UML TESTING PROFILE 2
A LANGUAGE FOR TEST AUTOMATION

Marc-Florian Wendland
MESCONF 2017, München, 26. Juni 2017

Fraunhofer FOKUS

UML TESTING PROFILE 2

Goal of this talk

- Understand the scope of UTP 2
- Find its place in the ocean of testing standards
- Become aware of its capabilities regarding the creation of model-based test specifications
- Learn more about test actions and arbitration specification to build reusable test specifications

AGENDA

1. The UML Testing Profile @ a Glance

2. What can I do with UTP 2?
   - Test Action and Procedural Elements
   - Arbitration Specifications

3. What has not yet been said

AGENDA

1. The UML Testing Profile @ a Glance

2. What can I do with UTP 2?
   - Test Action and Procedural Elements
   - Arbitration Specifications

3. What has not yet been said

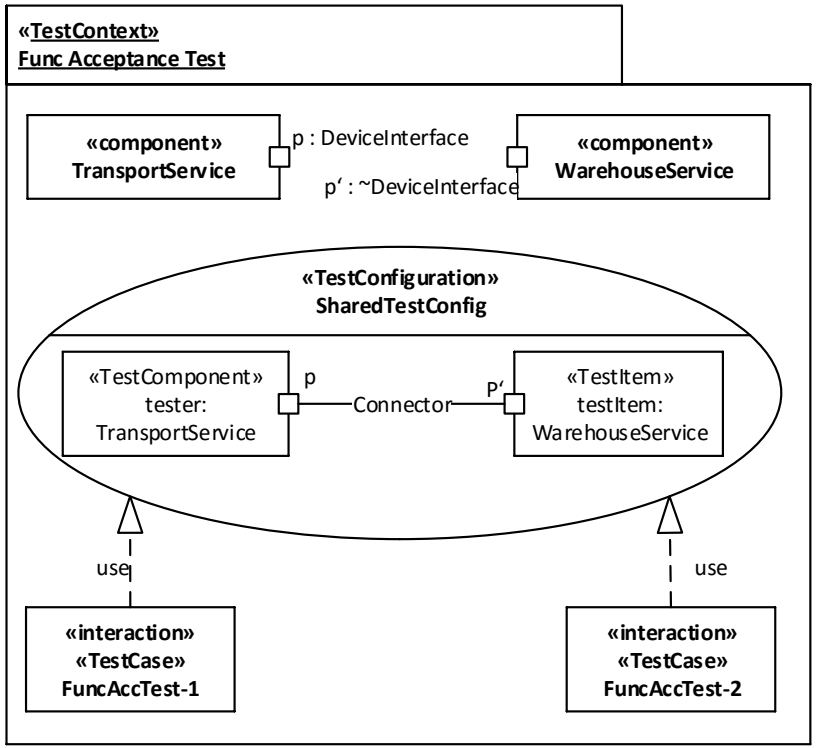
THE UML TESTING PROFILE @ A GLANCE

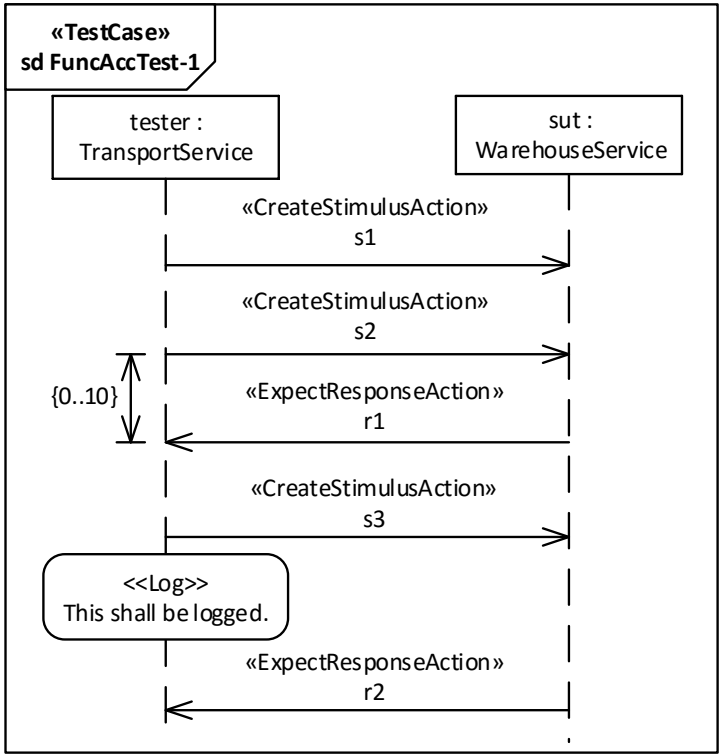
Understanding UTP

- A **test modeling language** based on UML
- Supports **(test) engineers** in carrying out (manual or automated) (dynamic) **test design activities**
- Specification of **test models and test logs**
- Facilitates (manual or automated) **test execution** and **evaluation**
- Simplifies **communication and understanding** among stakeholders
- Vendor- and methodology-independent (i.e., open) standard

UTP abides by the idea of model-driven engineering but for testing (test automation) purposes

THE UML TESTING PROFILE @ A GLANCE





THE UML TESTING PROFILE @ A GLANCE

Understanding UTP – Out of Scope

- Methodologies
- Modeling of test processes and/or higher-level test management concepts (such as test strategies, role concepts etc.)
- Static testing such as audits/reviews, static code analysis, etc.

