

SCODE Workbench 3.0

Release Notes

Copyright

The data in this document may not be altered or amended without special notification from ETAS GmbH. ETAS GmbH undertakes no further obligation in relation to this document. The software described in it can only be used if the customer is in possession of a general license agreement or single license. Using and copying is only allowed in concurrence with the specifications stipulated in the contract.

Under no circumstances may any part of this document be copied, reproduced, transmitted, stored in a retrieval system or translated into another language without the express written permission of ETAS GmbH.

© **Copyright 2021** ETAS GmbH, Stuttgart

The names and designations used in this document are trademarks or brands belonging to the respective owners.

MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See [mathworks.com/trademarks](https://www.mathworks.com/trademarks) for a list of additional trademarks.

SCODE Workbench V3.0 – Document R01 EN – 04.2021

Contents

1	Introduction.....	4
1.1	SCODE-ANALYZER.....	4
1.2	SCODE-CONGRA	4
1.3	References.....	4
1.4	Presentation of Supporting Information	4
1.5	User Documentation.....	5
2	Product Definition	6
2.1	Functions at a Glance.....	6
2.1.1	SCODE-ANALYZER.....	6
2.1.2	SCODE-CONGRA.....	6
2.2	General Description.....	6
2.2.1	System Prerequisites	6
2.2.2	Software Prerequisites	7
2.2.3	Release Test Configuration	7
2.3	Delivery	8
2.3.1	Used 3rd Party Software	9
2.4	Installation.....	9
2.5	Licensing	9
3	Changes.....	10
3.1	What's New in SCODE Workbench 3.0.....	10
3.1.1	General.....	10
3.1.2	SCODE-ANALYZER.....	10
3.1.3	SCODE-CONGRA.....	13
3.2	Compatibility to Earlier Releases.....	15
3.3	Known Issue Reports	15
3.4	Known Issues	15
3.4.1	SCODE-ANALYZER.....	15
3.4.2	SCODE-CONGRA.....	18
4	Notes.....	25
4.1	General.....	25
4.2	SCODE-ANALYZER.....	25
4.3	SCODE-CONGRA	25
4.4	MATLAB® Connection.....	25
5	Contact Information	27

1 Introduction

SCODE Workbench comprises the following two components:

- SCODE-ANALYZER
- SCODE-CONGRA

1.1 SCODE-ANALYZER

SCODE-ANALYZER employs the SCODE Essential Analysis approach, which is a structured method used to increase the understanding of a problem and to reduce the complexity in the subsequent software and systems design. Often the term SCODE is used as a synonym for the SCODE Essential Analysis method. However, in a more general way, SCODE stands for System Co-Design, thus the co-engineering across different disciplines, e.g. control, software and hardware engineering.

Focus is on mastering the complexity of the software in automotive systems. Drivers for the complexity are increasing functionality, real-time behavior, and customer variance. The Essential Analysis for physically dominated systems is an analysis method that allows reducing the system complexity to the unavoidable inherent problem complexity. This method has been applied successfully on several product relevant subsystems.

1.2 SCODE-CONGRA

SCODE-CONGRA is designed to help you define and analyze continuous systems, simulate them and generating code.

The novel approach is that the system is described purely in form of variables, relations, and equations.

The equations are "undirected" and depending on which variables are marked as inputs, the equations are solved in the corresponding direction and code is generated representing the result of this direction of equations.

SCODE Workbench is an Eclipse-based product. If you are familiar with using an Eclipse environment, then you should feel at home.

1.3 References

- SCODE Workbench Getting Started.pdf
- Safety Advice.pdf

Both are available in the installed product.

1.4 Presentation of Supporting Information

`OCI_CANTxMessage msg0` = Code snippets are presented in the Courier font. Meaning and usage of each command are explained by means of comments. The comments are enclosed by the usual syntax for comments.

Choose File > Open .	Menu commands are shown in boldface.
Click OK .	Buttons are shown in boldface.
Press <ENTER>.	Keyboard commands are shown in angled brackets in small caps.
The "Open File" dialog box is displayed.	Names of program windows, dialog boxes, fields, etc. are shown in quotation marks.
Select the file <code>setup.exe</code>	Text in drop-down lists on the screen, program code, as well as path- and file names are shown in the Courier font.
A <i>distribution</i> is always a one-dimensional table of sample points.	General emphasis and new terms are set in italics.

1.5 User Documentation

The SCORE Workbench user's documentation is provided as PDF (Getting Started) and online help. The online help is available at any time via the **Help → Help Contents** menu.

The Getting Started manual in PDF format can be found on the installation disk and in the `documents` subfolder of your installation directory.

2 Product Definition

2.1 Functions at a Glance

2.1.1 SCODE-ANALYZER

SCODE-ANALYZER yields a decomposition of the overall problem in several smaller sub-problems, which can be solved separately and more easily. The integration of the sub-problem solutions then provides the overall solution to the original problem. Main features of the tool are:

- An editor to define the input and output space by a Zwicky box in terms of Dimensions and Alternatives
- An editor to define modes by assigning input sets to output sets taking into account different analyses on input and output spaces
- An editor to define mode transitions to specify which context changes cause a transition between system modes
- A graphical editor to show the decision tree for modes and transitions which can be optimized to find the shortest transition between system modes
- Extensible code generators that can translate the system into code. This code can be executed or simulated, or post processed and integrated in real life production code.

