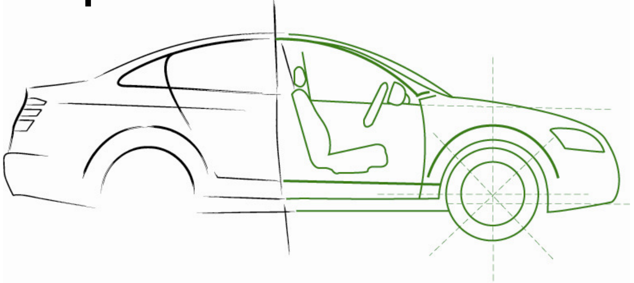
ReVVerT 2011

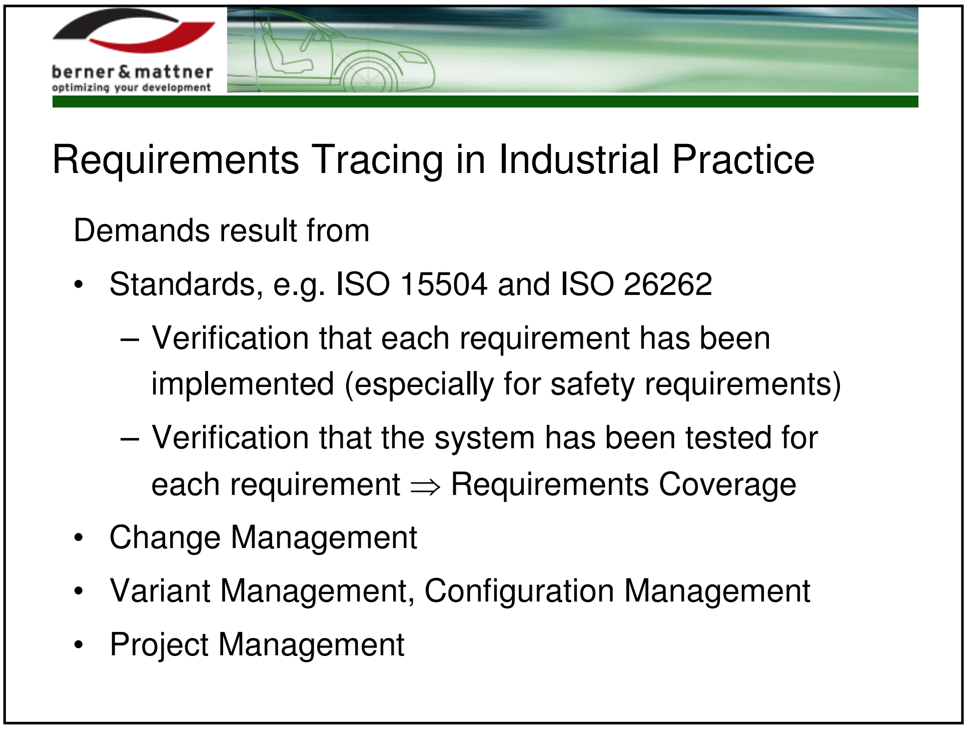
Tracing of Requirements and Test Cases



Dr. Joachim Wegener, Nico Beierle, Peter Kruse, Dr. Robinson-Mallett
Berner & Mattner Systemtechnik GmbH
joachim.wegener@berner-mattner.com

Agenda

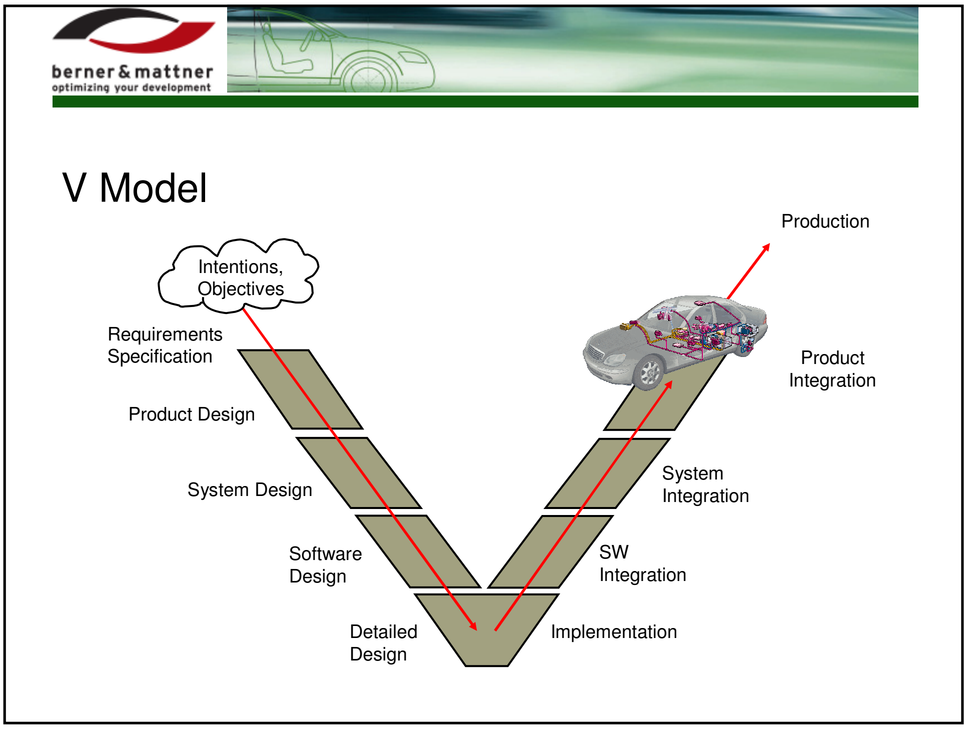
- Introduction
- Demands for Requirements Tracing in Industrial Practice
- Classification-Tree Method (CTM)
- Integration of Requirements Management and CTM
  - Linking Requirements and Test Cases
  - Visualization of Requirement Changes
- Summary and Future Work



Requirements Tracing in Industrial Practice

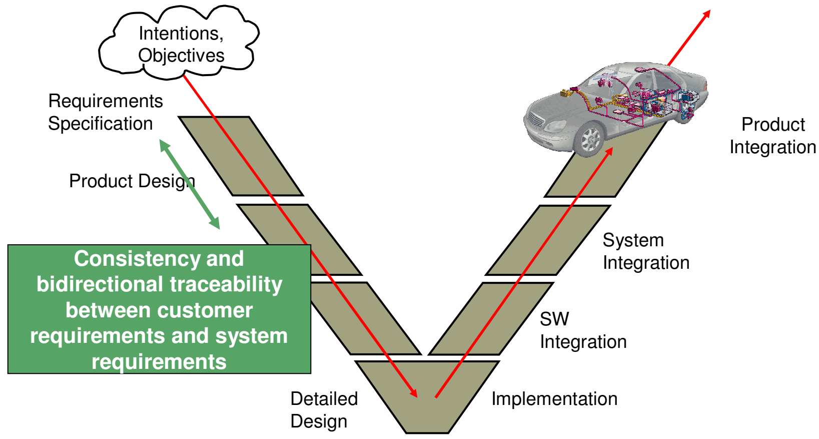
Demands result from

- Standards, e.g. ISO 15504 and ISO 26262
  - Verification that each requirement has been implemented (especially for safety requirements)
  - Verification that the system has been tested for each requirement ⇒ Requirements Coverage
- Change Management
- Variant Management, Configuration Management
- Project Management



