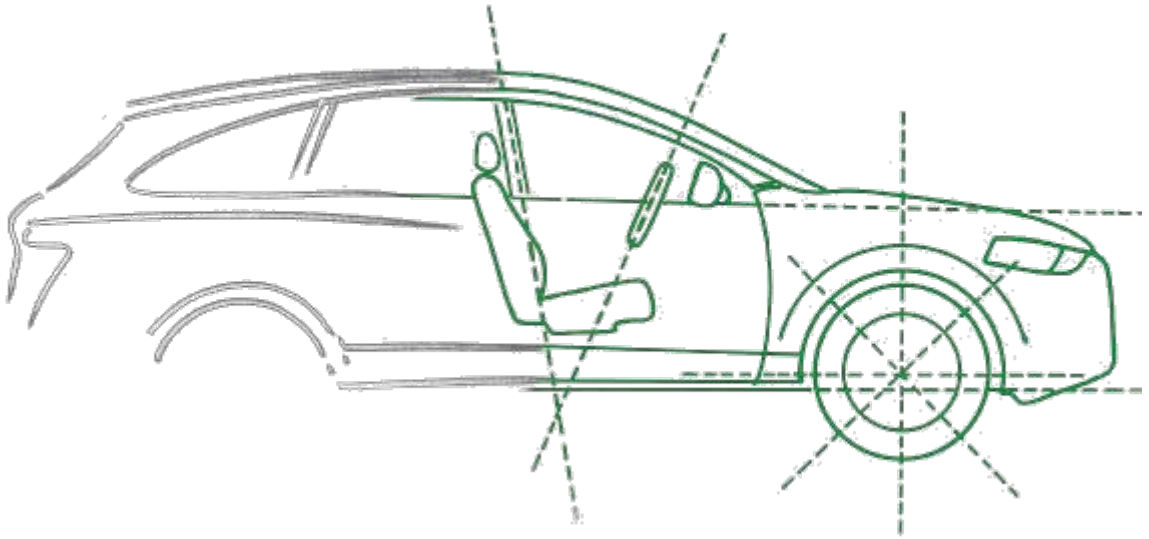
berner & mattner
optimizing your development

Modeling requirements to support testing of product-lines



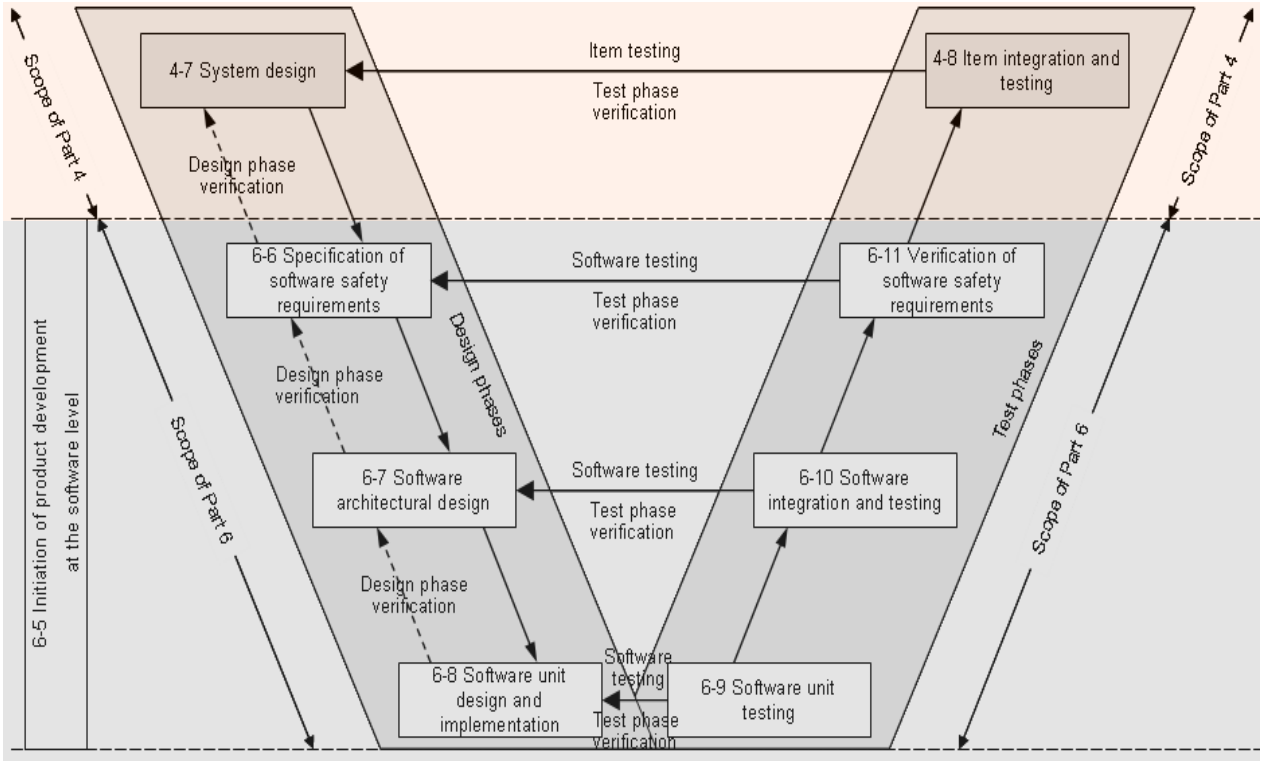
C. Robinson-Mallett, robinson-mallett@berner-mattner.com
Jens Köhnlein, jens.koehnlein@daimler.com
Matthias Grochtmann, matthias.grochtmann@berner-mattner.com
Joachim Wegener, joachim.wegener@berner-mattner.com
Steffen Kühn, steffen.kuehn@berner-mattner.com

