# Model-Based Testing of Complex Hybrid Aerospace Systems


Philipp Helle
EADS Innovation Works
philipp.helle@eads.net

Dr. Udo Brockmeyer
BTC Embedded Systems AG
udo.brockmeyer@btc-es.de



Innovate2013
The IBM Technical Summit

# Please note the following

IBM's statements regarding its plans, directions, and intent are subject to change or withdrawal without notice at IBM's sole discretion.

Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision.

The information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code or functionality. Information about potential future products may not be incorporated into any contract. The development, release, and timing of any future features or functionality described for our products remains at our sole discretion.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon many factors, including considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve results similar to those stated here.

# EADS at a glance


AIRBUS AN EADS COMPANY

EUROCOPTER AN EADS COMPANY

ASTRIUM AN EADS COMPANY

CASSIDIAN AN EADS COMPANY

# BTC Embedded Systems

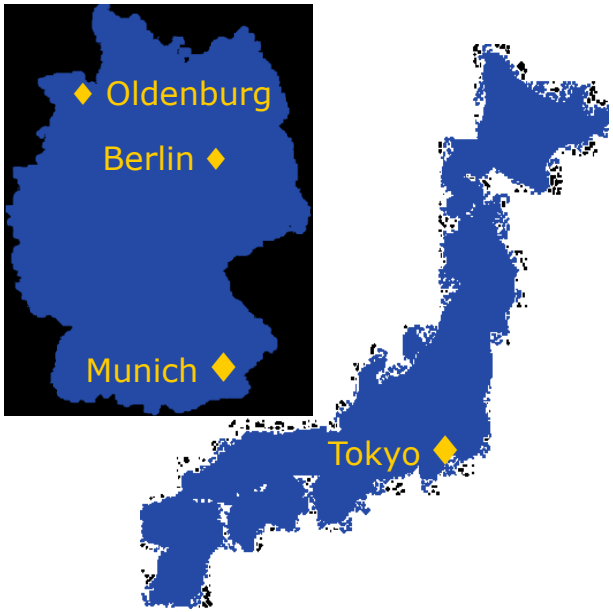
**Model-based Testing Company established in**

**1999**

**BTC-ES Headquarter in Oldenburg (D)**

Subsidiaries in Munich and Berlin (D)

BTC Japan Co., Ltd.




## Mission Statement:

Our mission is to enable customers to increase product quality in a shortened design phase by introducing automatic test and verification technology to the model-based systems & software development process.

## Main Customer Domains:

Automotive, Aerospace

# Agenda

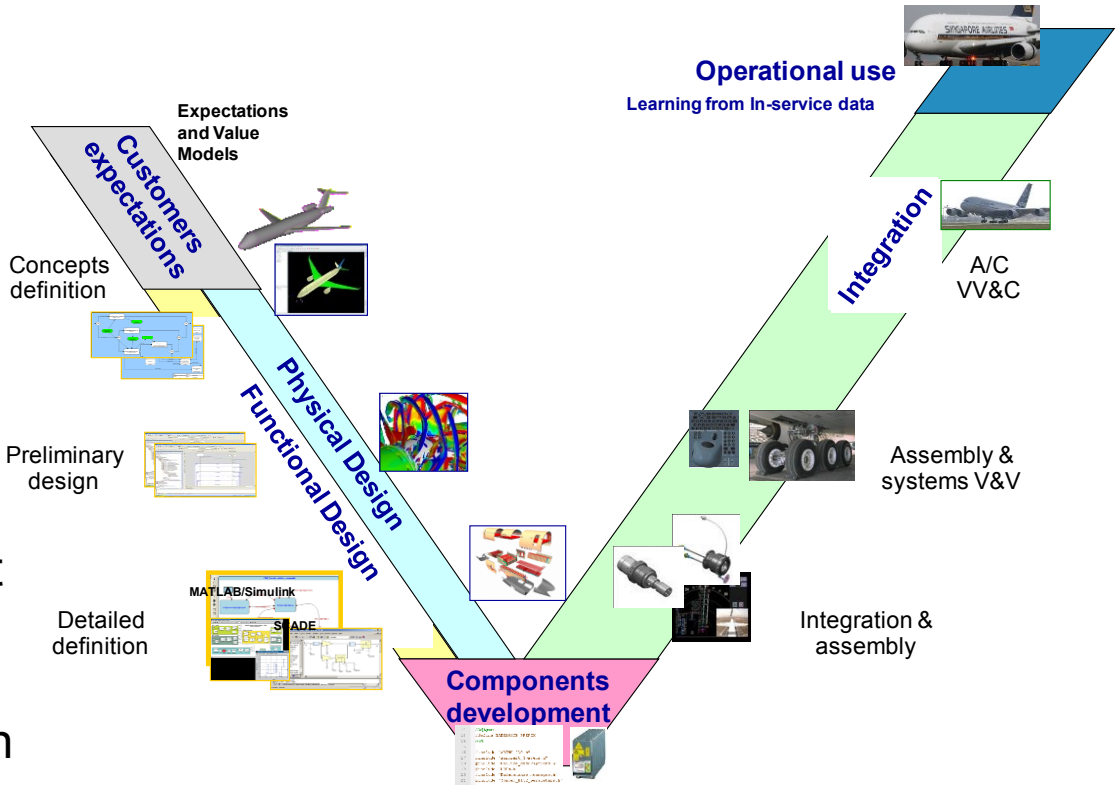
- EADS background and motivation
- Model-based Testing concept and artefacts
- Model-based Testing with Rational Rhapsody, Rhapsody ATG and Rhapsody TestConductor
- Example and Early Results
- Conclusion

# Agenda

- EADS background and motivation
- Model-based Testing concept and artefacts
- Model-based Testing with Rational Rhapsody, Rhapsody ATG and Rhapsody TestConductor
- Example and Early Results
- Conclusion

# Systems Engineering at EADS

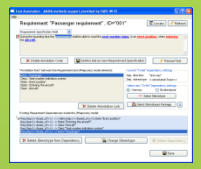
- EADS is typically doing systems specification and systems integration
- Duration and cost of system development efforts has experienced rapid exponential growth over time
- The system design of a product is a key driver for its lifecycle cost.
  Wrong / not optimal decisions in early phases are very costly to fix in later phases



