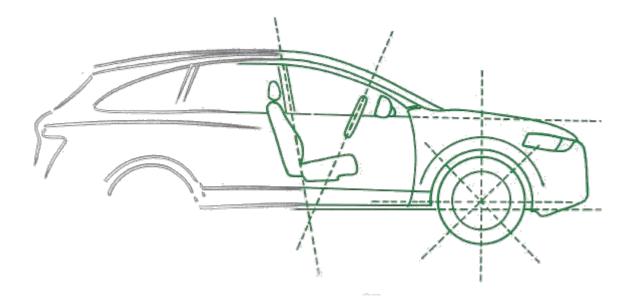


#### Modeling requirements to support testing of product-lines



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



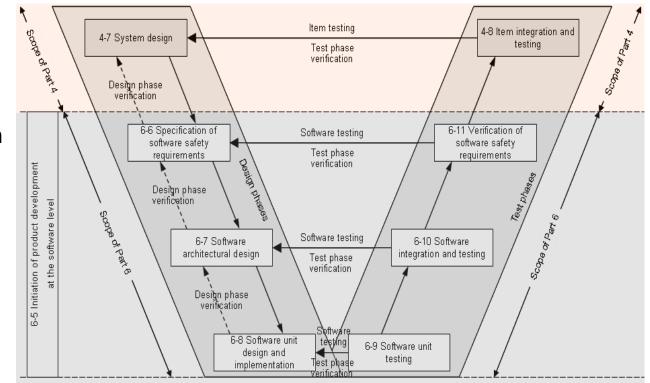
## Overview

- 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

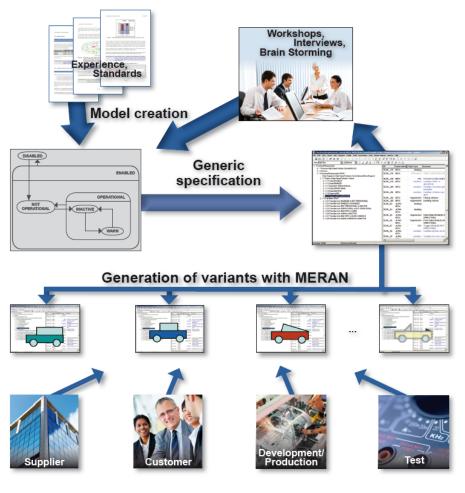
- ISO 26262 will soon be obligatory
- ISO 26262 part 4 regulates system analysis and validation
- Requirements are input for Item integration and testing analysis
- MBT and MBA shall be applied during Item Integration and Testing





#### 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



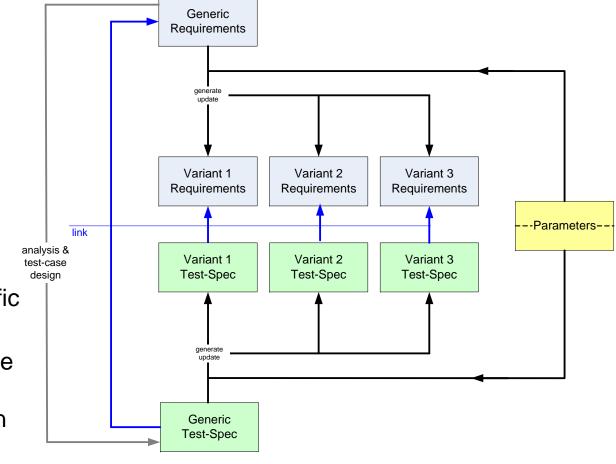
#### optional steps

- 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 testcases





### **Model-based Specification**

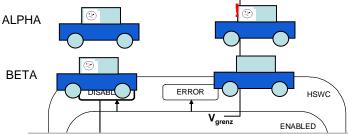
- Structure of a specification follows a functional model
- Semi-formal specification of states and transitions
- Semi-formal specification of conditions

#### **Advantages**

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

#### 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



## **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

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

#### conditional specification types

Туре	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

Туре	ALPHA	BETA	Name	Text
heading	Х	Х		X.1.5 State OPERATIONAL
information	Х	Х	State OPERATIONAL	The state OPERATIONAL represents the operational function HSWC.
requirement	Х	Х	Sub-states OPERATIONAL	State OPERATIONAL includes State INACTIVE and State ACTIVE.
requirement	Х	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	Х	Х	Condition HSWC is enabled	•The function HSWC is enabled; in State ENABLED.
condition	Х	Х	Condition Operation Speed exceeded	•The vehicle speed is greater or equals V <sub>Operation</sub> .
condition		Х	Condition No rear-seat passengers	•On the rear-seats no passengers are detected.
requirement	Х	Х	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

