



Bringing TDL to Users: A Hands-on Tutorial

Philip Makedonski, Gusztav Adamis, Martti Käärik,
Andreas Ulrich, Marc-Florian Wendland, Anthony Wiles

Overview

What is TDL?

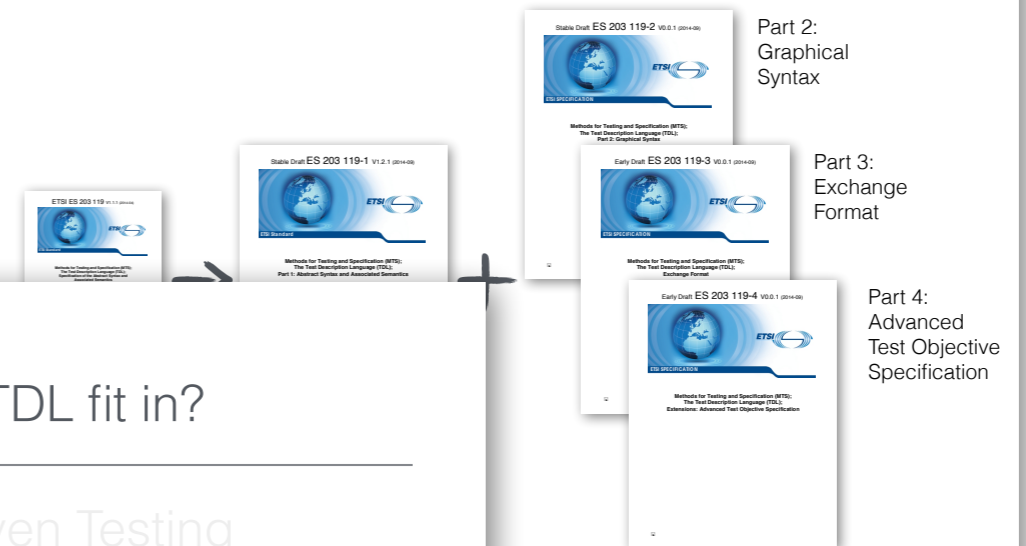
- Test Description Language
 - Design, documentation, and representation of formal test descriptions
 - Scenario-based approach
- Standardised at ETSI by TC MTS
- Initial work within STF 454
- Current efforts within STF 476



© ETSI 2014. All rights reserved

What is new in TDL?

Work in Progress



rights reserved

25

Where does TDL fit in?

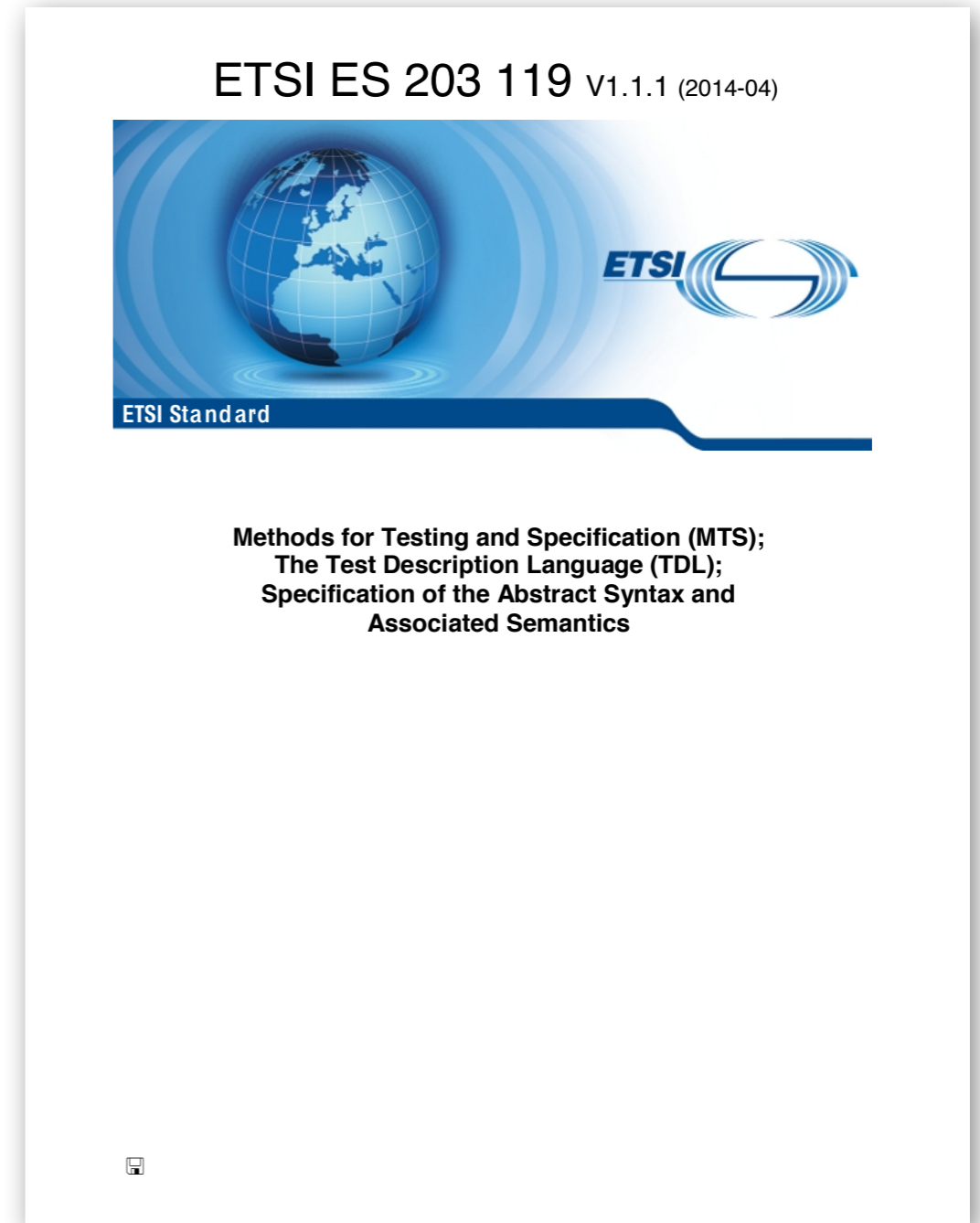


© ETSI 2014. All rights reserved

33

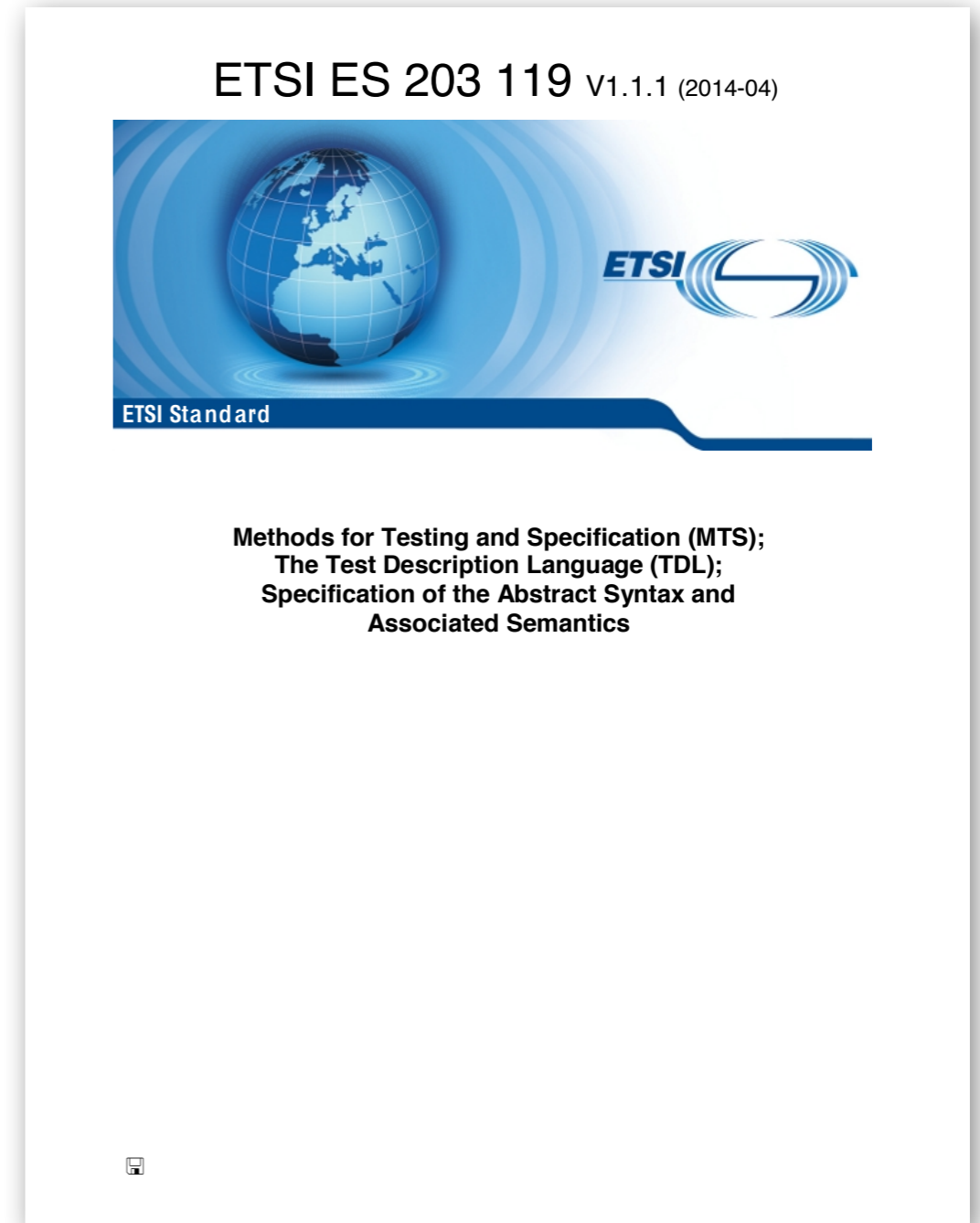
What is TDL?

- Test Description Language
 - Design, documentation, and representation of formal test descriptions
 - Scenario-based approach
- Standardised at ETSI by TC MTS
- Initial work within STF 454
- Current efforts within STF 476



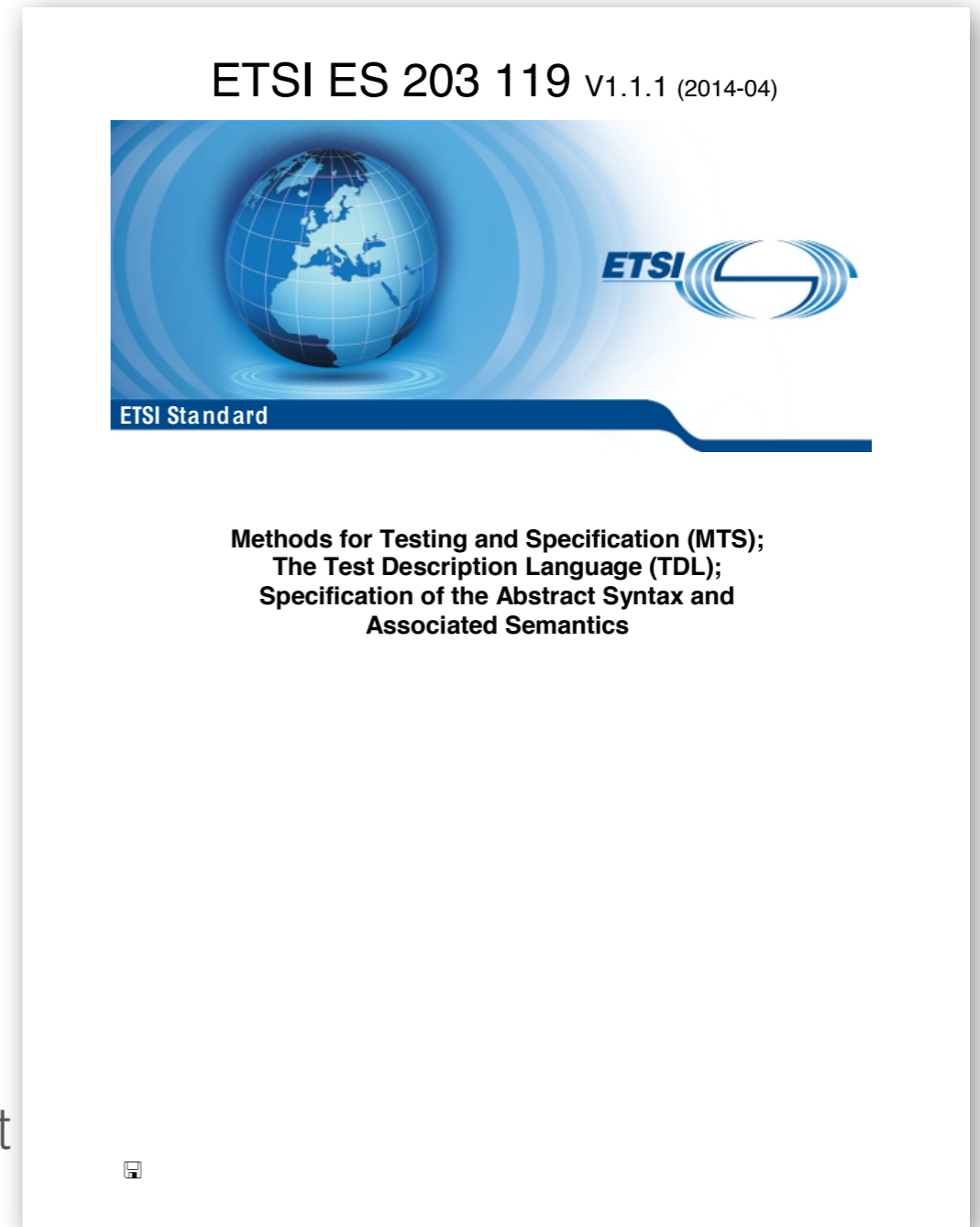
What is TDL?