2.1.2 SCODE-CONGRA

SCODE-CONGRA is a tool for designing system and generating code, both simulation and production code. The system can be described both graphically and textually. Main features of the tool are:

- A graphical editor enabling you to describe the content of your system using common graphical editing patterns as known from modeling tools
- A textual editor for the in the "System Equation Language" (syq) enabling you to describe the content of the system in an alternative way.
- An execution environment that answer important questions on the solvability, stability, as well as sensitivities in the specified system.
- Extensible code generators translate a given system into code. This code can be executed or simulated, or post processed and integrated in real life production code for e.g. control applications.
- A coupling with a simulation tool, which allows you to run the generated code on another platform and validate your model.

2.2 General Description

2.2.1 System Prerequisites

The following minimum system prerequisites must be met:

Required Hardware	2.0 GHz Dual-Core PC or equivalent 4 GB RAM DVD-ROM drive (applicable for DVD based installation only) Network adapter Graphics with a resolution of at least 1024x768
Required Operating System	Windows® 10 (x64)
Required Free Disk Space	800 MB (not including the size for application data)

The following system prerequisites are recommended:

Recommended Hardware	2,0 GHz Quad-Core PC or equivalent 8 GB RAM DVD-ROM drive (applicable for DVD based installation only) Network adapter Graphics with a resolution of 1920x1080
Recommended Operating System	Windows® 10 (x64)
Recommended Free Disk Space	>2,0 GB

2.2.2 Software Prerequisites

SCODE-CONGRA will require a Computer Algebra System (CAS) to execute most of its functionality. The free CAS Maxima is provided with the installer. If installed during the installation of SCODE-CONGRA, Maxima is also configured correctly in the tool.

An installation of MATLAB® is required for Simulink integration between SCODE Workbench and MATLAB® and Simulink®.

2.2.3 Release Test Configuration

2.2.3.1 Host Platform

SCODE Workbench has been tested on the following host platforms:

- Windows 10 (64-bit, versions 1903, 1909)

2.2.3.2 Java™ Platform

SCODE Workbench has been tested with the following JRE:

- openjdk version "11.0.9" 2020-10-20
- OpenJDK Runtime Environment AdoptOpenJDK (build 11.0.9+11)
- OpenJDK 64-Bit Server VM AdoptOpenJDK (build 11.0.9+11, mixed mode)

2.2.3.3 Eclipse Target Platform

SCODE Workbench has been tested on the following Eclipse platform releases:

Eclipse Platform Releases	Product Platform	Update Site
Eclipse 2020-09 (4.17)	✓	

Product platform indicates the version of the Eclipse workbench installed when you install SCODE Workbench from product installer.

Update site indicates the version(s) of Eclipse that are supported when installing SCODE Workbench from a compatible update site on the product DVD.

The Eclipse target platform configuration for each of these releases is provided on the DVD in the folder `EclipseTargetPlatforms`.

You can access the Eclipse target platform configuration for your using the following steps:

1. Install the product.
2. Start SCODE.
3. Go to **Help → About SCODE Workbench**.
4. Click "Installation Details".
5. Select the "Configuration Tab".

2.2.3.4 3rd Party Products

This release of SCODE Workbench has been tested using the following products:

Product	Version
MathWorks®: MATLAB® and Simulink®	MATLAB 2016b, 2017b, 2018b, 2019b
QTronic FMU SDK ¹	2.0.3

2.3 Delivery

The software is delivered with an installation routine on a DVD including SCODE Workbench software, documentation, tools, utilities, and further information. All software documentation is available in the Portable Document Format (PDF), which requires a PDF Reader.

The DVD contains the following items:

File or Directory	Description
<code>Autostart.exe</code>	The executable for starting the ETAS Product Installation program.
<code>Start.exe</code>	The ETAS Product Installation program.
<code>Start.ico</code>	The SCODE Workbench icon.
<code>install\ETAS SCODE Workbench 3.0</code>	Directory containing the software installation of SCODE Workbench.
<code>\Documentation</code>	Directory containing copies of the PDG user documentation.

¹ <https://www.qtronic.de/doc/fmusdk.zip>

\Documentation\Open Source Software	Directory containing information about 3rd party open source software used in or provided for use with SCODE Workbench. Including copies of the licenses under which this software is used.
\HTML	Directory containing text and images for the DVD.

2.3.1 Used 3rd Party Software

SCODE Workbench makes use of third-party software components. Licensing information for these components can be found in `Documentation\SCODE Workbench 3.0 OSS Attributions.pdf`.

2.4 Installation

SCODE Workbench is distributed as a standard Microsoft Windows installer. Run `setup.exe` and follow the on-screen instructions to install the SCODE Workbench product.

Installation procedures and hints can be found in `Documentation\SCODE Workbench 3.0 Getting Started.pdf`.

2.5 Licensing

Licenses can be ordered via your regional ETAS sales representative.

For evaluation of the product, there is the possibility to request evaluation licenses with a 3 months duration. Please contact your regional ETAS Sales Representative.

Starting from SCODE Workbench 3.0 Release, all machine based licenses for SCODE-ANALYZER and SCODE-CONGRA will be issued with a new technology Flex Net Embedded. For more details see https://www.etas.com/download-center-files/company/License_Management_FAQ_20200730.pdf.

3 Changes

This chapter describes changes with respect to the previous version of SCODE Workbench.

3.1 What's New in SCODE Workbench 3.0

3.1.1 General

- The underlying Eclipse platform is upgraded to 2020-09 4.17.
- File properties have been removed.
- Synchronization between Workspace preferences and project properties has been improved.
- SCODE specific workspace preferences can be exported and imported.
- A welcome view has been added to the product.
- Product examples have been added: SI Units, Plant model library, YA-KINDU Traceability®.