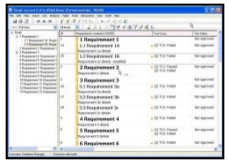

**Model-based Systems Engineering for early Virtual aircraft Functional and Physical View**

# Model-based specification

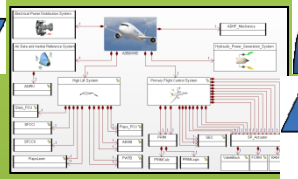
**Requirements analysis**



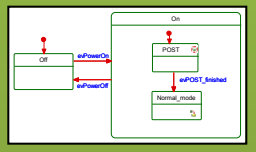
Traceability description



Textual requirements



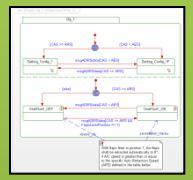
Hierarchy and context description



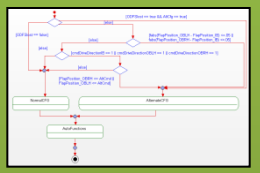

States & Modes
Functions description

**Specification modelling**

Structure description



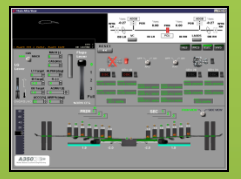
Behaviour description



Algorithm description

**Model usage**

Test scenarios



Integrated simulation

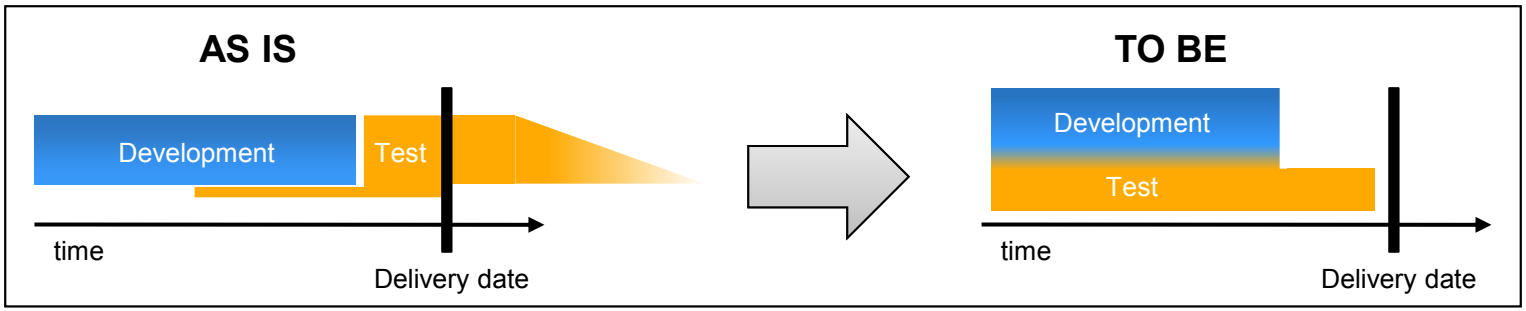
Document generation

Traceability for coverage and impact analysis

# Need for an integrated MBSE approach

- Development and testing are often enough separated in systems engineering resulting in an unnecessary high effort for testing while it is already seen as inseparable in software engineering
- An MBSE approach including model based testing allows **early validation of requirements and verification of the system design** by testing throughout the design phase and facilitates error tracing and impact assessment later on.
- The usage of the OMG UML Testing Profile in conjunction with SysML allows reusing the artefacts from the engineering stage at the testing stage of the development cycle and thereby enables **testing based on the system specification model**