- Design, documentation, representation?
 - ease development and review
 - improve productivity and quality
 - both industry and standardisation
 - reduce implementation details



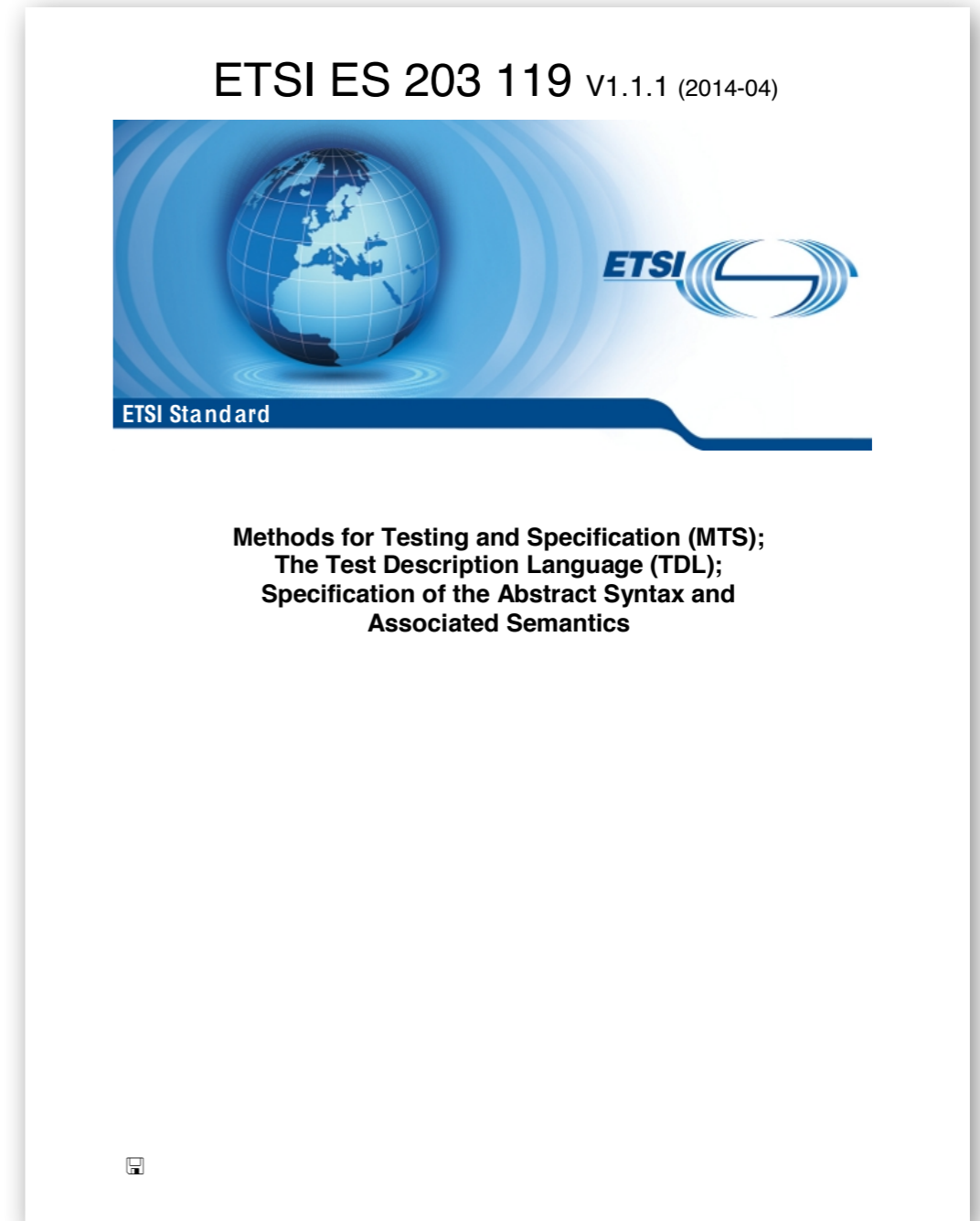
What is TDL?

- Scenario-based?
 - describe interactions with a system
 - attach test objectives to scenarios
 - derive and automate tests
- Reactive, distributed, real-time
 - common black-box testing concepts
 - domain adaptation, agile development



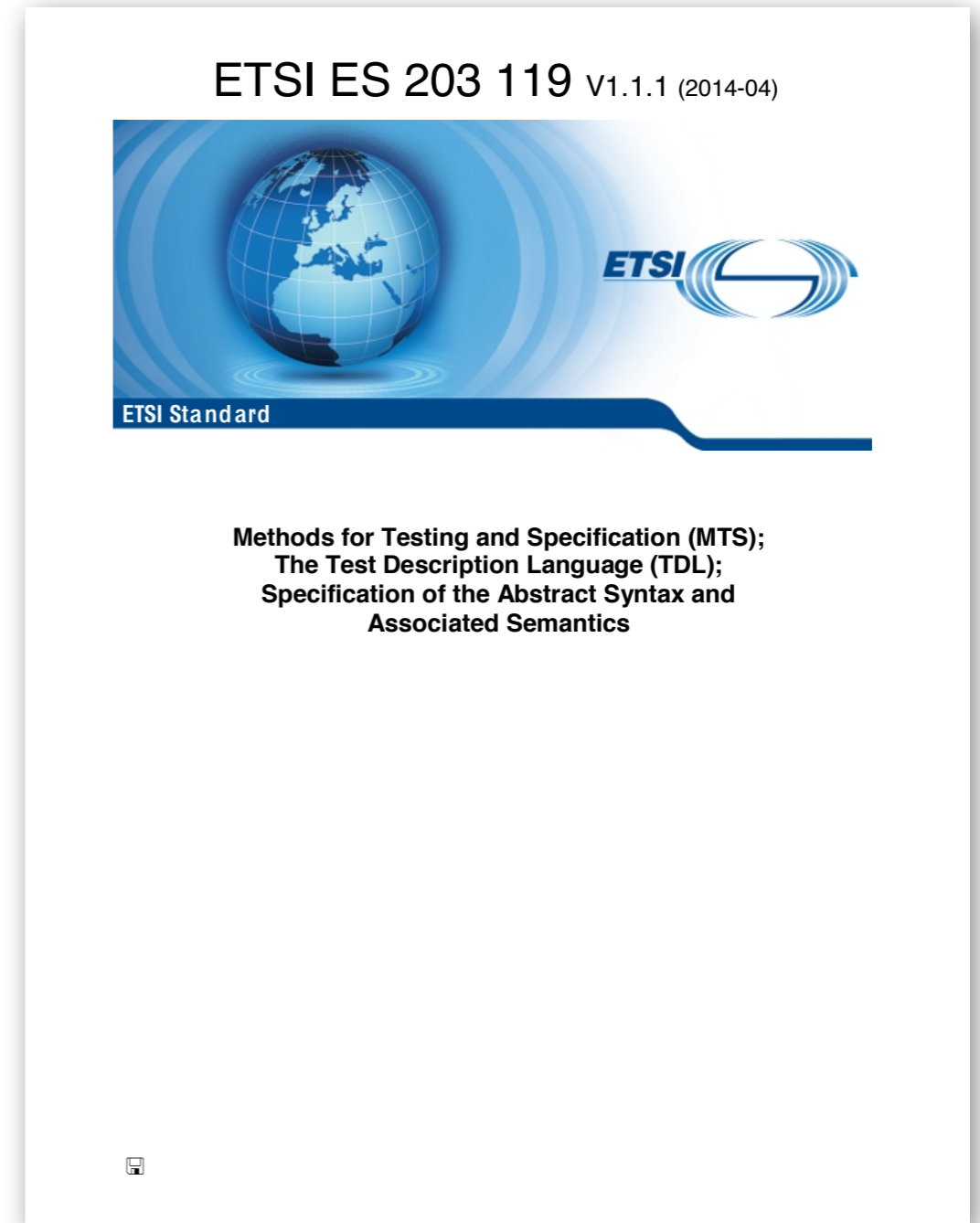
What is TDL?

- Standardised?
 - clear semantics
 - interoperability of tools and test specifications
 - updated with user needs
 - maintenance commitment



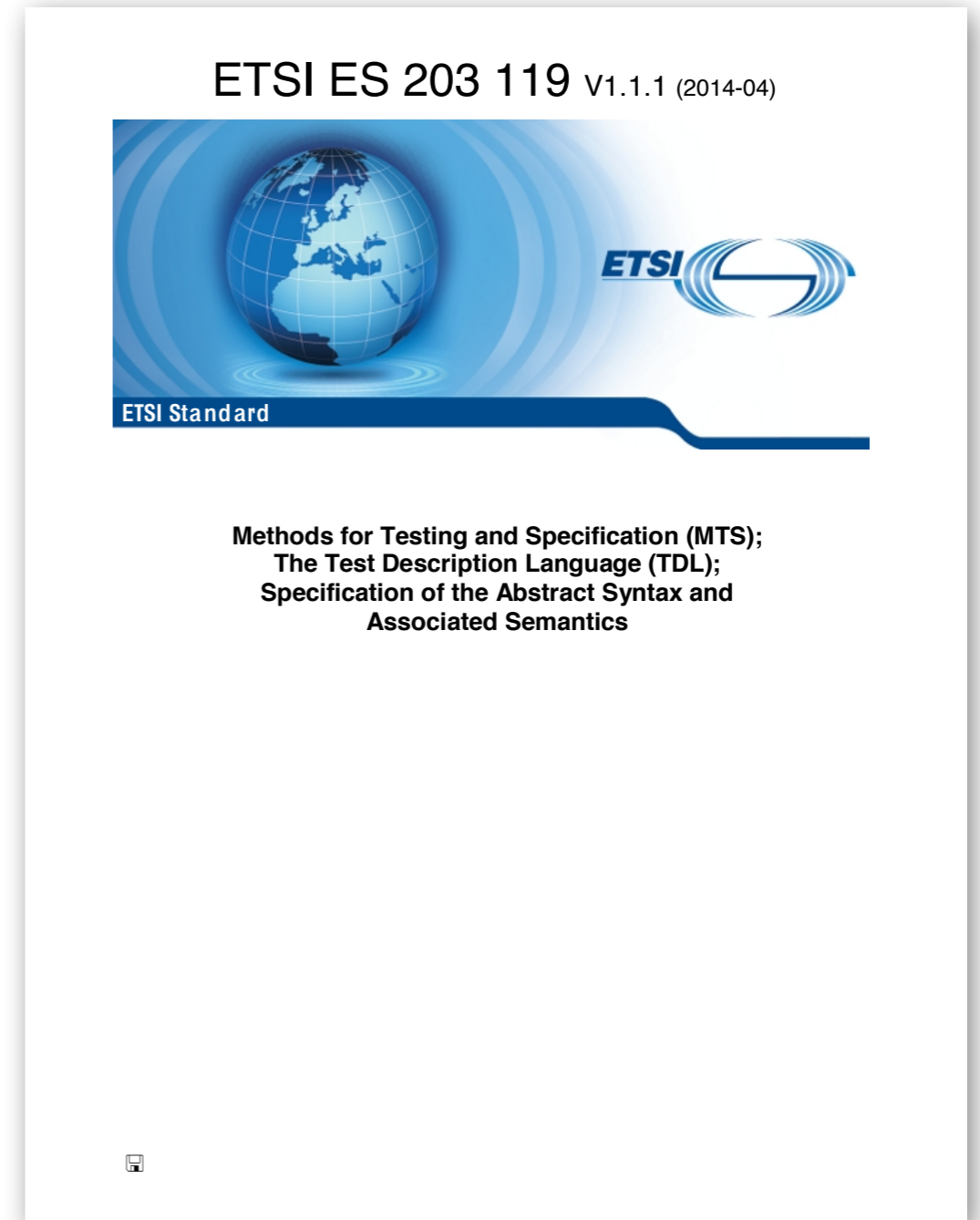
What is TDL?

- Contributions from:
 - Siemens AG, Ericsson Hungary
 - Fraunhofer FOKUS, ETSI CTI
 - University of Göttingen
 - OU Elvior, Cinderella ApS
- Guidance:
 - Steering Group, TC MTS



What is TDL?

- TDL main ingredients
 - Test data
 - Test configuration
 - Test behaviour
 - Test objectives



What is TDL?

- TDL main ingredients
 - Test data
 - Test configuration
 - Test behaviour
 - Test objectives

54 ETSI ES 203 119 V1.1.1 (2014-04)

Annex B (informative): Examples of a TDL Concrete Syntax

B.1 Introduction

The applicability of the TDL meta-model that is described in the main part of the present document depends on the availability of TDL concrete syntaxes that implement the meta-model (abstract syntax). Such a TDL concrete syntax can then be used by end users to write TDL specifications. Though a concrete syntax will be based on the TDL meta-model, it can implement only parts of the meta-model if certain TDL features are not necessary to handle a user's needs.

This annex illustrates an example of a possible TDL concrete syntax in a textual format that supports all features of the TDL meta-model, called "TDLan". Three examples are outlined below - two examples translated from existing test descriptions taken from [i.2] and [i.3], as well as an example illustrating some of the TDL data parameterization and mapping concepts. The examples are accompanied by a complete reference description of the textual syntax of TDLan given in EBNF.

B.2 A 3GPP Conformance Example in Textual Syntax

This example describes one possible way to translate clause 7.1.3.1 from TS 136 523-1 [i.2] into the proposed TDL textual syntax, by mapping the concepts from the representation in the source document to the corresponding concepts in the TDL meta-model by means of the proposed textual syntax. The example has been enriched with additional information, such as explicit data definitions and test configuration details for completeness where applicable.

```
//Translated from [i.2], Section 7.1.3.1
TDLan Specification Layer_2_DL_SCH Data Transfer {
//Procedures carried out by a component of a test configuration
//or an actor during test execution
Action precondition : "Pre-test Conditions:
RRC Connection Reconfiguration" ;
Action preamble : "Preamble:
The generic procedure to get UE in test state Loopback
Activated (State 4) according to TS 36.508 clause 4.5
is executed, with all the parameters as specified in the
procedure except that the RLC SDU size is set to return no
data in uplink.
(reference corresponding behaviour once implemented) ;

//User-defined verdicts
//Alternatively the predefined verdicts may be used as well
Verdict PASS ;
Verdict FAIL ;

//User-defined annotation types
Annotation TITLE ; //Test description title
Annotation STEP ; //Step identifiers in source documents
Annotation PROCEDURE ; //Informal textual description of a test step
Annotation PRECONDITION ; //Identify pre-condition behaviour
Annotation PREAMBLE ; //Identify preamble behaviour.

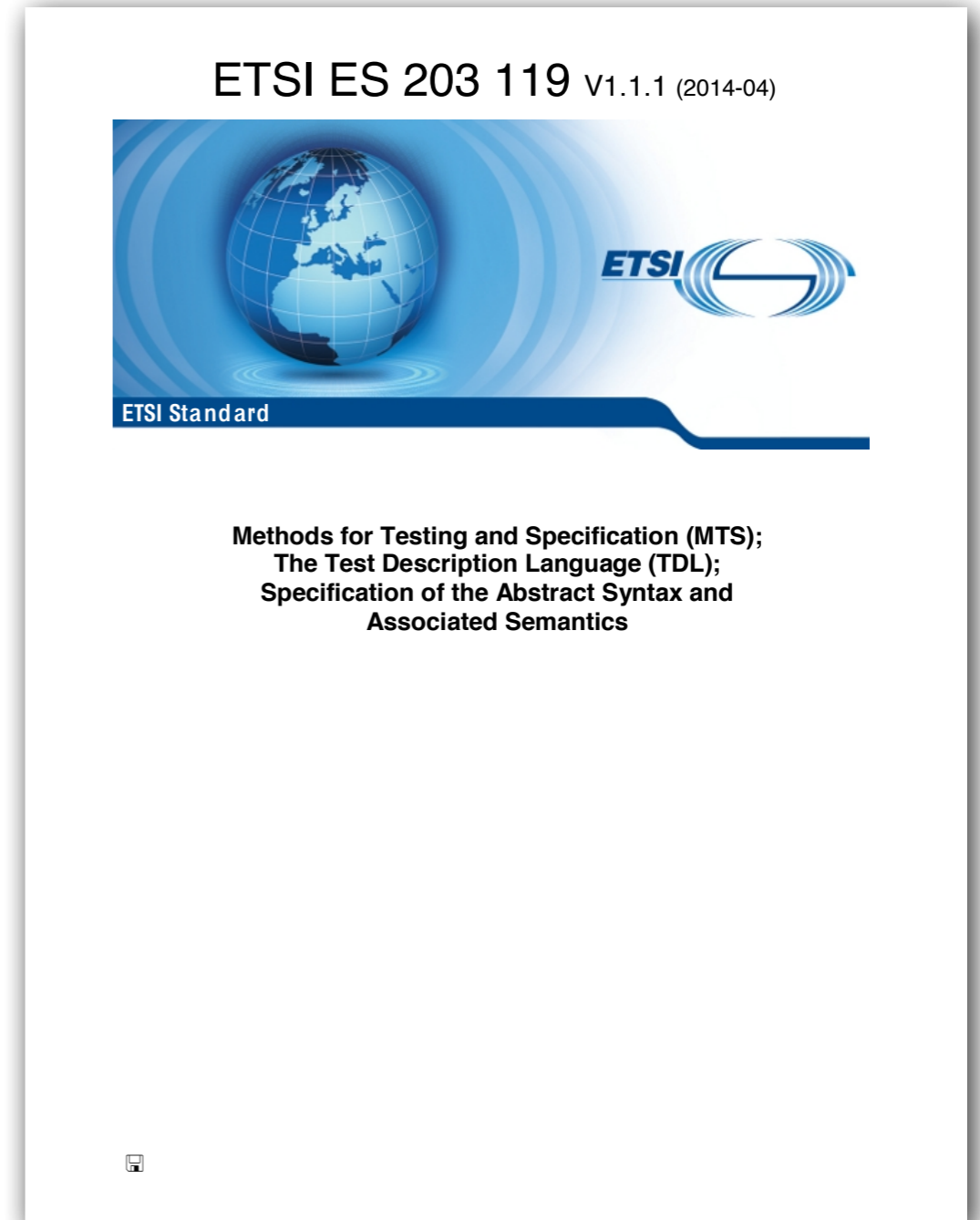
//User-defined time units
Time Unit seconds;

//Test objectives (copied verbatim from source document)
Test Objective TP1 {
from : "36523-1-a20_s07_01.doc:7.1.3.1.1 (1)" ;
description : "with { UE in E-UTRA RRC_CONNECTED state }
ensure that {
when { UE receives downlink assignment on the PDCCCH
for the UE's C-RNTI and receives data in the
associated subframe and UE performs HARQ
operation }
then { UE sends a HARQ feedback on the HARQ
process }
}" ;
}
}
```