3.1.2 SCODE-ANALYZER

3.1.2.1 New and modified options and features

Code Generation

- Code can only be generated when the model has been saved to disk.
- The format of generated code has been improved.
- Comparison with literals are normalized, so that literals are always placed on the right hand side of the operator.
- Configurable naming templates have been added to the generation preference page to allow customization of input argument names and enumeration types.
- Validations on implementation names and representation values have been added to ensure unambiguous naming while editing implementation views.
- There are now three options available for "*Dimension and mode representation*":
 - "*Enumeration*"
 - "*Use custom representation - default: Integer*"
 - "*Use custom representation - default: Enumeration*"



CAUTION

Usage of **custom representations** is at the user's own risk. Conflicts between competing definitions cannot and will not be detected, and can lead to false negative and false positive fingerprint verification results (see User Guide Chapter Code Generation for details).

- The option *"Use integer as default representation"* has been replaced with *"Use custom representation - default: Integer"*.

C code generation

- Generated C code is now compliant with the following MISRA C:2012 rules 16.3 and 12.1.

C++ code generation

- A C++ code generate has been added to SCORE-ANALYZER. The generated functions are encapsulated in stateless classes.

Decision Trees

- The option *"Hide Mode"* now also hides "dead" branches.
- An option *"Focus on Mode"* has been added to focus only on selected modes (the opposite operation to *"Hide mode"*).
- The *"Reload"* button has been reworked to set the model in an unsaved state.

ESDL code generation

- *"ASCET 6.x generation"* now supports AMD/AXL version 6.4.5.

Export

- A SCORE-ANALYZER model can now be exported to the JSON format.
- Multiple decision trees and the transition graph can now be exported in one step to SVG or graphML files.

Fingerprint computation

- The computation of the fingerprint can now be aborted.

MATLAB® and Simulink

- Connections between SCORE-ANALYZER and MATLAB R2019b can now be established. Supported MATLAB versions include R2016b, R2017b, R2018b, and R2019b.
- The structure in the Simulink library browser now provides a single entry point for SCORE Workbench library blocks.

MATLAB® code generation

- The model data can be exported to a **_modelData.m* file with a MATLAB function which contains the model data structure.
- Execution of a customer script can be triggered to automatically process this model data, e.g. import in a Simulink Library Data Dictionary (SLDD).

Mode Transition Matrix

- A new semantic of the mode transition behavior *"Based on target mode"* has been added to the tool (walkthroughs are not considered).
- It is now possible to use *"INFO"* dimensions in event rules.

Report generation

- An EHANDBOOK report generator has been added to the tool.
- Reports now also contain comments and textual representations of mode rules and event rules.

Requirements linking

- Installation of YAKINDU Traceability® by itemis is now supported to allow linking of requirements to model elements of SCORE-ANALYZER.

Simulink Integration

- The code for the ModeSelector of the MATLAB® function block has been relocated to a separate *m file*. The MATLAB function lying under the mask of the Simulink block now calls this separate file.
- Simulink S-function generation is now supported with a reworked code generation approach. This includes generation of an S-function in MATLAB syntax and, alternatively, as a MEX function.
- Two separate scripts are now generated to create new and update existing and connected MATLAB function and S-function blocks.
- A connection can be established with either a library block or one instance derived from it at a time.
- For output type *"Modes and Actions"* additional create and update scripts are generated for two new Simulink blocks: *Action function* and *SelectorAction function*. A MATLAB connection can be established between all three Simulink blocks and the SCORE-ANALYZER file from which they originate at a time.

TPT Export

- A new TPT exporter generates native TPT test suites which can be directly opened with the ETAS RT2 testing tool (version 7 and later) and the TPT tool by PikeTek® (version 7 and later).
- Test suite generation now supports walkthroughs.
- An option to generate separate TPT files for each source mode is provided to reduce the size of the single test suite.

3.1.2.2 Discontinued options and features

- Property settings on file level
- Support of *"Built-in"* symbols in implementation views
- Generation options:
 - *"Word size of target"*
 - *"Generate Infrastructure files"*
 - *"Generate despite the presence of errors"*
- Dimension representations:
 - *"Structs of Enumeration"*
 - *"Zwicky-Box scoped bit field (only for C)"*
- Rule generation type *"Using Rule Definition as single function"*
- ESDL code generator options:
 - *"Generate mode transition as state machine"*
 - AMD/AXL versions 6.2.1 and 6.3.1
- C code generator options:
 - *"Use Macros"*
 - *"Use Extended Macros"*

- *"Code Size Measurements"*
- Simulink integration (connection of blocks) with MATLAB version R2015b
- Mapping ANALYZER to CONGRA
- TPT file export
 - File format *"TEXT"*
 - Test suite type *"BFS"*

3.1.3 SCODE-CONGRA

3.1.3.1 New and modified options and features

C code generation

- User-specific service routines are considered in the generated code.
- Additional headers and libraries can be specified by the user.
- Generated C code is now compliant with selected MISRA C:2012 rules.

Code generation

- Triggering of the code generation for a specific project is accessible via context menu.
- Integrator and delay chains are now automatically resolved during code generation by adding chain variables.
- Variables that are input and output simultaneously are handled correctly by introducing forwarded variables.
- A configurable naming template for function inputs is provided.
- The format of generated code has been improved.
- Comparison with literals are normalized.
- Error handling in error cases, e.g. division by zero, can be configured in more detail.
- Options for code styling have been added.

EHANDBOOK report generator

- An EHANDBOOK report generator has been added to the tool.