THE UML TESTING PROFILE @ A GLANCE

From UTP 1.2 to UTP 2 – Reasons for a major revision

- UTP 1.0 was ahead of its time – in the meantime, things have changed
  - Incorporate experiences with model-based development and testing
  - Incorporate experiences of using UML and profiles
  - Incorporate new standards like ISO 29119 or ETSI ES 202 951 (MBT)
- Lack of/insufficiently elaborated concepts
  - Test design facility, test data values, test logging facility
- OMG policies to introduce new concepts in a minor revision are restrictive

UTP 2 is rather a technical modernization of the language instead of a reinvention of the wheel

THE UML TESTING PROFILE @ A GLANCE

UTP in the ocean of testing and domain-specific standards

ISO 29119-1

IEEE:ATML

ETSI TDL

ISTQB CMBT

ISO 26262

ETSI MBT

ETSI TTCN-3

ISO 29119-4

SoaML

Do-178C

TestIF

IEEE:829

ISO 29119-3

ISTQB TAE

UML

EN 50129

ISO 29119-5

ETSI TPLan

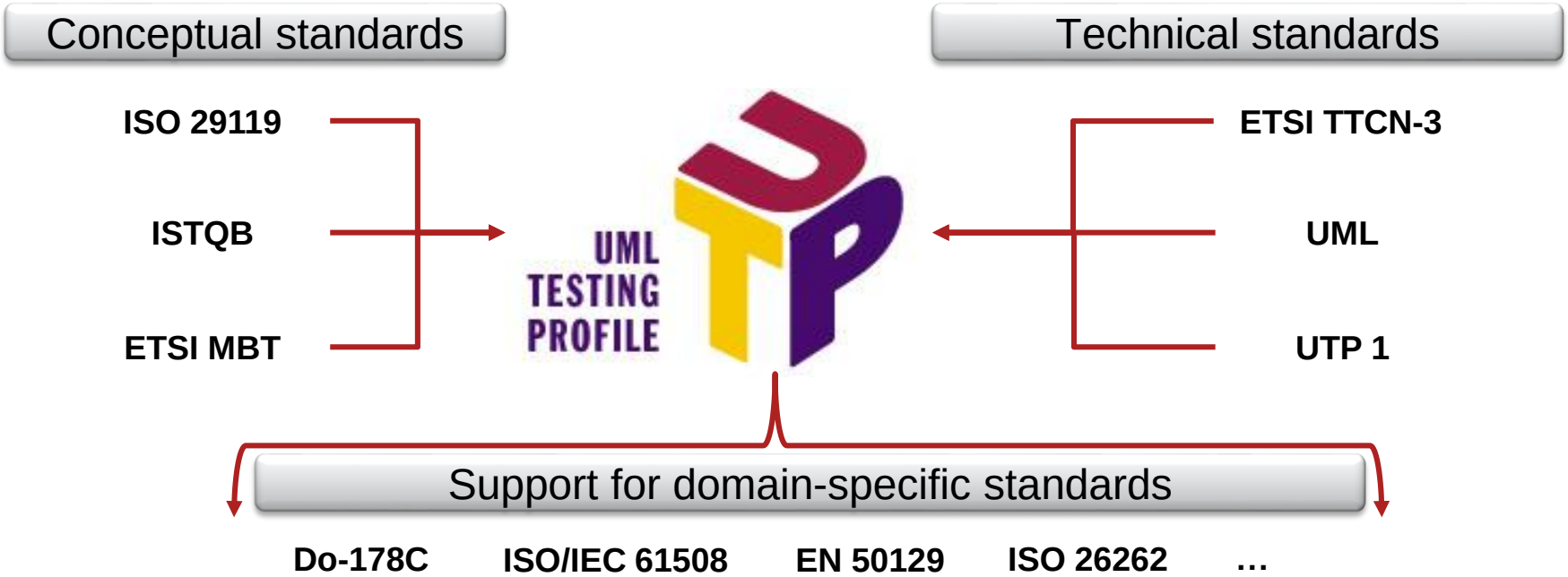
SysML

ISO/IEC 61508

UTP 1

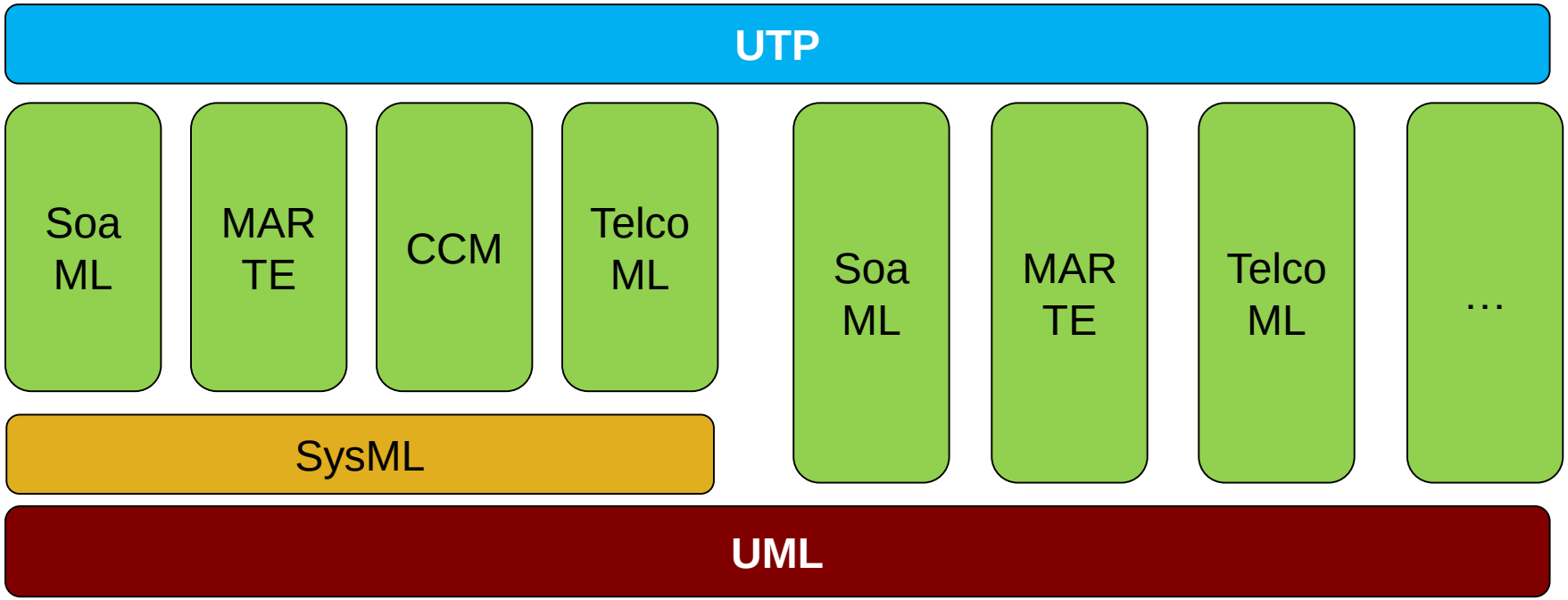
THE UML TESTING PROFILE @ A GLANCE

Influencing standards



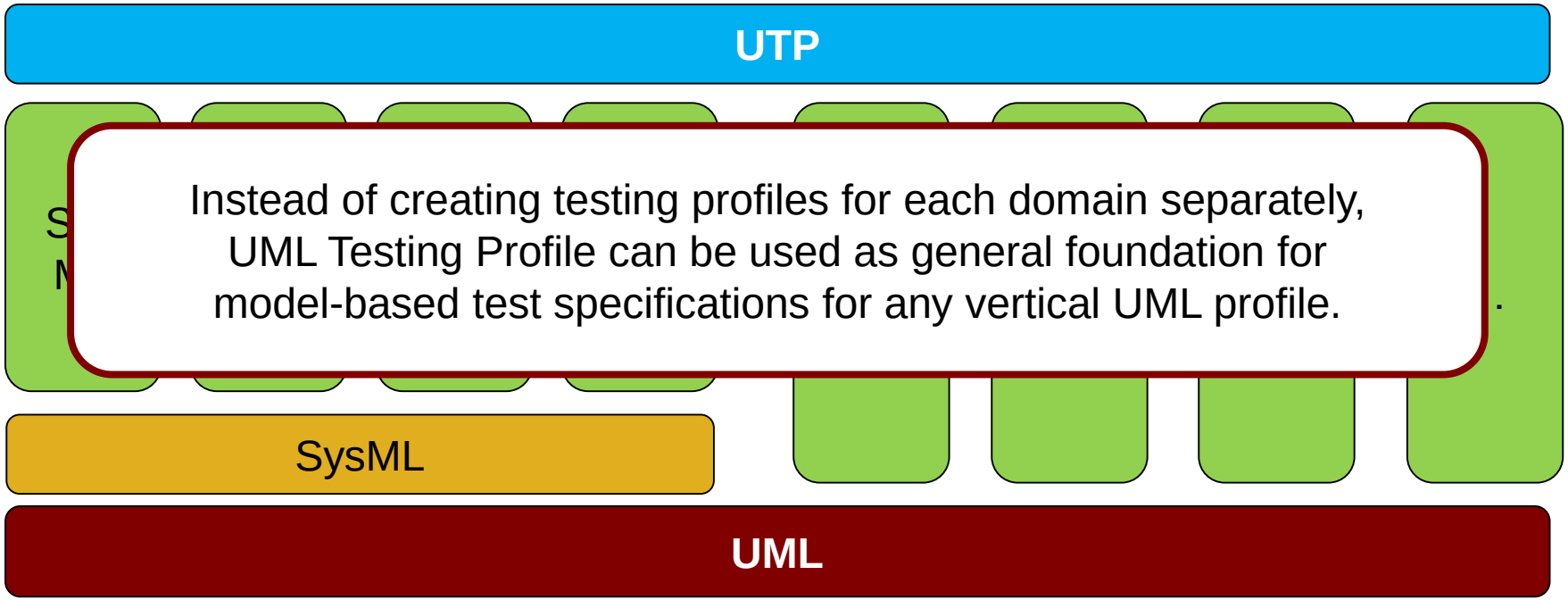
THE UML TESTING PROFILE @ A GLANCE

UTP in the UML ecosystem



THE UML TESTING PROFILE @ A GLANCE

UTP in the UML ecosystem



AGENDA

1. The UML Testing Profile @ a Glance

2. What can I do with UTP 2?

3. What has not yet been said

WHAT CAN I DO WITH UTP 2?