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

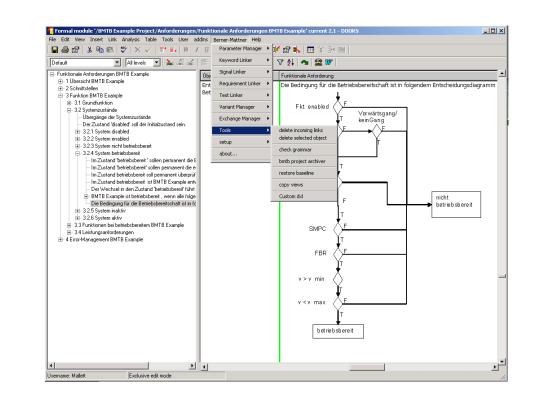
Туре	Mark		Purpose		Input	Expected Response	
Type	ALPHA	BETA	positive	negative	Input	Expected Response	
test-case	X	X	Х	Х			
pre-condition	Х	Х	Х	Х	HSWC is in state INACTIVE		
pre-condition	Х	Х	Х	Х	HSWC is operational		
test-step	Х	Х	Х		Open rear-windows.	Windows are opening.	
test-step	Х	Х		Х	Close any windows.	Windows are closed.	
test-step	Х	Х	Х		Accelerate car to a speed of <b>V</b> <sub>activation</sub>	Car speed is equal or greater <b>v</b> <sub>activation</sub> .	
test-step	Х	Х		Х	Accelerate car to a speed of (v <sub>activation</sub> -1)	Car speed is less than v <sub>activation</sub> .	
post-condition		Х	X	Х		Signal 'Close Windows' emitted.	
post-condition	Х		X	Х		Signal 'Close Windows' not emitted.	
post-condition	Х		Х	Х		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

unctional Requirements' current 0.0 in /BMTB State-t	oased Exar	mple Project/Red	quirements (Formal mo	dule) - DOORS	
Edit View Insert Link Analysis Table Tools Discuss	sions User	Add-Ons MERA			
🚔 💾 🗍 🗃 📾 🗊 🗍 B / U abe 🗄	建建一	] 📲 🚏 🖓 🔮	× +2 🎝 🗍 🖙 🚿	ej ej 😴	
v Edit View 🔽 📕 All levels 💌 🗍 📲		1 🔤 🖓 🗲	≣ 7 ₽ 🖉 🏚		
- State Diagram for High Speed Window Control #p	Variants	Object Type	Kurzname	Funktionale Anforderung	Referenzen
i⊟- 3.1 States of High Speed Window Control i⊞- 3.1.1 State DISABLED i⊟- 3.1.2 State ENABLED	ALPHA BETA	condition	Condition Activation speed not exceeded	The <u>system vehicle</u> speed is below #param[Lower     Activation Speed Bound].	SCWL_KEY_1: system vehicle
- The State ENABLED represents the active	ALPHA	heading		3.1.6 State WARN	
The State ENABLED contains the State 0     The initial state of State Enabled is State N     The function HSWC is in State ENABLED	ALPHA	information	State WARN	The state WARN represents the warning enabled by the function . HSWC, when exceeding the upper activation speed bound.	•
- 3.1.3 State NOT OPERATIONAL - The State NOT OPERATIONAL represent - The State NOT OPERATIONAL represent - The State NOT OPERATIONAL represent	ALPHA	AND	Invariant INACTIVE	The function HSWC is in <b>State WARN</b> , if every of the following conditions hold:	_
The function HSWC is in State NOT OPEF     The function HSWC is in State ENABL	ALPHA	condition	Condition HSWC is operational		SCWL_54: State OPERATIONAL
<ul> <li>The function HSWC is suppressed, if a</li> <li>The speed of the system vehicle is</li> </ul>	ALPHA	condition	Condition Activation speed exceeded	Activation Speed Bound].	SCWL_KEY_1: system vehicle
i⊟- The function HSWC is suppressed The rear-window is locked to a Ωn the rear-state suppressed	ALPHA	requirement	Enable Waring Message	An optical warning message is displayed on the <b>DDU</b> , while the function HSWC is in <b>State WARN</b> .	SCWL_61: State WARN SCWL_126: DDU
On the rear-seats passengers . 	BETA	heading		3.1.7 State CLOSE WINDOWS	
In State NOT OPERATIONAL the function  - 3.1.4 State OPERATIONAL  - 3.1.5 State INACTIVE	BETA	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.	•
3.1.6 State WARN	BETA	AND	Invariant CLOSE WINDOWS	The function HSWC is in State CLOSE WINDOWS, if every of the following conditions hold:	SCWL_176: State CLOSE WINDOWS
- 3.2 Transitions HSWC  - 3.2.1 Transition from DISABLED to NOT OPEF - 3.2.1 Transition from DISABLED to NOT OPEF	BETA	condition	Condition HSWC is operational		SCWL_54: State OPERATIONAL
⊕- 3.2.2 Transition from ENABLED to DISABLED     ⊕- 3.2.3 Transition from NOT OPERATIONAL to I     ⊕- 3.2.4 Transition from OPERATIONAL to NOT (	BETA	condition	Condition Activation speed exceeded	Activation Speed Bound].	SCWL_KEY_1: system vehicle
	BETA	condition	Condition Windows are open	• The <u>WCU</u> signals at least one open window.	SCWL_124: WCU SCWL_SIG_1:
	BETA	requirement	Closing windows	HSWC is in State CLOSE WINDOWS.	SCWL_176: State CLOSE WINDOWS
	RETA	requirement	Lovellina Volume	The volume of the storen evotem is levelled #norom[Storen Low	
e: Mallett Exclusive edit mode					
		J			