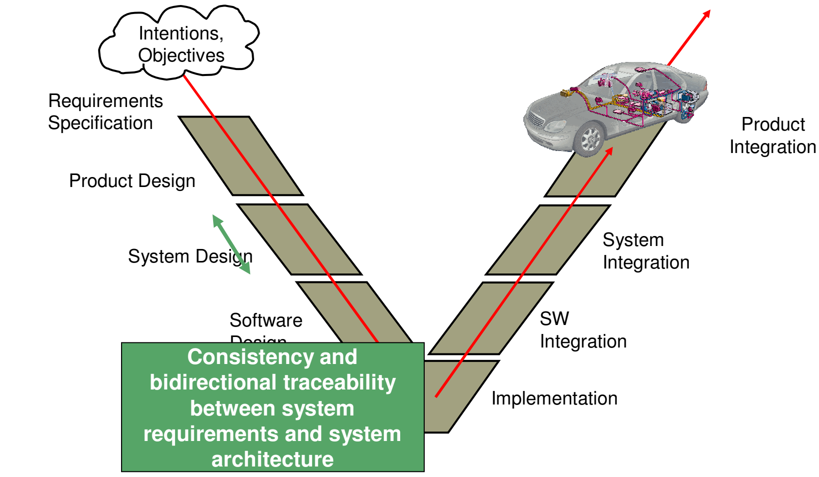
V Model

SPICE (Software Process Improvement and Capability Determination, ISO/IEC 15504)

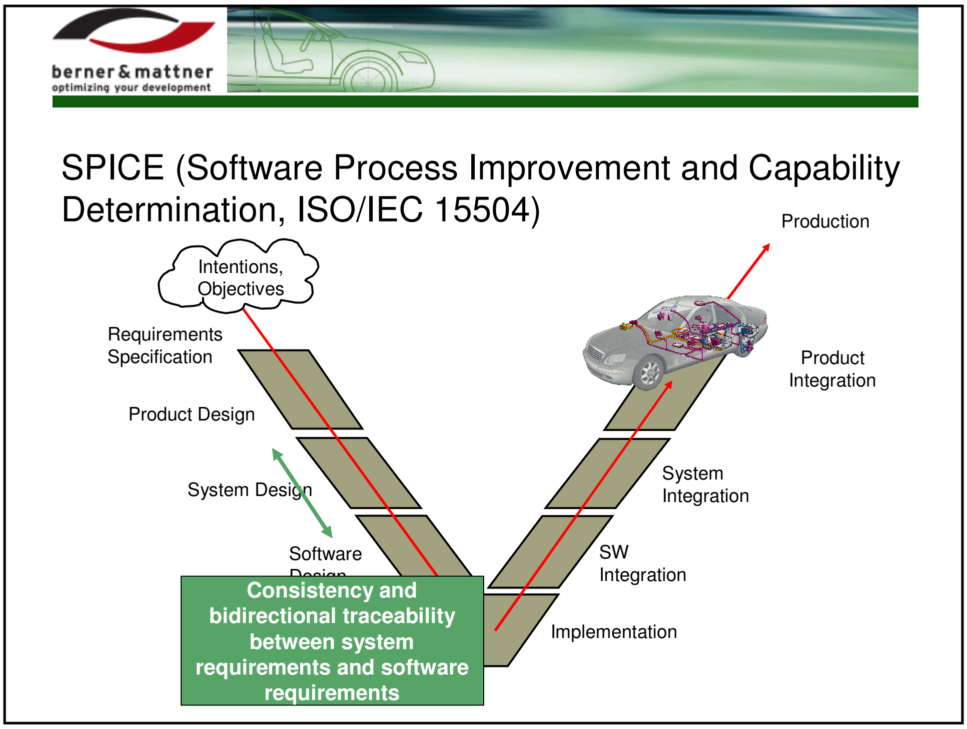


Consistency and bidirectional traceability between customer requirements and system requirements

SPICE (Software Process Improvement and Capability Determination, ISO/IEC 15504)



Consistency and bidirectional traceability between system requirements and system architecture



berner & mattner
optimizing your development

SPICE (Software Process Improvement and Capability Determination, ISO/IEC 15504)

Intentions, Objectives

Requirements Specification

Product Design

System Design

Software Design

Implementation

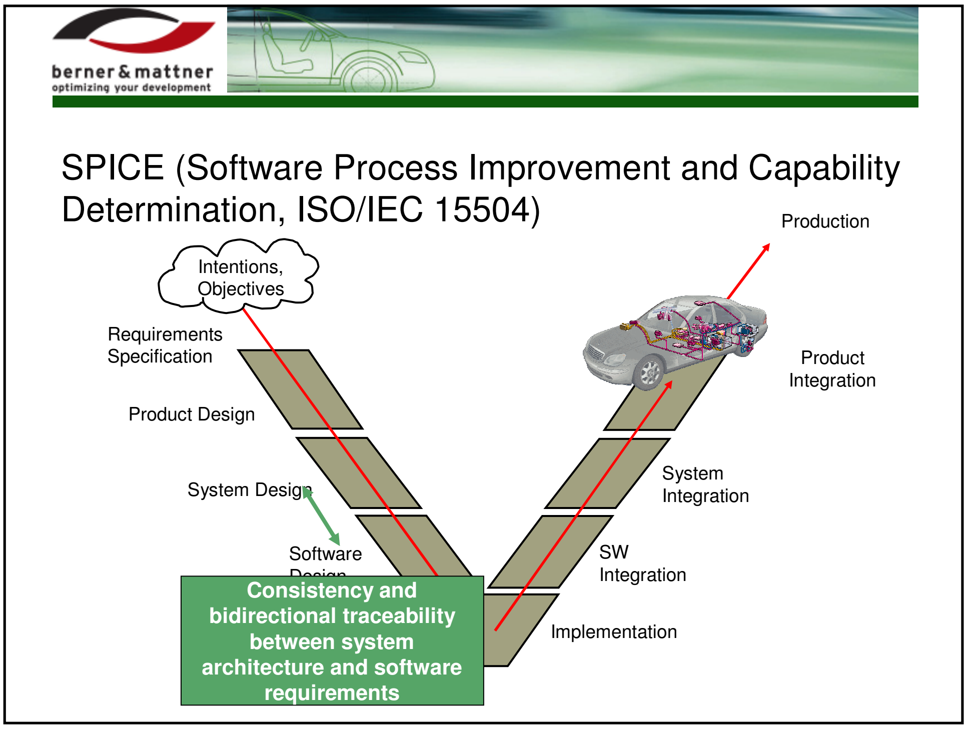
SW Integration

System Integration

Product Integration

Production

Consistency and bidirectional traceability between system requirements and software requirements



berner & mattner
optimizing your development

SPICE (Software Process Improvement and Capability Determination, ISO/IEC 15504)

Intentions, Objectives

Requirements Specification

Product Design

System Design

Software Design

Implementation

SW Integration

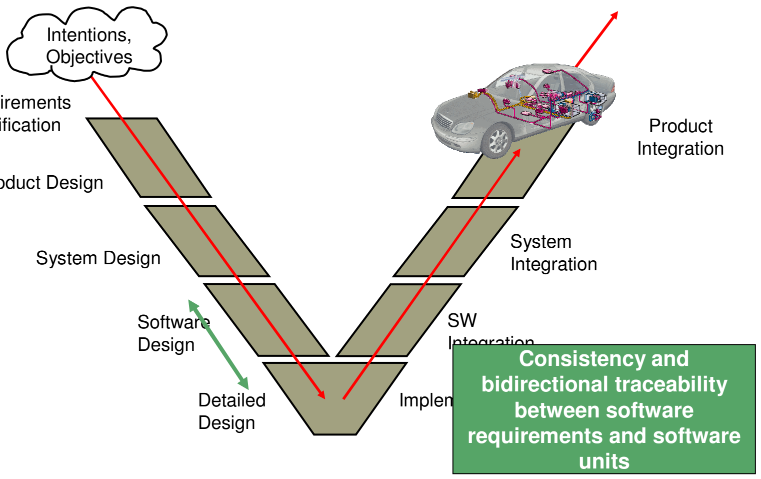
System Integration

Product Integration

Production

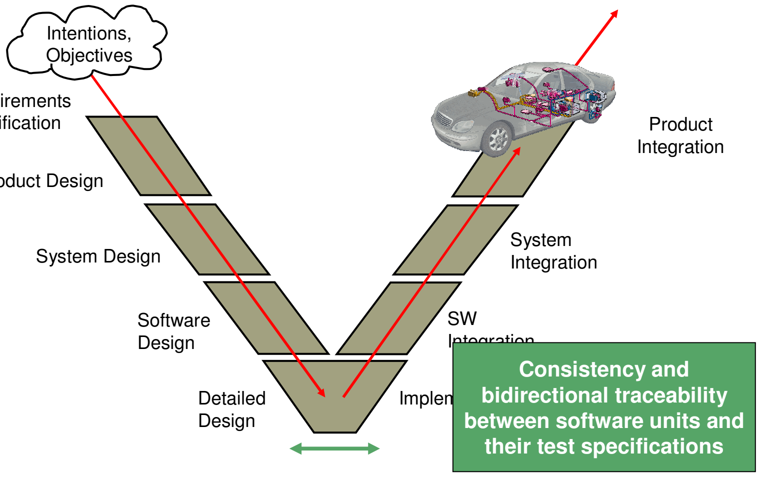
Consistency and bidirectional traceability between system architecture and software requirements