ESDL code generation

- The `amd` sub-folder for *"ASCET 6.x model"* export has been moved to the ESDL generation location.
- Support has been added for ASCET 6.4.5.
- Additional *"doStep"* is generated to perform timing updates.

Export graphs

- All graphs of a SCODE-CONGRA model can be exported in a single action in SVG or GraphML format.

FMU generation

- FMU can be generated for co-simulation and contain a compiled dll.

Graphical editor

- The context menu to set the type of a variable has been restructured, and multi-select is available.

MATLAB and Simulink

- Support for MATLAB 2019b has been added, support for MATLAB 2015b has been removed.
- The structure of SCODE blocks in the Simulink library browser has been reworked.

Requirements linking

- Installation of YAKINDU Traceability® by itemis is now supported to allow linking of requirements to model elements of SCODE-CONGRA.

Units

- Tearings using variables with units are not allowed.
- Code generated from models with units must not be used for productive applications.

3.1.3.2 Discontinued options and features

C code generator

- "Error code to return for invalid solution assumptions"

Code generation

- Descriptions for relations and variables via the annotation `@description("...")` are no longer translated as comments in the generated code.
- Code generation for models containing differentiate edges/`@state` annotations is no longer supported.
- "Separator for derivatives in derived artifacts"
- "Generate extended variable information"

Diagnosis

- The diagnosis feature has been removed.

MATLAB® code generator

- "Code generation for partial derivatives":
- "Generate Jacobian matrix for Extended State Space computations"
- "Add partial derivatives as outputs"
- "Use custom two-sided limiter function"
- "Generate tearing accuracy and iteration number as workspace parameters"

Verification

- "Generate inline verification"

General

- "Immediate structural analysis validation"

3.2 Compatibility to Earlier Releases

It is recommended to use a new workspace with each new release. Additional semantic checks and analyses introduced may cause models originally developed in earlier releases to generate new warnings and errors.

Update sites for import and conversion of out-dated mint and mxe files are no longer supported. For conversion from mint or mxe the user should use SCODE Workbench release 2.6.3 and the corresponding update sites.

Users may have to perform 'Configure' to convert existing projects from an older version to a SCODE-ANALYZER nature or a SCODE-CONGRA nature for this release.

3.3 Known Issue Reports

If a product issue develops, ETAS will prepare a Known Issue Report (KIR) and post it on the internet. The report includes information regarding the technical impact and status of the solution. Therefore, you must check the KIR applicable to this ETAS product version and follow the relevant instructions prior to operation of the product.

The Known Issue Report (KIR) can be found here: <https://www.etas.com/kir>

3.4 Known Issues

This section describes the set of known problems.

3.4.1 SCODE-ANALYZER

Issue Reference	Description
SCT-86	<p>Warnings are logged in the .log file when commands have no category defined</p> <p>The *.log file may contain warnings from an Eclipse plug-in that commands should have a category.</p> <p>Workaround: None, since these warnings have no functional impact.</p>
SCT-88 SCT-19489	<p>Decision tree: performing undo/redo multiple times leads to crash the layout</p> <p>Workaround: Click on Reorder Globally to retrieve the graph.</p>
SCT-762	<p>Dropdown for description font and size in the property page is shown only half</p> <p>Workaround: Clicking on the dropdown lets you choose the font and size.</p>
SCT-993	<p>Simulink integration may not work if multiple versions of SCODE Workbench are open at the same time</p> <p>Workaround: Close all running SCODE Workbench instances except the one to be used for Simulink integration.</p>

Issue Reference	Description
SCT-13163	<p>Folders and projects may be renamed by invalid names</p> <p>It is possible to use invalid names for folder and project names via F2 or rename . This may impede code generation.</p> <p>Workaround: Rename folders and projects watchfully.</p>
SCT-18088	<p>Code is generated to wrong target folder</p> <p>Code for the model <code>ab.c.scde</code> is generated to the same folder as the code for model <code>ab_c.scde</code>.</p> <p>Workaround: Do not use dots in filenames.</p>
SCT-18503	<p>Default transition behavior 'Based on target mode' does not consider walkthroughs</p> <p>Workaround: Define events to explicitly exclude the walkthrough target states.</p>
SCT-18774	<p>Workspace caches error states from deleted project</p> <p>Workaround: Disable 'Nested Projects' in the 'Content' tab of the 'Filters and Customization ...' menu of the project explorer</p>
SCT-18794	<p>'Copy-paste' and 'Rename (F2)' allow invalid project and model names</p> <p>Workaround: See SCT-13163.</p>
SCT-19127	<p>PDF report may have truncated table cell content</p> <p>Workaround: Create a WORD report, open the created report file with Microsoft Word and save the document as PDF.</p>
SCT-19153	<p>No rule suggestions for inconsistencies in non-system mode</p> <p>Workaround: Toggle the non-system mode to a system mode, fix the overlapping rules and toggle the system mode back to a non-system mode.</p>
SCT-19455	<p>Declared package does not match the expected package in generated ESDL code</p> <p>Target folders for ESDL code in SCODE Workbench are not recognized as a valid ESDL package.</p> <p>Workaround: Copy the folder that contains the generated artifacts to the desired ASCET-DEVELOPER workspace.</p>
SCT-19659	<p>Report may look strange in WORD and PDF format when YAKINDU Traceability is installed and active</p> <p>Workaround: Temporarily deactivate the YAKINDU Traceability license for report generation. Alternatively, use HTML or EHANDBOOK as report file type.</p>
SCT-19673	<p>Build view shows green flag even if no MATLAB code is generated</p> <p>SCODE Workbench does not report that code generation fails because MATLAB has locked an m-file.</p> <p>Workaround: Make sure MATLAB does not lock files which are expected to be replaced during code generation.</p>

