which are read and written by an ECU function are used to exchange data with other ECU functions.

**Calibrations** are Variables for which the values **can be changed / fine-tuned** with the help of calibration tools such as ETAS INCA. Calibrations can be e.g. single scalar values take the form of curves or maps which are interpreted by 1D or 2D interpolation routines.



	1	2	3	4	5	6
1	*	Mark	Xref	Class	Name	Description
2			IN	RAM	a	First summand
3			IN	RAM	b	Second summand
4			OUT	RAM	result	Sum (=result of the addition)
5						



	1	2	3	4	5	6
1	*	Mark	Xref	Class	Name	Description
2			LOC	VALUE	Param	A parameter
3						
4						
5						

## Directory-based EHB-CB for C-Code

### Textual Content

You can add textual documents in **RTF** or **AsciiDoc** format, together with the model file. This content is then transformed into the EHB container. Later it is shown in the function documentation window of the function component in EHB-NAV and describes verbally the functionality of the corresponding model. These documents can contain also images and screenshots of model hierarchies. You can add multiple documents per folder, even in both formats in parallel.

### Referencing to model screenshots

EHANDBOOK Container-Build can automatically generate a screenshots for block diagram visualizations derived from **C-Code** that can be embedded in the textual documentation.

To generate a **screenshot**, you have to create a model reference to the **C-function** you want to depict. This can be achieved with the EHANDBOOK model reference directive:

```
ehbmodelref:<hierarchy-path-to-c-function>[]
```

# EHANDBOOK-Container-Build

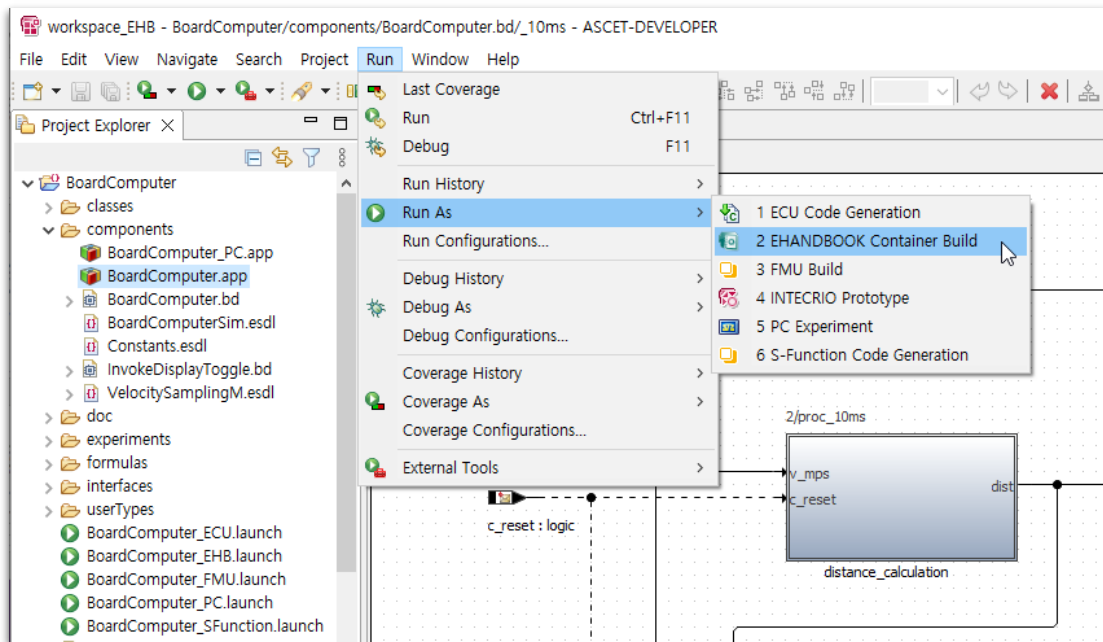
## ASCET-Developer



### Creating a launch configuration for EHANDBOOK Container-Build

To generate an EHANDBOOK container for the **application(\*.app)**, we need to create a new **launch configuration**.

Go to **Run -> Run As -> EHANDBOOK Container-Build ....**



### Configure a few settings:

**Name:** This will be used for the EHANDBOOK Container file name.

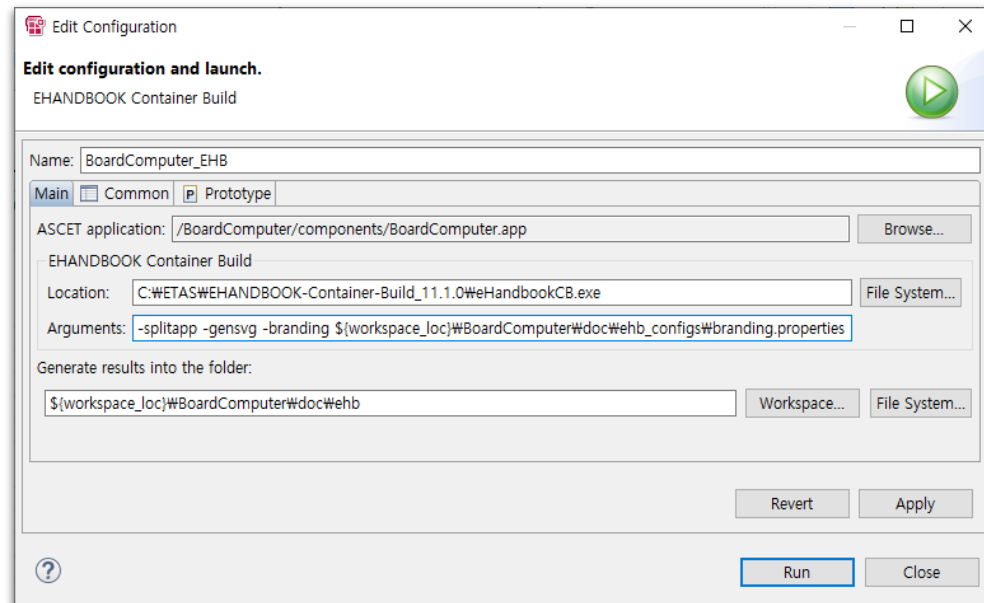
**ASCET application:** ASCET app file for which you want to generate an EHANDBOOK Container.