SPICE (Software Process Improvement and Capability Determination, ISO/IEC 15504)

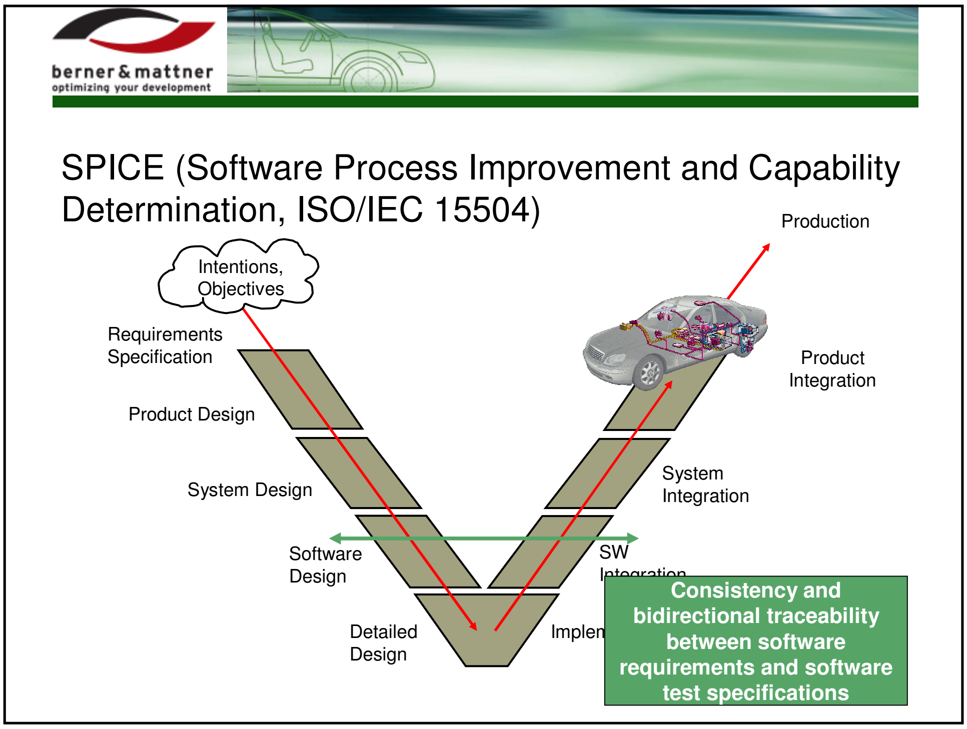


Consistency and bidirectional traceability between software requirements and software units

SPICE (Software Process Improvement and Capability Determination, ISO/IEC 15504)



Consistency and bidirectional traceability between software units and their test specifications



berner & mattner
optimizing your development

SPICE (Software Process Improvement and Capability Determination, ISO/IEC 15504)

Intentions, Objectives

Requirements Specification

Product Design

System Design

Software Design

Detailed Design

Implen

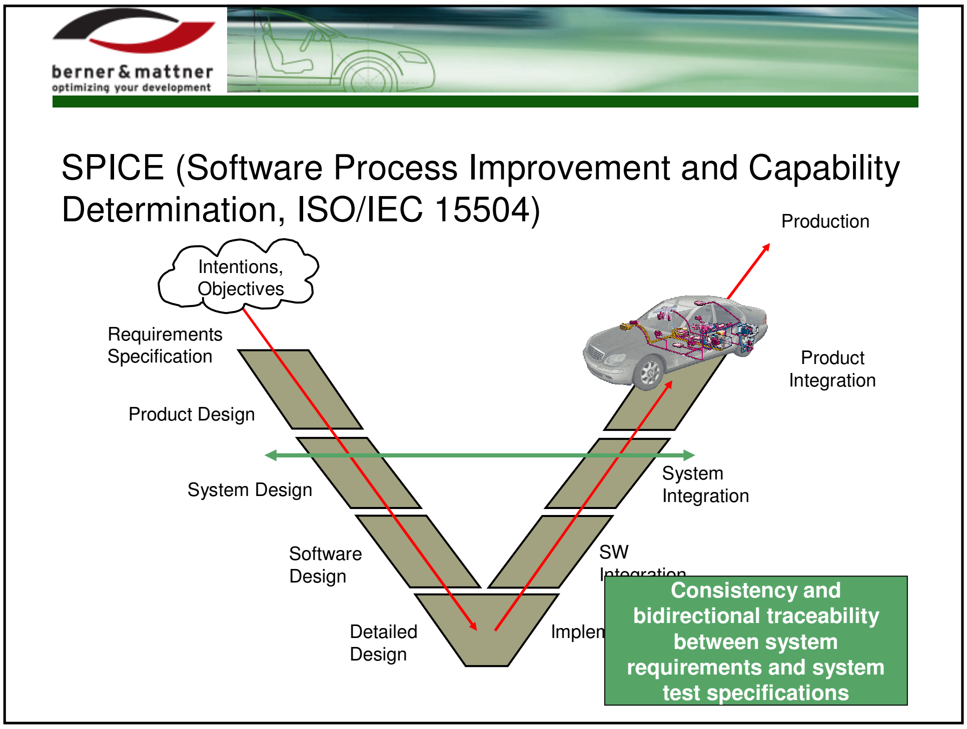
SW Integration

System Integration

Product Integration

Production

Consistency and bidirectional traceability between software requirements and software test specifications



berner & mattner
optimizing your development

SPICE (Software Process Improvement and Capability Determination, ISO/IEC 15504)

Intentions, Objectives

Requirements Specification

Product Design

System Design

Software Design

Detailed Design

Implen

SW Integration

System Integration

Product Integration

Production

Consistency and bidirectional traceability between system requirements and system test specifications

Road Vehicles – Functional Safety (ISO 26262)

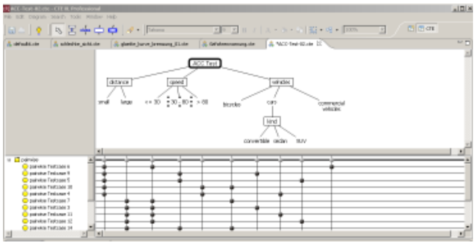
Subset of typical demands

- To evaluate the completeness of test cases and to demonstrate that there is no unintended functionality, the coverage of requirements at the software unit level shall be determined.
- To evaluate the completeness of tests and to obtain confidence that there is no unintended functionality the coverage of requirements at the software architectural level by test cases shall be determined.
- Each functional and technical safety requirement shall be verified (by test, if applicable) at least once in the complete integration subphase.