ETSI

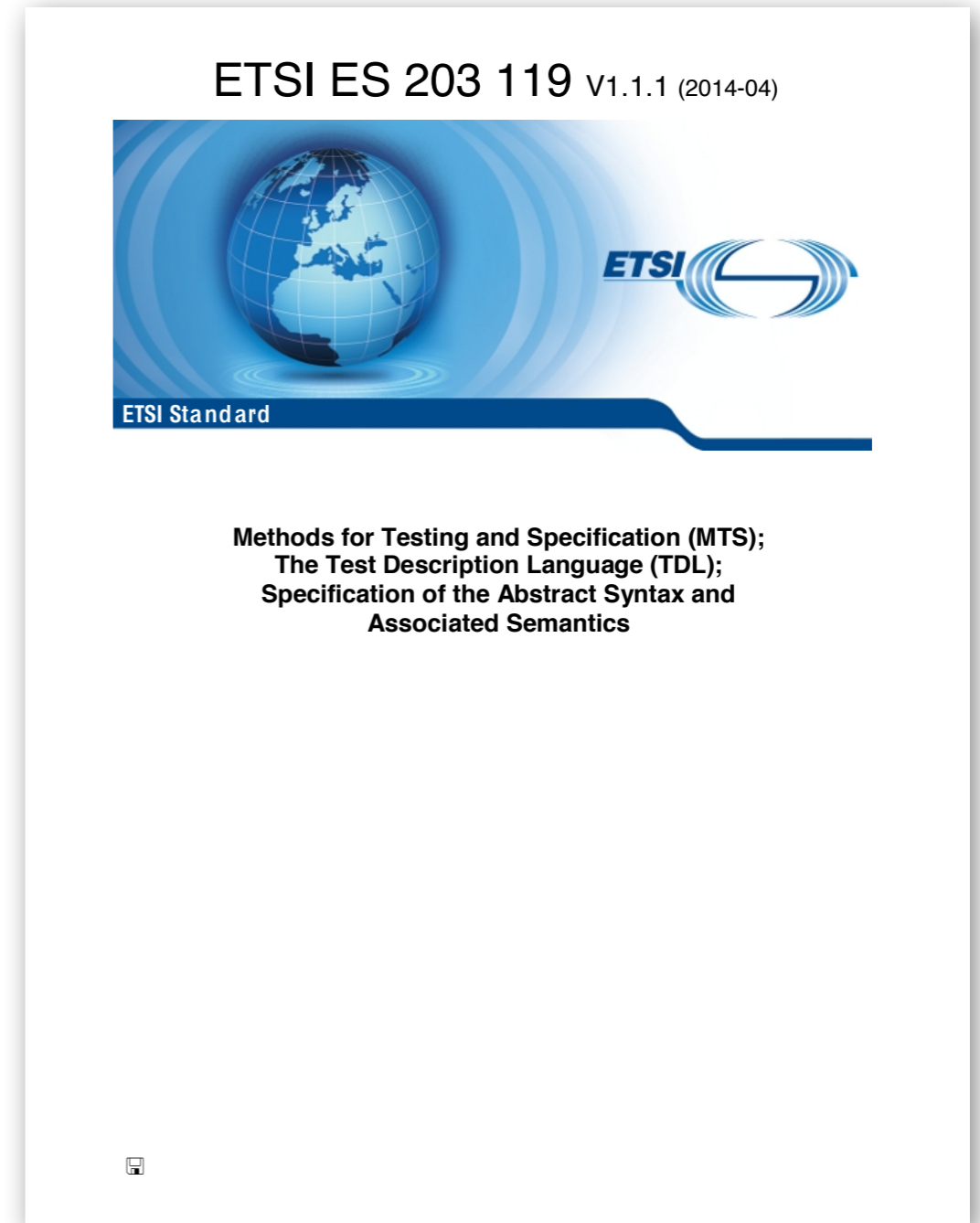
What is TDL?

- TDL main ingredients
 - Test data
 - Test configuration
 - Test behaviour
 - Test objectives



What is TDL?

- Test data
 - abstract symbols
 - can be composed by using parameters
 - mappable to concrete data



What is TDL?

```

template Login johnny_correct := {
  user := "johnny",
  password := "apple",
  hint := "seed",
  id := 1000
}
template Login johnny_incorrect := {
  user := "johnny",
  password := "orange",
  hint := "second favourite fruit",
  id := 2000
}

```

```

type record Login {
  charstring user,
  charstring password,
  charstring hint,
  integer id
} with {
  encode "xpath=//div[@id='login']";
  encode (user) "relative=/div/dd[3]";
  encode (password) "relative=/div/dd[4]";
};

```

```

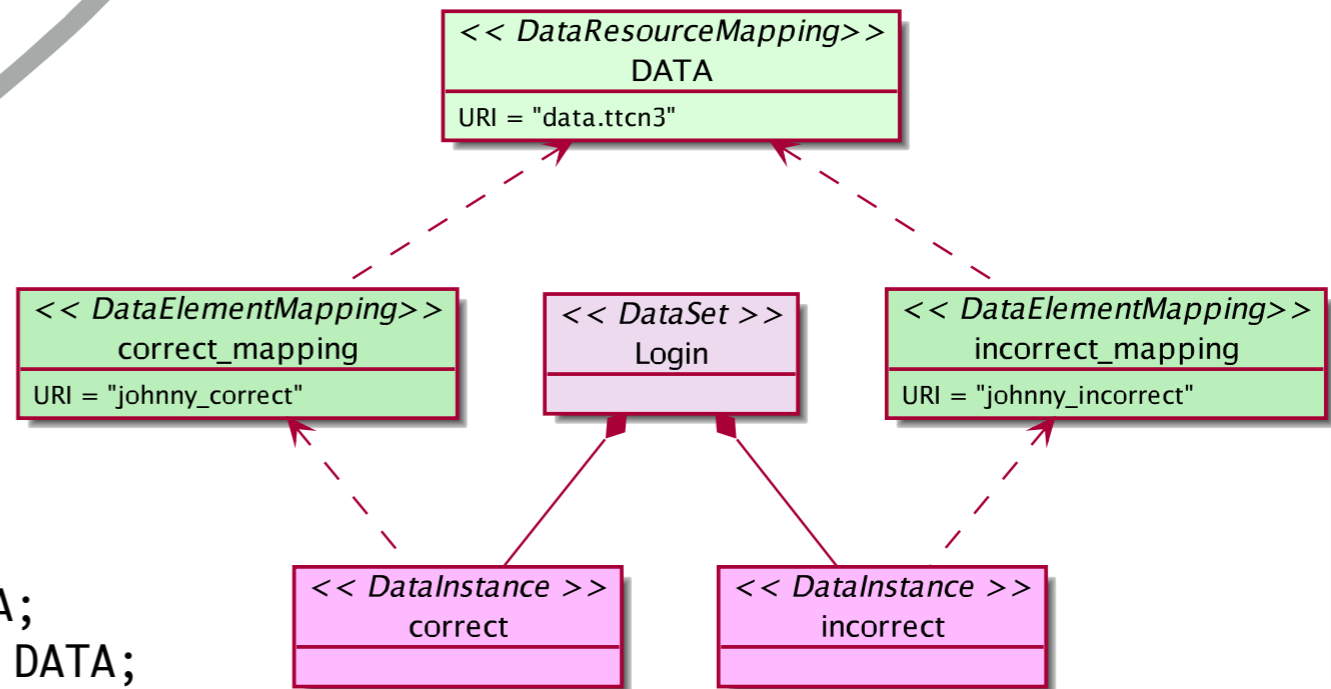
Data Set Login {
  instance correct;
  instance incorrect;
}

```

```

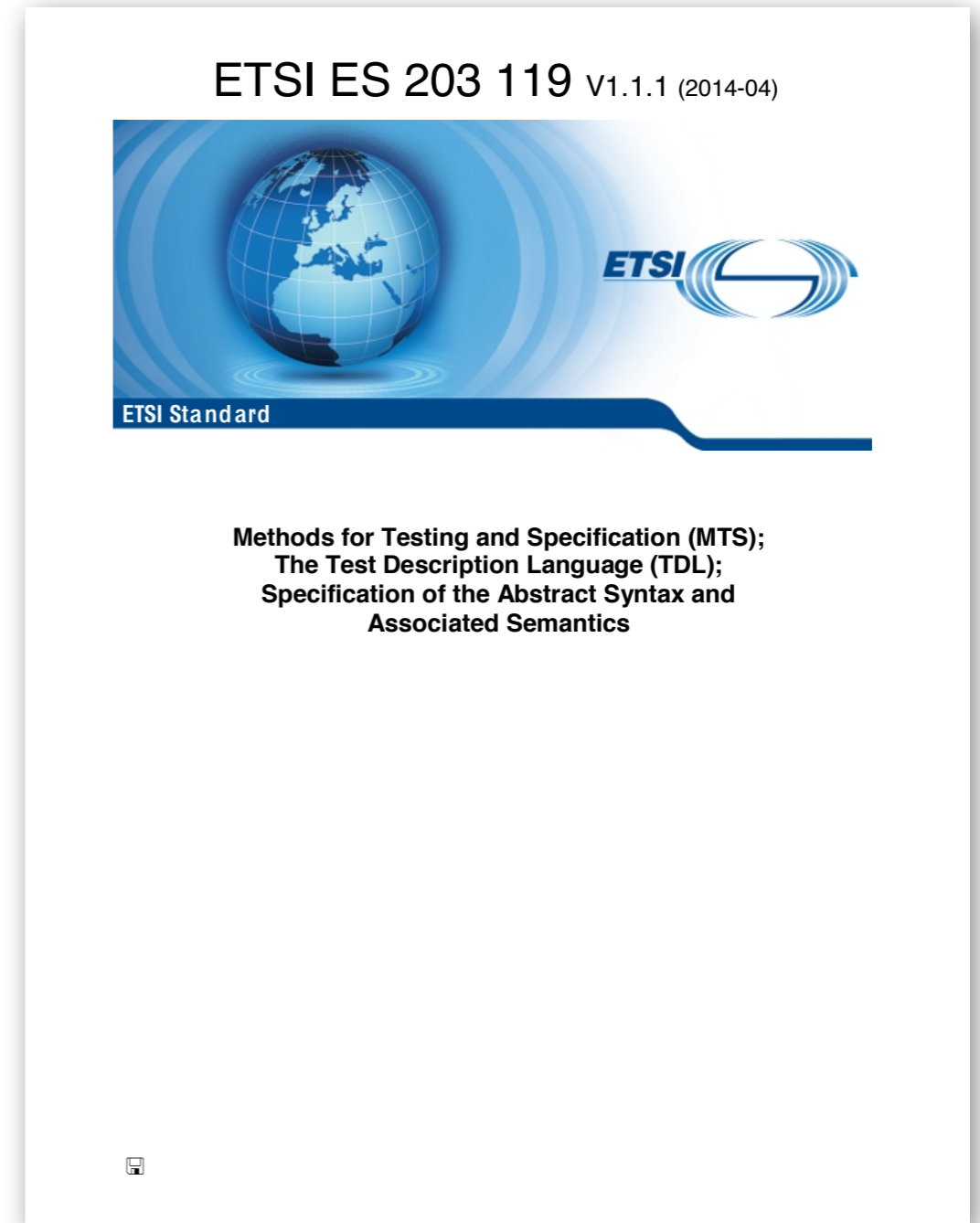
Use "data.ttcn3" as DATA ;
Map correct to "johnny_correct" in DATA;
Map incorrect to "johnny_incorrect" in DATA;

```



What is TDL?

- Test configuration
 - typed components and gates
 - connections among gates
 - component roles

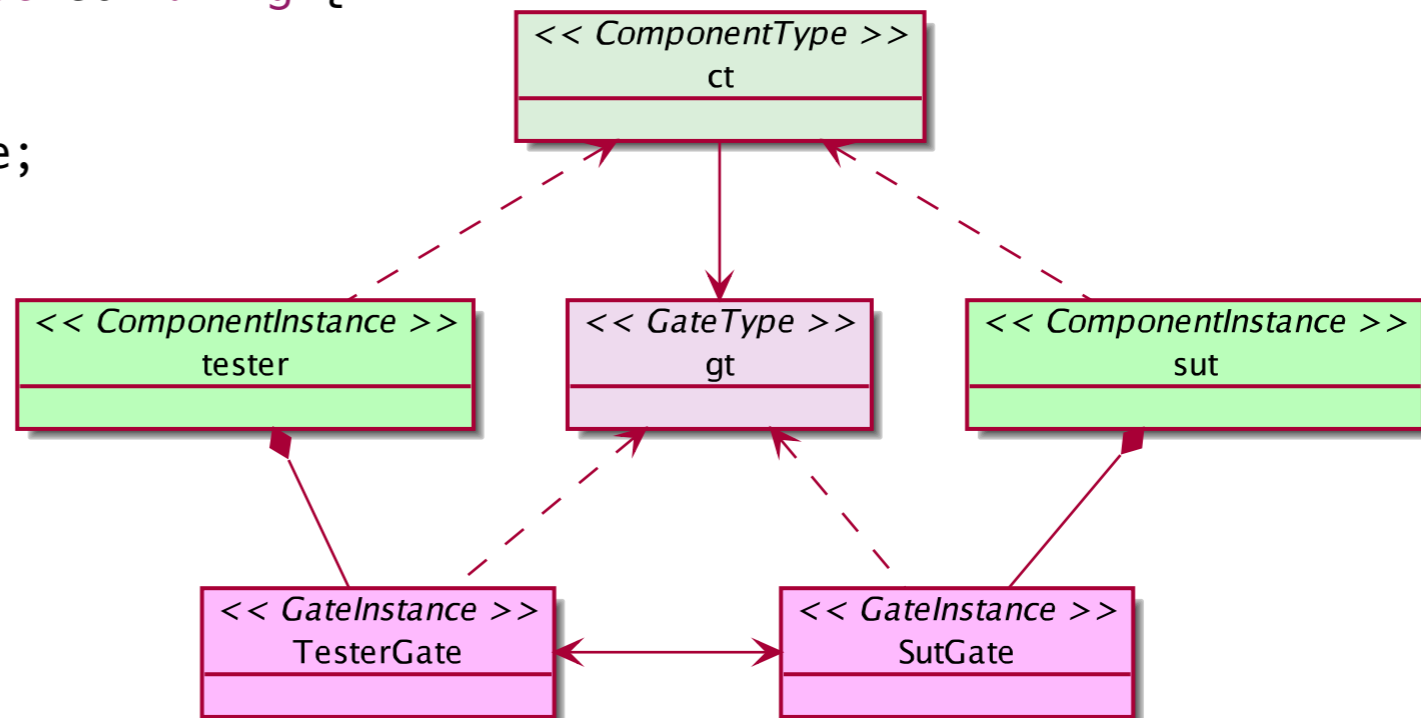
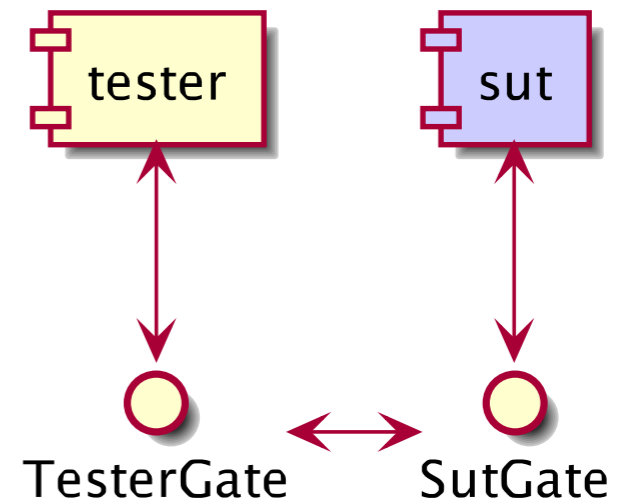