#### EHANDBOOK Container Build

**Location :** This path must point to the **eHandbookCB.exe** of the EHANDBOOK Container-Build tool we installed earlier.

**Argument :** Here you can provide additional **command-line parameters** to the EHANDBOOK Container-Build tool.

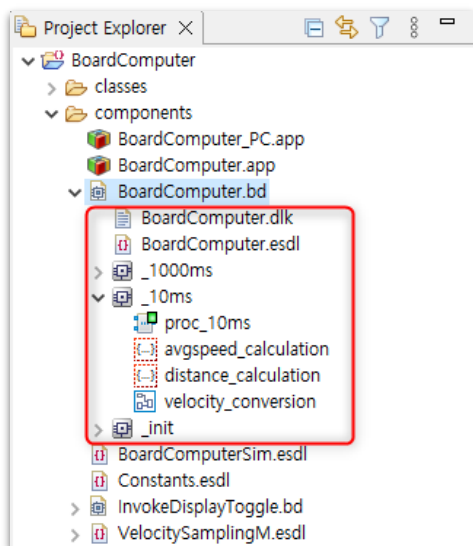
**Generate results into folder:** EHB Container-Build **results** shall be generated. This can be a location within the workspace or project, or another location on your system.



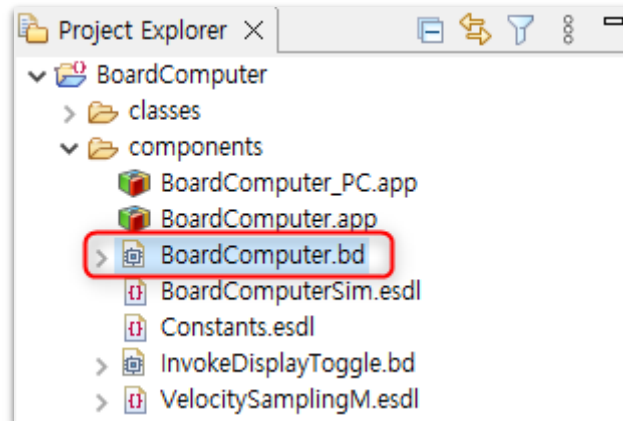
### ASCET-DEVELOPER authoring styles

EHANDBOOK supports two styles for how to author and interpret documentation contents which are attached to ASCET-DEVELOPER apps. In both approaches, the **documentation link files** (.dlk) files provided by ASCET-DEVELOPER keep track of the linking between model elements and documentation files.

#### Snippet mode

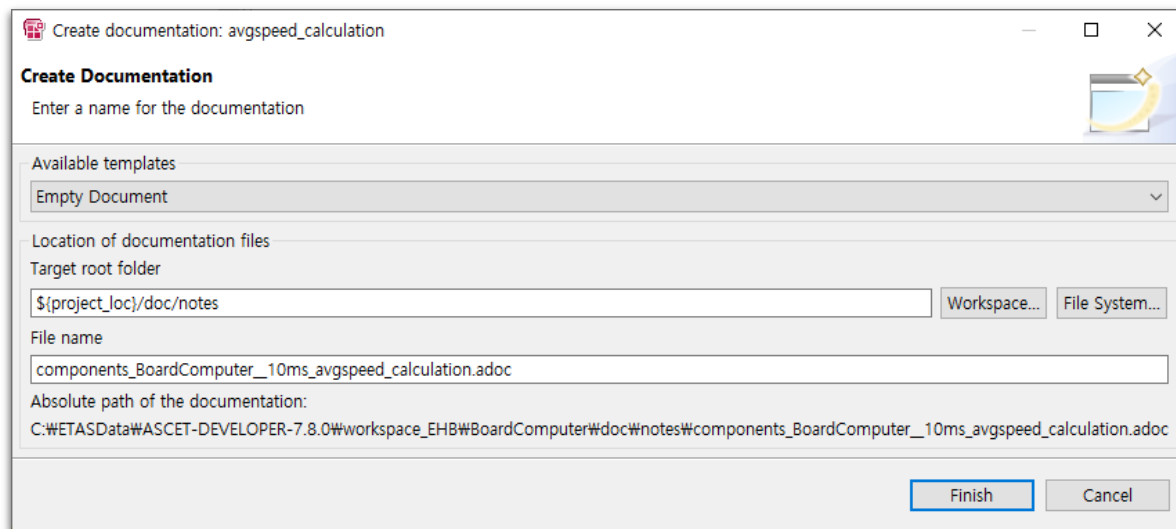
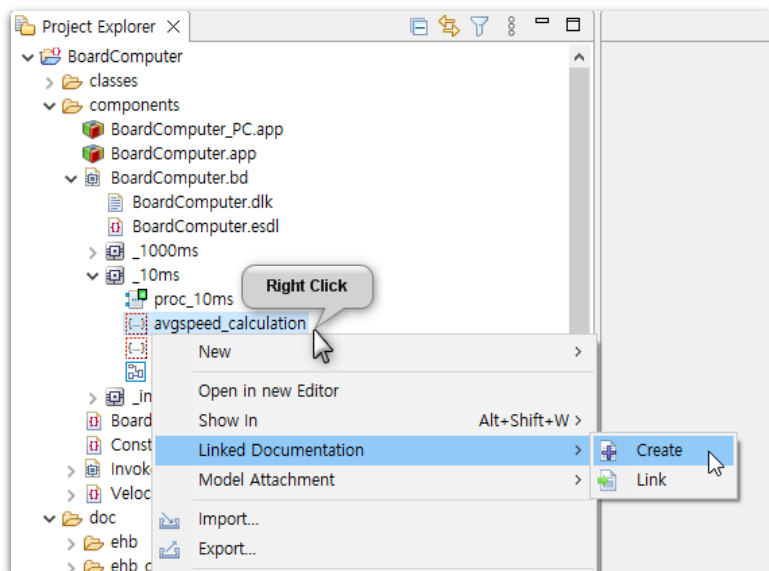