#### Appendix 2: Parameter module HSWC

	0.0 in /BMTB State-based Example					
	Link Analysis Table Tools Discussi					
🛃 🏟 💾   🖃	■• ■•   B I <u>U</u> =be   Ξ	建建 📗	" 먉 년 왕 백 동   @ ※ @ @ @			
View Default	💌 🛛 All levels 💌 🗍 🔐		12 🕫 ←⊂ 🖪 🔻 🖓 🖓 🖓			
Function	Parameter Name	Default Value	BETA Value	ALPHA Value	Parameter Type	Parameter Description 🖻
HSWC parameters		-				
	State Diagram		DEABLED EMABLED OPERATO NAL OPERATO NAL OPERATO NAL OUDSE WINDOWS	D RABLED		
				110	[km/h]	The speed limit which
	Lower Operational Speed Bound			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					
o somano. maliott	Jenolasino Cali mode					



### Appendix 3: Variant module ALPHA

🛃 🦣 💾 📗 🗃 📾 🗊 📗 B I 🛄 abe 🗄 🚎	🛊 🛛 📲	말 🕘 序 🕫 3	þ. 🛛 co oð cij cij 😴 🚽		
View Edit View 🔽 📕 All levels 🔽 🗍 📲 📲	1 🚠 🗍 📑	🕂 🗲 🗄 🛛	√ <sup>2</sup> n 2 2↓		
Functional Requirements variant ALPHA     in 1 Overview High Speed Window Control (HSWC)		Object Type	Kurzname	Funktionale Anforderung	Referenzen
- The goal of the function HSWC is to close the windows of	WL_59	0.0	Invarianci i Accitive	the following conditions hold:	30002_30. 30808 MACHIVE
<ul> <li></li></ul>	ALPHA_SC WL_147	condition	Condition HSWC is operational	• The function HSWC is in State OPERATIONAL .	SCWL_54: State OPERATIONAL
<ul> <li>3 Functional Requirements HSWC</li> <li>State Diagram for High Speed Window Control</li> <li>31 Output of Uter Speed Window Control</li> </ul>	ALPHA_SC WL_153	condition	Condition Activation speed not exceeded	• The system vehicle speed is below 120 [km/h].	SCWL_KEY_1: system vehicle
i⊟- 3.1 States of High Speed Window Control i⊞- 3.1.1 State DISABLED i⊞- 3.1.2 State ENABLED	ALPHA_SC WL_60	heading		3.1.6 State WARN	
⊕- 3.1.2 State ENABLED ⊕- 3.1.3 State NOT OPERATIONAL ⊕- 3.1.4 State OPERATIONAL ⊕- 3.1.5 State INACTIVE	ALPHA_SC WL_61	information	State WARN	The state WARN represents the warning enabled by the function HSWC, when exceeding the upper activation speed bound.	
⊡- 3.1.6 State WARN     ⊡- 3.2 Transitions HSWC	ALPHA_SC WL_149	AND	Invariant INACTIVE	The function HSWC is in <b>State WARN</b> , if every of the following conditions hold:	SCWL_61: State WARN
⊕- 3.2.1 Transition from DISABLED to NOT OPERATION     ⊕- 3.2.2 Transition from ENABLED to DISABLED	ALPHA_SC WL_150	condition	Condition HSWC is operational	• The function HSWC is in State OPERATIONAL.	SCWL_54: State OPERATIONAL
⊕- 3.2.3 Transition from NOT OPERATIONAL to INACTI\ ⊕- 3.2.4 Transition from OPERATIONAL to NOT NOT OPERATIONAL TO NOT OPERATIONAL TO NOT OPERATIONAL TO NOT OPE	ALPHA_SC WL_151	condition	Condition Activation speed exceeded	• The system vehicle speed exceeds 130 [km/h].	SCWL_KEY_1: system vehicle
⊕- 3.2.5 Transition from INACTIVE to WARN ⊕- 3.2.6 Transition from WARN to INACTIVE	ALPHA_SC WL_155	requirement	Enable Waring Message	An optical warning message is displayed on the <b>DDU</b> , while the function HSWC is in <b>State WARN</b> .	SCWL_61: State WARN SCWL_126: DDU
	ALPHA_SC WL_63	heading		3.2 Transitions HSWC	
	ALPHA_SC WL_64	heading		3.2.1 Transition from DISABLED to NOT OPERATIONAL	
	ALPHA_SC WL_65	requirement	Initial State DISABLED-NOT OPERATIONAL	This transition starts in State DISABLED.	SCWL_43: State DISABLED
	ALPHA_SC	requirement	Final State DISABLED-NOT	This transition ends in State INACTIVE.	SCWL_58: State INACTIVE