Overview

- RE/RM and test-management in practice (with DOORS)
- Model-based specification and analysis
- Variant-management for specifications
- Berner & Mattner MERAN for IBM Rational DOORS
- Applications and experiences

Automotive Industry Context of MBA and MBT of Systems

- ISO 26262 will soon be obligatory
- ISO 26262 part 4 regulates system analysis and validation
- Requirements are input for Item integration and testing analysis
- MBT and MBA shall be applied during *Item Integration and Testing*

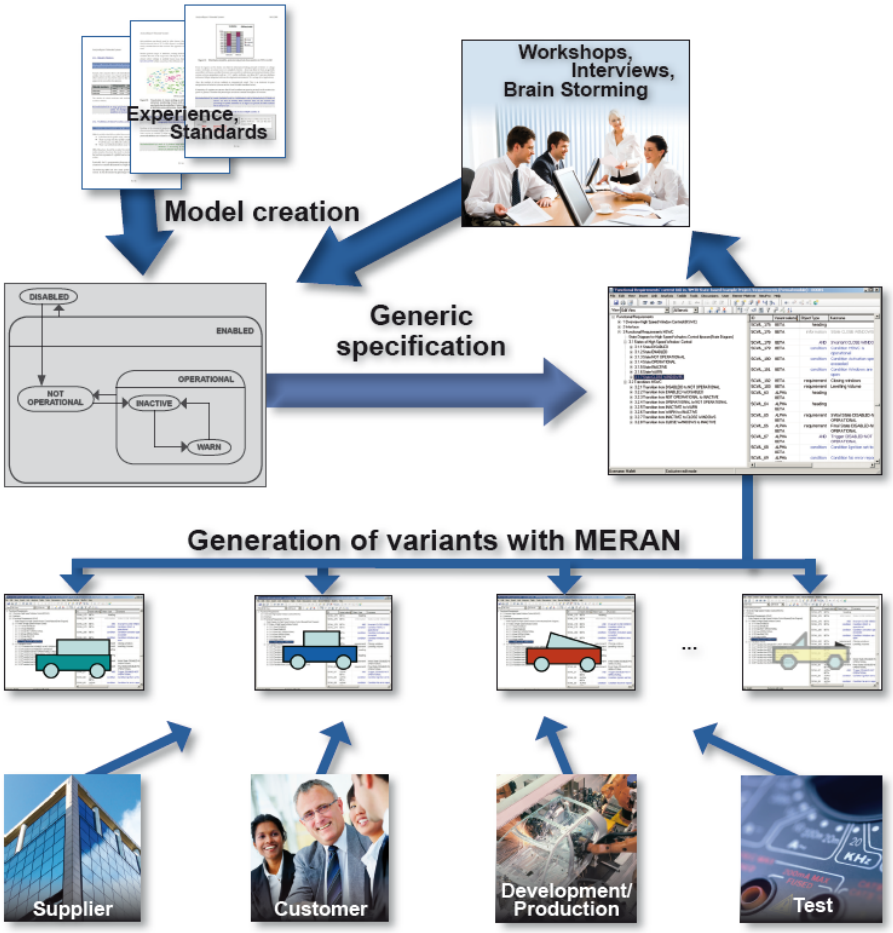


Model-based requirements writing process

- Interviews and Workshop
- Model definition (e.g. Statecharts)
- Specification structure follows model
- Elicitation and integration of requirements
- Iterative analysis, refinement and completion of requirements