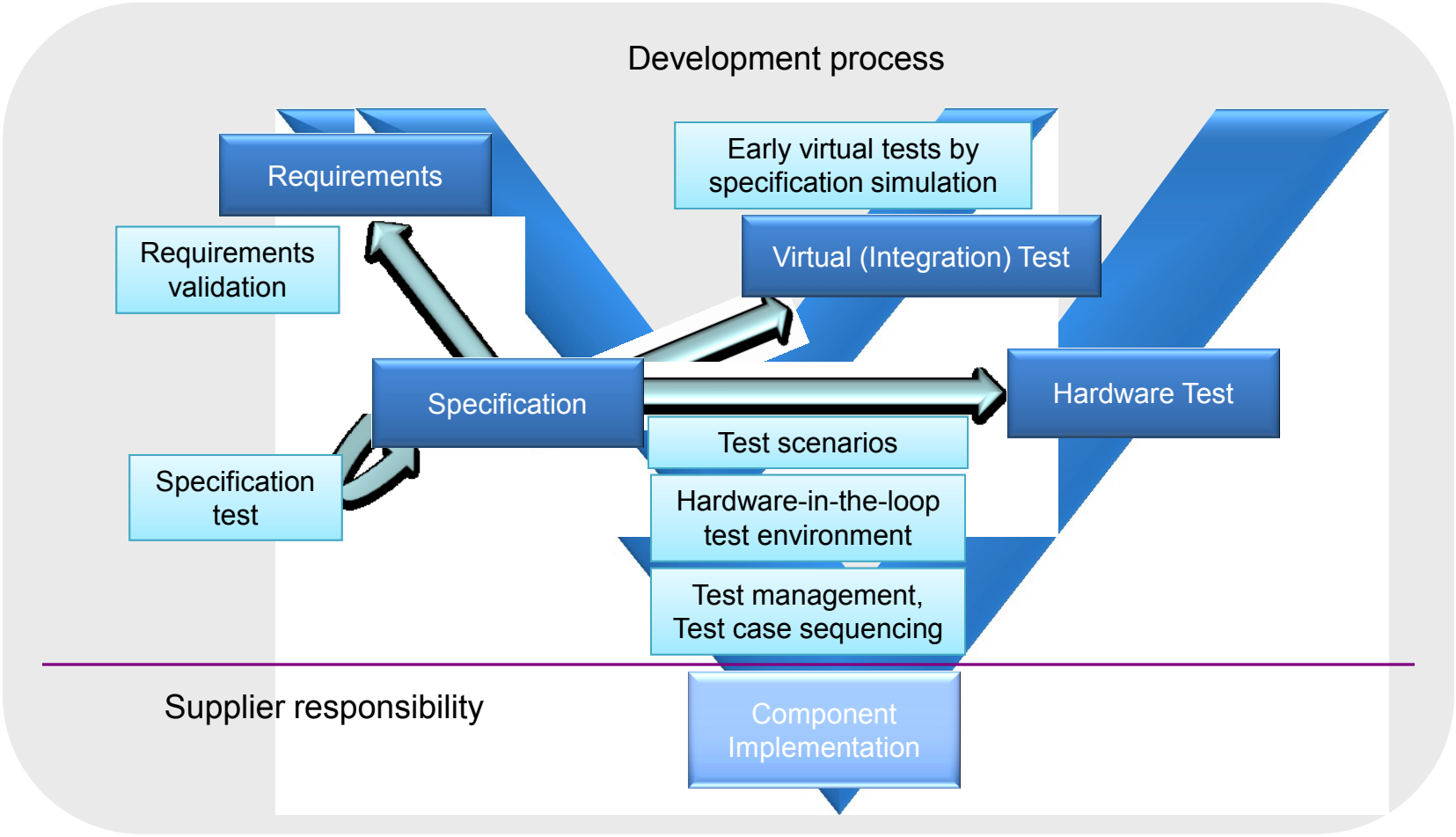
**Model-based testing**



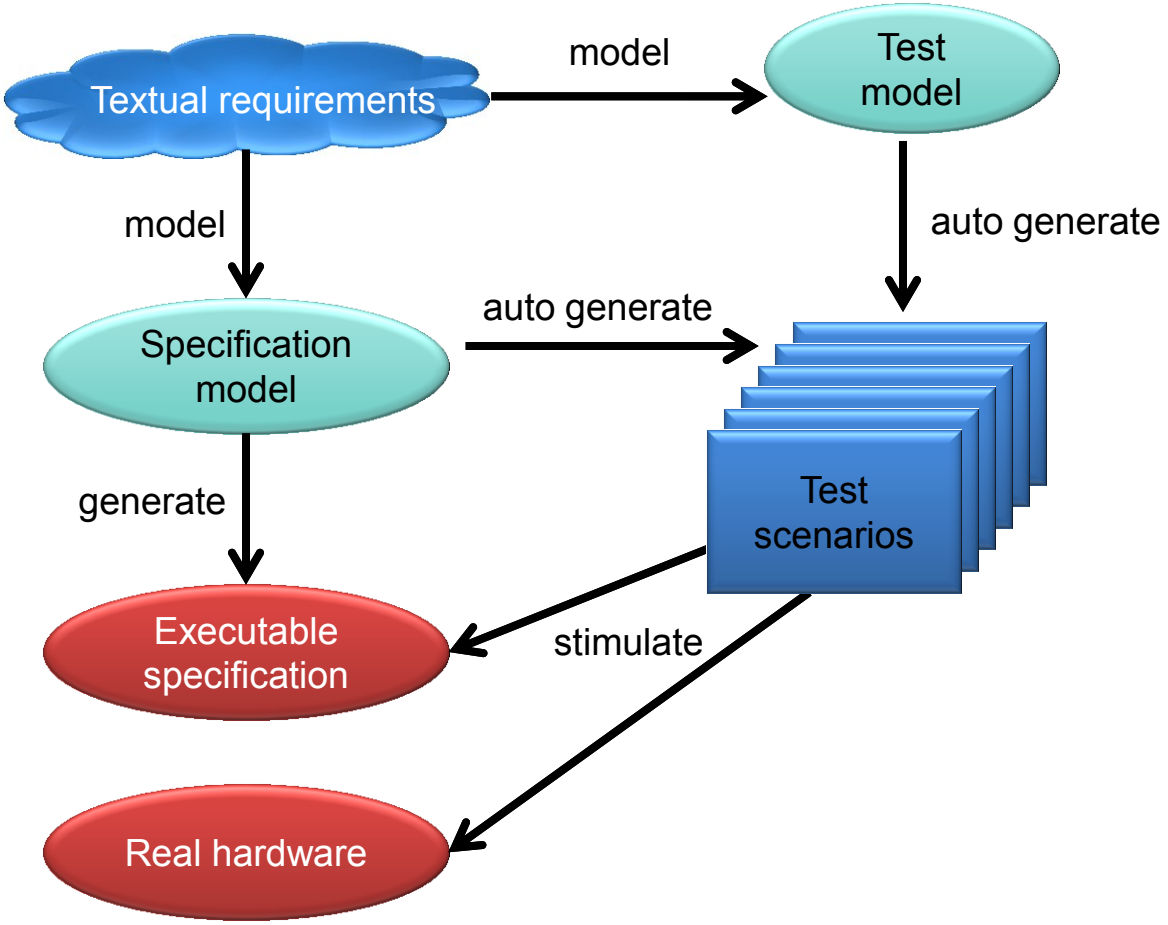
# Agenda

- EADS background and motivation
- Model-based Testing concept and artefacts
- Model-based Testing with Rational Rhapsody, Rhapsody ATG and Rhapsody TestConductor
- Example and Early Results
- Conclusion

# From model-based specification to model-based testing

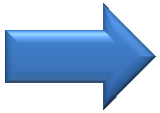


# Artefacts and relations



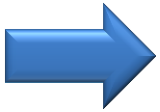
# Benefits of model-based testing

- Seamless traceability from the initial requirements through the system and subsystem functions to test cases and test results



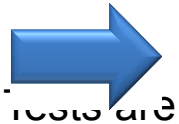
**Better impact analysis**

- Test scenarios can be automatically generated from the specification model based on an in-depth automatic white-box model analysis



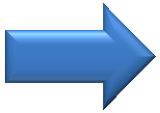
**Higher test coverage**

- Test plans are developed in the context of the SuT, the resources available and the coverage that can be delivered



**Better test management**

- Tests are created earlier in the development process, automatically executed, used for specification testing as well as system testing and reused when the design evolves

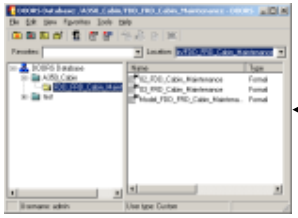


**Overall test effort reduction**

# Agenda

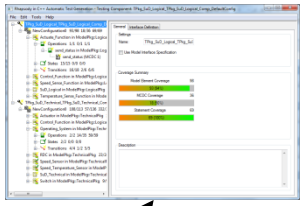
- EADS background and motivation
- Model-based Testing concept and artefacts
- Model-based Testing with Rational Rhapsody, Rhapsody ATG and Rhapsody TestConductor
- Example and Early Results
- Conclusion

# EADS Model-based Tool chain



DOORS

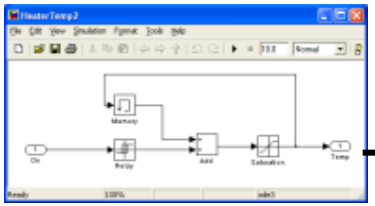
Gateway/Reqtify



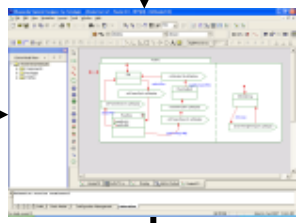
ATG

Traceability

Test Case Generation



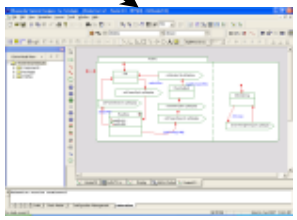
MATLAB/Simulink



Rhapsody

Test Execution

Code generation



TestConductor

Integrated simulation

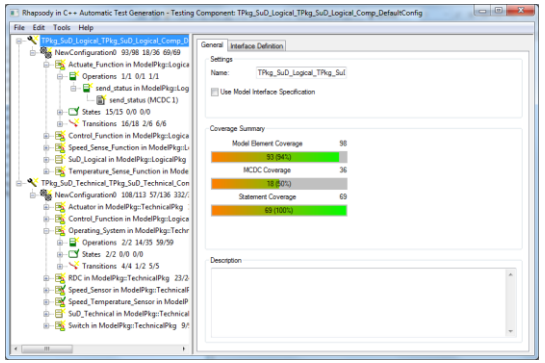
**Tools:**

- **DOORS (IBM)**
  - Requirements Management
- **Rhapsody Gateway (IBM/Geensoft)**
  - Requirements Traceability
- **Rhapsody (IBM)**
  - Specification modelling
  - Model Integration
  - Simulation Execution
- **ATG (IBM/BTC-ES)**
  - Automatic Test Case/Scenario generation
- **TestConductor (IBM/BTC-ES)**
  - Automatic test execution
  - Coverage report generation
- **MATLAB/Simulink (MathWorks)**
  - Physical Behaviour
  - Environment Simulation

# Rhapsody ATG and Rhapsody TestConductor

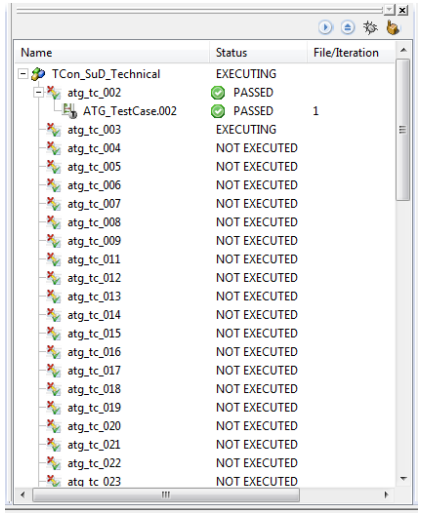
- **IBM® Rational® Rhapsody® Automatic Test Generation (ATG) Add On:**

- Model-based test case generation using UML/SysML test models
- Model coverage, statement coverage, MC/DC
- Incremental creation of test suites