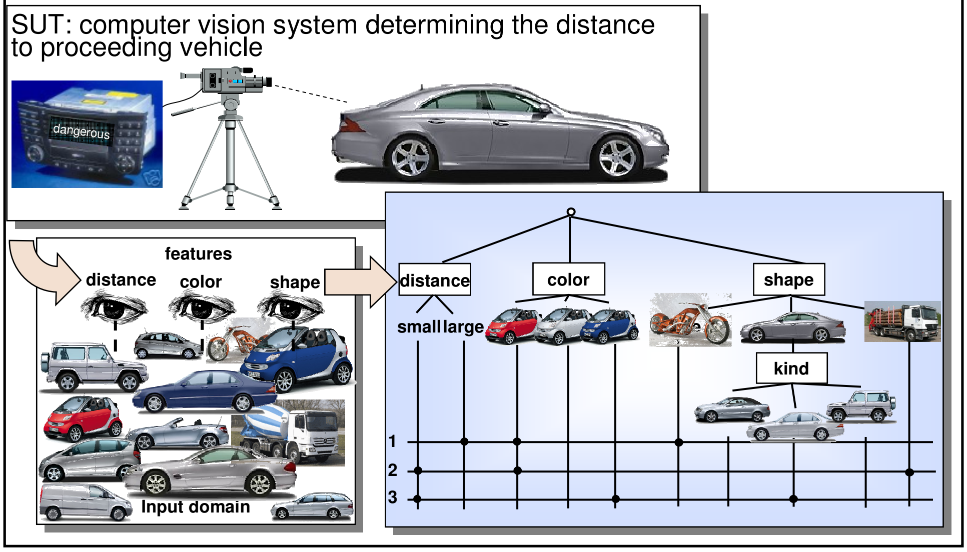
Similar requirements in DOD2167A, IEC 61508 etc.

Classification-Tree Method

- All-purpose test method for specification-based test case design
  - Independent of test phase (from unit to system testing)
  - Independent of application domain (technical systems as well as IT systems)
  - Independent of certain test objects
- Comprehensive and easily understandable test documentation
- Good abstraction
- Systematic procedure, proven in use
- Clear graphical representation of test complexity and amount
- Widely used
- Recommended by standards like ISTQB Certified Tester
- Tool support (CTE XL, CTE XL Prof.)



Classification-Tree Method

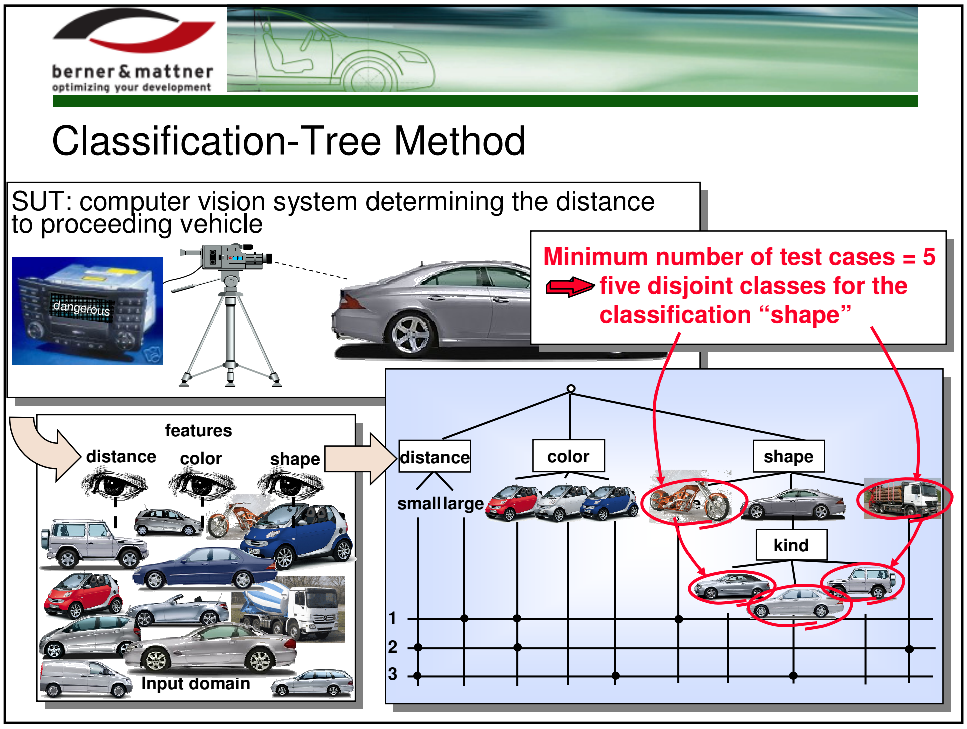


SUT: computer vision system determining the distance to proceeding vehicle

Classification-Tree Method

Additional Aspects

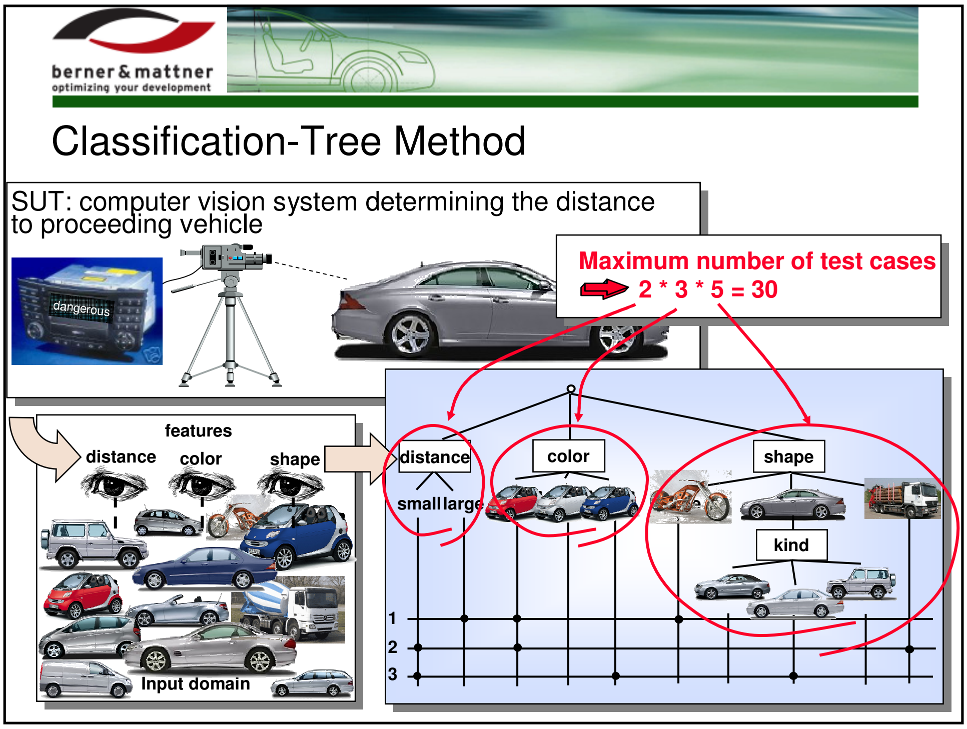
- Vehicle speed of system vehicle
- Relative speed between target vehicle and system vehicle
- Weather conditions: clear, rain, snowfall
- Daytime: night, morning / evening, noon
- Lighting: Low sun angle, oncoming vehicles with high beam
- User action: braking, override, none
- …



Classification-Tree Method

SUT: computer vision system determining the distance to proceeding vehicle

Minimum number of test cases = 5
⇒ five disjoint classes for the classification "shape"

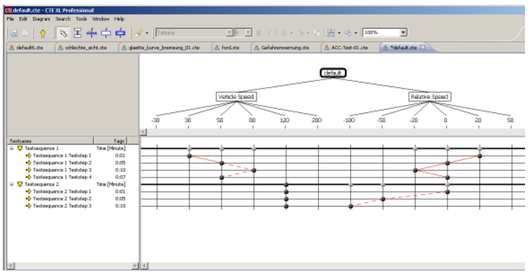


Classification-Tree Method

SUT: computer vision system determining the distance to proceeding vehicle

Maximum number of test cases
⇒ 2 * 3 * 5 = 30

Classification-Tree Method



Testcases	Tags
Testsequence 1	Time [Minute]
Testsequence 1 Teststep 1	0:01
Testsequence 1 Teststep 2	0:05
Testsequence 1 Teststep 3	0:10
Testsequence 1 Teststep 4	0:07
Testsequence 2	Time [Minute]
Testsequence 2 Teststep 1	0:01
Testsequence 2 Teststep 2	0:05
Testsequence 2 Teststep 3	0:10

Weightings for classes

Logical dependency rules
Truck ⇒ not *high speed*

Generation rules

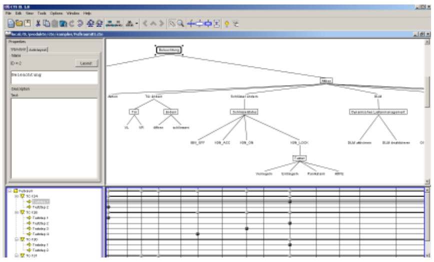
- Pairwise (*distance*, *shape*)
- Prioritized Pairwise (*distance*, *shape*)
- (*distance* * *shape*) + *color*

Test sequences

- Sequence of test steps with timing information
- Function definitions for value changes between test steps

Classification-Tree Editor CTE XL Prof.