Issue Reference	Description
SCT-19696	<p>Global and local ordering of decision tree can not be changed in directed acyclic graph (dag)</p> <p>Workaround: Toggle to tree view, rearrange order and toggle back to dag.</p>
SCT-19705	<p>Decision tree for outgoing transitions of a mode does not show condition dimension nodes which are involved in outgoing transitions when all other alternatives of this condition dimension belong to the 'No transition' event of this mode.</p> <p>Workaround: Click the 'Show non-transitions' button in the toolbar of the decision tree to display all nodes involved in the decision tree.</p>
SCT-19706	<p>Rule proposals don't copy to the clipboard</p> <p>A rule proposal in the 'Analysis Details' view is not copied to the clipboard using Ctrl+C and right click 'Copy'.</p> <p>Workaround: Drag the rule from the 'Analysis Details' view to the rule editor; or right click the rule, select 'Show In Rule Editor' and click the 'Add Include/Exclude Rule' button; or click in any other view before copying the rule.</p>
SCT-19728	<p>SCODE Workbench editors do not return GDI objects</p> <p>The performance of the tool may degenerate when the number of GDI objects approaches 10.000 (see task manager).</p> <p>Workaround: Close and restart SCODE Workbench when performance decreases or the number of GDI objects is high.</p>
SCT-19782	<p>Fingerprint mismatch when model has an empty mode</p> <p>Verification code reports a fingerprint mismatch for generation source 'Mode Transition Matrix' when model has empty mode because no code is generated in order to avoid dead code.</p> <p>Workaround: Delete all empty modes or use the 'Purge Zwicky Box Container' command.</p>
SCT-19784	<p>Updating the decision tree does not set model in an unsaved state</p> <p>Clicking the link "Results may not be up to date. Click here to update" on a saved model does not make the "Save" command available.</p> <p>Workaround: Drag any node in the tree to a different position or click the "Reload and shuffle tree" button.</p>
SCT-19798	<p>'Purge Zwicky Box Container' command is unavailable</p> <p>The 'Purge Zwicky Box Container' command does not detect that a model is not clean when it has empty modes but no empty event.</p> <p>Workaround: Add an empty event, save the model and then execute the 'Purge Zwicky Box Container' command.</p>

3.4.2 SCODE-CONGRA

Issue Reference	Description
SCT-9, SCT-19088	<p>Rename of folder causes errors in SYQ file</p> <p>Renaming a folder (aka package) the package name in the SYQ files does not update and causes errors.</p> <p>Workaround: Manually copy the files to the new package folder and rename the package internally also manually.</p>
SCT-96	<p>Execution Environment does not support nested tearing</p> <p>Computations with nested tearings cannot be executed in the execution environment.</p> <p>Workaround: Use execution environments for the generated code, e.g. of the MATLAB or C code.</p>
SCT-192	<p>Solutions manually provided to the cache may be ignored</p> <p>When providing manually a solution to the cache, if the request is a conditional equation then the solution has to be conditional otherwise it will be ignored.</p> <p>Workaround: Provide the same solution in the else expression and the solution will be used.</p>
SCT-211	<p>Integrate/Delay relations with arguments of libraries is not possible</p> <p>Modelling of such relations will work in the tool, even though the generated code might be erroneous.</p> <p>Workaround: Do not use integrate/delay relations with arguments of libraries.</p>
SCT-646	<p>MATLAB® licencing dialog will block SCODE-CONGRA</p> <p>This is a similar issue to SCT-7400 (please see also this description).</p> <p>Solution: Please install a license of MATLAB first before using MuPAD as a solver.</p>
SCT-896	<p>Computation can be invalid</p> <p>It can happen that a generated computation is marked as faulty by the IDE, primarily, if it contains a library call. Then no further code that depends on the computation will be generated.</p> <p>Workaround: Open the computation in the editor. It is then indexed again, which leads to a correct file. Thus, the code generation process continues.</p>
SCT-920	<p>Corrupt SYQ file after adding flow or system</p> <p>If the textual model file (SYQ) ends with an end of line comment without a line break, then, when adding a flow or system via the context menu in the graphical editor, this might render a wrong SYQ file after saving. The declaration of the new flow (or system) might start in the comment line.</p> <p>Workaround: Manually insert a line break in the SYQ file, and always be sure to use a line break after the last comment line.</p>

Issue Reference	Description
SCT-922	<p>Unchanged diagrams are marked as dirty</p> <p>In case several graphical diagrams are opened, and only one of them is changed, all diagrams are marked with the "*" for being "dirty".</p> <p>Workaround: None.</p>
SCT-923	<p>Renaming a flow is not reflected in the execution environment</p> <p>The execution environment and the specification of objects are closely synchronized, and the execution environment gets updated with changes in the specification. This however does not hold true if you rename a flow or computation.</p> <p>Workaround: Reopen the renamed computation/flow in the execution environment. You will have to redo any changes you did (in particular values).</p>
SCT-924	<p>Clean Project operation not complete</p> <p>In some cases, artefacts generated by the generator are not completely removed by the "Clean Project" operation</p> <p>Workaround: manually delete the "src-gen" folder</p>
SCT-925	<p>Removing packages from a project has side effects on other packages</p> <p>The deletion of a package in a project that does contain multiple packages does also temporarily remove the model and flow nodes in the project explorer under other SYQ files in other packages.</p> <p>Workaround: Close the project and open it again.</p>
SCT-927	<p>MuPAD® returns negative result when reaching maximum depth for nested procedure calls</p> <p>When passing complex solving commands to MuPAD that involve algebraic loops with a large number of relations, variables and assumptions MuPAD might return a negative result.</p> <p>Workaround: Try to reduce the number of assumptions, to manually break/tear the affected algebraic loops or to use a different solver.</p>
SCT-933	<p>Generated resources cannot be referenced, especially subcomputations in case of subsystem libraries</p> <p>The generated resources (e.g. *.syq files containing computations) are not put into the index at point of generation. Consequence is that the importing of objects in these generated files (especially: computations of generated "library" systems) does not work ad hoc.</p> <p>Workaround: Close and reopen the project containing the generated files (i.e. the library) after building and generation.</p>