#### Expert authoring mode



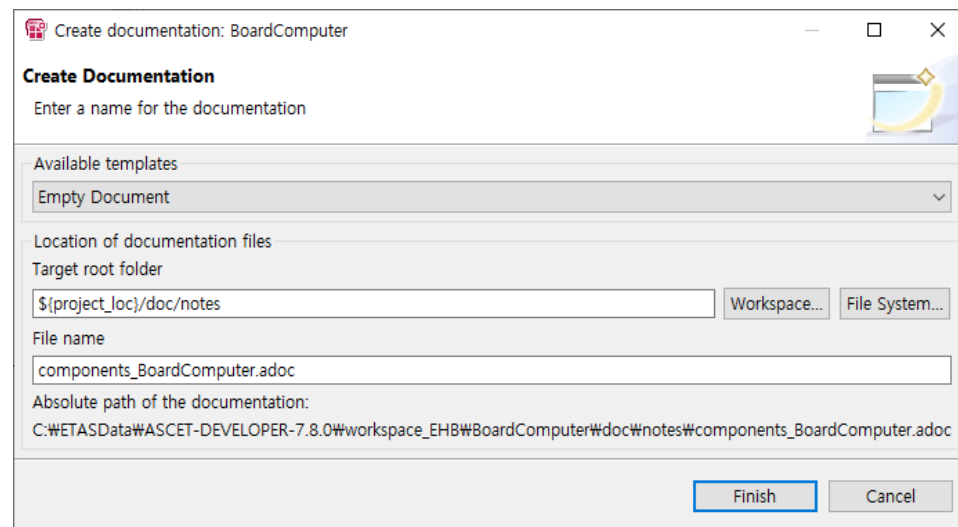
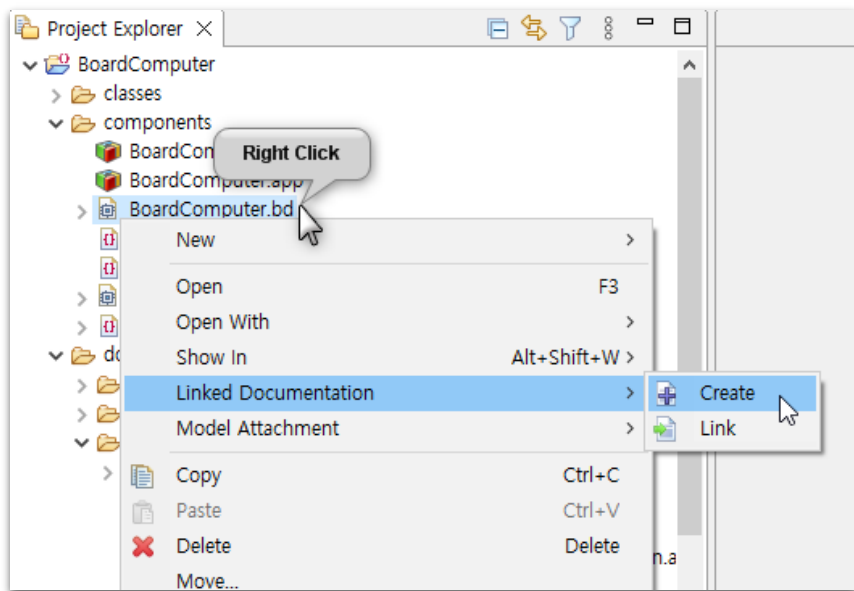
### ASCET-DEVELOPER authoring styles : **Snippet mode**

With ASCET-DEVELOPER, you can attach documentation snippets to all kinds of **model artifacts** such as ESDL files, block diagrams, specifications, containers or statement blocks, and even methods. Through this, small pieces of information are attached to the individual model elements that shall be documented.