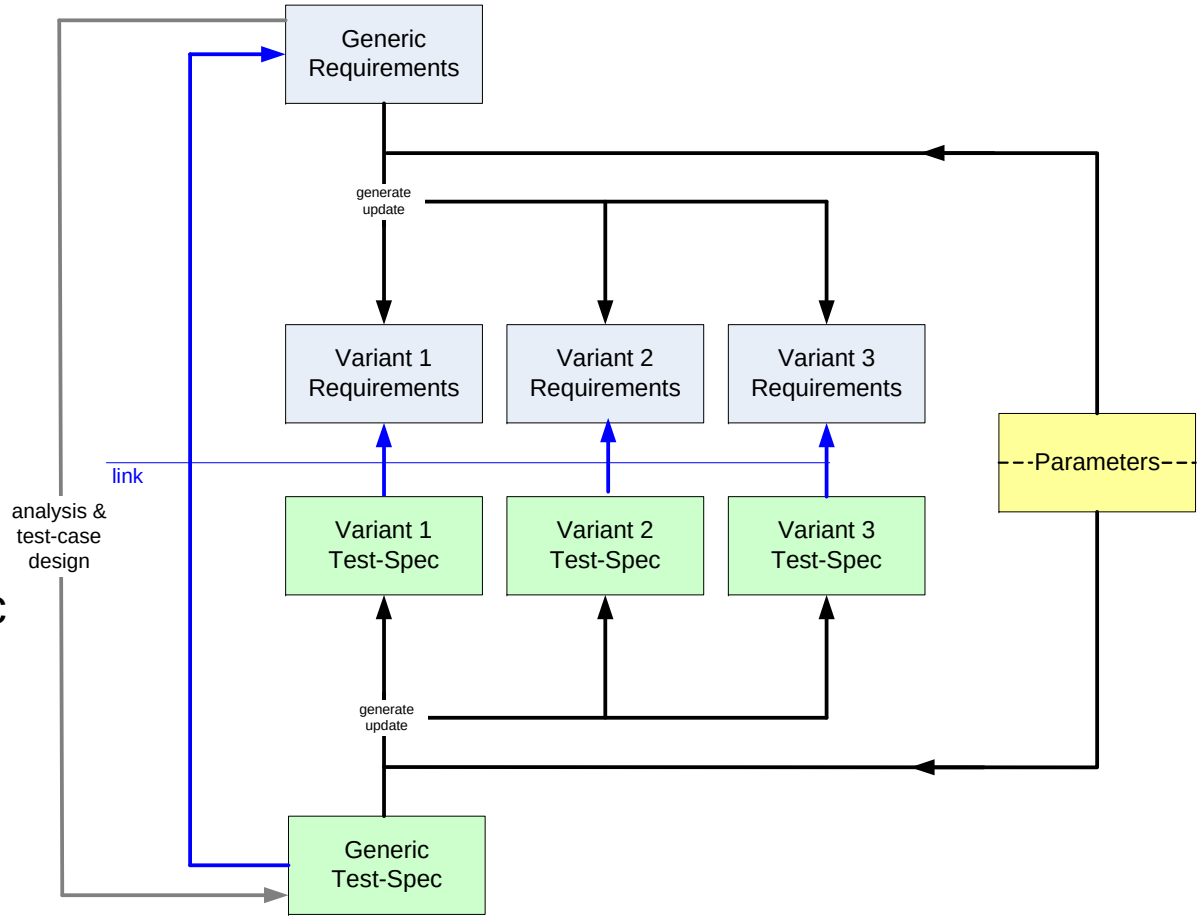
optional steps

- Specification of *complete* test-cases
- Implementation of executable test-cases
- Formal analyses and approvals



Specification and testing of product-lines

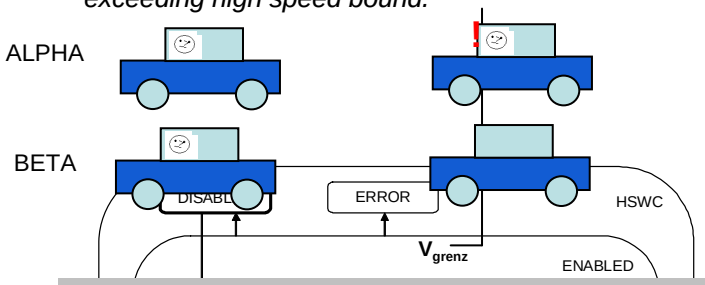
- generic requirements specification is created for product-line
- Options
  - create generic test-case specs and generate specific test-cases
  - create specific test-cases for specific requirements
- a mutual parameter base may reduce risk of inconsistencies between requirements and test-cases



Model-based Specification

- Structure of a specification follows a functional model
- Semi-formal specification of states and transitions
- Semi-formal specification of conditions

Example HSWC:
Function High Speed Window Control in variant ALPHA warn on opened window exceeding high speed bound.
Function HSWC in variant BETA, closes windows autonomously on exceeding high speed bound.