Issue Reference	Description
SCT-935	<p>TX_GraphicalEdit: The positions are translated when removing or adding an element</p> <p>When a node gets added in the graphical editor and a different node gets deleted in the textual editor without saving in between, the nodes might change their index and they get re-ordered after saving.</p> <p>Workaround: Always save after a single change. Or Save before switching the editor type.</p>
SCT-936	<p>Bug at Manual Workflow and Libraries</p> <p>The current implementation (as also cannot be changed on short notice) is that the build is based on files. Using the "manual workflow" just builds one item (a file can contain several items (flows, multiple systems)). As the build is started on a file, the first step in the build (as can currently not be influenced by us, at least not on short notice), is that all generated artefacts for that file are deleted. This leads to the behavior, that only one flow gets computed and stored in the "src-gen" folder, if there are multiple flows contained in one source SYQ file.</p> <p>Workaround: If several items are in one file, split the file into several files, such that each file only contains one item (flow, system).</p>
SCT-937	<p>Sensitivity on lines into subsystem not shown correctly</p> <p>When using sensitivity analysis, the lines to and from subsystems may not be displayed correctly in terms of thickness.</p> <p>Workaround: none</p>
SCT-944	<p>Outline view does not update the arrow style</p> <p>If during a sensitivity analyses or simulation values are changing in the execution environment then the outline view gets not updated.</p> <p>Workaround: None</p>
SCT-946	<p>Border node arrangement might not work as expected if there are many border nodes</p> <p>If there are many border nodes, the "arrange" border nodes algorithm might produce wrong result in case the calculated positions of the border nodes intersect with each other.</p> <p>Workaround: Repeat the arrange command, and/or use manual rearrangement of the border nodes or use the "Layout->Linked Border Nodes" context menu.</p>
SCT-949	<p>Comments in the SYQ file might get deleted</p> <p>When doing modifications to the model in the graphical editor, esp. when deleting elements, also comments in the SYQ file might get deleted (Root cause is that these comments are associated with the deleted elements)</p> <p>Workaround: you might want to use the "local history" to check on inadvertent deletion of comments.</p>

Issue Reference	Description
SCT-958	<p data-bbox="628 293 1353 353">Inconsistent graph for sub-system and sub-system "pre-view"</p> <p data-bbox="628 365 1353 427">The graphical preview (mouse-over over a subsystem) does not update in case of adding a tearing edge in a subsystem.</p>
SCT-987	<p data-bbox="628 450 1353 510">Equation solving operation in external CAS tools cannot be cancelled</p> <p data-bbox="628 521 1353 674">The solving of equations in an external CAS tool cannot be cancelled. Depending on the nature of the system/flow to be solved, the time taken by the CAS Tool (e.g. MuPAD) can be several minutes (esp. in case of algebraic loops). This process cannot be interrupted.</p> <p data-bbox="628 685 1353 745">Workaround: Wait until the CAS tool is finished, or manually terminate SCODE-CONGRA in the Task Manager.</p>
SCT-993	<p data-bbox="628 768 1353 828">Simulink integration may not work if different versions of ETAS tool is installed</p> <p data-bbox="628 840 1353 936">When there are several tools of SCODE-CONGRA in an older version installed, it might happen, that from Simulink opening the latest SCODE-CONGRA tool does not work.</p> <p data-bbox="628 947 1353 1003">Workaround: Launch SCODE-CONGRA first and then perform a double click on the Simulink block</p>
SCT-1002	<p data-bbox="628 1025 1353 1055">Maxima solver is using a lot of resources</p> <p data-bbox="628 1066 1353 1218">In cases of complex equations, the maxima solver uses a lot of resources and may not terminate in time. In some cases, SCODE-CONGRA tries to reinvoke Maxima, which leads to concurrently running maxima instances consuming even more memory.</p> <p data-bbox="628 1229 1353 1294">Workaround: Use an alternative solver (e.g. MuPAD or cache solvers).</p>
SCT-1467	<p data-bbox="628 1317 1353 1346">TX_Solver: Endless loop when solving with MuPAD</p> <p data-bbox="628 1357 1353 1386">For some projects solving ends up in an endless loop.</p> <p data-bbox="628 1397 1353 1485">Workaround: The user has also the computation (means the generated SYQ file) open in the editor. This file needs to be closed then the loop will terminate.</p>
SCT-13064	<p data-bbox="628 1507 1353 1536">Char table parameters are only created once</p> <p data-bbox="628 1547 1353 1644">If char tables are defined in subsystems their parameters are only created once even though the subsystem is used several times.</p> <p data-bbox="628 1655 1353 1675">Workaround: Use instance-specific char tables.</p>
SCT-13163	<p data-bbox="628 1697 1353 1727">Renaming allows invalid folder names</p> <p data-bbox="628 1738 1353 1834">When renaming a folder (which represents a package in the CONGRA model) it is possible to use names that are invalid in the CONGRA grammar.</p> <p data-bbox="628 1845 1353 1899">Workaround: Don't use special characters and reserved keywords in folder names and check for error markers.</p>