### ASCET-DEVELOPER authoring styles : **Expert authoring mode**

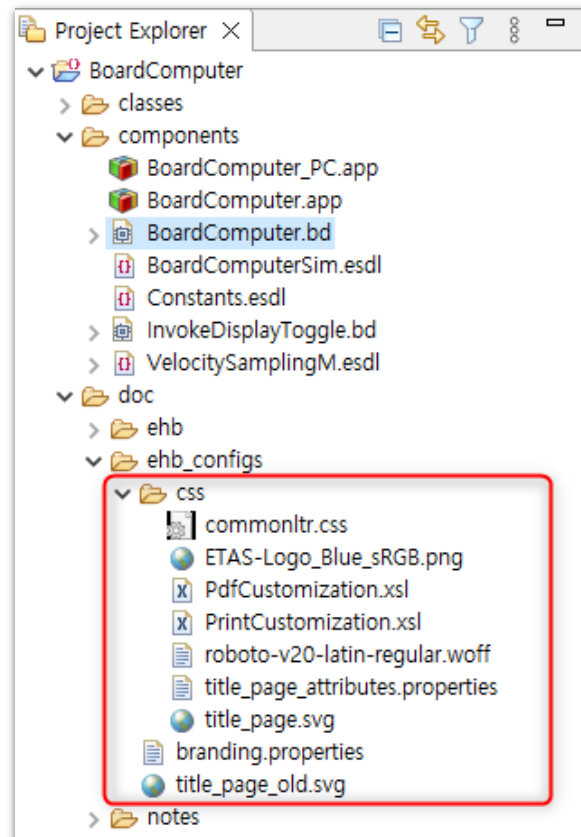
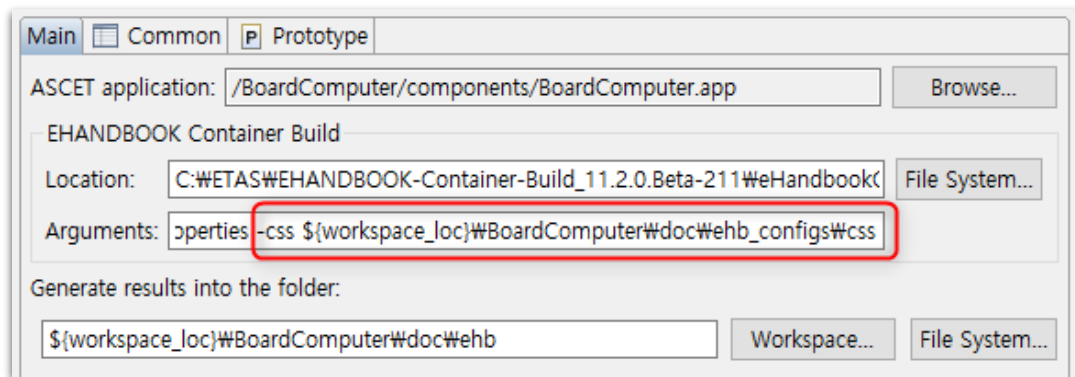
**A single documentation file** is attached to the block diagram or ESDL class that represents a **static entry point class**. The documentation file contains all your contents for the complete static entry point class. As an expert author, you have the complete degree of freedom for how to structure the contents of your documentation chapter.



### Applying custom branding and styling

The ECU software documentation files generated with EHANDBOOK Container-Build can be customized to adhere to the corporate identity of the documentation provider by applying the corresponding branding and styling as well as by providing a custom title page.

The customization is achieved by passing additional configuration files to EHB-CB via the **command-line interface**.



# EHANDBOOK-Container-Build ASCET

## Directory-based EHB-CB for ASCET

### Setting up the input data

To generate EHB Containers using the **Directory-based** EHANDBOOK Container-Build tools, the input data such as models, interface and label descriptions as well as textual documentation must be arranged in a directory structure. The directory structure must follow a specific convention:

- **A root directory:** This directory is passed to EHB-CB tool-chain as input directory
- **One or more sub-directories:** Each sub-directory represents a separate function
- **Within each sub-directory :** files with model data, interface and label description as well as textual description is placed

```
└── ECU_SW_XYZ
    ├── Function1
    │   ├── function_1_model.xml
    │   ├── interface_and_labels.xlsx
    │   └── textual_description.adoc
    .
    .
    .
    └── Function_n
        ├── function_n_model.xml
        ├── interface_and_labels.xlsx
        └── textual_description.adoc
```