With UTP 2, I as a test engineer would want to

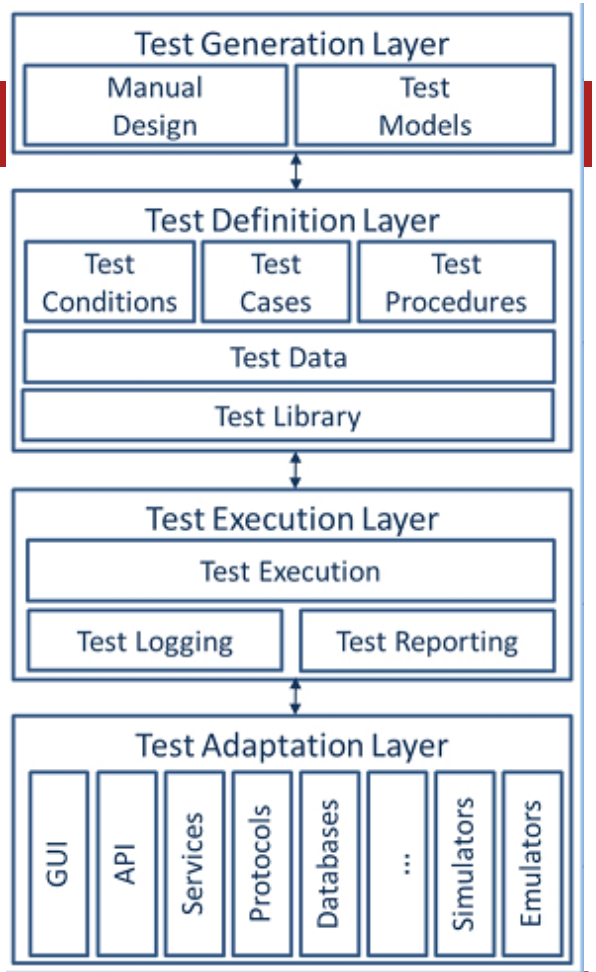
- Specify test (automation) architectures in a technology-independent manner…
- (Automatically) design test cases, test data and test schedules …
- Visualize test cases, test data and test schedules …
- Specify and reuse test environments …
- Capture test execution results for further test evaluation…
- Specify matching mechanisms for actual and expected responses…
- Specify arbitration rules for verdict arbitration…
- Generate executable test scripts and test results for a dedicated target platform…
- Produce test reports in a desired format…

… so that comprehensibility and communication among stakeholders are improved, important knowledge is preserved and the degree of automation is increased

WHAT CAN I DO WITH UTP 2?

Building test architectures with UTP 2

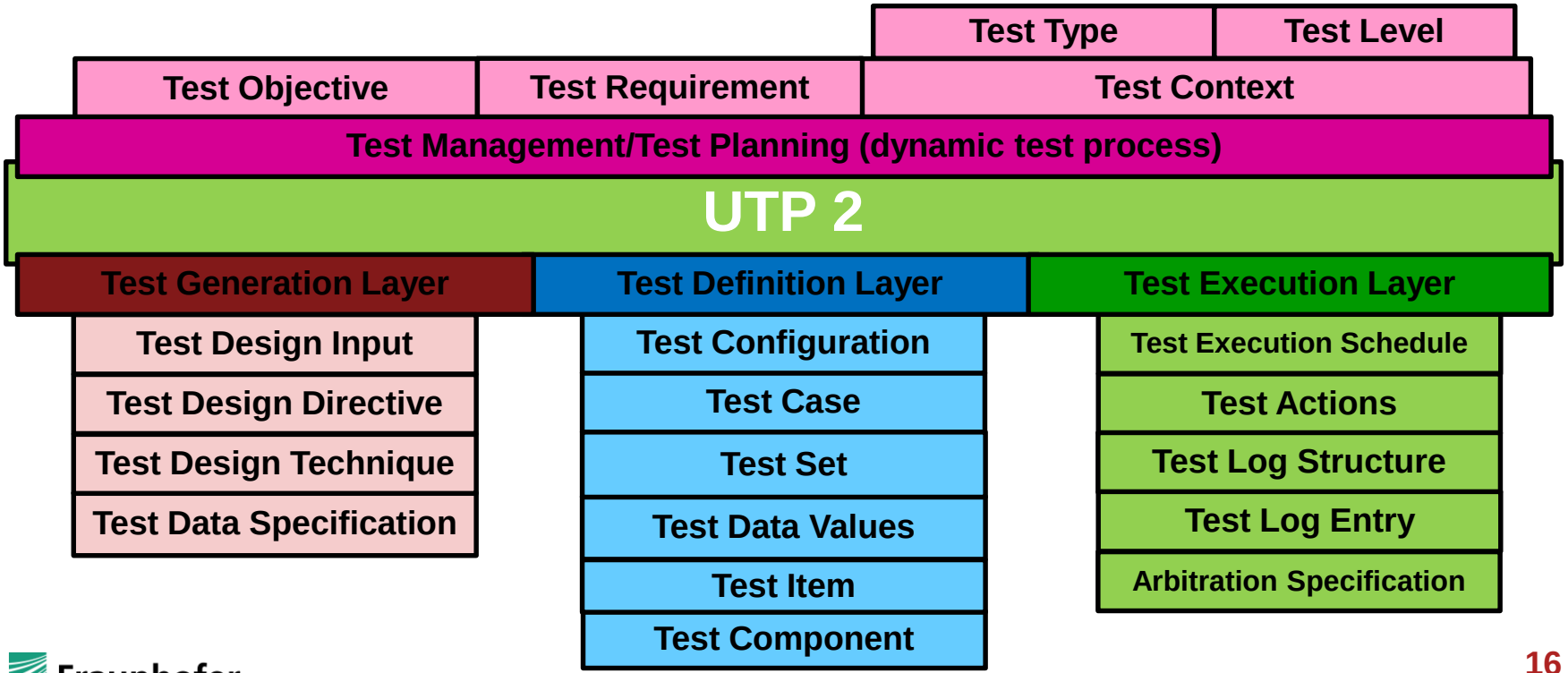
- Test Generation Layer
  Manual / automated design of test cases / test data
- Test Definition Layer
  Specification of test cases, test data, test procedures…
- Test Execution Layer
  Execution of test cases, logging of test execution, test evaluation & verdict arbitration
- Test Adaptation Layer
  Establishing communication with the system under test in order stimulate and observes it