Chapter X Functional requirements HSWC
Chapter X.1 States
Chapter X.1.1 State HSWC
Chapter X.1.2 State DISABLED
Chapter X.1.3 State ENABLED
Chapter X.1.4 State NOT OPERATIONAL
Chapter X.1.5 State OPERATIONAL
Chapter X.1.6 State INACTIVE
Chapter X.1.7 State ACTIVE
Chapter X.1.8 State ERROR
Chapter X.2 Transitions
Chapter X.2.1 Transition DISABLED-NOT OPERATIONAL
Chapter X.2.2 Transition ENABLED-DISABLED
Chapter X.2.3 Transition NOT OPERATIONAL-INACTIVE
Chapter X.2.4 Transition OPERATIONAL-NOT OPERATIONAL
Chapter X.2.5 Transition INACTIVE-ACTIVE
Chapter X.2.6 Transition ACTIVE-INACTIVE
Chapter X.2.7 Transition ENABLED-ERROR

Advantages

- Intuitively comprehensible, functional structure
- Improved
  - changeability
  - analysability
  - testability
- accepted redundancies only
- great basis for following model-based development, testing and analyses

Typed Specifications

- Specifications contain
  - identifyable
  - atomic
  - typed objects
- Requirements, must be distinguishable from structure, explanations and comments
- Typing allows application of automated analysis and processing

basic specification types

Type	description
heading	structure elements
information	explanations or comments
requirement	separated required product property

conditional specification types

Type	description
XOR	exclusive disjunctive composition of conditions
OR	disjunctive composition of conditions
AND	conjunctive composition of conditions
condition	atomic conditional requirement

State Specification

- State specification template
  - explanatory information including short name
  - optional sub-states and initial state
  - optional state invariant
  - additional state-related requirements

Example HSWC State Operational

Type	ALPHA	BETA	Name	Text
heading	X	X		X.1.5 State OPERATIONAL
information	X	X	State OPERATIONAL	The state OPERATIONAL represents the operational function HSWC.
requirement	X	X	Sub-states OPERATIONAL	State OPERATIONAL includes State INACTIVE and State ACTIVE.
requirement	X	X	Initial State OPERATIONAL	The initial sub-state in State OPERATIONAL is State INACTIVE.
AND	X	X	Condition Operation HSWC	The function HSWC is operational, if every of the following conditions hold:
condition	X	X	Condition HSWC is enabled	•The function HSWC is enabled; in State ENABLED.
condition	X	X	Condition Operation Speed exceeded	•The vehicle speed is greater or equals $V_{Operation}$.
condition		X	Condition No rear-seat passengers	•On the rear-seats no passengers are detected.
requirement	X	X	Additional Requirement	HSWC shall ...

Transition Specification

- Transition Specification Template
  - explanatory information including short name
  - initial state
  - final state
  - trigger condition
  - transition-related requirements

Example HSWC Transition Inactive to Active

Type	ALPHA	BETA	Name	Text
heading	X	X		X.1.4 Transition from INACTIVE to ACTIVE
information	X	X	Transition INACTIVE-ACTIVE	The transition from state INACTIVE to state ACTIVE represents the activation of HSWC.
requirement	X	X	Initial State	The initial state is State INACTIVE.
requirement	X	X	Final State	The final state is State ACTIVE.
AND	X	X	Condition Activation HSWC	The function HSWC is activated if every of the following conditions hold:
condition	X	X	Condition HSWC is operational	•The function HSWC is operation; the function is in State OPERATIONAL.
condition	X	X	Condition Windows are open	•At least one window is open.
condition	X	X	Condition Vehicle exceeding v_{act}	•The vehicle speed is greater or equals $v_{Activation}$.
requirement		X	Command closing windows.	On activation of HSWC windows must be commanded to close.
requirement	X		Activation Message	On activation of HSWC a message must be displayed to the driver.
requirement	X	X	Activity Report	On activation of HSWC the activity bit has to be set.

Test-case specification

- Test-case specification template
  - Optional pre-conditions
  - Test-steps

    Attributes
    - Precondition
    - Action
    - Measurement
    - Postcondition
    - ...
  - Optional post-conditions
- Further Extensions
  - Code-Steps
  - Test-runs

type	description
test-case	containing pre-/post-conditions, and test-steps
pre-condition	initial system state
test-step	separated input and response pairs
post-condition	final system state

Generic Test-case Specification

- Test-case specification may reuse variant definitions
- Test-case specifications may extend variant definitions
- Test-case design
  - Functional Coverage
  - Combinatorial Testing
  - Sequence Enumeration
  - Search-based Testing