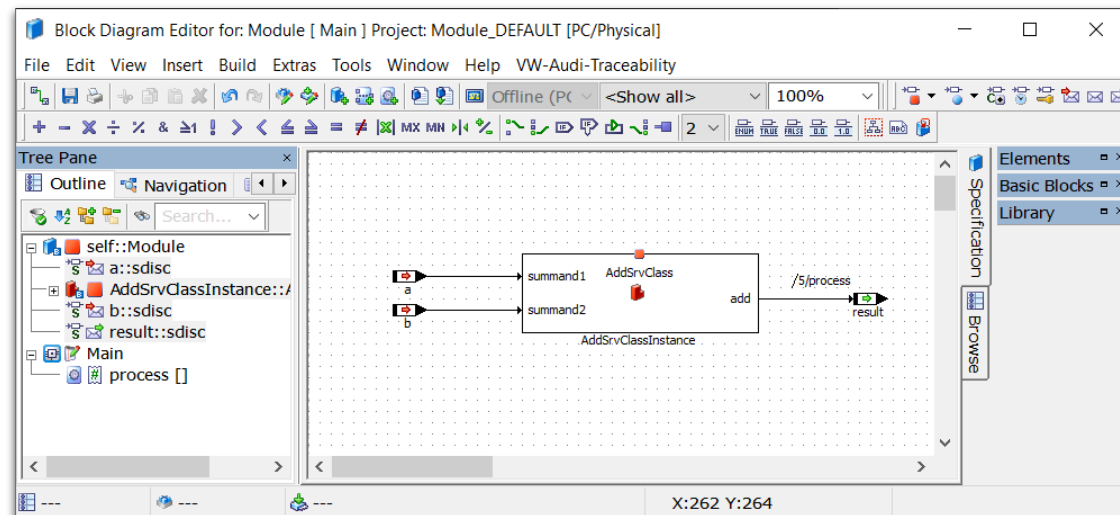
## Directory-based EHB-CB for ASCET

### ASCET model

EHANDBOOK Container-Build tools support generating interactive models for ECU functions developed in ASCET.

When using the EHANDBOOK Container-Build tool-Chain, ASCET models can be used as specification models for the algorithmic behavior of individual ASAM MDX-based ECU functions or AUTOSAR (Classic) software components.

Besides this, ETAS has developed customized add-ons for various customers to directly generate EHANDBOOK Containers from ASCET.



## Directory-based EHB-CB for ASCET

### Input data for label information and interface specification

To document information about the labels used by the ECU function as well as the interface specification in the EHB Container, this has to be described in the form of an **Excel** file which is stored in the same directory as the **ASCET** model sources.

The Excel file template comes with two sheets: one for **signals** and one for **calibrations**.

**Signals** are Global variables which **can be measured** with the help of measurement tools such as ETAS INCA.

Signals which are read and written by an ECU function are used to exchange data with other ECU functions.

**Calibrations** are Variables for which the values **can be changed / fine-tuned** with the help of calibration tools such as ETAS INCA. Calibrations can be e.g. single scalar values take the form of curves or maps which are interpreted by 1D or 2D interpolation routines.

	1	2	3	4	5	6
1	*	Mark	Xref	Class	Name	Description
2			IN	RAM	a	First summand
3			IN	RAM	b	Second summand
4			OUT	RAM	result	Sum (=result of the addition)
5						

	1	2	3	4	5	6
1	*	Mark	Xref	Class	Name	Description
2			LOC	VALUE	Param	A parameter
3						
4						
5						

## Directory-based EHB-CB for ASCET

### Textual Content

You can add textual documents in **RTF** or **AsciiDoc** format, together with the model file. This content is then transformed into the EHB container. Later it is shown in the function documentation window of the function component in EHB-NAV and describes verbally the functionality of the corresponding model. These documents can contain also images and screenshots of model hierarchies. You can add multiple documents per folder, even in both formats in parallel.

### Referencing to model screenshots

EHANDBOOK Container-Build can automatically generate a screenshots from **ASCET** based interactive models that can be embedded in the textual documentation.

To generate a **screenshot**, you have to create a model reference to the **ASCET** model and the subsystem you want to depict. This can be achieved with the EHANDBOOK model reference directive:

```
ehbmodelref:<ascet-6-model/hierarchy-path>[]
```

# EHANDBOOK-Container-Build

## ASCET for HMC

## EHB-CB for ASCET (for HMC)

### Installation

- ETAS EHANDBOOK-Container-Build for HMC  
EHANDBOOK-Container-Build\_XX.X.X\_HMC\_setup.exe