### Appendix 4: Variant module BETA

'Functional Requirements variant BETA' current 0.0 in /B				Formal module) - DOORS	_	
ile Edit View Insert Link Analysis Table Tools Discussio				-8		
	■ <u> </u> = <u> </u> = " <b>₽ .X</b>    [		┱⋣ਸ਼ਗ਼	ji in the second se		
View Edit View 🗾 All levels 🗾 🗍 👬 🖃 Functional Requirements variant BETA					[- ·	
Functional Requirements variant BETA     TA     Toverview High Speed Window Control (HSWC)	<u> </u>	Object Type	Kurzname	Funktionale Anforderung	Referenzen	
<ul> <li>I Overview High Speed Window Control (HSWC)</li> <li>The goal of the function HSWC is to close the windows</li> <li>1.1 Model range and variants</li> </ul>	BETA_SC WL_153	condition	Condition Activation speed not exceeded	<ul> <li>The system vehicle speed is below 110 [km/h].</li> </ul>	SCWL_KEY_1: system vehicle	
⊟-2 Interface ⊕-2.1 System Level	BETA_SC WL_175	heading		3.1.6 State CLOSE WINDOWS		
<ul> <li>              £.2.2 Component Level      </li> <li>             5 Functional Requirements HSWC         </li> <li>             State Diagram for High Speed Window Control         </li> </ul>	BETA_SC WL_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.		
∴ 3.1 States of High Speed Window Control     ∴ 3.1.1 State DISABLED	BETA_SC WL_178	AND	Invariant CLOSE WINDOWS	The function HSWC is in <b>State CLOSE WINDOWS</b> , if every of the following conditions hold:	SCWL_176: State CLOSE WINDOWS	
⊕ 3.1.2 State ENABLED     ⊕ 3.1.3 State NOT OPERATIONAL     ⊕ 3.1.4 State OPERATIONAL	BETA_SC WL_179	condition	Condition HSWC is operational	• The function HSWC is in State OPERATIONAL.	SCWL_54: State OPERATIONAL	
	BETA_SC WL_180	condition	Condition Activation speed exceeded	• The system vehicle speed exceeds 120 [km/h].	SCWL_KEY_1: system vehicle	
3.2 Transitions HSWC      3.2.1 Transition from DISABLED to NOT OPERATIC	BETA_SC WL_181	condition	Condition Windows are open	• The WCV signals at least one open window.	SCWL_124: WCU SCWL_SIG_1:	
	BETA_SC WL_182	requirement	Closing windows	Any open windows are closed automatically, while the function HSWC is in <b>State CLOSE WINDOWS</b> .	SCWL_176: State CLOSE WINDOWS	
	BETA_SC WL_183	requirement	Levelling Volume	The volume of the stereo system is levelled 50 [dB].		
	BETA_SC WL_63	heading		3.2 Transitions HSWC		
	BETA_SC WL_64	heading		3.2.1 Transition from DISABLED to NOT OPERATIONAL		
	BETA_SC WL_65	requirement	Initial State DISABLED- NOT OPERATIONAL	This transition starts in State DISABLED.	SCWL_43: State DISABLED	
	BETA_SC WL_66	requirement	Final State DISABLED- NOT OPERATIONAL	This transition ends in State INACTIVE.	SCWL_58: State INACTIVE	
Þ	•					
rname: Mallett Exclusive edit mode						