What is TDL?

```
Gate Type gt accepts Login;
```

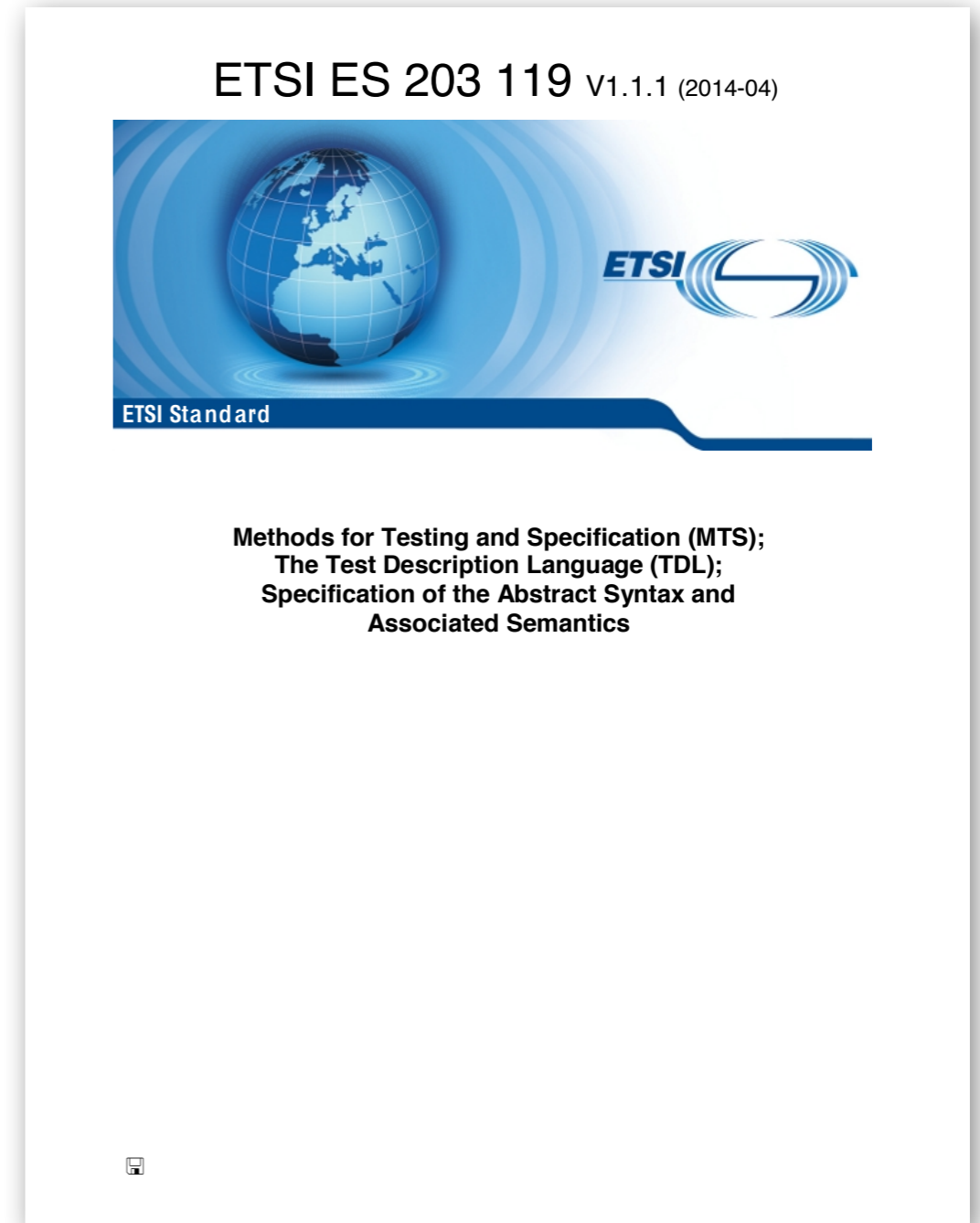
```
Component Type ct {  
  gate types : gt;  
}
```

```
Test Configuration tc {  
  instantiate tester as Tester of type ct having {  
    gate TesterGate of type gt;  
  }  
  instantiate sut as SUT of type ct having {  
    gate SutGate of type gt;  
  }  
  connect TesterGate to SutGate;  
}
```



What is TDL?

- Test behaviour
 - defines expected behaviour
 - failure upon deviations by default
 - actions and interactions
 - alternative, parallel, iterative, conditional
 - defaulting, interrupting, breaking



What is TDL?

```

Test Description td (p from Login) {
  use configuration : tc;
  {
    TesterGate sends instance incorrect to SutGate;
    alternatively {
      SutGate sends instance failure to TesterGate;
      set verdict to pass;
    } or {
      SutGate sends instance success to TesterGate;
      set verdict to pass;
    }
  }
}

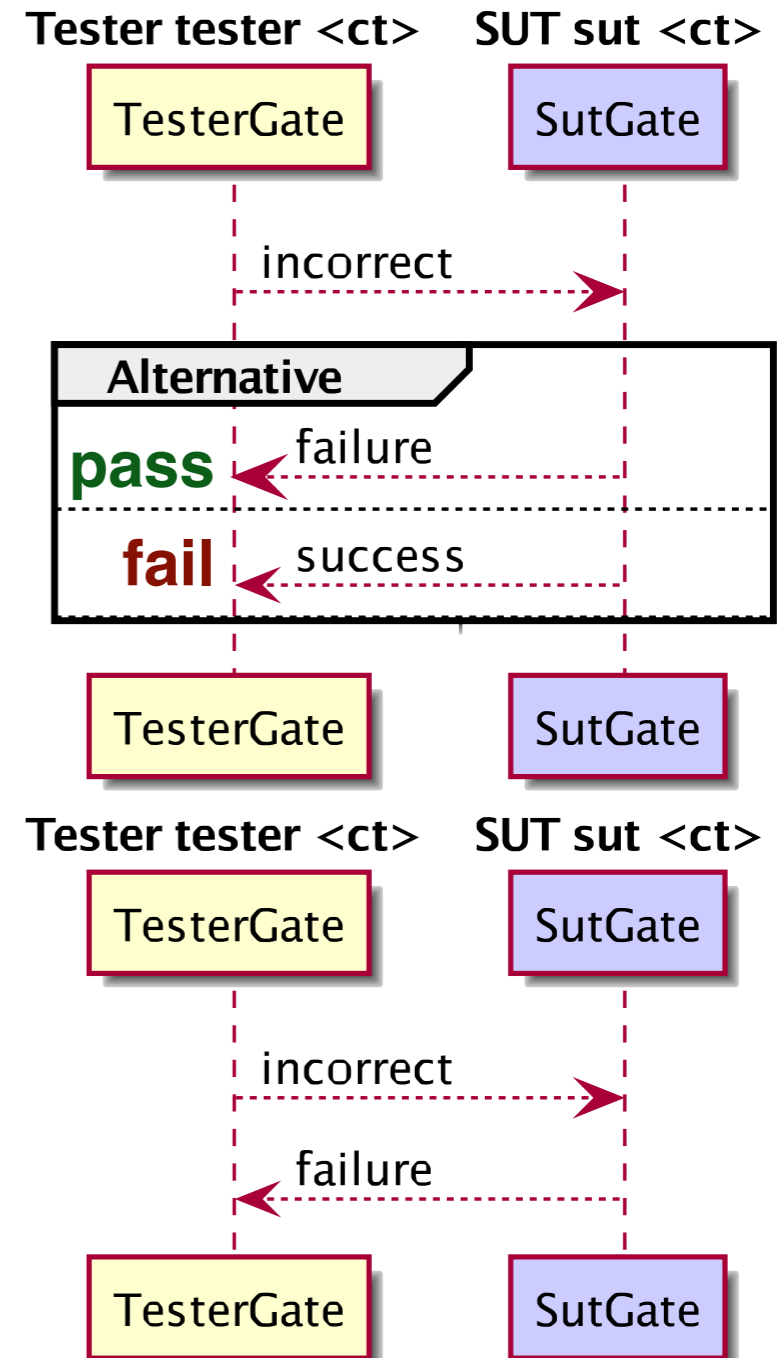
```

or simply (relying on the default semantics):

```

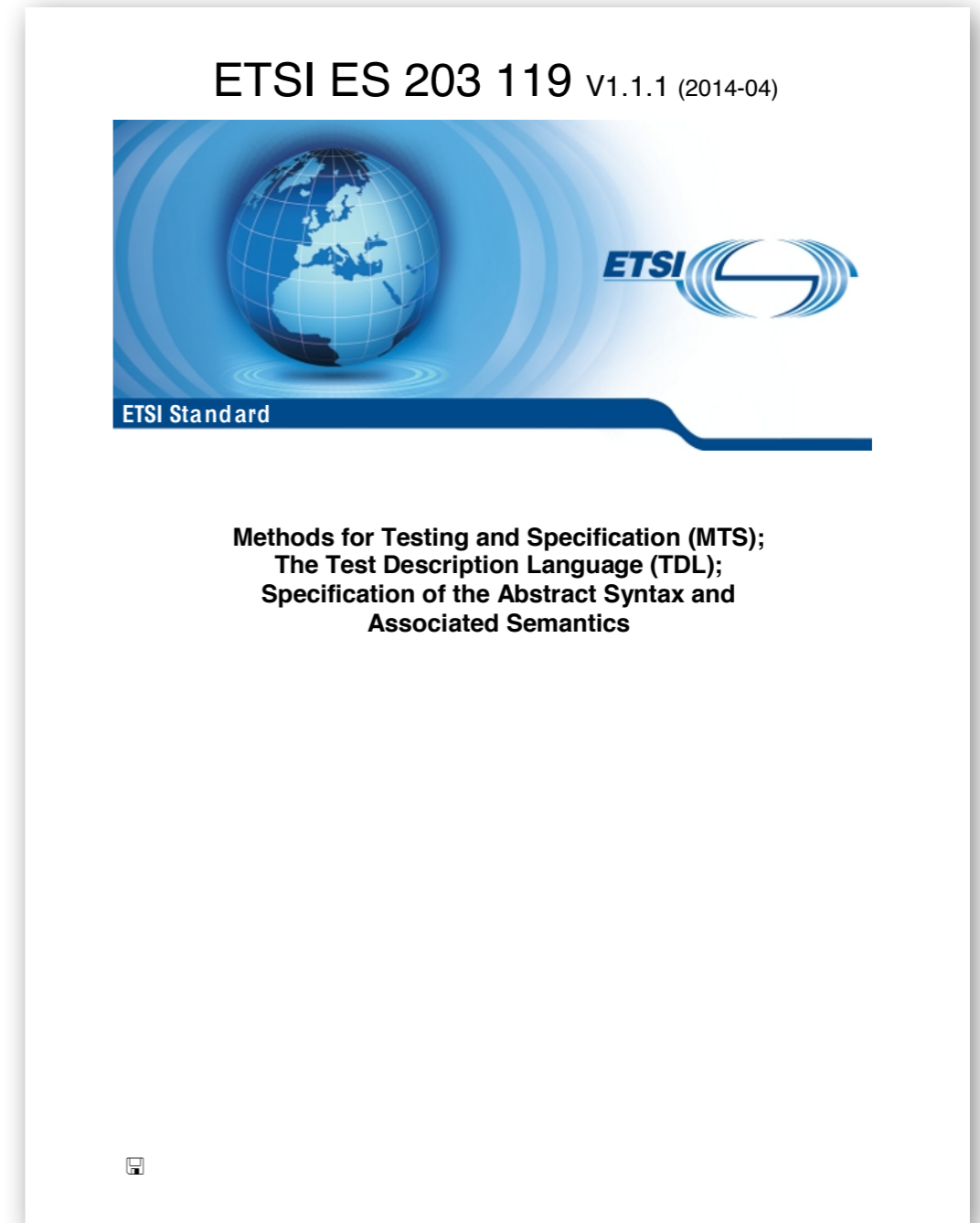
Test Description td (p from Login) {
  use configuration : tc;
  {
    TesterGate sends instance incorrect to SutGate;
    SutGate sends instance failure to TesterGate;
  }
}

```



What is TDL?

- Test objectives
 - may be attached to any realiser
 - behaviour
 - test description
- contain description and reference



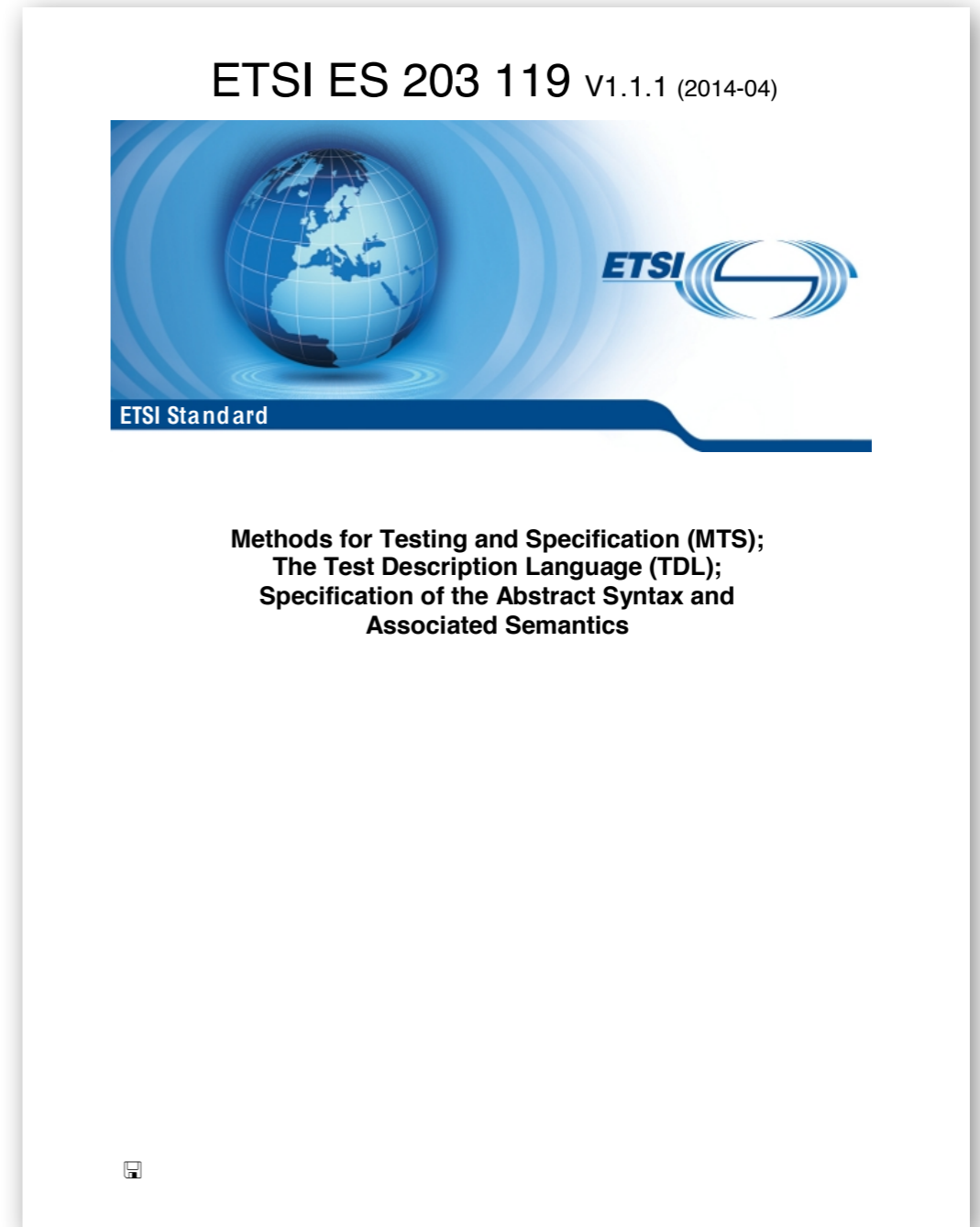
What is TDL?

```
Test Objective tp {
  description : "ensure that
                when incorrect login is provided
                a failure response is sent";
}
Test Description td (p from Login) {
  use configuration : tc;
  {
    TesterGate sends instance incorrect to SutGate;
    alternatively {
      SutGate sends instance failure to TesterGate with {
        test objectives : tp;
      };
      set verdict to pass;
    } or {
      SutGate sends instance success to TesterGate;
      set verdict to pass;
    }
  }
}
```



Getting started with TDL?