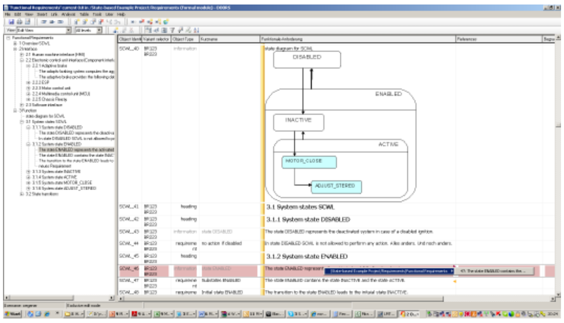
- Syntax-oriented, context sensitive graphical editor supporting the classification-tree method
- Hierarchical structuring of large classification trees and large numbers of test cases and test sequences
- Automatic verification of test cases against dependency rules
- Automatic test case generation according to generation rules
- Modeling of test sequences
- Interfaces for DOORS, TESSY, QualityCenter, MESSINA, TPT etc.
- Statistics
- Tag concept for annotation of information



DOORS

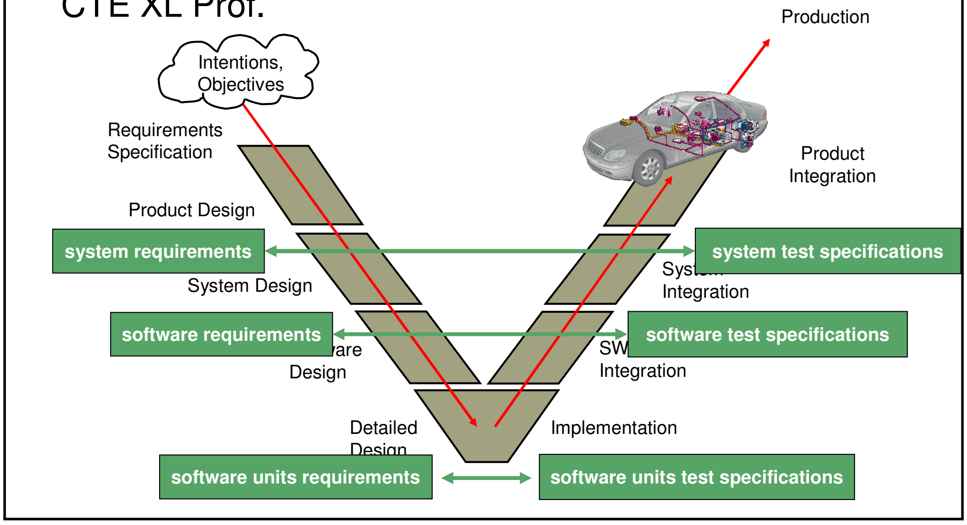
- Product of IBM Rational
- Industry standard for requirements management
- Supports requirements exchange between project partners (OEM – Suppliers) during the requirements negotiation process
- Unique identification of requirements
- Linking of requirements
- User management
- Baselining, Histories

- often used for test management, too


'Functional Requirements' current 0.8 in /State-based Example Project/Requirements (Formal module) - DOORS
File Edit View Insert Link Analysis Table Tools User Help
Functional Requirements
1 Overview SCWL
2 Interface
3 Function
state diagram for SCWL
DISABLED
ENABLED
INACTIVE
ACTIVE
MOTOR_CLOSE
ADJUST_STERED

Object Identifier	Variant selector	Object Type	Kurzname	Funktionale Anforderung
SCWL_40	BR123 BR223	information		state diagram for SCWL
SCWL_41	BR123 BR223	heading		3.1 System states SCWL
SCWL_42	BR123 BR223	heading		3.1.1 System state DISABLED
SCWL_43	BR123 BR223	information	state DISABLED	The state DISABLED represents the deactivated system in case of a disabled ignition.
SCWL_44	BR123 BR223	requirement	no action if disabled	In state DISABLED SCWL is not allowed to perform any action. Alles anders. Und noch anders.
SCWL_45	BR123 BR223	heading		3.1.2 System state ENABLED
SCWL_46	BR123 BR223	information	state ENABLED	The state ENABLED represent
SCWL_47	BR123 BR223	requirement	Substates ENABLED	The state ENABLED contains the state INACTIVE and the state ACTIVE.
SCWL_48	BR123	requirement	Initial state ENABLED	The transition to the state ENABLED leads to the initial state INACTIVE.

Horizontal Requirements Tracing using DOORS and CTE XL Prof.



Production

Intentions, Objectives

Requirements Specification

Product Integration

Product Design

system requirements

system test specifications

System Design

System Integration

software requirements

software test specifications

Software Design

SW Integration

Detailed Design

Implementation

software units requirements

software units test specifications

Target Elements for Requirements Tracing

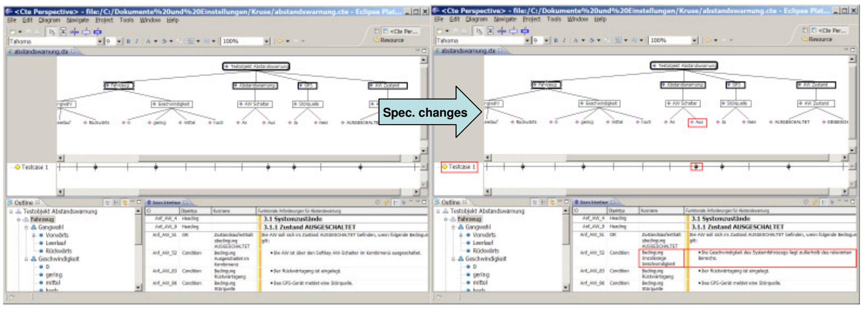
- Elements of the classification tree
  - Classifications
    - Example: the distance must be controlled continuously ⇒ *distance*
  - Classes
    - Example when the distance falls below speed/2,5 meters for more than a second, send warning ⇒ *small, large*
- Dependency rules
  - Example: the system must be inactive for small speeds below 30 km/h ⇒ *speed: <30* ⇒ *state: inactive*

Target Elements for Requirements Tracing

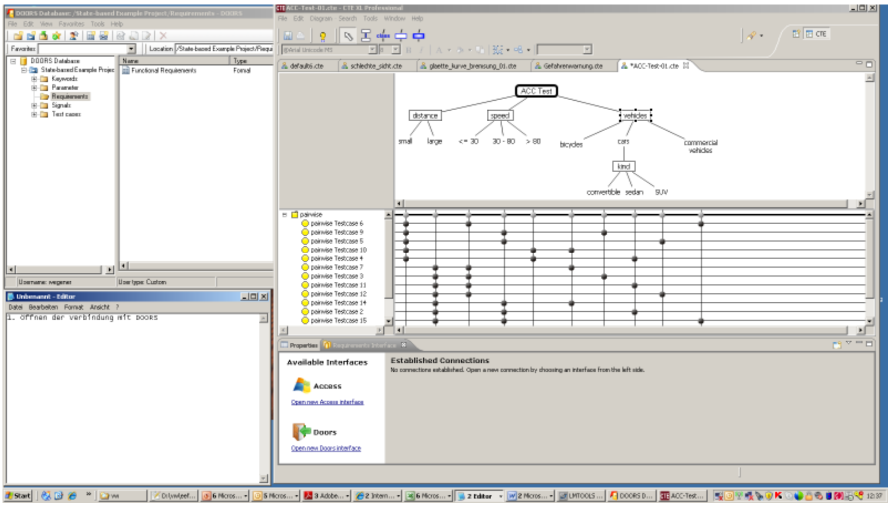
- Generation rules
  - Example: all preceding vehicles have to be detected independent of the system vehicle‘s speed ⇒ *speed * vehicle kinds*
- Elements of the combination table
  - Test cases
    - Example *speed: high*, *distance: small*, *vehicle: truck*, *color: black*, …)
  - Test sequences
    - set of test steps
  - Test steps
    - Example: when the speed falls below 30 km/h for more than a second the system has to be deactivated, a corresponding info message has to be displayed ⇒ step1: *speed: 50 km/h*, step2: *speed: <30 km/h*)

Integration of DOORS and CTE XL Prof.