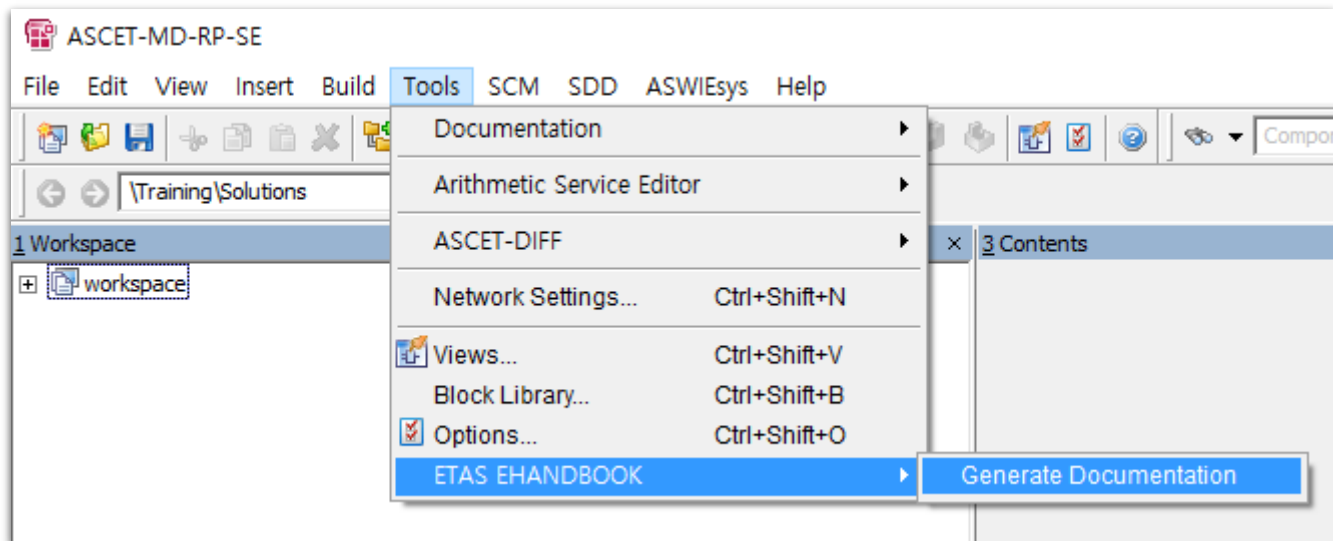
### Software Requirement

- ETAS ASCET V6.1  
EHB-CB requires ETAS ASCET V 6.1 installed on the target system before running the EHB-CB installer. After installation, EHB-CB is part of the Tools folder of your ASCET V6.1 installation.

## EHB-CB for ASCET (for HMC)

### EHANDBOOK Container and PDF Generation

- Open ASCET V6.1. Application
- Select the model
- Execute Generate Documentation : Tools > ETAS EHANDBOOK > Generate Documentation



### Generate Documentation

Generate documentation: EHANDBOOK Container-Build 11.2.1

Help

Documentation information

Project title

Software version

Specification version

Generation date12/ 1/2023

Build information

Notes folder

Styling folder

Output folder\*

EHB file name\*

ASCET view information

ASCET viewNo valid ASCET view file found. Check documentation.

Post Processing

☐ Encrypt Container

☐ Generate PDF

Generate

Cancel

Parameter	Description
Project title	Contains initially the name of the model selected from ASCET.
Software version	Specifies the software version of the documentation input.
Specification version	Specifies the specification version of the documentation input.
Generation date	Date of container and documentation creation.

### Generate Documentation

Generate documentation: EHANDBOOK Container-Build 11.2.1

Help

Documentation information

Project title

Software version

Specification version

Generation date12/ 1/2023

Build information

Notes folder

Styling folder

Output folder\*

EHB file name\*

ASCET view information

ASCET view

No valid ASCET view file found. Check documentation.

Post Processing

☐ Encrypt Container

☐ Generate PDF

Generate

Cancel

Parameter	Description
Note folder	Location of Notes folder.
Styling folder	Location of folder containing the styling information (title page, fonts, company logo etc...) For more details refer to the document <i>EHB-CB Styling Guide</i> . If no custom styles are provided, the default styling is used in EHB-NAV and PDF respectively.
Output folder	Output folder location of generated documentation.
EHB file name	Name of the container file name.



### Generate Documentation

Generate documentation: EHANDBOOK Container-Build 11.2.1

Help

Documentation information

Project title

Software version

Specification version

Generation date12/ 1/2023

Build information

Notes folder

Styling folder

Output folder\*

EHB file name\*

ASCET view information

ASCET viewNo valid ASCET view file found. Check documentation.

Post Processing

☐ Encrypt Container

☐ Generate PDF

Generate

Cancel

Parameter	Description
ASCET view	Select a view, which appears in the drop down list of views populated from the view file given in eHandbookCB.ini. Selecting <Show all> implies that there is no view applied and the model will not be reduced. Default option when a valid file is added is <Show all>.
Encrypt Container	Unchecked (default): No encryption of the container. Checked: Encrypt container with provided user password.  Password length: 4 - 255 characters Password characters: alphanumeric and special characters are allowed. Container gets the file extension .ehbcrypt
Generate PDF	Unchecked (default): No PDF documentation generated Checked: Generate PDF documentation in addition.

## EHB-CB for ASCET (for HMC)

### PDF Documentation

As an optional post processing step during the container generation, a PDF document can be created in addition. The PDF content is extracted out of the text panes per functional component of the container content (texts, model screenshots and other graphics).

### ASCET view configuration

For know-how protection purposes, ASCET allows designing the models for different views, e.g. it is possible to hide some element for a defined view. EHB supports this concept. With the GUI option “ASCET View” you are setting, for which view the model and its screenshots should be generated. The command `–DascetView= config-file.xml` in `eHandbookCB.ini` locates the file, where for every view name the corresponding model appearance is configured in a given format.