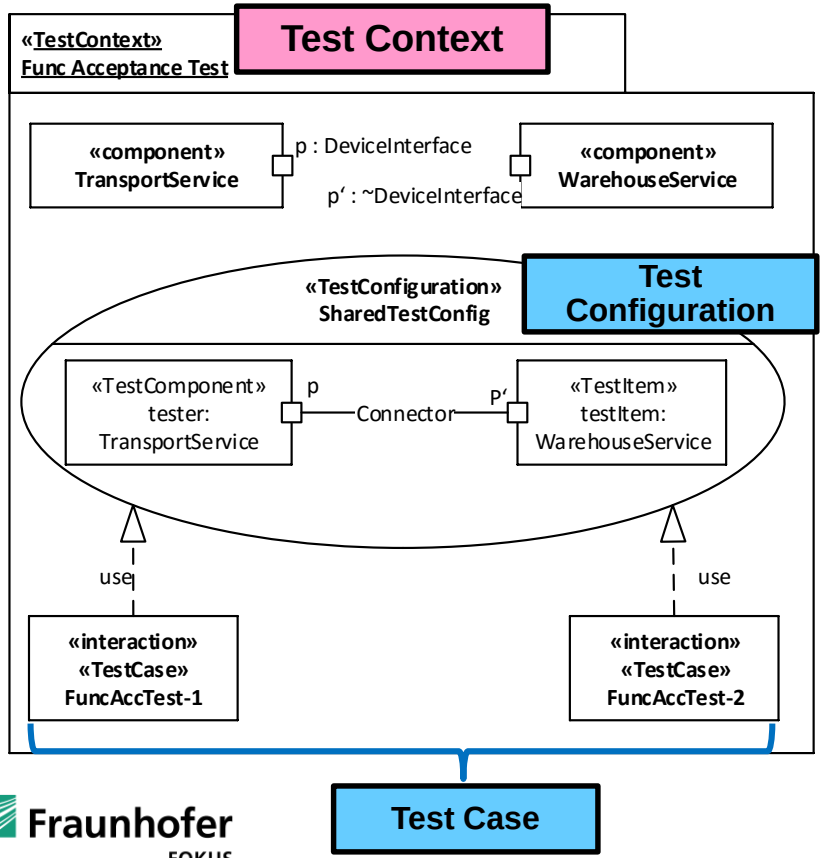
UTP 2 offers explicit concepts for the test generation, test definition and test execution layer

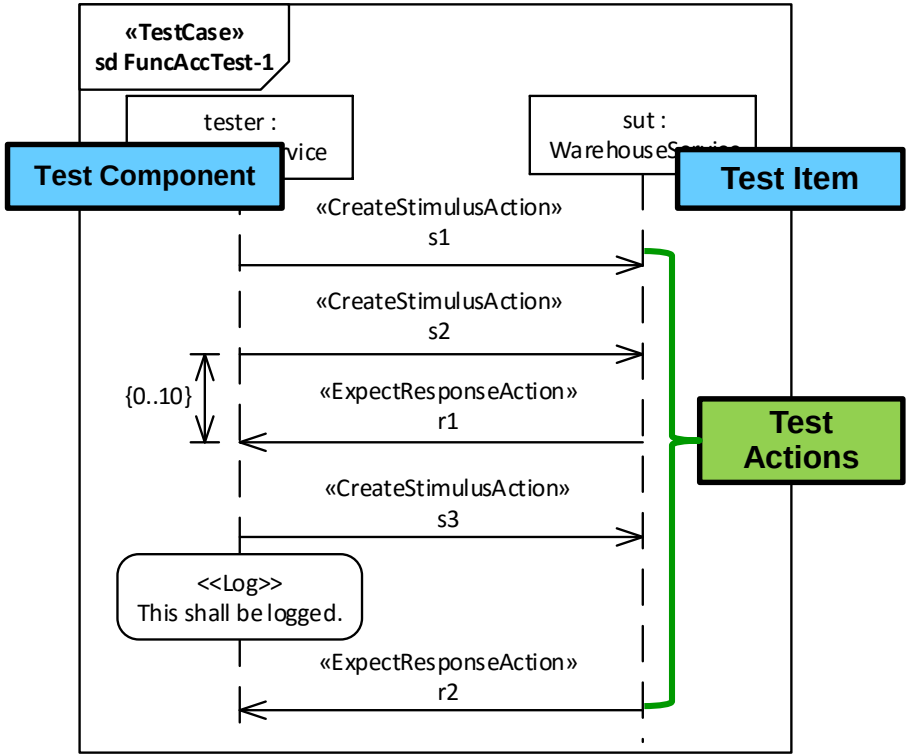
WHAT CAN I DO WITH UTP 2?