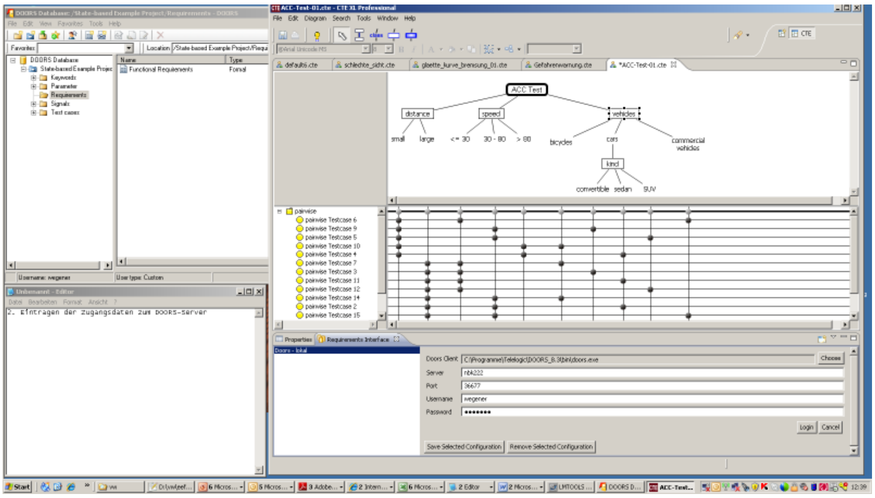
- Linking requirements to target elements of the classification tree and combination table
- Automatic monitoring of requirement changes
- Highlighting of target elements necessary to review



Integration of DOORS and CTE XL Prof.



Integration of DOORS and CTE XL Prof.



Username: wegener

User type: Custom

Doors Client: C:\Programme\Telelogic\DOORS_8.3\bin\doors.exe

Server: nbk222

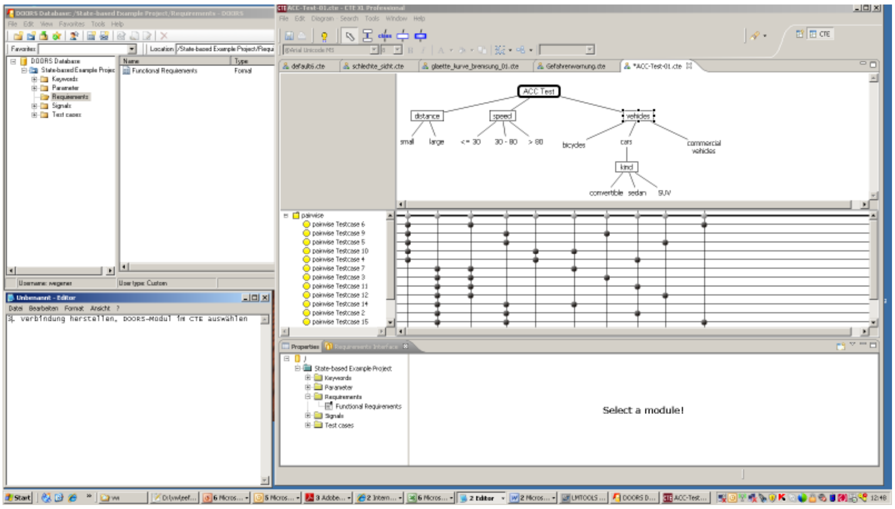
Port: 36677

Username: wegener

Password: ••••••••

```
2. Eintragen der Zugangsdaten zum DOORS-Server
```

Integration of DOORS and CTE XL Prof.



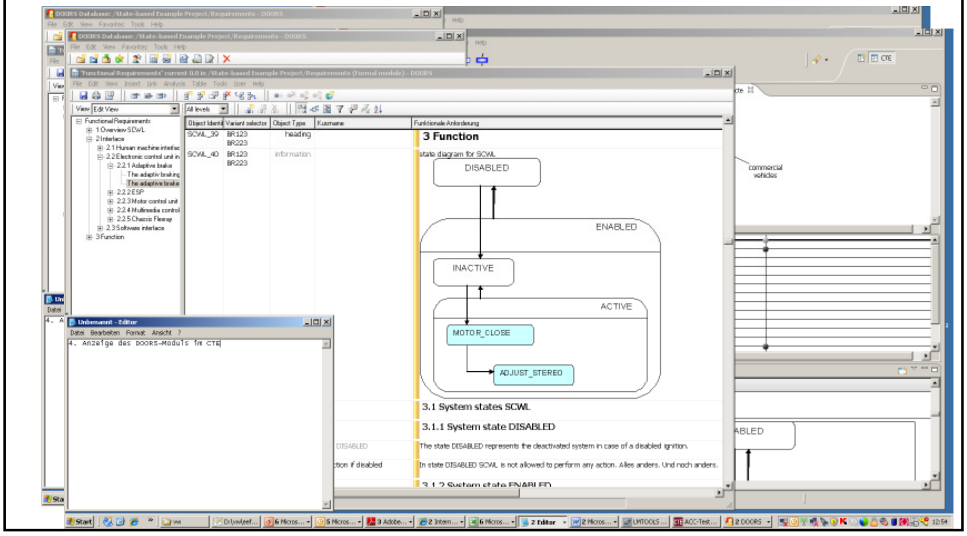
Username: wegener

User type: Custom

Select a module!

```
3. Verbindung herstellen, DOORS-Modul im CTE auswählen
```

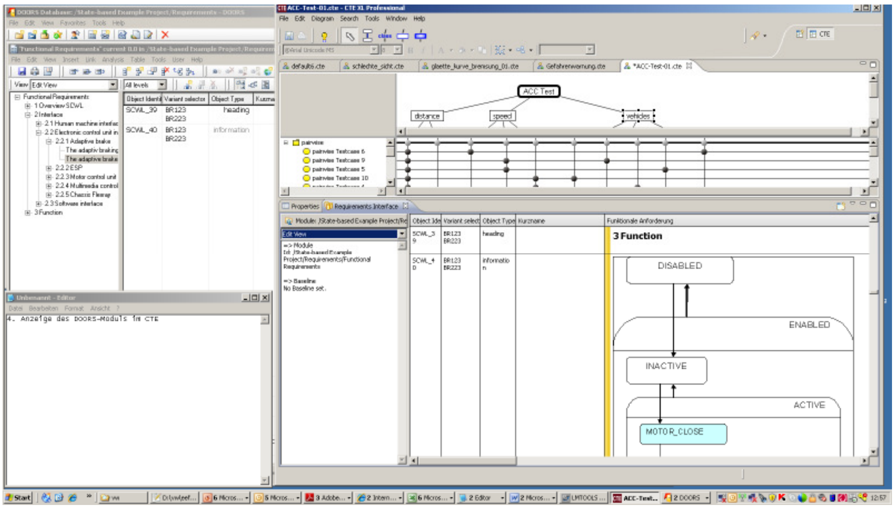
Integration of DOORS and CTE XL Prof.