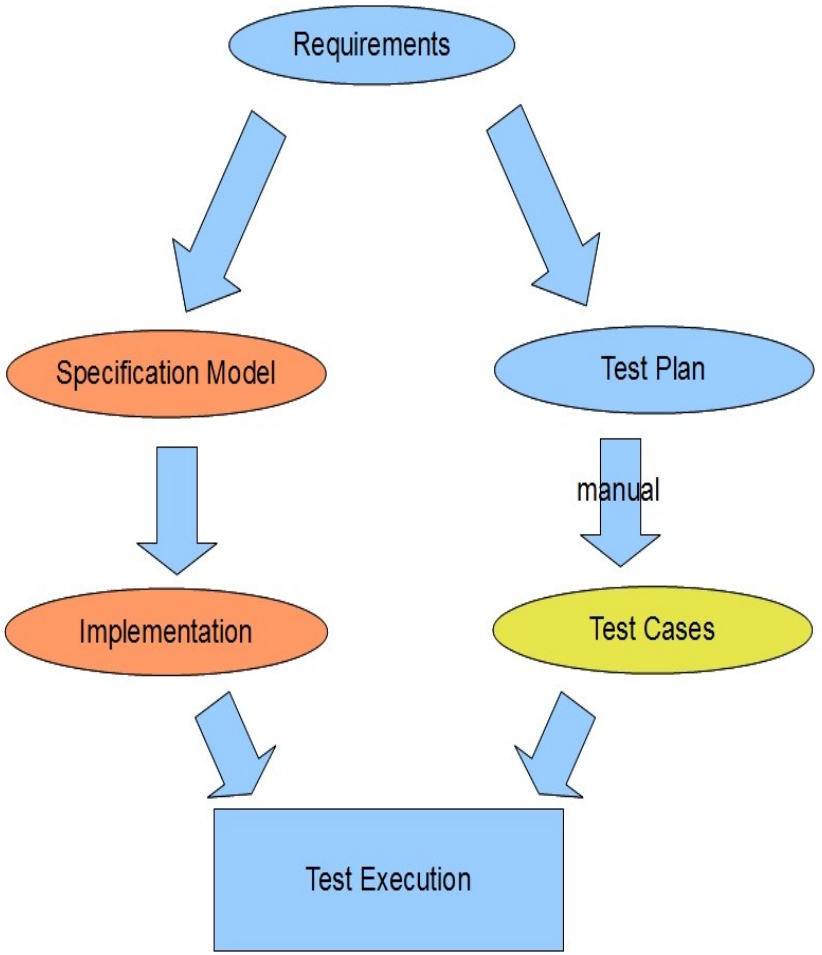
- **IBM® Rational® Rhapsody® TestConductor Add On:**

- Model-based test case specification and exeution using the UML Testing Profile
- Test can be defined graphically using UML sequence diagrams, flowcharts, statecharts or code
- Error analysis using color coded sequence diagrams



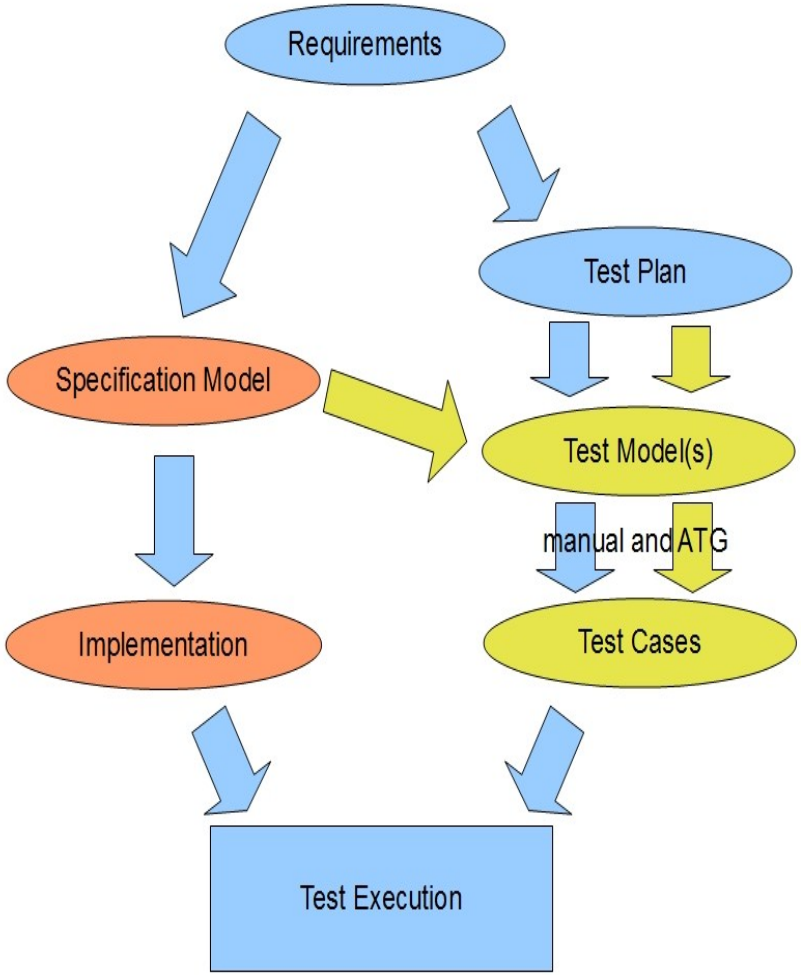
# Test Models and Automatic Test Generation (ATG)

- Model based development using a specification model (lef branch)
- creating tests directly from a mostly textual te plan (right branch)
- Development benefits from modelling but test are mostly manually developed and maintained outside of models



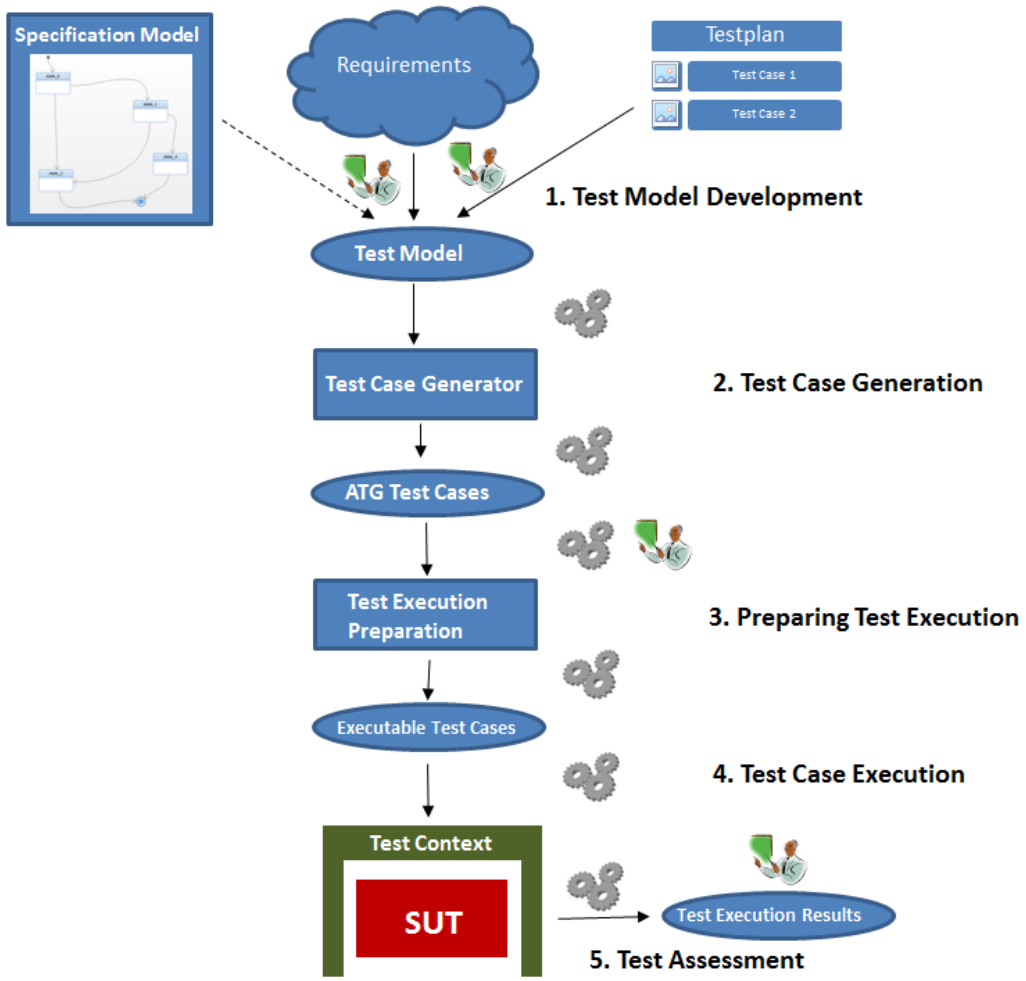
# Test Models and Automatic Test Generation (ATG)