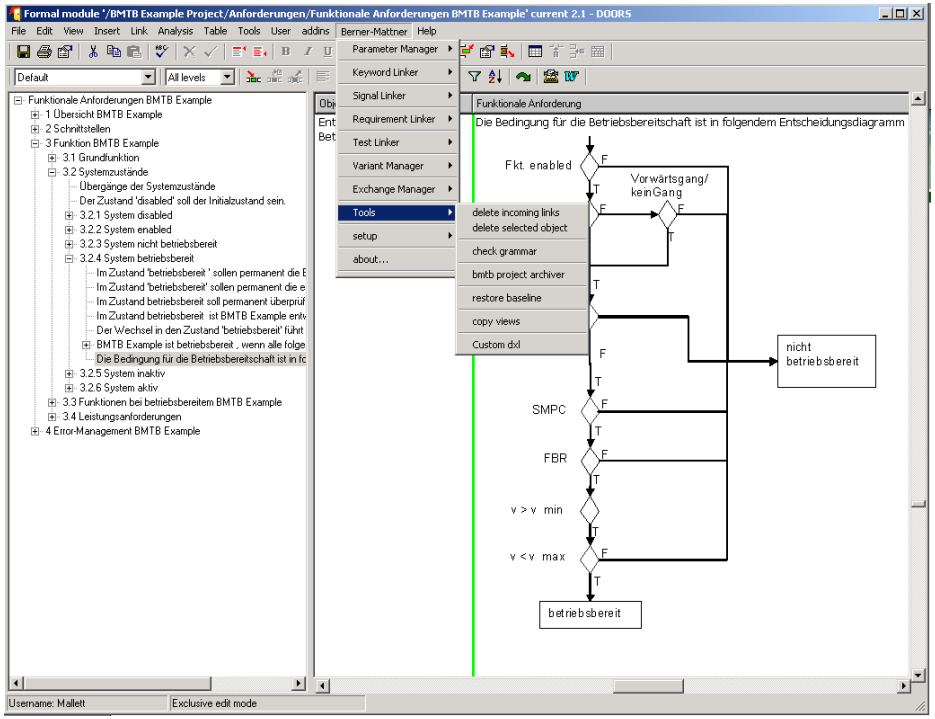
Example Test-case for HSWC Transition Inactive to Active

Type	Mark		Purpose		Input	Expected Response
	ALPHA	BETA	positive	negative		
test-case	**X**	**X**	**X**	**X**		
pre-condition	X	X	X	X	HSWC is in state INACTIVE	
pre-condition	X	X	X	X	HSWC is operational	
test-step	X	X	X		Open rear-windows.	Windows are opening.
test-step	X	X		X	Close any windows.	Windows are closed.
test-step	X	X	X		Accelerate car to a speed of $v_{activation}$	Car speed is equal or greater $v_{activation}$.
test-step	X	X		X	Accelerate car to a speed of ($v_{activation}$-1)	Car speed is less than $v_{activation}$.
post-condition		X	X	X		Signal 'Close Windows' emitted.
post-condition	X		X	X		Signal 'Close Windows' not emitted.
post-condition	X		X	X		Warning message is displayed.

MERAN for IBM Rational DOORS

Features

- Variant-Management
- Automatised management of keywords and signals lists
- Automatised indexing
- Automatised linking
- Test-management over a whole product-line
- Test-automation in DOORS
- Improved DOORS handling
- Automatic generation of statistics
- Extended data exchange, e.g. Excel, mind-maps, Visio
- Combinatorial testing tool
- Model-based analysis and testing tool



Experiences and Applications

Experiences

- theoretical expense reduction (n-1), where n is the number of variants, is barely producable in practive, but we
  - frequently reduce specification size by 30 % due to model-based approach
  - frequently reduce expenses by up to 50 % due to variant-management
- these figures are useful planning guidelines, based on experiences from more than 20 projects
- empirical validation has to be done