Conceptual overview



WHAT CAN I DO WITH UTP 2?





AGENDA

1. The UML Testing Profile @ a Glance

2. What can I do with UTP 2?

- **Test Action and Procedural Elements**
- Arbitration Specifications

3. What has not yet been said

WHAT CAN I DO WITH UTP 2?

Dedicated test actions in UTP 2

- Each UTP 2 **test case** consists of a **test procedure**
- A **test procedure** consists of **procedural elements**
- Procedural elements can be **atomic** (e.g., send a stimulus) or **non-atomic** (e.g. looping behavior)
- A special kind of atomic procedural actions are **test actions**

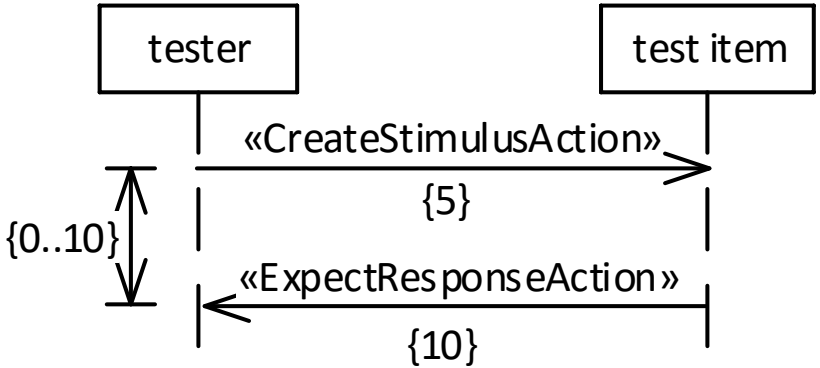
  > *„An atomic procedural element that is an instruction to the tester that needs to be executed as part of a test procedure within some time frame.” [UTP2]*

- „Tester“ stands for both automated (i.e., test components, test drivers, test stubs) and manual testers

WHAT CAN I DO WITH UTP 2?

Test action: Expect Response Action (with timing)

“A test action that instructs the tester to check the occurrence of one or more particular responses from the test item within a given time window.”



TTCN-3 equivalent

```
tester.send(Stimulus:{5});
timer.start(10);
alt{
[]tester.receive(Response:{10})
        {…}
[]timer.timeout
        {…}
}
```

WHAT CAN I DO WITH UTP 2?

Test actions (and procedural elements)

- The following test action (special kind of procedural elements) are provide in order to:
  - Stimuly the test item → **Create Stimulus Action**
  - Observe an expected response → **Expect Response Action**
  - Check some internal properties of the test item → **Check Property Action**
  - Submit a verdict to the arbitration specification → **Suggest Verdict Action**
  - Write something into the test log → **Create Log Entry Action**
- There are more procedural elements (e.g., loops, parallel, alternatives, procedure invocations etc.) that are used for building test procedures

UTP procedural elements are applicable to Interactions, State Machines and Activities

AGENDA

1. The UML Testing Profile @ a Glance

2. What can I do with UTP 2?

- Test Action and Procedural Elements
- **Arbitration Specifications**

3. What has not yet been said

WHAT CAN I DO WITH UTP 2?

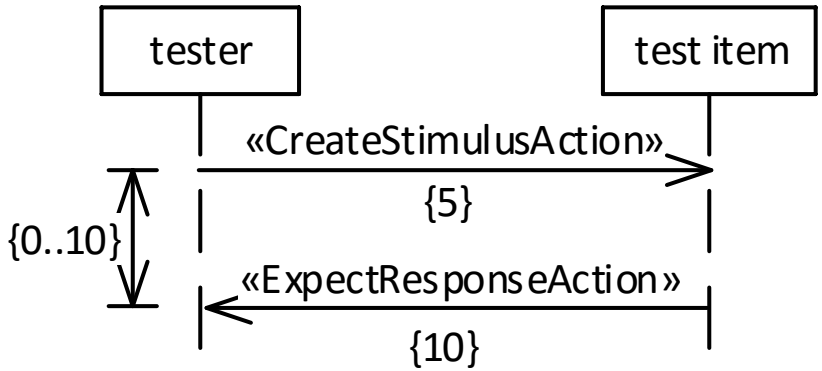
Introduction: Arbitration Specifications

- An arbitration specification (AS) is a **specification** of the rules that reasoning about verdicts
- Arbitration specifications can be defined for **test sets**, **test cases** and **test actions**
  - If no arbitration specification is set, a **default** one is set
  - Arbitration specifications can be **replaced** for certain test actions, test cases and test sets
- Arbitration specifications help keeping the test cases and test procedures **agnostic** of any verdict-related information

WHAT CAN I DO WITH UTP 2?