- Model based development **and** model based testing
- Automatically generate tests from test model using ATG
- Test maintenance means maintaining test models



# Test Models and Automatic Test Generation (ATG)

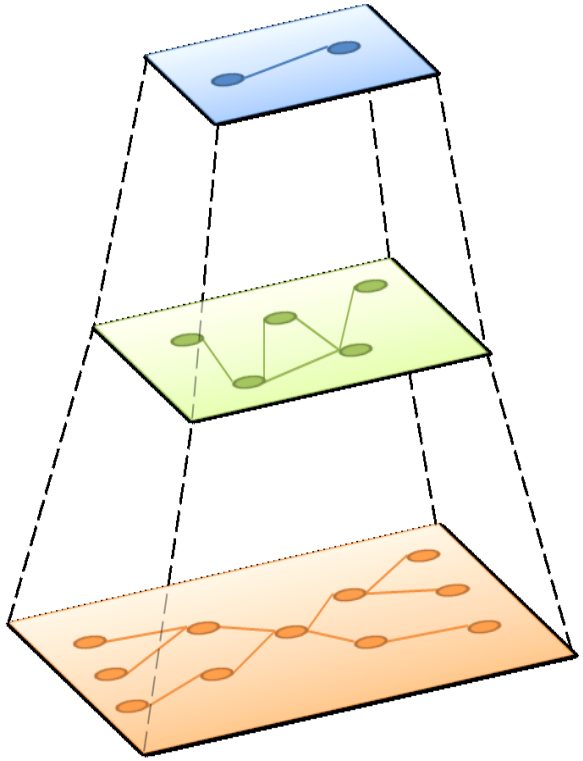
- Test models capture test objectives and test strategies
- ATG generates test cases
- ATG test cases are tailored to the concrete test environment, e.g. TestConductor
- Tests are executed and results are analyzed



# Agenda

- EADS background and motivation
- Model-based Testing concept and artefacts
- Model-based Testing with Rational Rhapsody, Rhapsody ATG and Rhapsody TestConductor
- Example and Early Results
- Conclusion

# Generic example model and application results



**Functional model**
Descriptive
5 functions
4 functional links

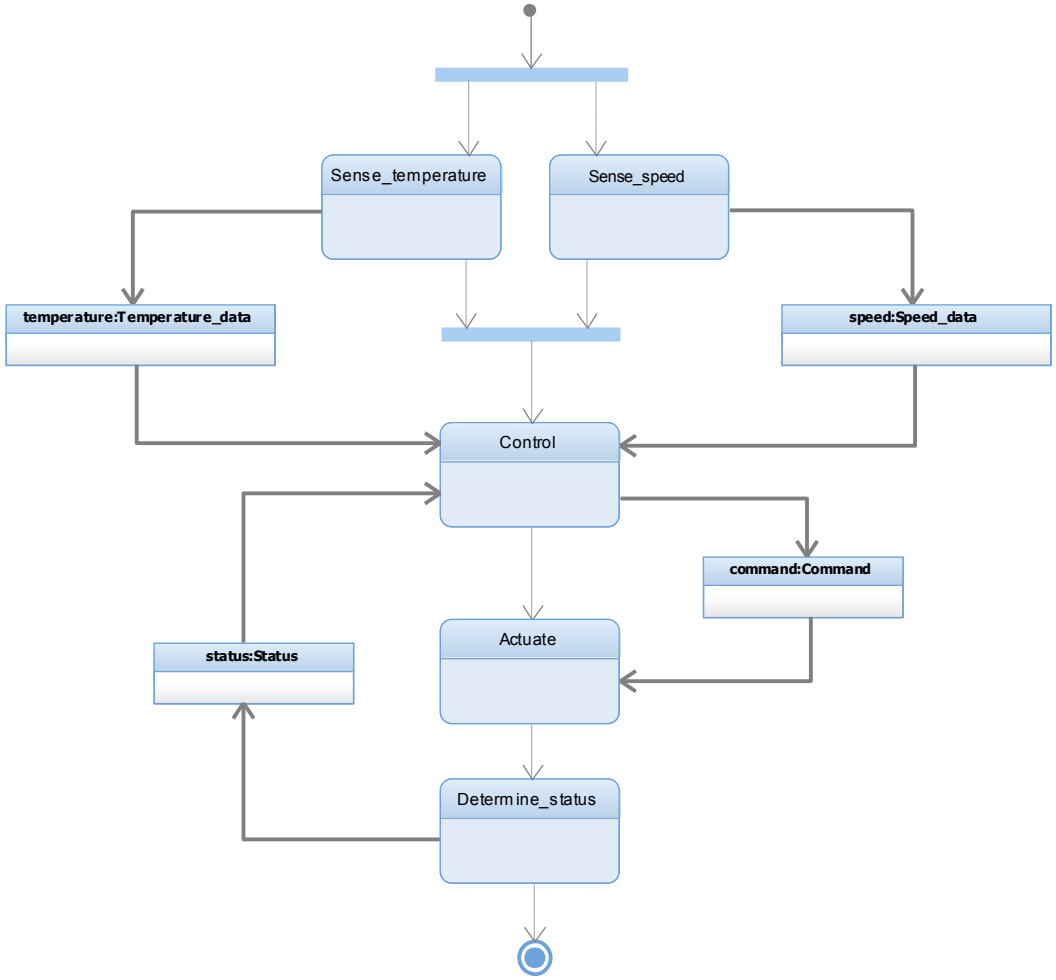
**Logical model**
Executable
4 logical block classes
9 logical block instances
14 logical links