Applications

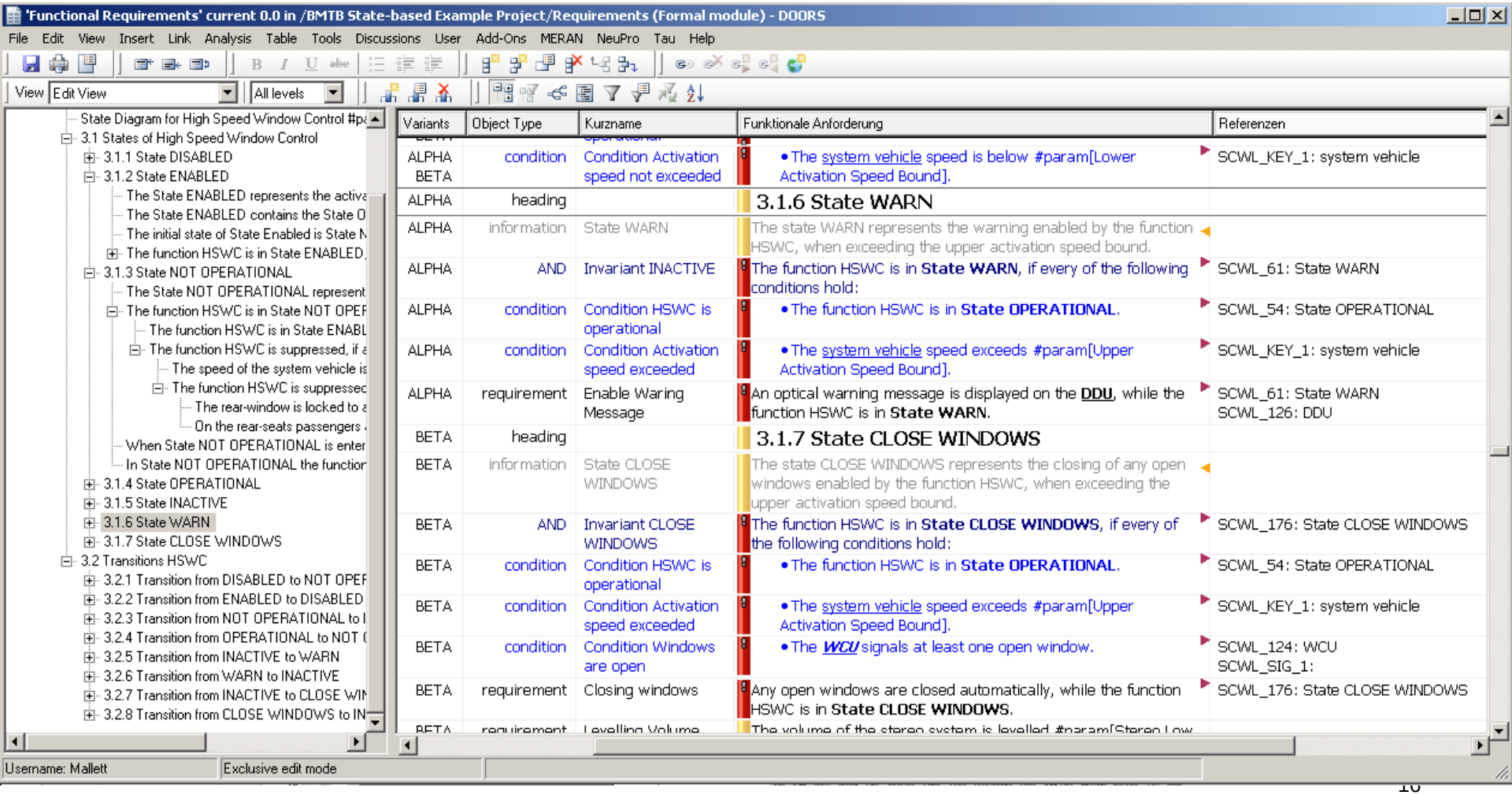
- Variant-Management of **Parking Systems** at **Daimler AG**
- Model-based specification and variant-management of several brake-assistance systems at **Daimler AG**
- Development and deployment of a car-platform- and company wide test-management system for the functional integration at **Volkswagen AG**
- Variant-management for the specification of a flight-system at **Airbus**

Summary

- Specifications often of low quality
- High expenses for adopting MBA/MBT
- Model-based approach may result in better requirements and enables efficient variant-management
- Effort savings in requirements and test specifications

End.

Appendix 1: Generic module HSWC

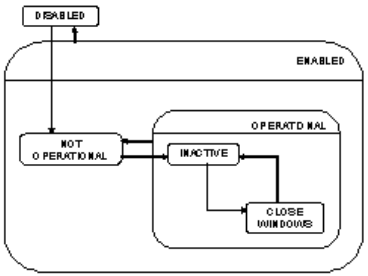
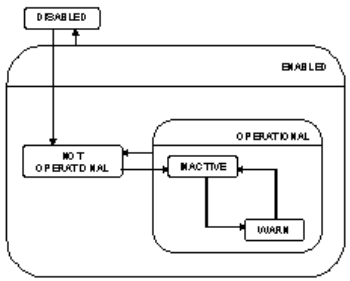


Appendix 2: Parameter module HSWC

'Parameter' current 0.0 in /BMTB State-based Example Project/Parameter (Formal module) - DOORS