# EHANDBOOK-Container-Build

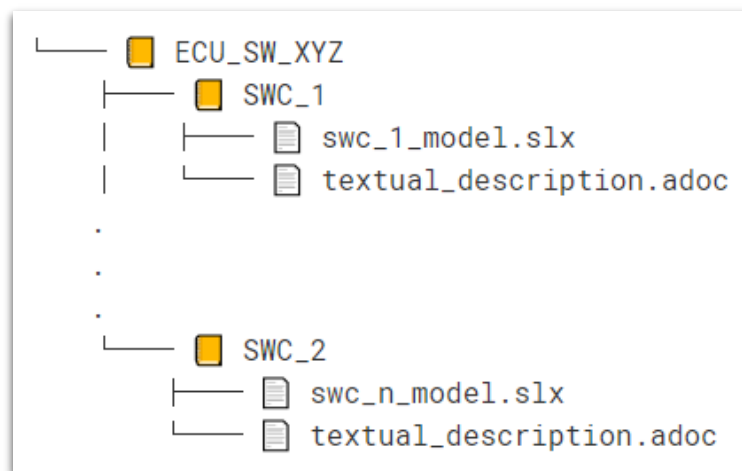
## AUTOSAR (Classic)

## Directory-based EHB-CB for AUTOSAR(Classic) .arxml files only

### Setting up the input data

To generate EHB Containers using the **Directory-based** EHANDBOOK Container-Build tools, the input data such as models, interface and label descriptions as well as textual documentation must be arranged in a directory structure. The directory structure must follow a specific convention:

- **A root directory:** This directory is passed to EHB-CB tool-chain as input directory
- **One or more sub-directories:** Each sub-directory represents a separate function
- **Within each sub-directory :** files with model data, interface and label description as well as textual description is placed



## Directory-based EHB-CB for AUTOSAR(Classic) .arxml files only

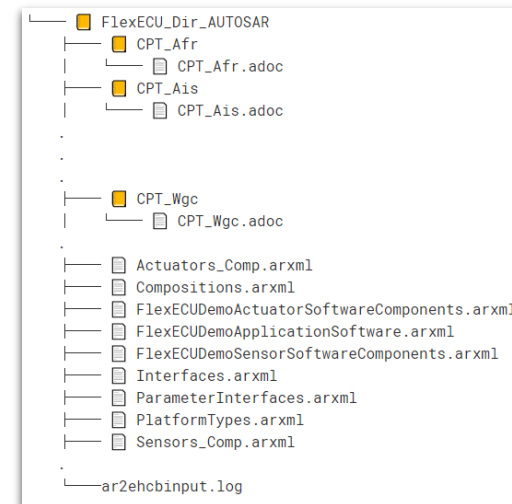
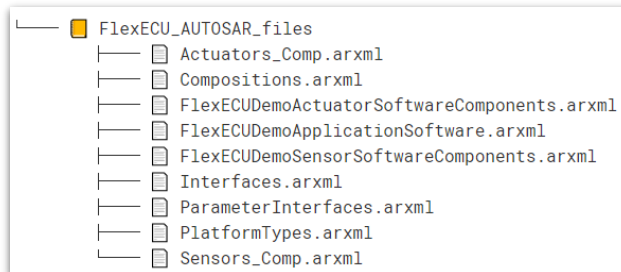
### Generate directory structure for EHB-CB from AUTOSAR .arxml files

For AUTOSAR (Classic)-based projects, the **directory structure can be automatically generated** with the help of EHANDBOOK Container-Build tools from the AUTOSAR .arxml files of the project.

Create a directory with the name you want. This will be the directory where EHANDBOOK Container-Build will create the directory structure for all the software components in the .arxml files.

The command-line option **-ar2ehbcbinput** is used to initialize and generate EHANDBOOK folder structure from AUTOSAR files.

```
C:\ETAS\EHANDBOOK-Container-Build...\eHandbookCB.exe ^  
-i "FlexECU_AUTOSAR_files" ^  
-n "FlexECU_Dir_AUTOSAR" ^  
-o "FlexECU_Dir_AUTOSAR" ^  
-ar2ehbcbinput ←
```



## Directory-based EHB-CB for AUTOSAR(Classic) .arxml files only

### **The contents of the generated directory structure can be further customized:**

- Delete folders of software components that shall not be documented
- Add the corresponding ASCET or Simulink model file to obtain a specification for the internal behavior of the software component.
- Add the corresponding C-Code sources to obtain an interactive model for the implementation of the internal behavior of the software component
- Alter the textual contents of the generated AsciiDoc files