**ATG**
17 Test cases
94% model element coverage (93/98)
50% MC/DC coverage (18/36)
100% statement coverage (69/69)

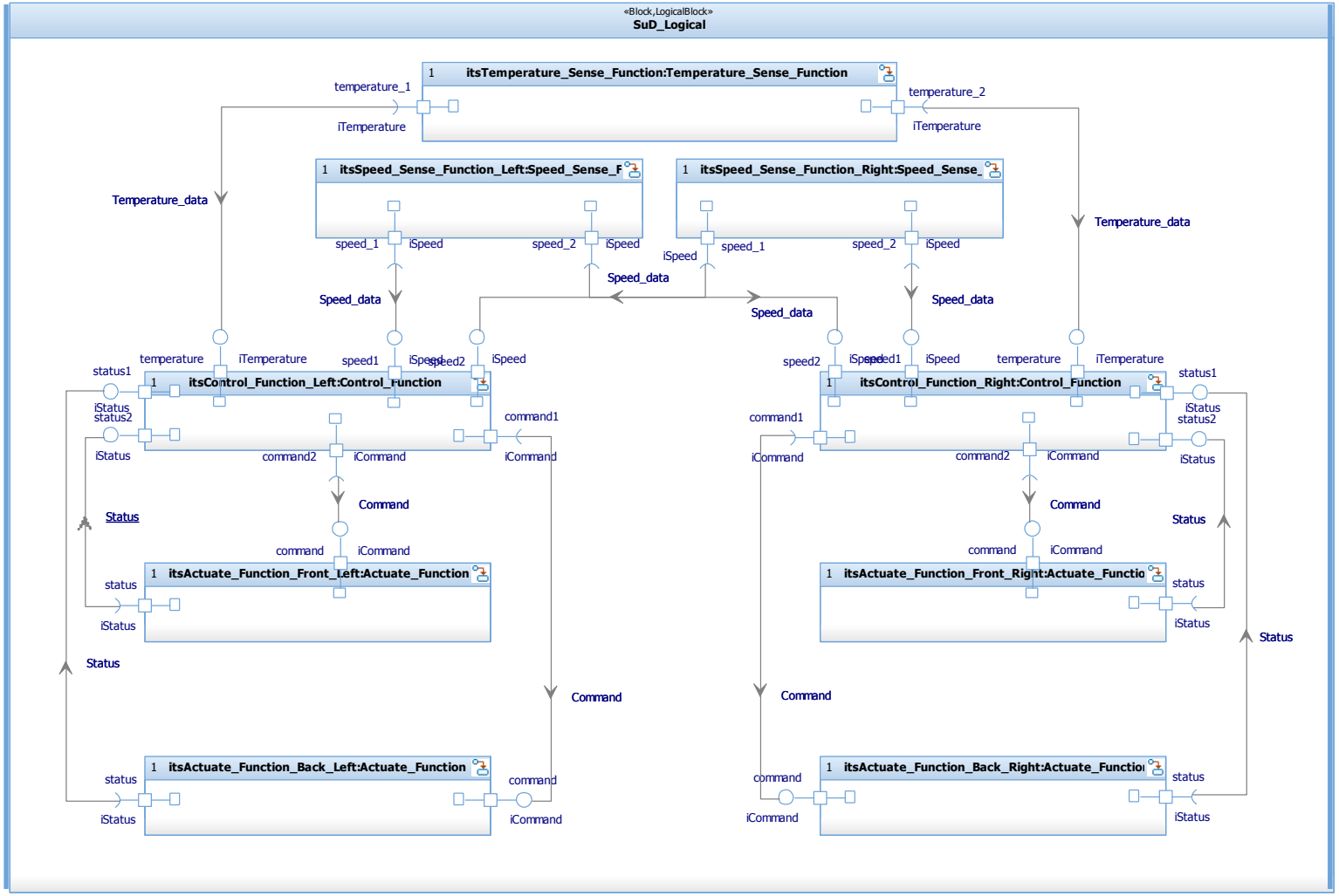
**Technical model**
Executable
6 technical block classes
18 technical block instances
36 technical links