Object Identi	Variant selector	Object Type	Kurzname	Funktionale Anforderung
SCWL_39	BR123 BR223	heading		3 Function
SCWL_40	BR123 BR223	information		state diagram for SCWL
				3.1 System states SCWL
				3.1.1 System state DISABLED
			DISABLED	The state DISABLED represents the deactivated system in case of a disabled ignition.
			ction if disabled	In state DISABLED SCWL is not allowed to perform any action. Alles anders. Und noch anders.
				3.1.2 System state ENABLED

```
4. Anzeige des DOORS-Moduls im CTE
```

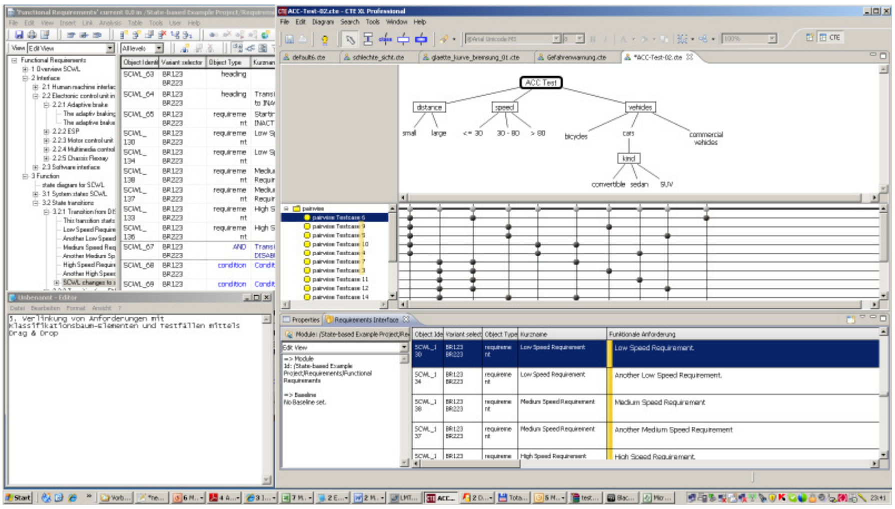
Integration of DOORS and CTE XL Prof.



Object Ide	Variant select	Object Type	Kurzname	Funktionale Anforderung
SCWL_39	BR123 BR223	heading		3 Function
SCWL_40	BR123 BR223	information		

```
4. Anzeige des DOORS-Moduls im CTE
```

Integration of DOORS and CTE XL Prof.



Unbenannt - Editor

5. Verlinkung von Anforderungen mit Klassifikationsbaum-Elementen und Testfällen mittels Drag & Drop

Module: /State-based Example Project/Re

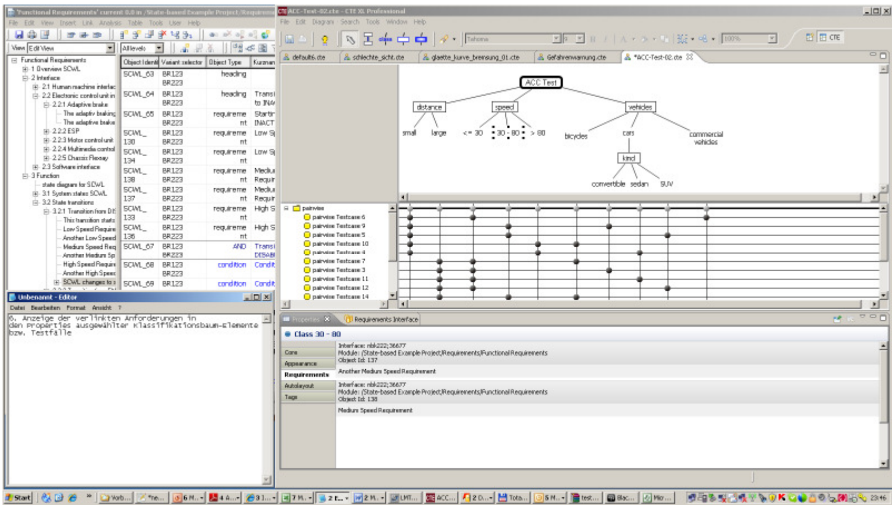
Edit View

=> Module
Id: /State-based Example Project/Requirements/Functional Requirements

=> Baseline
No Baseline set.

Object Ide	Variant select	Object Type	Kurzname	Funktionale Anforderung
SCWL_1 30	BR123 BR223	requirement	Low Speed Requirement	Low Speed Requirement.
SCWL_1 34	BR123 BR223	requirement	Low Speed Requirement	Another Low Speed Requirement.
SCWL_1 38	BR123 BR223	requirement	Medium Speed Requirement	Medium Speed Requirement
SCWL_1 37	BR123 BR223	requirement	Medium Speed Requirement	Another Medium Speed Requirement
SCWL_1	BR123	requireme	High Speed Requirement	High Speed Requirement.

Integration of DOORS and CTE XL Prof.



Unbenannt - Editor

6. Anzeige der verlinkten Anforderungen in den Properties ausgewählter Klassifikationsbaum-Elemente bzw. Testfälle

Class 30 - 80

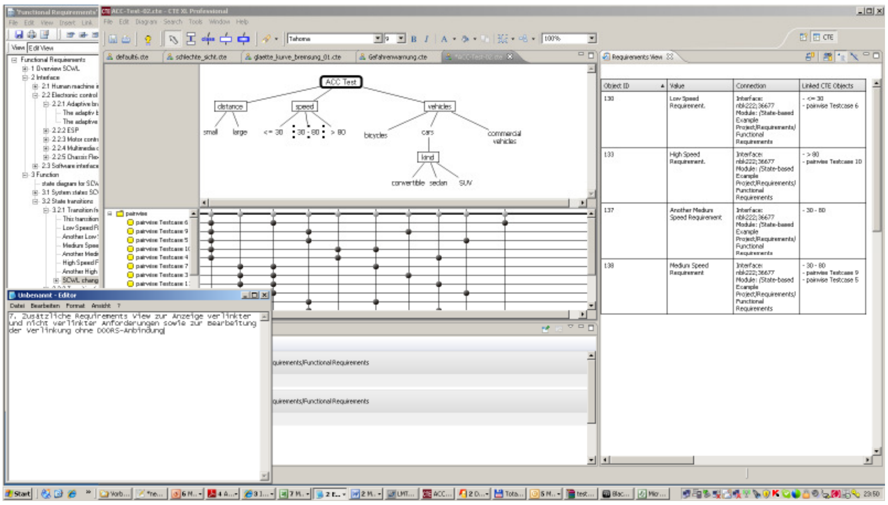
Interface: nbk222;36677
Module: /State-based Example Project/Requirements/Functional Requirements
Object Id: 137

Another Medium Speed Requirement

Interface: nbk222;36677
Module: /State-based Example Project/Requirements/Functional Requirements
Object Id: 138

Medium Speed Requirement

Integration of DOORS and CTE XL Prof.