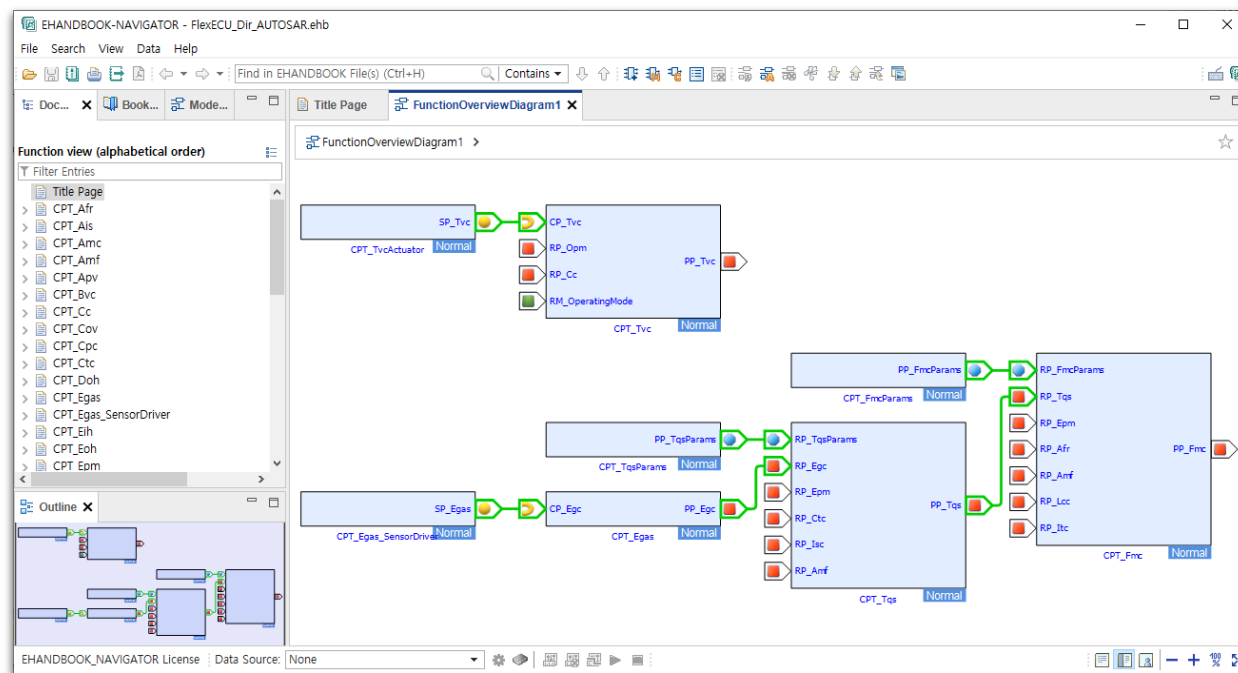
## Directory-based EHB-CB for AUTOSAR(Classic) .arxml files only

### Generate EHANDBOOK Container

To generate the EHANDBOOK, run the following command:

```
C:\ETAS\EHANDBOOK-Container-Build...\eHandbookCB.exe ^  
-i "FlexECU_Dir_AUTOSAR" ^  
-o "FlexECU_Dir_AUTOSAR_Output" ^  
-n "FlexECU_Dir_AUTOSAR" ^
```

- i : Input directory for generating the EHANDBOOK Container
- o : Output directory
- n : Name of generated EHANDBOOK Container file



# EHANDBOOK-Container-Build

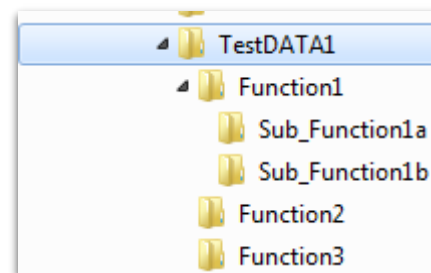
## How to guides



## Directory Based EHB-CB

### Input Directory Structure

This structure is 1:1 used to generate the table of contents in NAV.  
Two TOCs will be created: SW architecture and flat alphabetical.



#### Model could be:

- Preprocessed ANSI c-code as \*.c and \*.h in ASCII text, saved in UTF-8 encoding format (multiple files allowed)
- C-Code models packed as \*.zip also possible
- Simulink model as \*.mdl with MATLAB version R2016b - R2020b or \*.slx with MATLAB version R2016b - R2022b
- Simulink models packed all together as one .zip wherein zip name is exactly same as main model name (.slx or \*.mdl) also possible
- ASCET model as \*.axl out of ASCET V5.2.1 – V6.4.3

## How to provide attachments to EHB-CB

### Directory Based EHB-CB

In the directory-based EHB-CB approach, all files which shall be added as attachments must be stored in a sub-directory named **\_attachments**. During EHANDBOOK Container-Build, the files are picked up and added as attachments.

```
Tqs
|_ Tqs.adoc
|_ Tqs.slx
|_ Tqs.xslx
|_ _attachments
|_   Tqs_Calibration_Guide.pdf
|_   Tqs_Initial_Calibration_Values.dcm
|_   Tqs_Variables_List.lab
```

### ASAM Based EHB-CB

In the ASAM-based EHB-CB approach, an <ABLOCK> with category **ATTACHMENTS** has to be created and linked to the documentation chapter to which the attachments shall be associated.

```
<ABLOCK>
...
<AREFS>
  <AREF ID-REF="ID_Tqs_Attachments" />
</AREFS>
...
</ABLOCK>

<ABLOCK ID="ID_Tqs_Attachments">
  <SHORT-NAME>Tqs_Attachments</SHORT-NAME>
  <CATEGORY>ATTACHMENTS</CATEGORY>
  <FILES>
    <FILE>_attachments\Tqs_Calibration_Guide.pdf</FILE>
    <FILE>_attachments\Tqs_Initial_Calibration_Values.dcm</FILE>
    <FILE>_attachments\Tqs_Variables_List.lab</FILE>
  </FILES>
</ABLOCK>
```

## How to customize EHB Containers

### Container Encryption

Encrypted containers can be generated by using the command parameter `-epwd` or `-epwdf`. The password can be passed either in the command line e.g. `-epwd "qwertz"` or within a password file e.g. `-epwdf ./my_password_file.txt`

The password should satisfy the following credentials:

- Password length: minimum 4; maximum 255 characters
- Password characters: alphanumeric, special characters are allowed
- Password pattern: No special pattern is required

Thank you!