**ATG**
30 Test cases
95% model element coverage (108/113)
41% MC/DC coverage (57/136)
99% statement coverage (332/335)

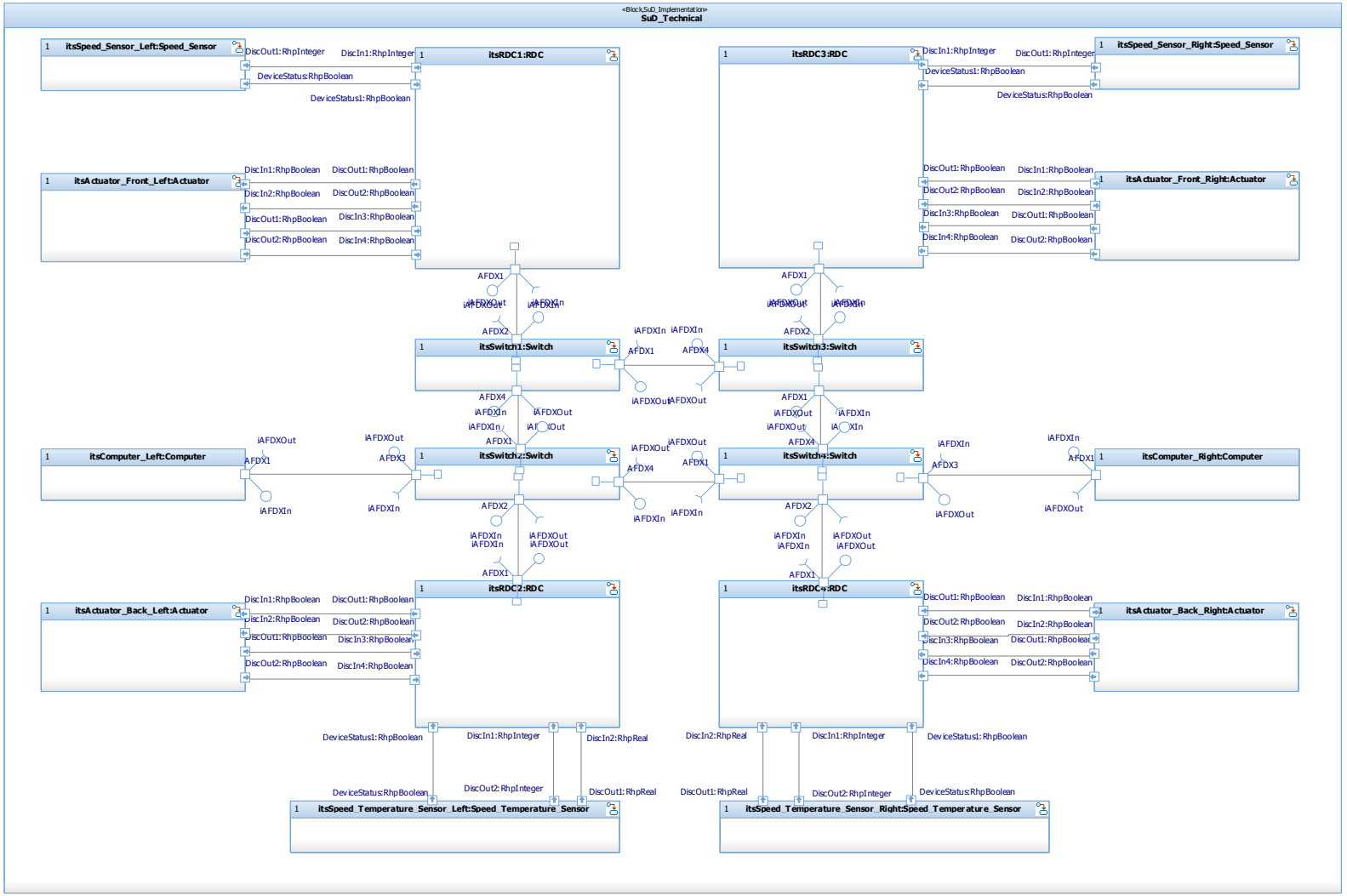
# Functional model



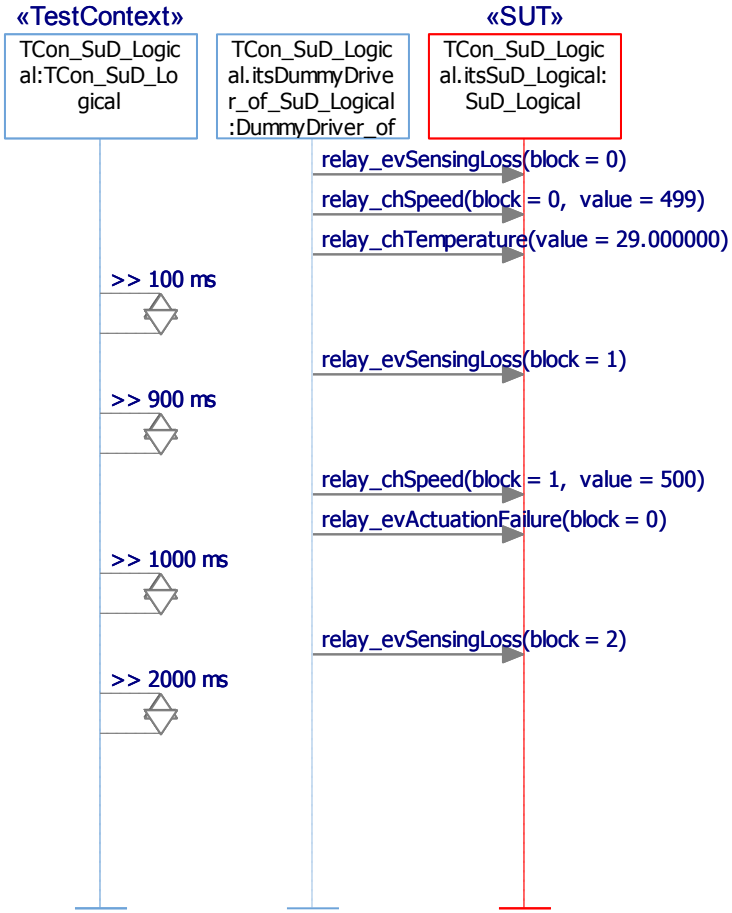
# Logical model



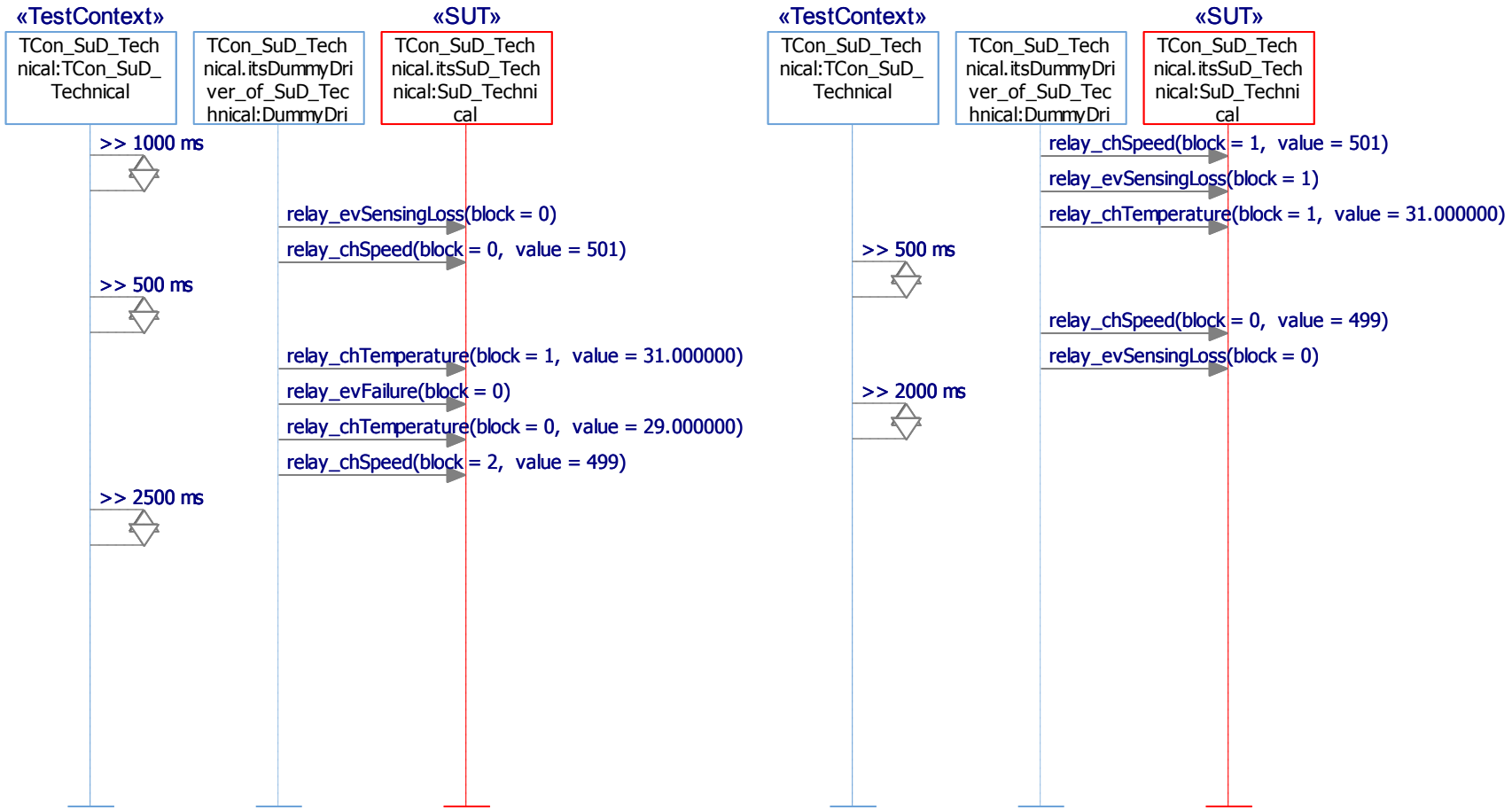
# Technical model



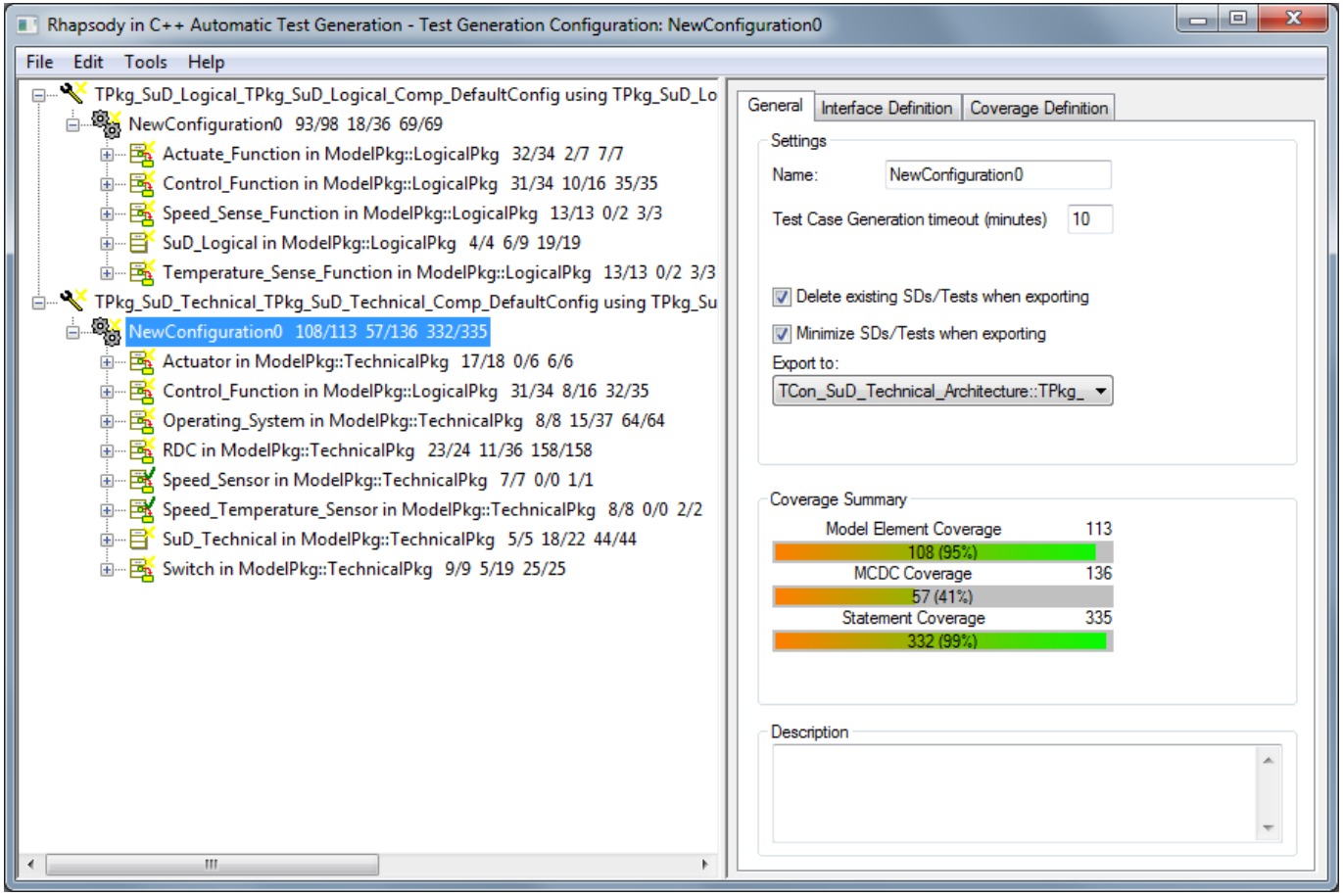
# ATG Test scenario examples – logical model