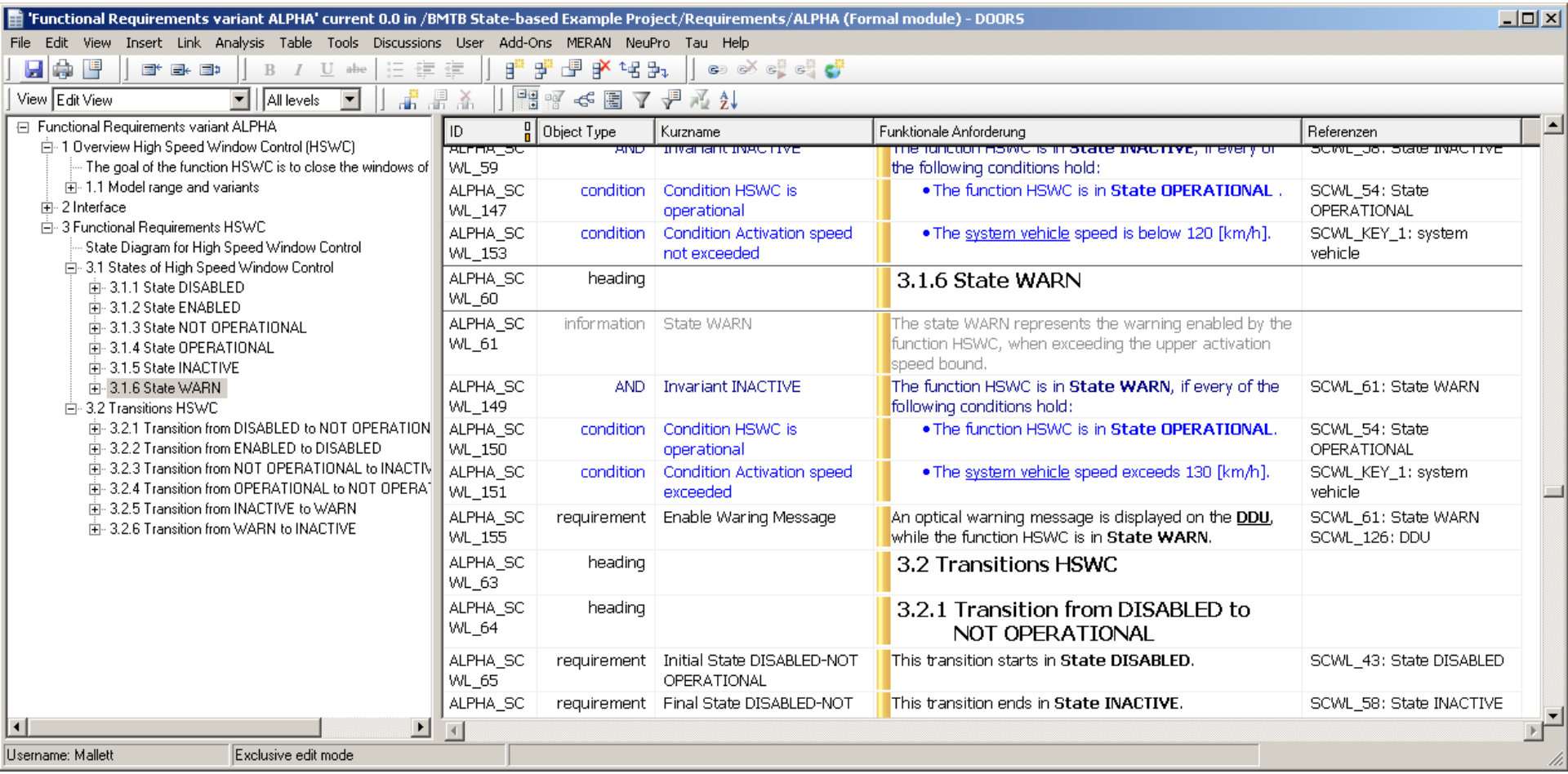
File Edit View Insert Link Analysis Table Tools Discussions User Add-Ons MERAN NeuPro Tau Help