- “How to cook up your own tooling”
- You’ll need
 - TDL meta-model
 - Eclipse Modelling Tools and Plugins
 - Xtext (EMFText)
 - PlantUML (Sirius/GMF/Graphiti/Spray)
 - Epsilon (any query/transform tech)



What is TDL?

The screenshot shows the Eclipse IDE with two code editors and a UML diagram. The left editor shows TDL code for a 'Data Set Login' and 'Data Set Response'. The right editor shows a corresponding XML schema for a 'Login' record. The bottom editor displays a UML diagram illustrating the relationships between these elements.

```

2 Data Set Login {
3   instance johnny;
4   instance jimmy;
5   instance any(jimmy,johnny);
6 }
7 Data Set Response {
8   instance success;
9   instance failure;
10 }
11 Verdict pass;
12 Verdict fail;
13
14 Use "data.ttcn3" as DATA ;
15 Map johnny to "johnny_appleseed" in DATA as johnny_ttcn3;
16
3 type record Login {
4   charstring user,
5   charstring password,
6   charstring hint,
7   integer id
8 } with {
9   encode "xpath=//div[@id='login']";
10  encode (user) "relative=/div/dd[3]";
11  encode (password) "relative=/div/dd[4]";
12 };
13
14 template Login johnny_appleseed := {
15   user := "johnny",
16   password := "apples"

```

```

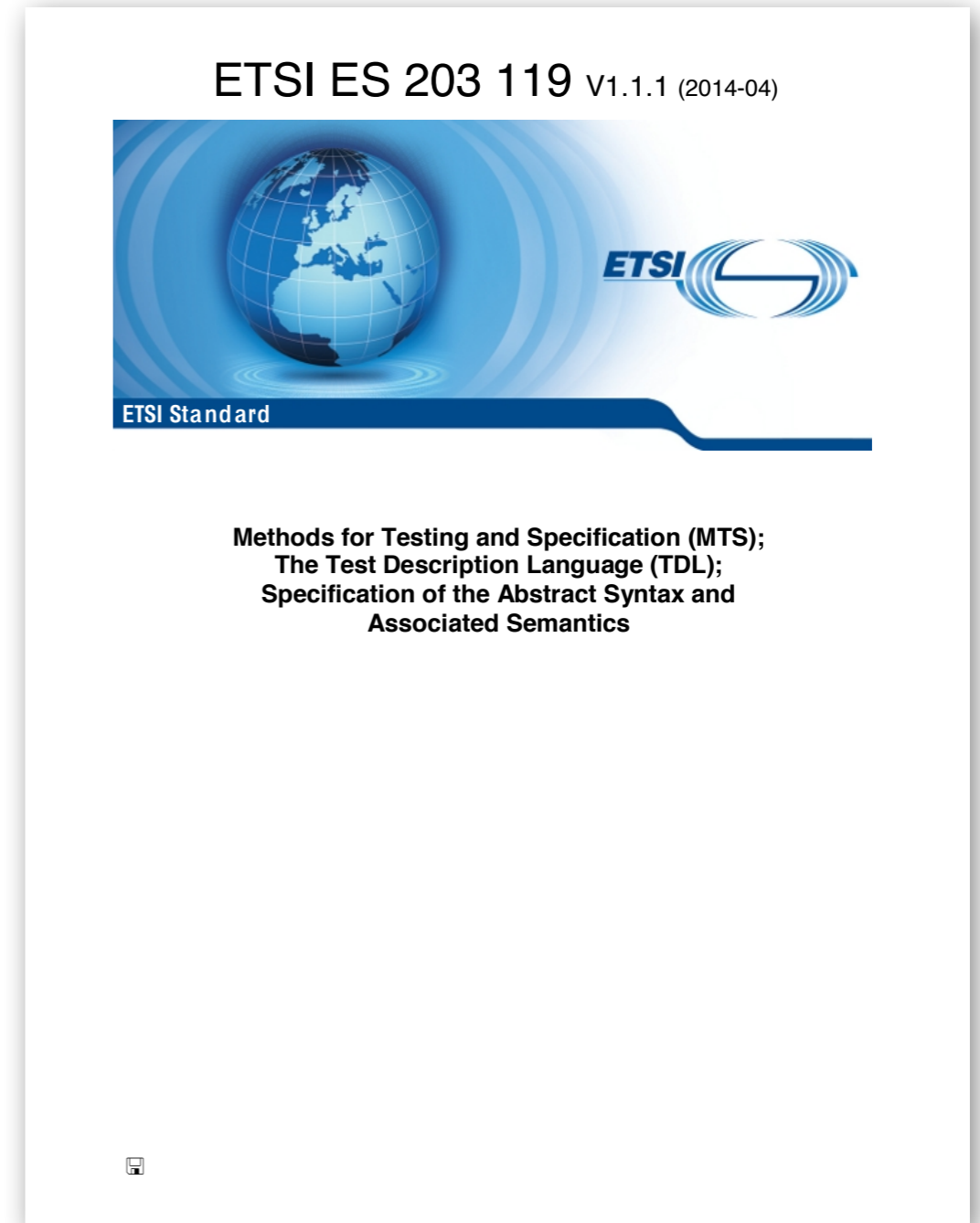
classDiagram
    class DATA["<< DataResourceMapping >> DATA"] {
        URI = "data.ttcn3"
    }
    class Response["<< DataSet >> Response"]
    class Login["<< DataSet >> Login"]
    class johnny_mapping["<< DataElementMapping >> johnny_mapping"] {
        URI = "johnny_appleseed"
    }
    class johnny["<< DataInstance >> johnny"]
    class jimmy["<< DataInstance >> jimmy"]
    class any["<< DataInstance >> any"]
    class success["<< DataInstance >> success"]
    class failure["<< DataInstance >> failure"]

    DATA ..> johnny_mapping
    johnny_mapping ..> Login
    johnny ..> Login
    jimmy ..> Login
    any ..> Login
    Login ..> Response
    success ..> Response
    failure ..> Response

```

What is TDL?

- From this morning's session...
 - maturity
 - proof by implementation
 - validation by tests
 - transparent change management



What is TDL?



ETSI's Bug Tracker

Logged in as: *makedonski* (Philip Makedonski - manager)

13-09-2014 22:00 IST

Project: TDL TDL

[Main](#) | [My View](#) | [View Issues](#) | [Report Issue](#) | [Change Log](#) | [Roadmap](#) | [Summary](#) | [Manage](#) | [My Account](#) | [Logout](#)

Issue #

Search

[\[Advanced Filters \]](#) [\[Create Permalink \]](#)

Viewing Issues (1 - 10 / 10) [\[Print Reports \]](#) [\[CSV Export \]](#) [\[Excel Export \]](#)

| | P | ID | # | Project | Severity | Status | Updated | Summary | |
|--------------------------|---|----|---------|---------|----------------|---------|---|------------|---|
| <input type="checkbox"/> | | | 0006768 | 1 | TDL meta-model | major | assigned (Andreas Ulrich) | 01-08-2014 | New MM element as the starting point of the Behaviour Description of a Test Description |
| <input type="checkbox"/> | | | 0006773 | | TDL | feature | assigned (Andreas Ulrich) | 31-07-2014 | Accessing DataProxy arguments |
| <input type="checkbox"/> | | | 0006765 | | TDL meta-model | major | assigned (Andreas Ulrich) | 31-07-2014 | Time Observation |
| <input type="checkbox"/> | | | 0006764 | 1 | TDL meta-model | minor | assigned (Andreas Ulrich) | 31-07-2014 | Description of VerdictType shall be modified |
| <input type="checkbox"/> | | | 0006763 | | TDL meta-model | minor | resolved (Andreas Ulrich) | 31-07-2014 | Blocks of ParallelBehaviour should be able to declare Guards |
| <input type="checkbox"/> | | | 0006767 | 1 | TDL | minor | resolved (Andreas Ulrich) | 11-07-2014 | Allow to reference test descriptions that run on a different test (sub-) configuration |
| <input type="checkbox"/> | | | 0006772 | | TDL | feature | assigned (Andreas Ulrich) | 10-06-2014 | Variable assignment from Interaction and ActionReference |
| <input type="checkbox"/> | | | 0006771 | | TDL | feature | assigned (Andreas Ulrich) | 10-06-2014 | Component variables |
| <input type="checkbox"/> | | | 0006770 | | TDL | feature | assigned (Andreas Ulrich) | 10-06-2014 | Named parameters |
| <input type="checkbox"/> | | | 0006769 | | TDL | feature | assigned (Andreas Ulrich) | 10-06-2014 | Move parameters from DataInstance to DataSet |

Select All

new feedback acknowledged confirmed assigned resolved closed

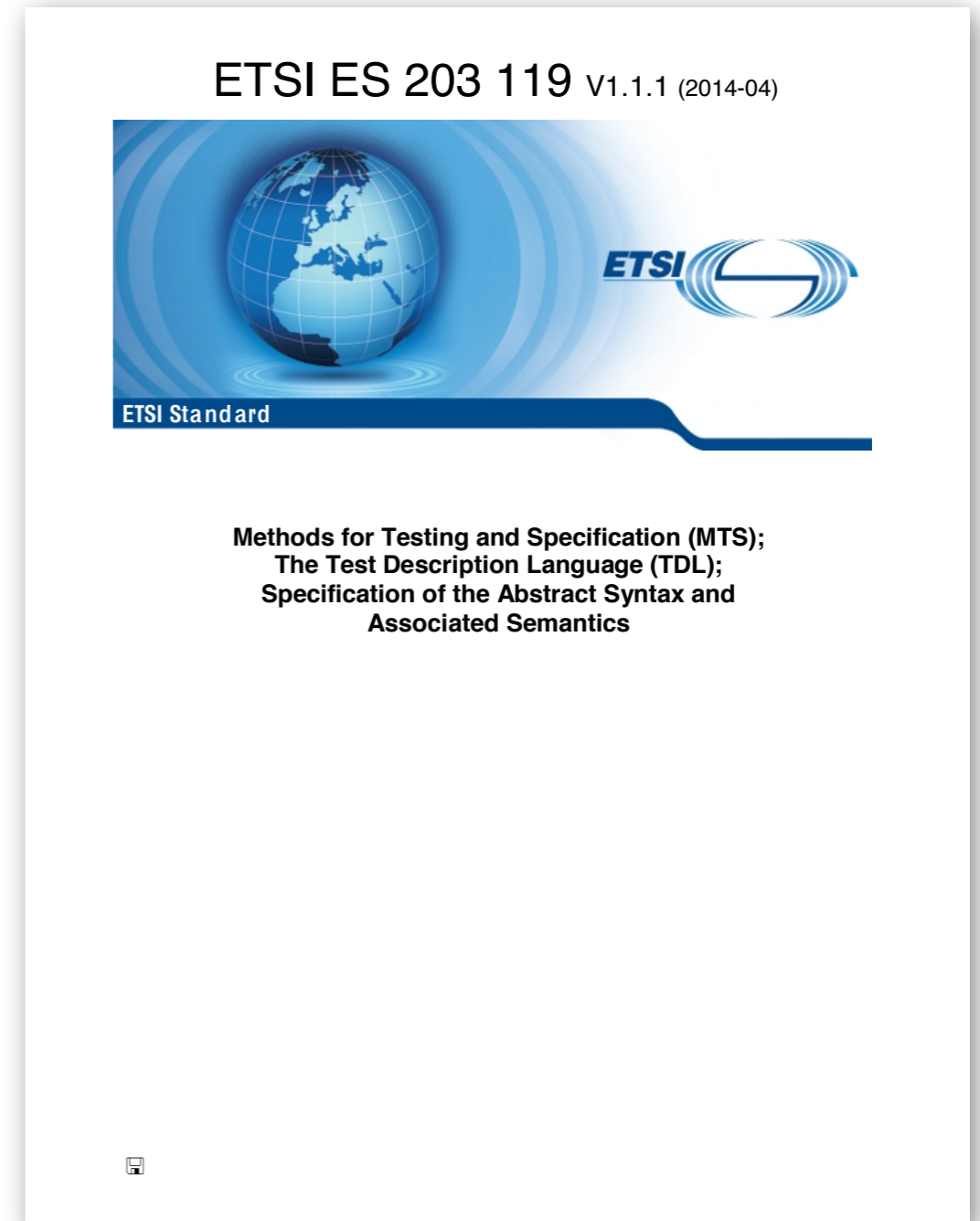
MantisBT 1.2.14 [^]
Copyright © 2000 - 2014 MantisBT Team
denis.filatov@etsi.org



TDL Project at http://forge.etsi.org/mantis/view_all_bug_page.php

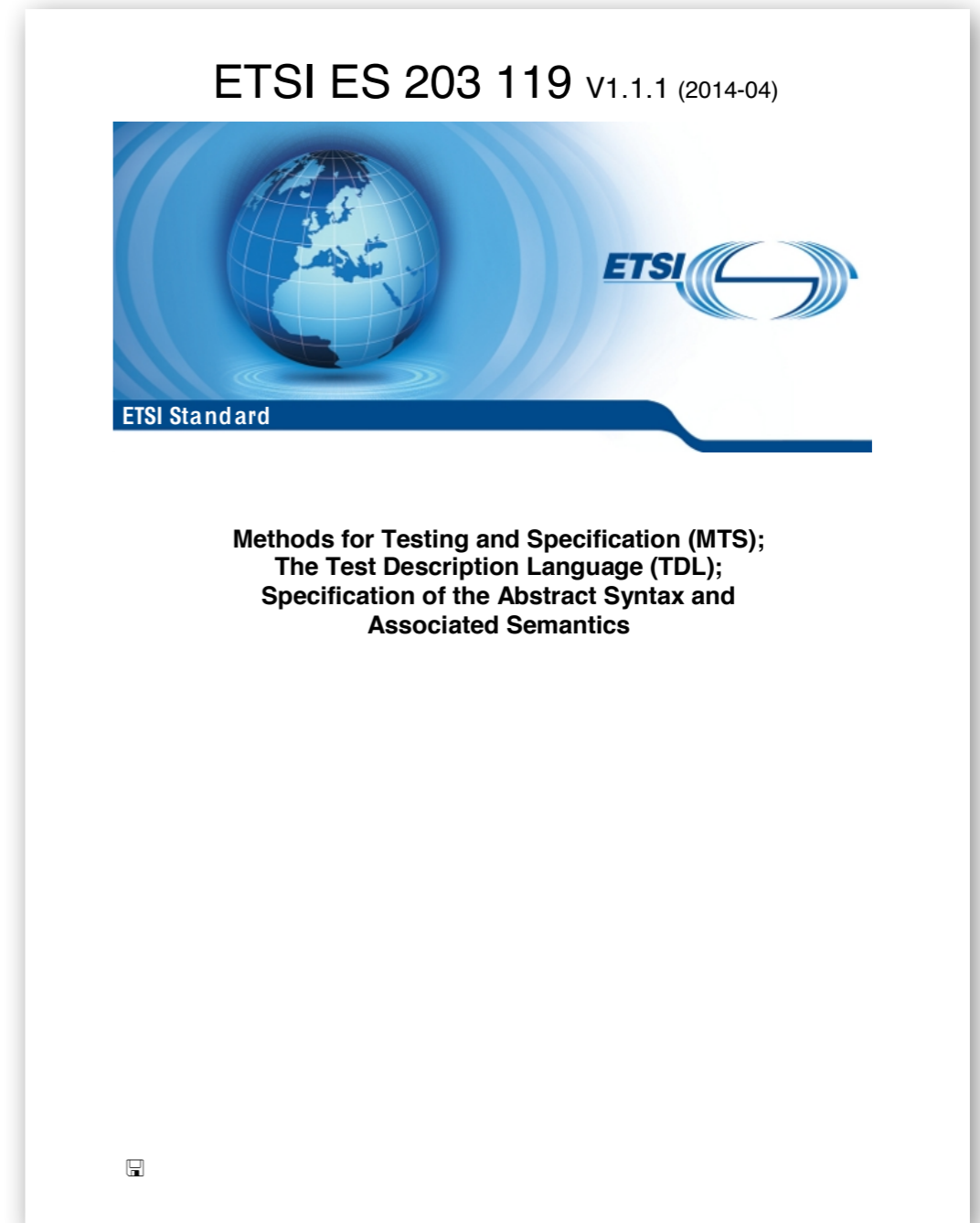
What is TDL?