Issue Reference	Description
SCT-14243	<p>Gradient for conditional equations within not symbolically solvable algebraic loops is computed as zero</p> <p>In this case the gradient should not be calculated at all, but the resulting computation shows zero as result.</p> <p>Workaround: None.</p>
SCT-14284	<p>Constraints on result variables are not passed to the MuPAD solver</p> <p>This might lead to more and more complex solutions return by the MuPAD solver.</p> <p>Workaround: Manually select the adequate solution or use the solution cache to use simplified solutions.</p>
SCT-14669	<p>Border node position are lost when subsystem is exchanged</p> <p>If a subsystem used within a relation is exchanged for another one, the border nodes are relocated.</p> <p>Workaround: Rearrange the border nodes manually or using the context menu.</p>
SCT-14875	<p>For small values "fsolve" does not deliver the correct result.</p> <p>The default tolerance used by the MATLAB routine is too large.</p> <p>Workaround: Manually edit the generated MATLAB code to make the solver deliver correct result by decreasing the tolerance.</p>
SCT-14918	<p>Multiple SCODE Workbench installations share the same update site location</p> <p>This leads to the unexpected result that when (un-)installing plugins for one installation this might affect other installations. This only applies to concurrent installations of the same SCODE version.</p> <p>Workaround: Do not have several installations of the same SCODE Workbench version.</p>
SCT-15178	<p>Error case within a relation tearing cannot be handled by using a replacement value</p> <p>If the computation of the slack variable within a relation tearing happens to include a potential error case (such as division by zero), no replacement value can be given.</p> <p>Workaround: Use "Report error/abort execution" as error case handling, or adjust the model to eliminate the error case.</p>

Issue Reference	Description
SCT-15576	<p data-bbox="624 286 1367 353">Naming conflict between variables and functions for ESDL code generation (case sensitivity)</p> <p data-bbox="624 360 1367 461">In CONGRA you can use the same name for a variable and a function, given that there is a difference in cases. The ESDL generation will not work though for such models.</p> <p data-bbox="624 468 1367 533">Workaround: Be careful to not reuse names, even with different cases.</p>
SCT-17133, SCT-19525	<p data-bbox="624 548 1367 616">Erroneous models might not be displayed using the correct colors and analysis results</p> <p data-bbox="624 622 1367 723">Graphs for models with an error might not be using the correct colors. In addition, parts that should be inactive can be displayed as active if errors exist in the model.</p> <p data-bbox="624 730 1367 795">Workaround: Fix the errors to get the correct colors and active parts of the graph.</p>
SCT-17447, SCT-19087	<p data-bbox="624 810 1367 848">Opening a graph in the editor will mark it as modified</p> <p data-bbox="624 855 874 878">Workaround: None</p>
SCT-18544	<p data-bbox="624 907 1367 945">Rewind button in execution environment has no effect</p> <p data-bbox="624 952 874 974">Workaround: None.</p>
SCT-18673	<p data-bbox="624 996 1367 1064">Pantelides algorithm does not work for states that are also outputs</p> <p data-bbox="624 1070 874 1097">Workaround: None.</p>
SCT-19122	<p data-bbox="624 1120 1367 1187">Analysis not updated when applying tearing during manual workflow</p> <p data-bbox="624 1193 1367 1261">When a tearing violates the intrinsic direction of at least one relation, errors will only be visible after saving the model.</p> <p data-bbox="624 1267 1367 1332">Workaround: Save and redirect after model changes during the manual workflow.</p>
SCT-19139	<p data-bbox="624 1344 1367 1411">Evaluation of tearing boundaries might lead to evaluation errors</p> <p data-bbox="624 1417 1367 1512">Solution: Users have to take care to use only expressions for the boundaries of the tearing interval that can be safely evaluated.</p>
SCT-19582	<p data-bbox="624 1523 1367 1561">No variable matching during subsequent discretizations</p> <p data-bbox="624 1568 1367 1668">While performing automatic discretizations of integrate edges that reference to the same variables, the variables might not be reused correctly, but duplicated.</p> <p data-bbox="624 1675 1367 1736">Workaround: Manually delete the duplicated variables and reconnect the edges.</p>
SCT-19691	<p data-bbox="624 1747 1367 1814">Block update during Simulink integration cannot be performed after model changes</p> <p data-bbox="624 1821 1367 1921">A block update only affects the main SCODE block. If names or types of parameters or states are changed, the update cannot be performed.</p> <p data-bbox="624 1928 1367 1991">Workaround: Recreate the Simulink block(s) using the create script.</p>

Issue Reference	Description
SCT-19701	Diagram layout from flow editor not properly persisted When laying out the flow diagram, the changes might sporadically not be persisted until the project is closed. Workaround: Use the system editor to initially layout the diagram.
SCT-19710, SCT-19739	Units are not correctly resolved during analysis and code generation In some cases the units are not correctly scaled in the generated code. Workaround: Do not use code generated from model with units for production
SCT-19730	Some solutions returned by a CAS are not recognized Involved solutions returned from a CAS might not be used by SCODE-CONGRA, leading to unsolved computation steps. Workaround: Use a different solver, reformulate the equation, solve numerically or use the solution cache.