View Default All levels

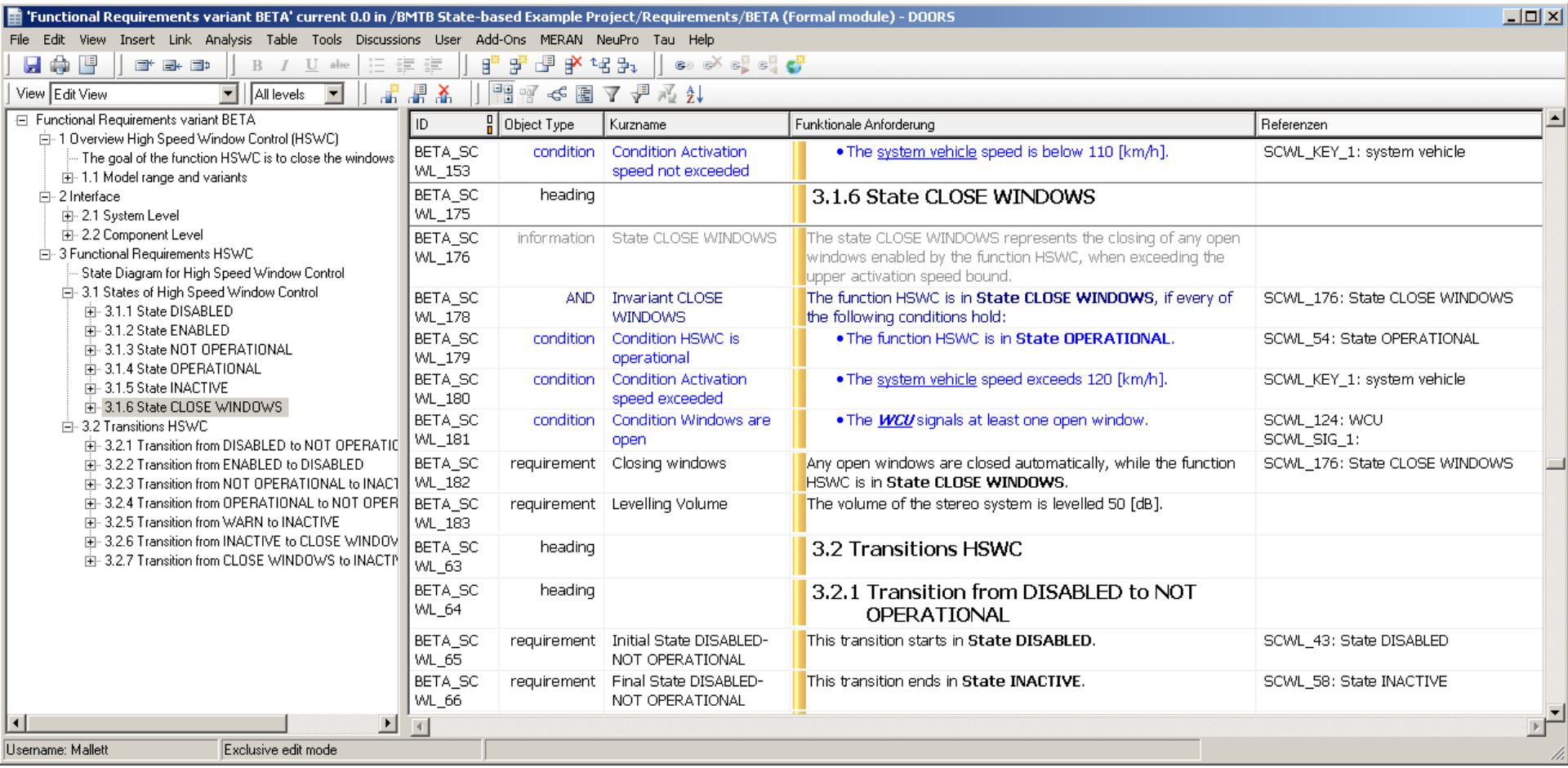
Function	Parameter Name	Default Value	BETA Value	ALPHA Value	Parameter Type	Parameter Description
HSWC parameters		-				
	State Diagram					
	Upper Operational Speed Bound	100		110	[km/h]	The speed limit which
	Lower Operational Speed Bound	90		100	[km/h]	The speed limit which
	Upper Activation Speed Bound	130	120		[km/h]	The volume to set aft of the rage from 0-10
	Lower Activation Speed Bound	120	110		[km/h]	The volume to set aft of the rage from 0-10
	Stereo Low Level	60	50		[dB]	

Username: Mallett | Exclusive edit mode

Appendix 3: Variant module ALPHA



Appendix 4: Variant module BETA



ID	Object Type	Kurzname	Funktionale Anforderung	Referenzen
BETA_SCWL_153	condition	Condition Activation speed not exceeded	• The system vehicle speed is below 110 [km/h].	SCWL_KEY_1: system vehicle
BETA_SCWL_175	heading		3.1.6 State CLOSE WINDOWS	
BETA_SCWL_176	information	State CLOSE WINDOWS	The state CLOSE WINDOWS represents the closing of any open windows enabled by the function HSWC, when exceeding the upper activation speed bound.	
BETA_SCWL_178	AND	Invariant CLOSE WINDOWS	The function HSWC is in **State CLOSE WINDOWS**, if every of the following conditions hold:	SCWL_176: State CLOSE WINDOWS
BETA_SCWL_179	condition	Condition HSWC is operational	• The function HSWC is in **State OPERATIONAL**.	SCWL_54: State OPERATIONAL
BETA_SCWL_180	condition	Condition Activation speed exceeded	• The system vehicle speed exceeds 120 [km/h].	SCWL_KEY_1: system vehicle
BETA_SCWL_181	condition	Condition Windows are open	• The ***WCU*** signals at least one open window.	SCWL_124: WCU SCWL_SIG_1:
BETA_SCWL_182	requirement	Closing windows	Any open windows are closed automatically, while the function HSWC is in **State CLOSE WINDOWS**.	SCWL_176: State CLOSE WINDOWS
BETA_SCWL_183	requirement	Levelling Volume	The volume of the stereo system is levelled 50 [dB].	
BETA_SCWL_63	heading		3.2 Transitions HSWC	
BETA_SCWL_64	heading		3.2.1 Transition from DISABLED to NOT OPERATIONAL	
BETA_SCWL_65	requirement	Initial State DISABLED-NOT OPERATIONAL	This transition starts in **State DISABLED**.	SCWL_43: State DISABLED
BETA_SCWL_66	requirement	Final State DISABLED-NOT OPERATIONAL	This transition ends in **State INACTIVE**.	SCWL_58: State INACTIVE