Arbitration Specifications

– What is the semantics of the following test case with respect to its verdict?



Semantics is given by the applied **arbitration specification**.

Questions

- What verdict shall be set, if the expected response is received?
- What verdict shall be set, if another response is received before?
- What is the initial verdict of the test case?
- Is there a precedence rule of verdicts similar to TTCN-3 (i.e., none < pass < inconclusive < fail < error)?

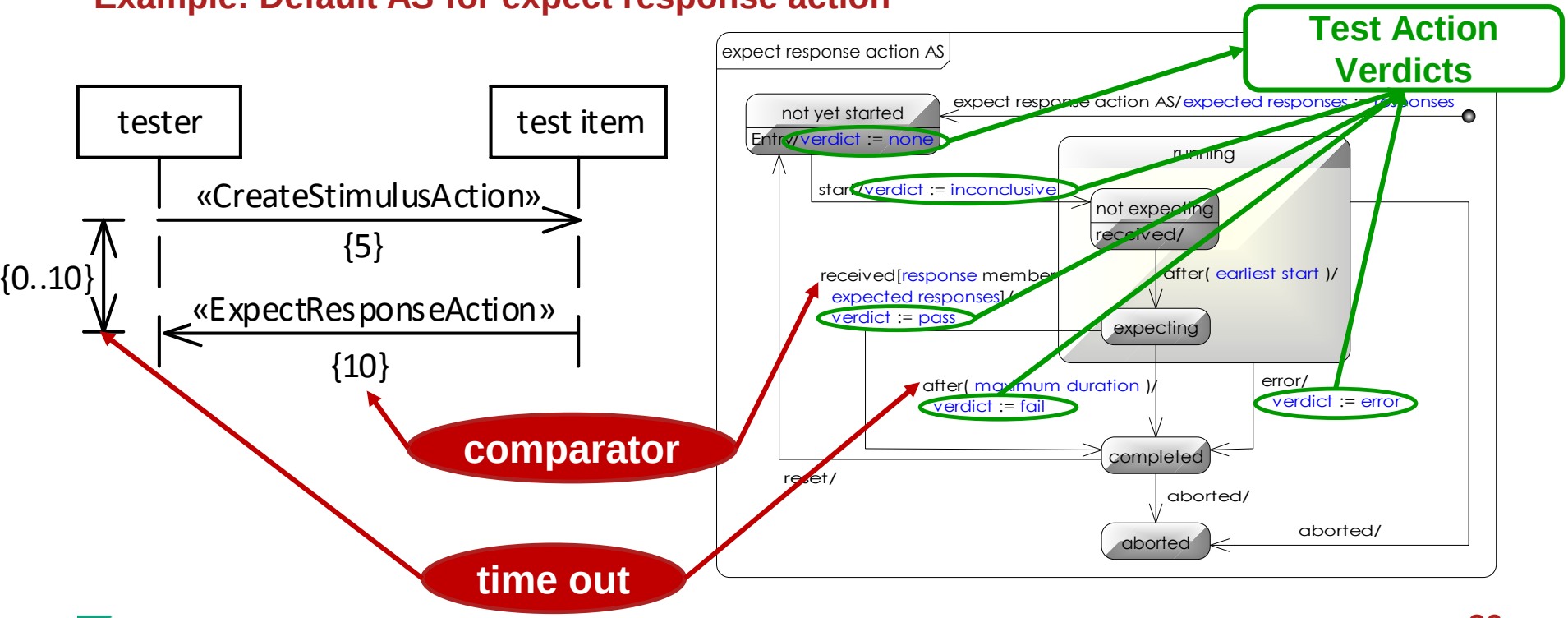
WHAT CAN I DO WITH UTP 2?

Interplay of arbitration specifications on different levels

- Test action, test case and test set represent ascending composition levels
- Each arbitration specification provides a verdict
  - Test action AS → test action verdict **(are summarized by)**
  - Test case AS → test case verdict **(are summarized by)**
  - Test set AS → test set verdict
- Test action verdicts result from the evaluation of atomic test actions
- Test action verdicts are conveyed to the test case AS that is responsible to calculate the test case verdict
- Test case verdicts are conveyed to the test set AS (if set) that is responsible to calculate the test set verdict

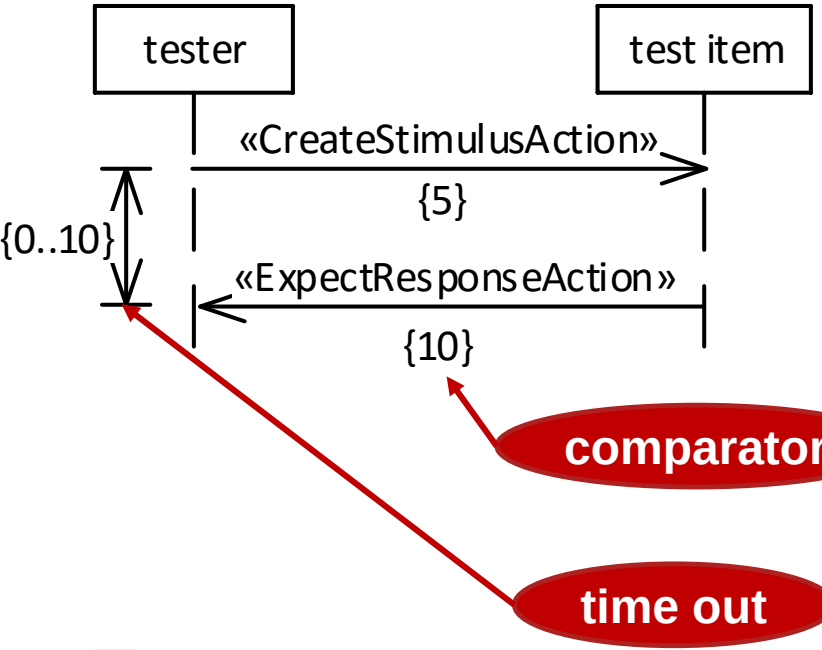
WHAT CAN I DO WITH UTP 2?