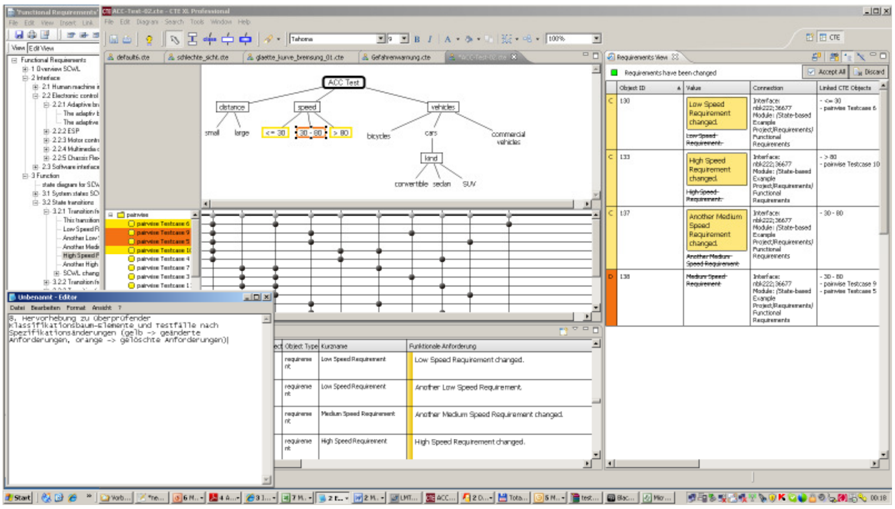


7. Zusätzliche Requirements View zur Anzeige verlinkter und nicht verlinkter Anforderungen sowie zur Bearbeitung der Verlinkung ohne DOORS-Anbindung

Object ID	Value	Connection	Linked CTE Objects
130	Low Speed Requirement.	Interface: nbk222; 36677 Module: /State-based Example Project/Requirements/Functional Requirements	- <= 30 - pairwise Testcase 6
133	High Speed Requirement.	Interface: nbk222; 36677 Module: /State-based Example Project/Requirements/Functional Requirements	- > 80 - pairwise Testcase 10
137	Another Medium Speed Requirement	Interface: nbk222; 36677 Module: /State-based Example Project/Requirements/Functional Requirements	- 30 - 80
138	Medium Speed Requirement	Interface: nbk222; 36677 Module: /State-based Example Project/Requirements/Functional Requirements	- 30 - 80 - pairwise Testcase 9 - pairwise Testcase 5

berner & mattner
optimizing your development

Integration of DOORS and CTE XL Prof.



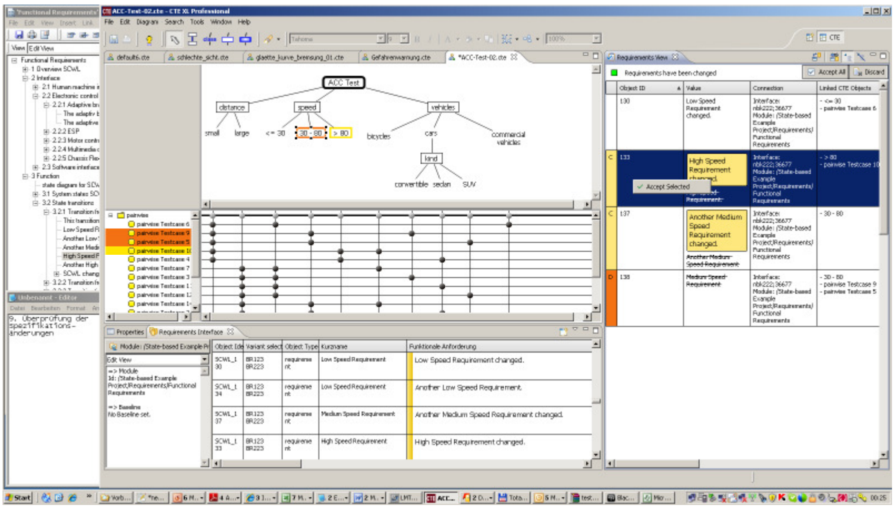
8. Hervorhebung zu überprüfender Klassifikationsbaum-Elemente und Testfälle nach Spezifikationsänderungen (gelb -> geänderte Anforderungen, orange -> gelöschte Anforderungen)

Requirements have been changed

	Object ID	Value	Connection	Linked CTE Objects
C	130	Low Speed Requirement changed. ~~Low Speed Requirement.~~	Interface: nbk222; 36677 Module: /State-based Example Project/Requirements/Functional Requirements	- <= 30 - pairwise Testcase 6
C	133	High Speed Requirement changed. ~~High Speed Requirement.~~	Interface: nbk222; 36677 Module: /State-based Example Project/Requirements/Functional Requirements	- > 80 - pairwise Testcase 10
C	137	Another Medium Speed Requirement changed. ~~Another Medium Speed Requirement~~	Interface: nbk222; 36677 Module: /State-based Example Project/Requirements/Functional Requirements	- 30 - 80
D	138	~~Medium Speed Requirement~~	Interface: nbk222; 36677 Module: /State-based Example Project/Requirements/Functional Requirements	- 30 - 80 - pairwise Testcase 9 - pairwise Testcase 5

Object Type	Kurzname	Funktionale Anforderung
requirement	Low Speed Requirement	Low Speed Requirement changed.
requirement	Low Speed Requirement	Another Low Speed Requirement.
requirement	Medium Speed Requirement	Another Medium Speed Requirement changed.
requirement	High Speed Requirement	High Speed Requirement changed.

Integration of DOORS and CTE XL Prof.



Requirements have been changed

	Object ID	Value	Connection	Linked CTE Objects
	130	Low Speed Requirement changed.	Interface: nbk222; 36677 Module: /State-based Example Project/Requirements/ Functional Requirements	- <= 30 - pairwise Testcase 6
C	133	High Speed Requirement changed. ~~High Speed Requirement~~	Interface: nbk222; 36677 Module: /State-based Example Project/Requirements/ Functional Requirements	- > 80 - pairwise Testcase 10
C	137	Another Medium Speed Requirement changed. ~~Another Medium Speed Requirement~~	Interface: nbk222; 36677 Module: /State-based Example Project/Requirements/ Functional Requirements	- 30 - 80
D	138	~~Medium Speed Requirement~~	Interface: nbk222; 36677 Module: /State-based Example Project/Requirements/ Functional Requirements	- 30 - 80 - pairwise Testcase 9 - pairwise Testcase 5

Module: /State-based Example Pr

Edit View

=> Module
Id: /State-based Example Project/Requirements/Functional Requirements

=> Baseline
No Baseline set.

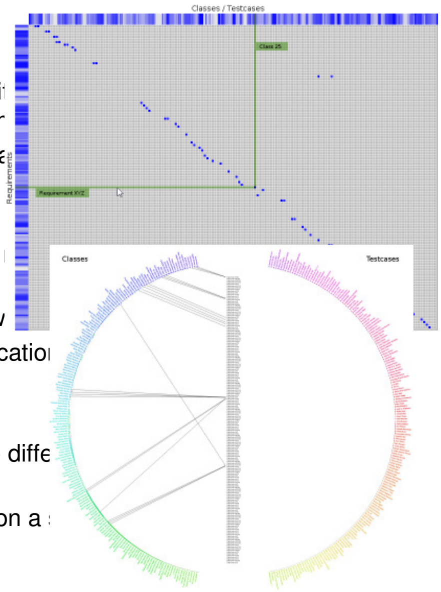
Object Ide	Variant select	Object Type	Kurzname	Funktionale Anforderung
SCWL_1 30	BR123 BR223	requirement	Low Speed Requirement	Low Speed Requirement changed.
SCWL_1 34	BR123 BR223	requirement	Low Speed Requirement	Another Low Speed Requirement.
SCWL_1 37	BR123 BR223	requirement	Medium Speed Requirement	Another Medium Speed Requirement changed.
SCWL_1 33	BR123 BR223	requirement	High Speed Requirement	High Speed Requirement changed.

Summary

- Requirements Tracing demanded in most development standards
  - Vertical tracing
  - Horizontal tracing
- DOORS and CTE XL common tools for requirements management and test case design
- Integration of DOORS and CTE XL Prof. provides a powerful support for horizontal tracing

berner & mattner
optimizing your development

Future Work



- MERAN is a tool to extend DOORS wi modelling capabilities (selection, parar
  - ➢ supporting consistent variant mana test
  - ➢ supporting model-based testing
- Usually, several classification trees to specification module
  - ➢ Additional tooling required to show
  - ➢ Alternatively, export of test specificatio DOORS
- Overview of requirements linking
- Use of natural language processing to diffe corrections
- Closing the gap for horizontal tracing on a “formalized” specification catalogues
- Support of other tools, e.g. MKS