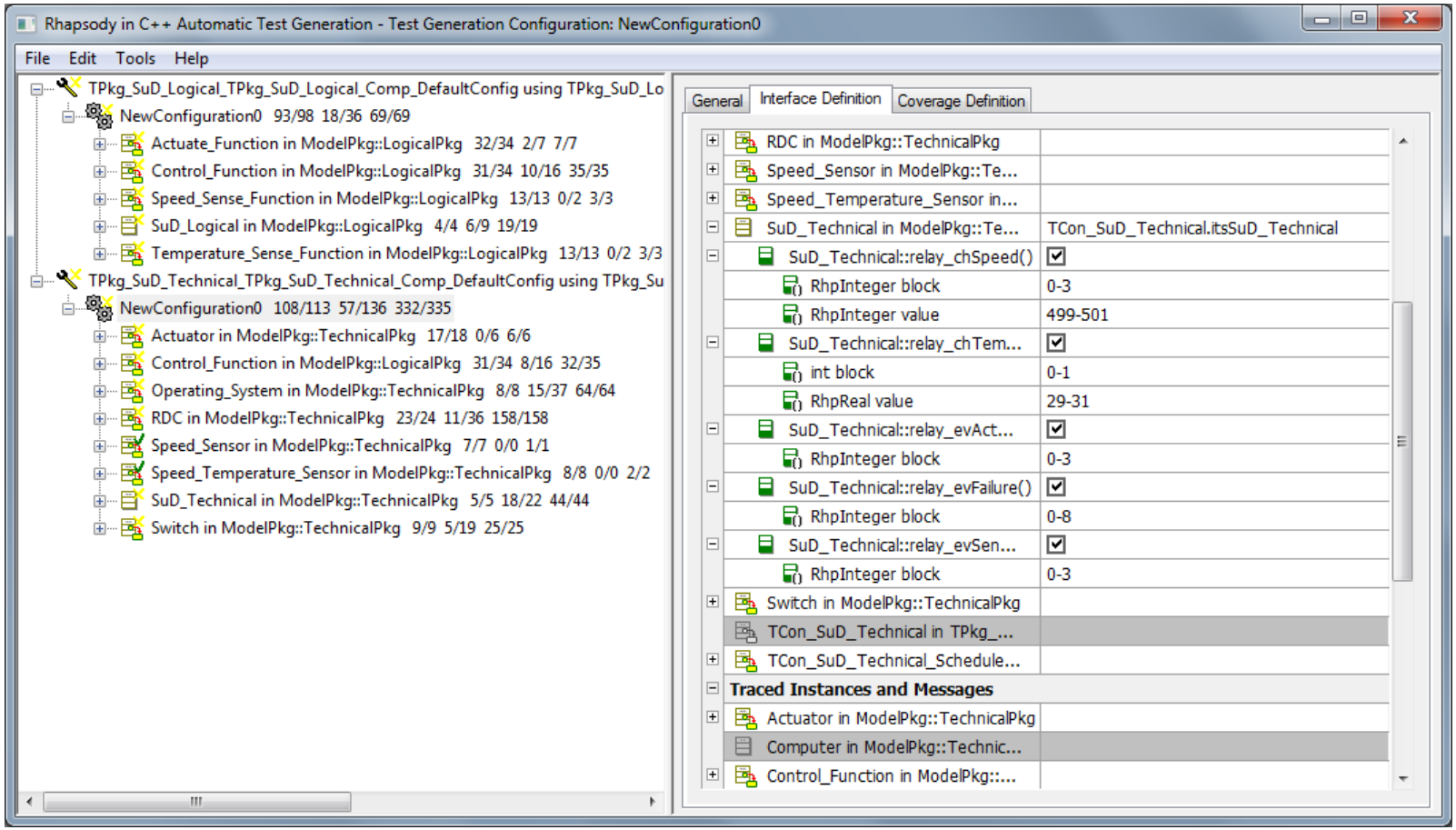

# ATG Test scenario examples – technical model



# ATG screenshot - coverage



# ATG screenshot – interface definition



# Agenda

- EADS background and motivation
- Model-based Testing concept and artefacts
- Model-based Testing with Rational Rhapsody, Rhapsody ATG and Rhapsody TestConductor
- Example and Early Results
- Conclusion

# Conclusion

- ATG…
  - works on a SysML specification model
  - achieves a high coverage even for complex models
  - does not have confining modelling restrictions as model checking based test case generators do
  - Helps improving testing performance

- Model-based Testing…
  - Supports in doing a better impact analysis
  - Leads to higher test coverage, resulting in higher product quality
  - Improves test management
  - Reduces overall test efforts

# EADS and BTC-ES are partners in the European MBAT research project

MBAT - Combined Model-based static Analysis and dynamic Testing of Embedded Systems


MBAT will increase the competitiveness of European key players in transportation domain by *

- reducing V&V costs for embedded systems by at least **20 percent**
- shortening time-to-market by at least **20 percent**
- increasing the coverage of the embedded system under V&V by at least **30 percent**
- signicantly increasing the probablitity to uncover errors
- enabling higher quality embedded systems

*the ARTEMIS Joint Undertaking under grant agreement no 269335 (ARTEMIS project MBAT) and from the German BMBF.*

# Questions

# Acknowledgements and disclaimers

**Availability**: References in this presentation to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates.

The workshops, sessions and materials have been prepared by IBM or the session speakers and reflect their own views. They are provided for informational purposes only, and are neither intended to, nor shall have the effect of being, legal or other guidance or advice to any participant. While efforts were made to verify the completeness and accuracy of the information contained in this presentation, it is provided AS-IS without warranty of any kind, express or implied. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, this presentation or any other materials. Nothing contained in this presentation is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software.

All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics may vary by customer. Nothing contained in these materials is intended to, nor shall have the effect of, stating or implying that any activities undertaken by you will result in any specific sales, revenue growth or other results.

# Thank You



# Innovate2013

## The IBM Technical Summit