- Test Description Language
 - Design, documentation, and representation of formal test descriptions
 - Scenario-based approach
- Standardised at ETSI by TC MTS
- Initial work within STF 454
- Current efforts within STF 476

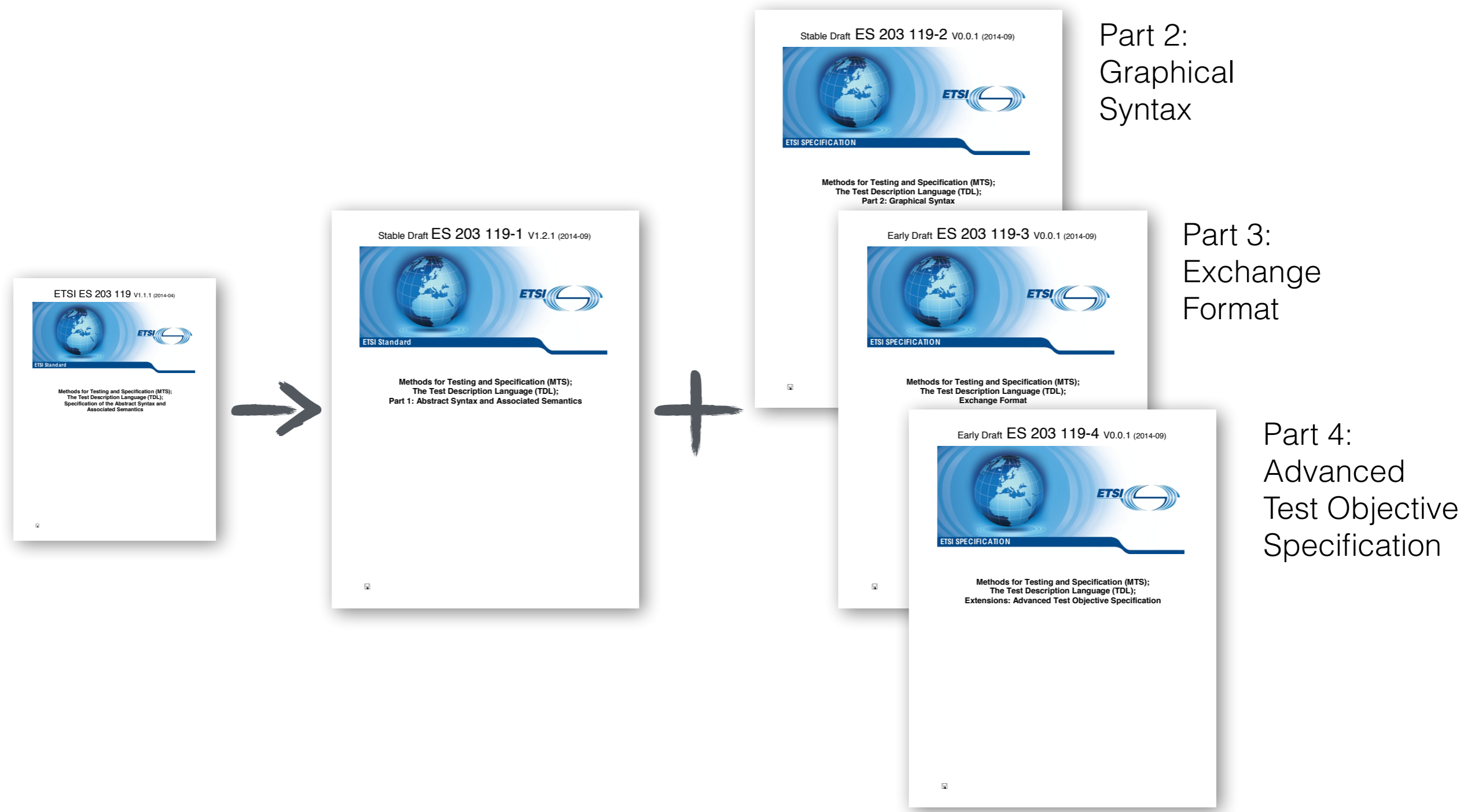


What is TDL?

- Test Description Language
 - Design, documentation, and representation of formal test descriptions
 - Scenario-based approach
- Standardised at ETSI by TC MTS
- Initial work within STF 454
- Current efforts within STF 476




What is new in TDL?



What is new in TDL?

- Towards automation support
 - Data handling
 - Test configurations
 - Time concepts


Stable Draft **ES 203 119-1** V1.2.1 (2014-09)



ETSI Standard

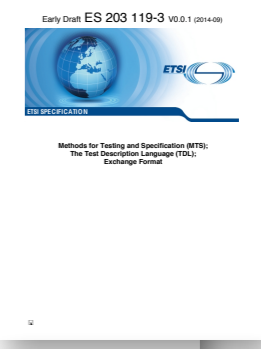
**Methods for Testing and Specification (MTS);
The Test Description Language (TDL);
Part 1: Abstract Syntax and Associated Semantics**

Stable Draft **ES 203 119-2** V0.0.1 (2014-09)




Methods for Testing and Specification (MTS);
The Test Description Language (TDL);
Part 2: Graphical Syntax

Early Draft **ES 203 119-3** V0.0.1 (2014-09)



Methods for Testing and Specification (MTS);
The Test Description Language (TDL);
Exchange Format

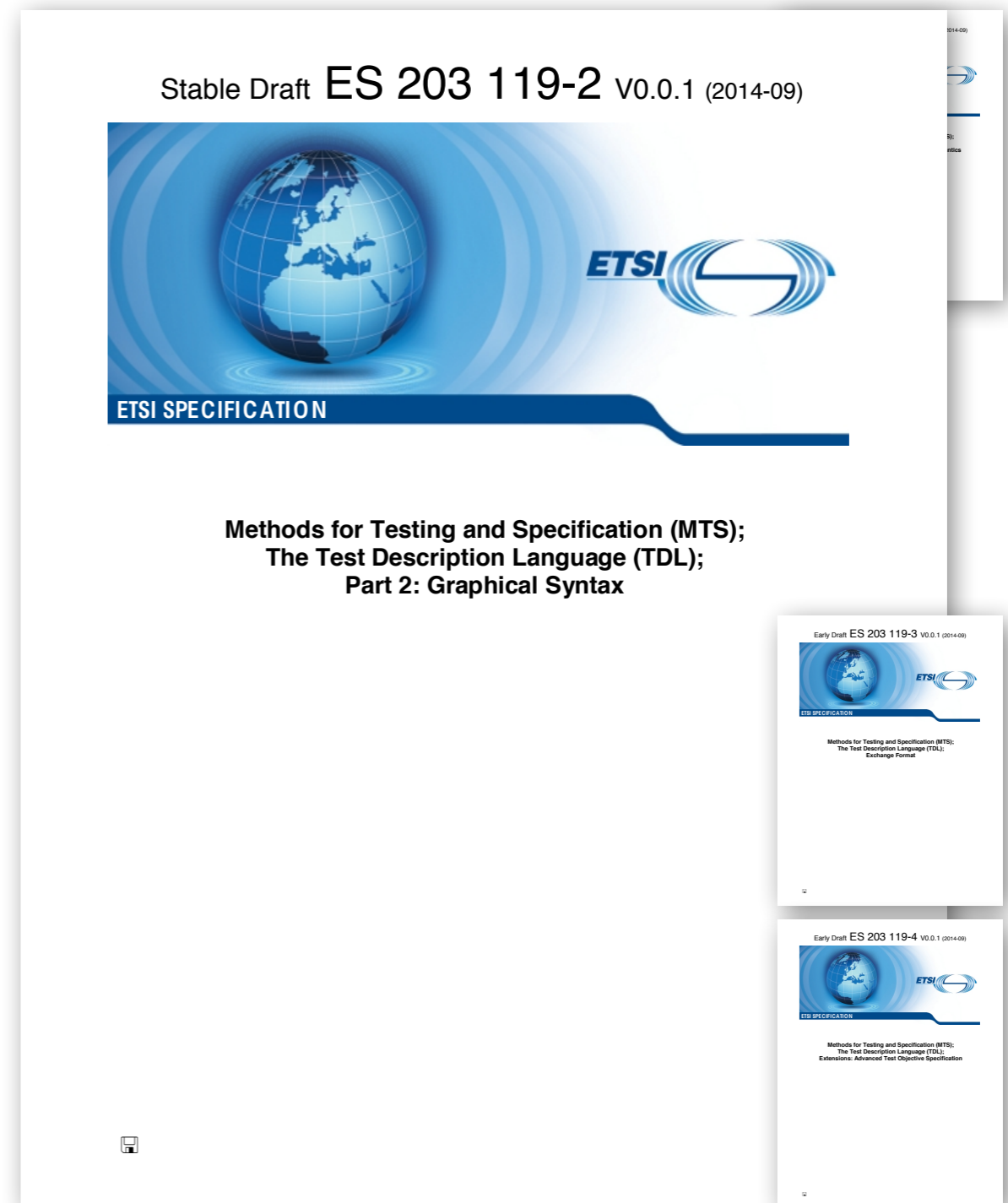
Early Draft **ES 203 119-4** V0.0.1 (2014-09)



Methods for Testing and Specification (MTS);
The Test Description Language (TDL);
Extensions: Advanced Test Objective Specification

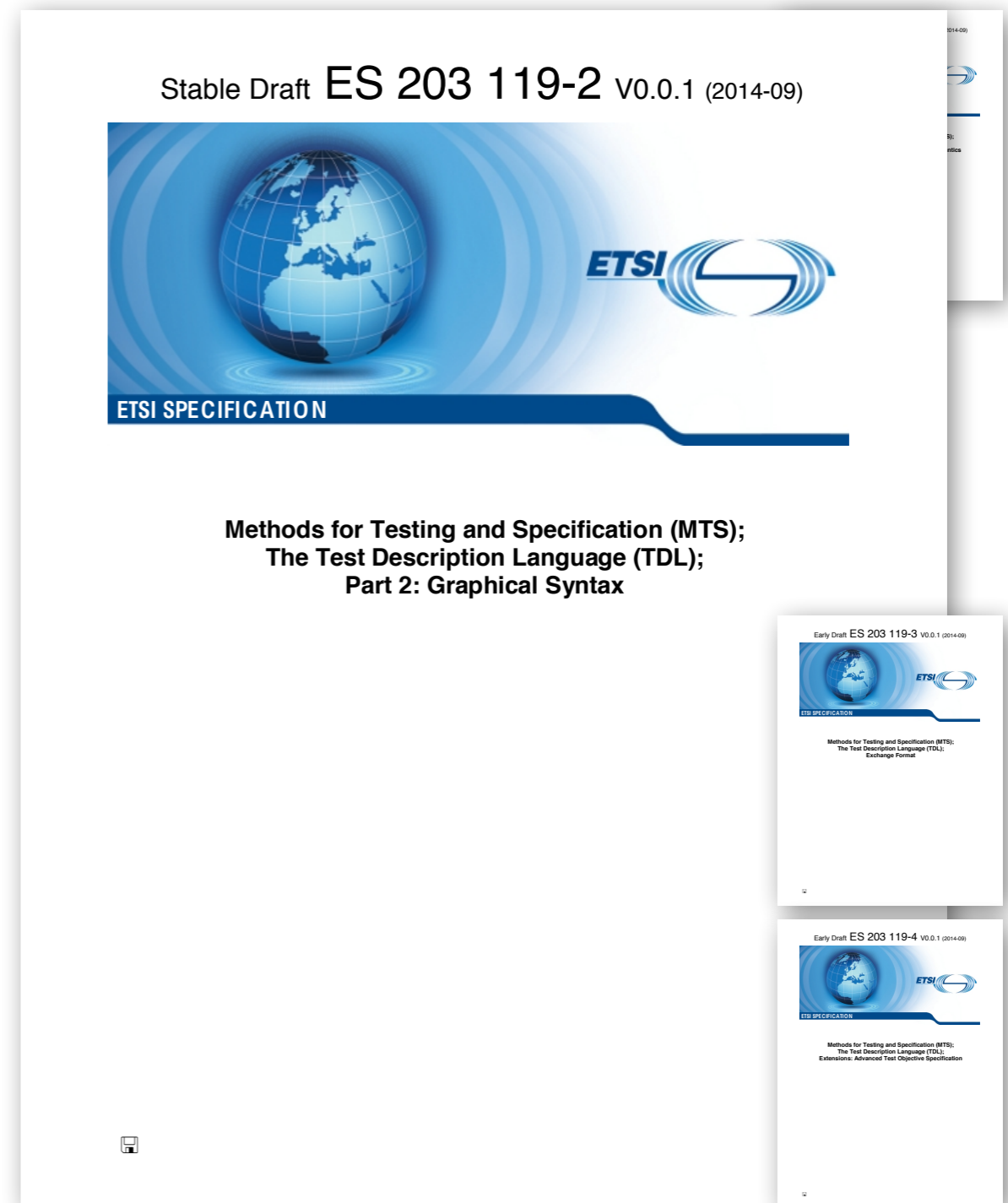
What is new in TDL?

- Graphical languages
 - Common in (test) modelling
 - Ease communication
- TDL Graphical Syntax
 - hybrid graphical language
 - simple shapes, compartments
 - textual visualisation of contents

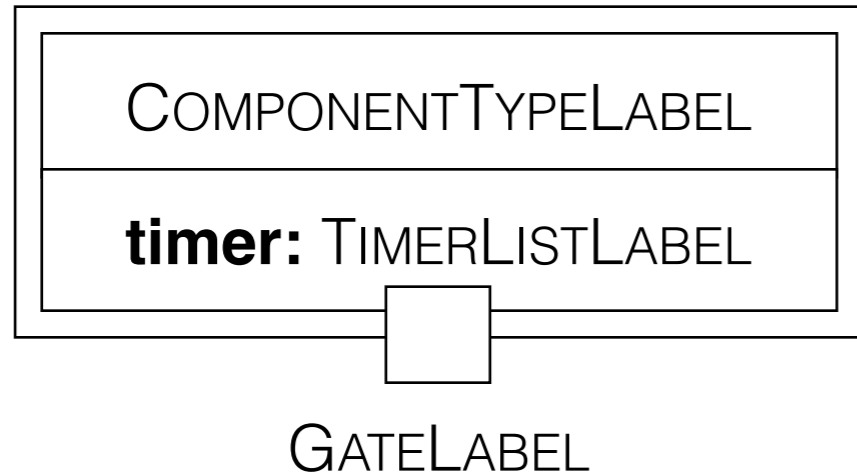


What is new in TDL?