4 Notes

4.1 General

- After importing projects that were created with earlier versions it may be necessary to configure them to the correct nature using **Configure > Convert to SCODE-ANALYZER project** or **Configure > Convert to SCODE-CONGRA project**, respectively.
- Importing existing SCODE-ANALYZER models using the mint format and existing SCODE-CONGRA models using the mxe format is not supported for SCODE Workbench. Please use SCODE Workbench 2.6.3 or earlier versions to import such models.

4.2 SCODE-ANALYZER

- 'Edit' functionality in mode transition graph is not supported.
- 'Undo' on decision tree is not supported for all operations.
- Large models may lead to performance issues while calculating analysis results and fingerprints.
- Non-system modes are not excluded while calculating the decision tree for outgoing transitions of a mode.
- The mode transition graph is not updated dynamically. The graph must be exported after mode transitions are changed.
- The *.sdgm and *.scode files are tightly coupled by the name. Renaming of the *.scode file will make the nodes and edges in the *.sdgm diagram file appear with (X).
- C Code is compiled and verified on MinGW
- Update scripts for Simulink integration do not update the delay element for generation source 'Mode Transition Matrix'. The delay element is only created by running the create script in MATLAB.

4.3 SCODE-CONGRA

- FMU generation does not support 32-bit floating-point numbers
- Units are an incubation feature and might not be handled correctly. Do not use models with units to generate production code!
- Update scripts for Simulink integration do not update the additional blocks or workspace parameters. Use the create script to generate all related objects.

4.4 MATLAB® Connection

- For round trip engineering between SCODE Workbench and MATLAB, connection between the tools should be established.
- If an exception occurs or the connection between MATLAB and SCODE Workbench is lost, both tools need to be restarted.

- A connected Simulink block is updated upon 'Save' operation on *.scode and *.syq files.
- Before uninstalling SCODE Workbench all connections to MATLAB versions must be removed in the MATLAB/Simulink section of **Windows > Preferences** in order to be able to establish connections with a newer release of the product.

5 Contact Information

ETAS Headquarters

ETAS GmbH

Borsigstraße 24

70469 Stuttgart

Germany

Phone: +49 711 3423-0

Fax: +49 711 3423-2106

Internet: www.etas.com

ETAS Subsidiaries and Technical Support

For details of your local sales office as well as your local technical support team and product hotlines, take a look at the ETAS website:

ETAS subsidiaries Internet: www.etas.com/en/contact.php

ETAS technical support Internet: www.etas.com/en/hotlines.php

Index

3

3rd-party software9

C

C code generation
 SCODE-ANALYZER11
 SCODE-CONGRA13
C++ code generation
 SCODE-ANALYZER11
Code generation
 SCODE-ANALYZER10
 SCODE-CONGRA13
Compatibility15
Contact information27

D

Decision tree11
Discontinued features
 SCODE-ANALYZER12
 SCODE-CONGRA14

E

Eclipse platform10
EHANDBOOK report
 SCODE-ANALYZER11
 SCODE-CONGRA13
ESDL code generation
 SCODE-ANALYZER11
 SCODE-CONGRA13
Export options
 JSON export11
 SCODE-ANALYZER11

F

FMU generation
 SCODE-CONGRA13

G

Graphical editor
 SCODE-CONGRA13

I

itemis
 YAKINDU Traceability@..... 11, 14

K

KIR15
Known issues
 SCODE-ANALYZER15
 SCODE-CONGRA18
Known issues report15

L

Licensing9

M

MATLAB®
 MATLAB function12
 MEX function12
 S-function12
 Supported versions 11, 14
MATLAB® code generation
 SCODE-ANALYZER11
 SCODE-CONGRA14
MATLAB® connection25
Maxima7
MISRA C 201213
Mode transition matrix11

P

PikeTek®
 TPT export12
Preferences
 Import/export10
Prerequisites
 Software7
 System6
Product examples10

R

Release test
 3rd party products8
 Eclipse target platform7
 Host platform7
 Java platform7
Reports
 EHANDBOOK 11, 13
 SCODE-ANALYZER11
Requirements linking

SCODE-ANALYZER	11	Graphical editor.....	13
SCODE-CONGRA	14	Known issues.....	18
S		Notes	25
SCODE Workbench		Requirements linking.....	14
Compatibility to earlier releases.....	15	Units	14
Preferences import/export	10	What's new	13
What's new	10	Simulink integration	
SCODE-ANALYZER	4	SCODE-ANALYZER	12
C code generation.....	11	SCODE-CONGRA	14
C++ code generation.....	11	Simulink Library Data Dictionary.....	11
Code generation.....	10	Simulink®	
Decision tree.....	11	Supported versions	11, 14
Discontinued features.....	12	Software prerequisites	7
EHANDBOOK report.....	11	MATLAB®.....	7
ESDL code generation	11	System prerequisites.....	6
Export options.....	11	T	
Functions	6	TPT export.....	12
INFO dimensions	11	U	
JSON export	11	Units	
Known issues.....	15	SCODE-CONGRA	14
MATLAB® code generation.....	11	W	
Mode transition matrix	11	What's new	
Notes	25	SCODE-CONGRA	13
Reports	11	Y	
Requirements linking.....	11	YAKINDU Traceability®	11, 14
Simulink integration.....	12	Product examples	10
TPT export.....	12	SCODE-ANALYZER	12
What's new	10	SCODE-CONGRA	14
SCODE-CONGRA	4	Z	
C code generation.....	13	Zwicky box.....	6
Code generation.....	13		
Discontinued features.....	14		
EHANDBOOK report.....	13		
ESDL code generation	13		
Export graphical elements.....	13		
FMU generation	13		
Functions	6		