Example: Default AS for expect response action



WHAT CAN I DO WITH UTP 2?

Example: Default AS for expect response action



TTCN-3 equivalent

```
tester.send(Stimulus:{5});
timer.start(10);
alt{
     []tester.receive(Response:{10})
        {setverdict(pass);}
     []tester.receive
        {repeat;}
     []timer.timeout
        {setverdict(fail);}
}
```

WHAT CAN I DO WITH UTP 2?

Summary: Arbitration Specifications

- Arbitration specifications have been **newly introduced** (in contrast to Arbiter) by UTP 2
- Help keeping the test case **clean** of verdict/arbitration-related logic
- UTP 2 **provides default** arbitration specification for test actions, test case and test sets
  - If no explicit arbitration is given, the default one will be taken by definition
  - Easy replacement of arbitration specifications through tagged values
- Arbitration specifications **do not have** to be expressed in a formal, yet executable way → UTP 2 provides a formal semantics for its default AS

AGENDA

1. The UML Testing Profile @ a Glance

2. What can I do with UTP 2?

3. Miscellaneous

MISCELLANEOUS

Many more concepts offered

- **Test design facility**: specify the test design techniques and there coverage goals to guide the test design process
- **Data specifications**: specify, modify and reuse data partitions, data specifications, data pools; optimized to describe and handle large sets of data for test data generation, test data selection and test case execution
- **ValueSpecification Extensions**: extensions to the UML ValueSpecifications for regual expression, range values, enumerated values, collections, complemented values
- **Test logging facility**: concepts to formalise, represents and/or visualize test execution traces; enables for post-execution comparison, test results harmonisation and integration etc.

MISCELLANEOUS

Relationship of UTP to SysML

- UTP and SysML are related in two ways
  - SysML re-specified/re-implemented the (UTP 1.x) concepts *test case* and *verdict*
  - UTP 2 re-specified/re-implemented the SysML (1.x) concept *verifies*

  →Technical compatibility ensures that both profiles could be applied simutaneously
- UTP 2 changed the concept test case and verdict
  - compatbility with UTP 2 is not given anymore
  - SysML 2 WG is interested in a liason with UTP to ensure compatibility and avoid unnecessary redundancy

MISCELLANEOUS

Timeline and Roadmap

- **June 2017**: Successful submission of revised submission; adoption by OMG as beta standard; charter of finalization task force (FTF)

- **June 2018**: Submission of FTF; release of UTP 2.0 by OMG expected; charter of UTP 2.1 revision task force

- **June 2019:** Release of UTP 2.1 expected

MISCELLANEOUS

Summary

- UTP 2 is a graphical modelling language based on ÚML
- A graphical modelling language to support test design activities
- Terminology in particular influenced by ISO 29119 and ISTQB
- Concepts provided to describe (parts of) test automation architectures
- Just a specification language! Transformations not part of UTP 2

WE ARE OVER AND DONE…

Thank you very much for your attention.
Questions?

CONTACT

Fraunhofer FOKUS
Kaiserin-Augusta-Allee 31
10589 Berlin, Germany
www.fokus.fraunhofer.de

Marc-Florian Wendland
Senior Researcher, SQC
marc-florian.wendland@fokus.fraunhofer.de
Phone +49 (0)30 3463-7395

REFERENCES

[1] Object Management Group (OMG), Unified Modeling Language (UML), http://ww.omg.org/spec/UML

[2] International Software Testing Qualifications Board (ISTQB), GISTQB Glossary V3.01, http://www.istqb.org/downloads/finish/20/206.html

[3] Internation Standardisation Organisaton (ISO), ISO 29119, Software and systems engineering - Software testing

[4] Object Management Group, UML Testing Profile (UTP), http://ww.omg.org/spec/UTP

[5] European Telecommunications Standardisation Institute (ETSI): ES 202 951: Requirements for Modeling Notations. ETSI Standard, Methods for Testing and Specification (MTS); Model-Based Testing (MBT). V1.1.1 (2011-07)

[6] Wendland, Marc-Florian, Abstractions on Test Design Techniques, in Proc.: Federated Conference on Computer Science and Information Systems (FedCSIS), 2014

[7] Object Management Group (OMG), Request for Proposal for UML Testing Profile 2.0, Doc#: ad/2013-11-02

[8] Wendland, Marc-Florian, Towards UTP 2, presented at 2nd User Conference on Advanced Automated Testing (UCAAT), 2014