- Aligned with UML
 - distinct where semantics differ
- One diagram to rule them all!
- BNF-like label specification
- Considers both ease of use **and** implementation
- Prototyped with Sirius



What is new in TDL?



context: ComponentType

COMPONENTTYPELABEL ::= self.name

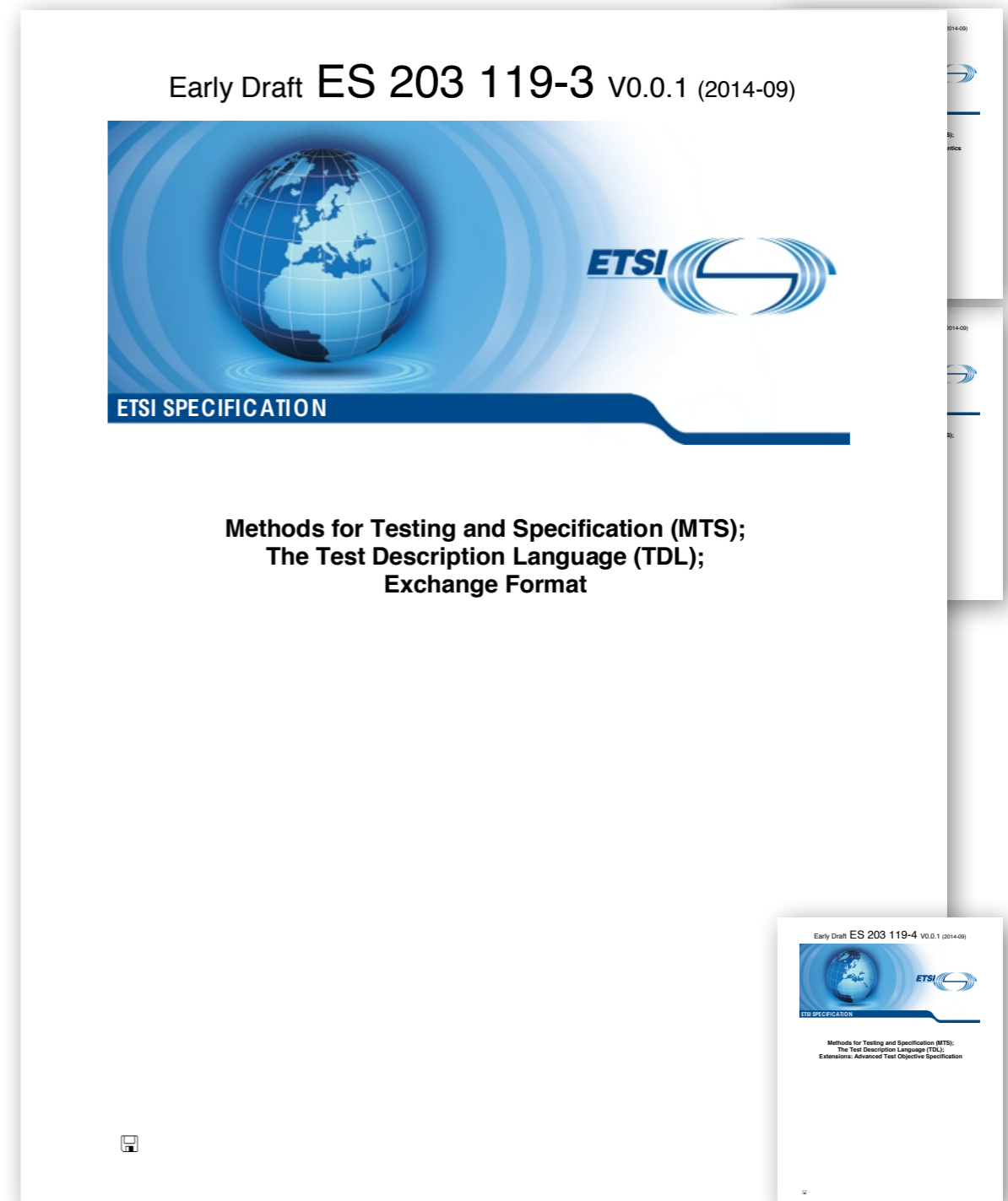
TIMERLISTLABEL ::= self.timer.name

...

The image shows the cover of an ETSI specification document. At the top, it reads 'Stable Draft ES 203 119-2 V0.0.1 (2014-09)'. Below this is a blue header with a globe and the ETSI logo. The text 'ETSI SPECIFICATION' is written in white on a blue background. The main title is 'Methods for Testing and Specification (MTS); The Test Description Language (TDL); Part 2: Graphical Syntax'. To the right, there are two smaller thumbnail images of other ETSI specification covers: 'Early Draft ES 203 119-3 V0.0.1 (2014-09) Methods for Testing and Specification (MTS); The Test Description Language (TDL); Exchange Format' and 'Early Draft ES 203 119-4 V0.0.1 (2014-09) Methods for Testing and Specification (MTS); The Test Description Language (TDL); Extensions: Advanced Test Objective Specification'.

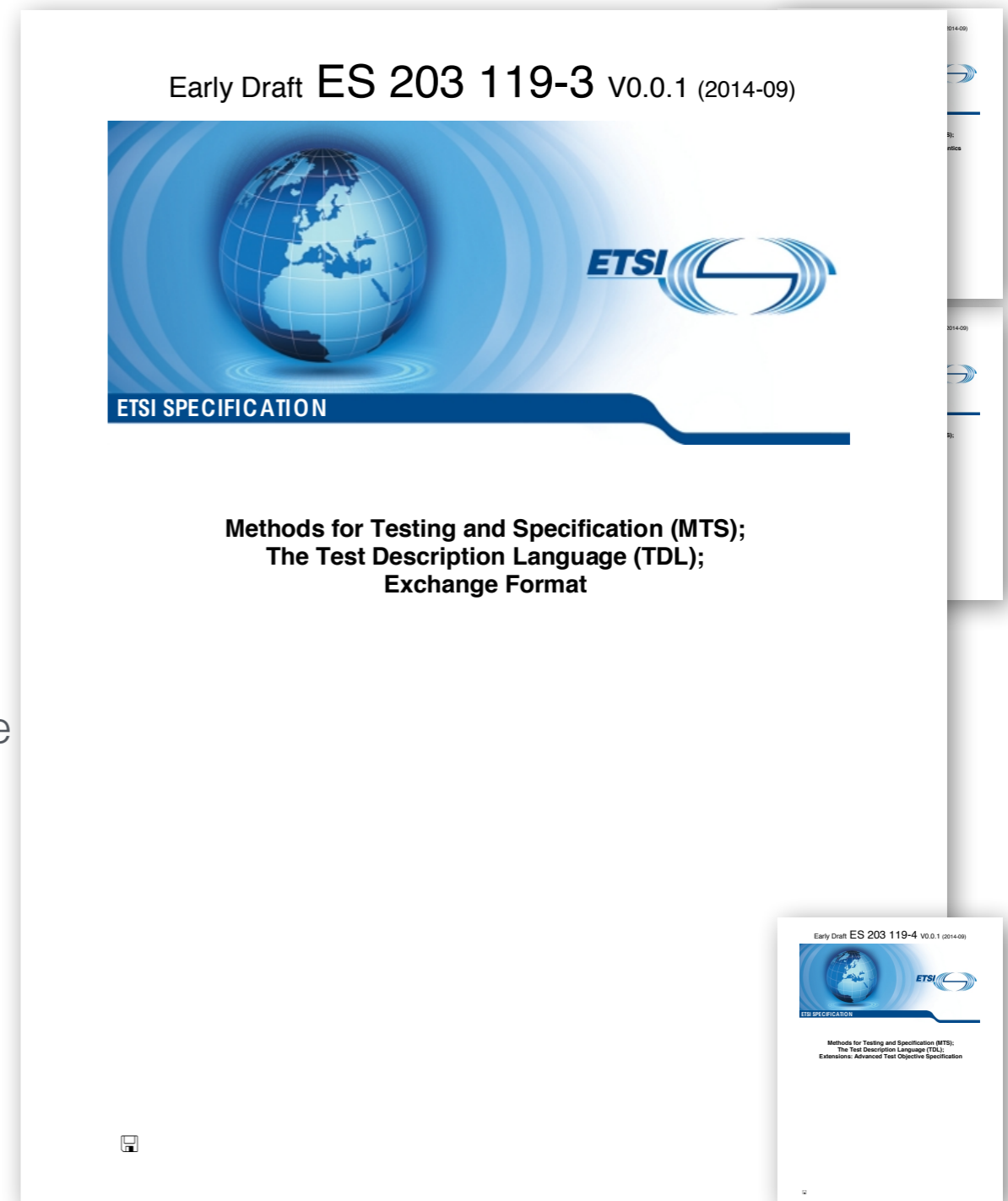
What is new in TDL?

- Based on OMG XMI
 - XML: Metadata Interchange
 - Serialisation of MOF models
 - Exchange among MOF tools
- XMI concerns
 - complex, many options



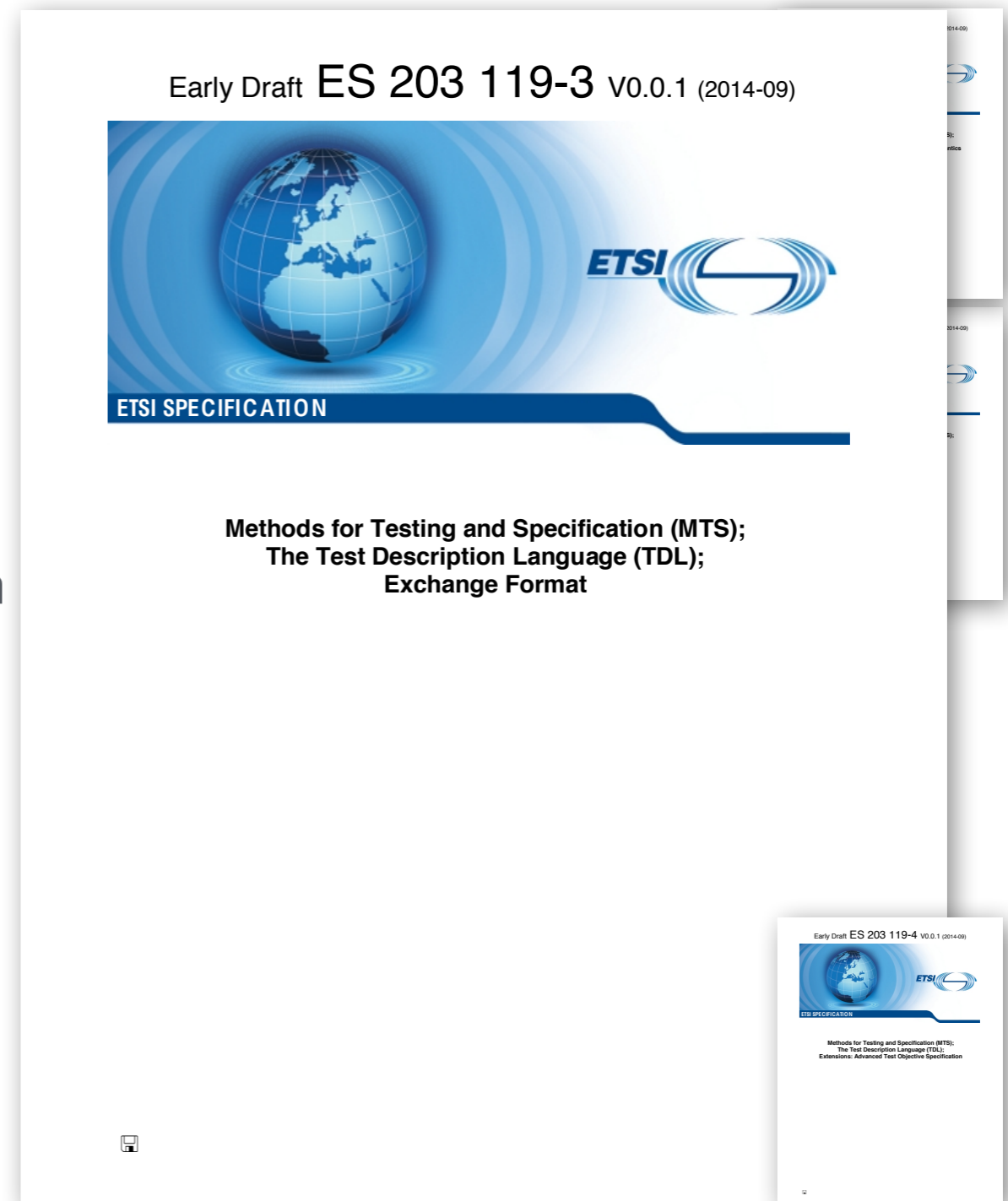
What is new in TDL?

- TDL specific XMI structure
 - exchange of TDL models
 - canonical TDL XMI structure
 - meta-class representations
 - multiplicity, associations, inheritance
- restrict flexibility of XMI
- syntactical validity only!

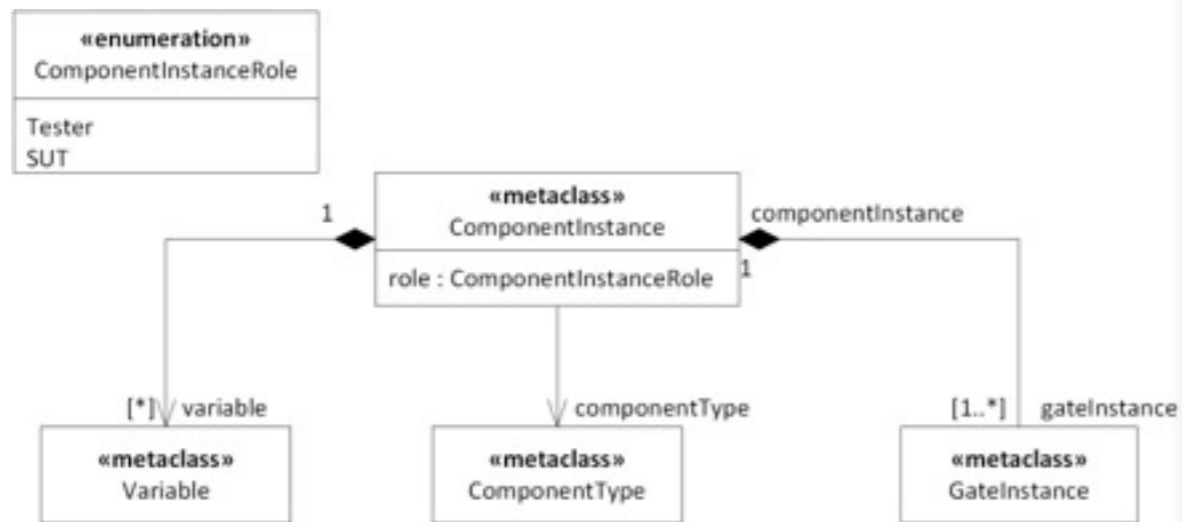


What is new in TDL?

- Syntactical validity only?
 - two-step validation
 - syntax: XMI Schema
 - semantics: MOF model validation



What is new in TDL?



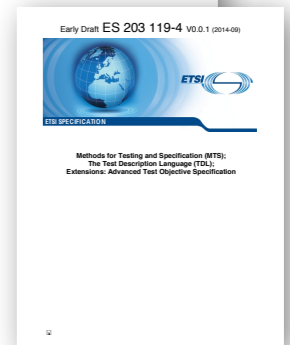
```

<xsd:complexType name="ComponentInstance">
  <xsd:complexContent>
    <xsd:extension base="tdl:Element">
      <xsd:choice maxOccurs="unbounded" minOccurs="0">
        <xsd:element name="gateInstance" type="tdl:GateInstance"/>
        <xsd:element name="variable" type="tdl:Variable"/>
      </xsd:choice>
      <xsd:attribute name="componentType" type="xsd:anyURI">
      <xsd:attribute name="role" type="tdl:ComponentInstanceRole">
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
  
```

Early Draft ES 203 119-3 V0.0.1 (2014-09)

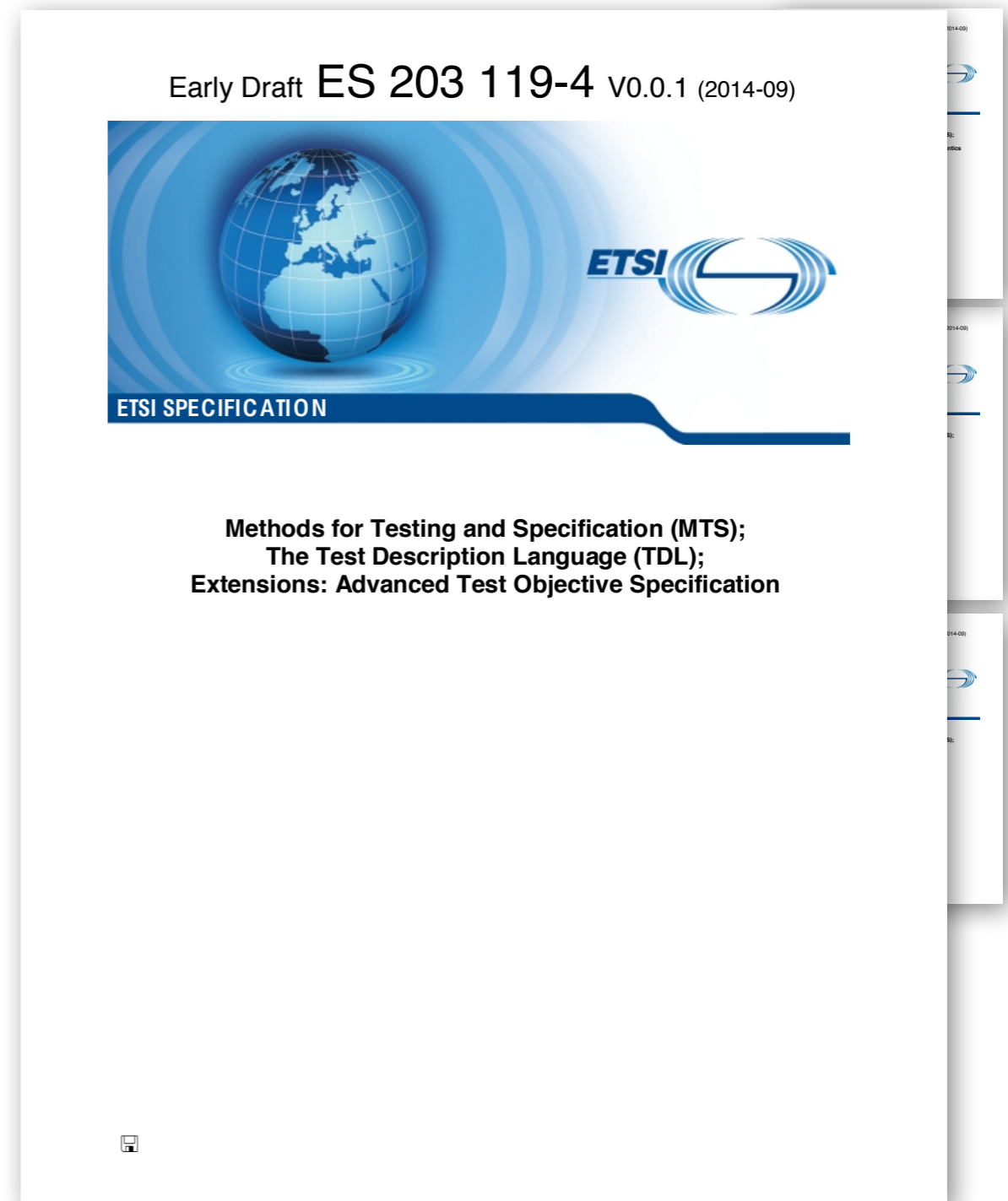


Methods for Testing and Specification (MTS);
The Test Description Language (TDL);
Exchange Format



What is new in TDL?

- Based on TPLan
 - refine test objectives
 - formalise specification
 - integrate and unify test description and test purpose specification



What is new in TDL?

Base Standard Specification

Identification of Requirements

Creation of ICS/IFS

Definition of TSS

Specification of Test Purposes

Specification of Test Descriptions

Specification of Test Cases

Validation

Early Draft **ES 203 119-4** V0.0.1 (2014-09)



**Methods for Testing and Specification (MTS);
The Test Description Language (TDL);
Extensions: Advanced Test Objective Specification**



What is new in TDL?

Base Standard Specification

Identification of Requirements

Creation of ICS/IFS

Definition of TSS

Specification of Test Purposes

Specification of Test Descriptions

Specification of Test Cases

Validation

Early Draft **ES 203 119-4** V0.0.1 (2014-09)



**Methods for Testing and Specification (MTS);
The Test Description Language (TDL);
Extensions: Advanced Test Objective Specification**



What is new in TDL?

Base Standard Specification

Identification of Requirements

Creation of ICS/IFS

Definition of TSS

Specification of Test Purposes

Specification of Test Descriptions

Specification of Test Cases

Validation

Early Draft **ES 203 119-4** V0.0.1 (2014-09)



**Methods for Testing and Specification (MTS);
The Test Description Language (TDL);
Extensions: Advanced Test Objective Specification**



What is new in TDL?

```

TestPurpose {
  TP Id "TP/CAM/INA/DOP/BV/02"
  Test objective "Checks that CAM message includes
                 DoorOpen information 30s after closed"
  Reference "TS 102 637-2 [1], clauses 7.1 and 7.2"
  PICS Selection PICS_PUBTRANSVEH
  Initial conditions:
  with {
    the IUT having reached an initial_state
    and
    the IUT having sent a new valid CAM message
    containing DoorOpen TaggedValue
  }
  Expected behaviour :
  ensure that {
    when {
      the door is closed
    }
    then {
      the IUT sends a new CAM message
      containing DoorOpen TaggedValue
    }
  }
}

```

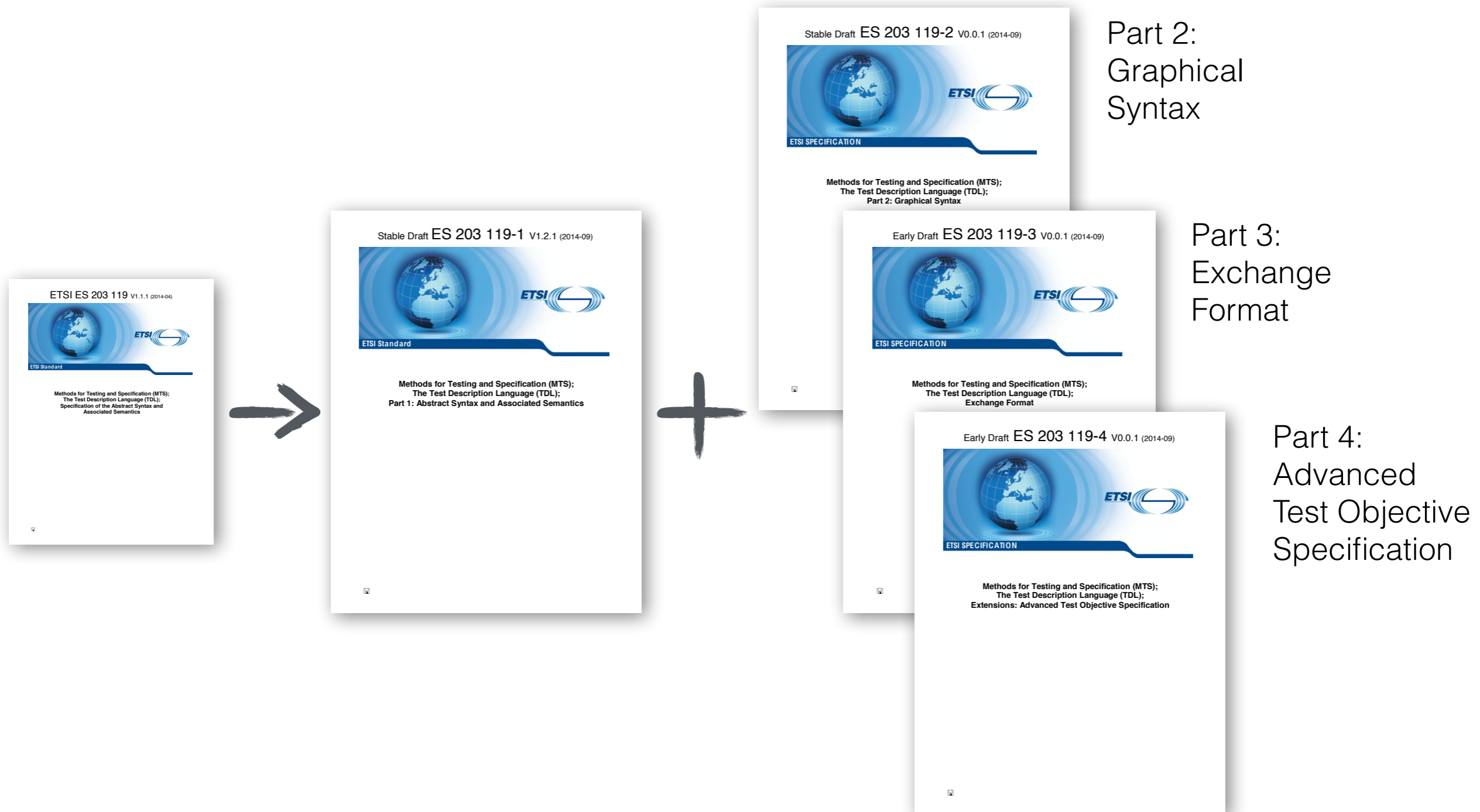
Early Draft ES 203 119-4 V0.0.1 (2014-09)



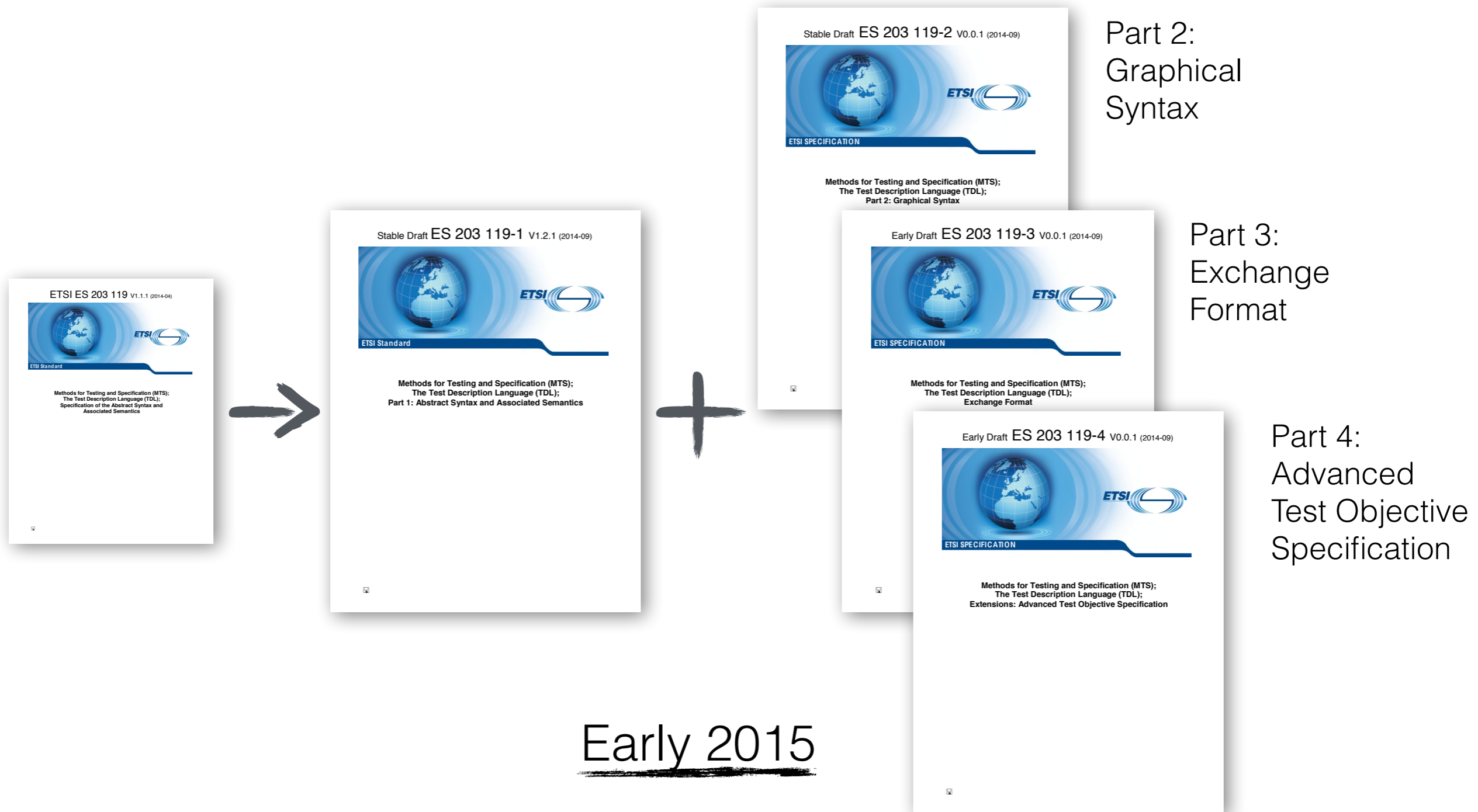
**Methods for Testing and Specification (MTS);
The Test Description Language (TDL);
Extensions: Advanced Test Objective Specification**

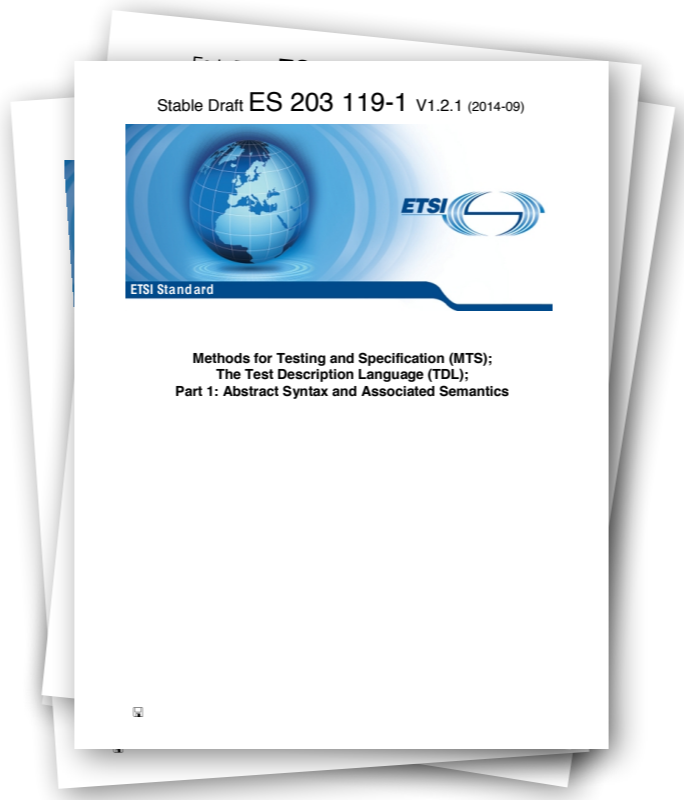


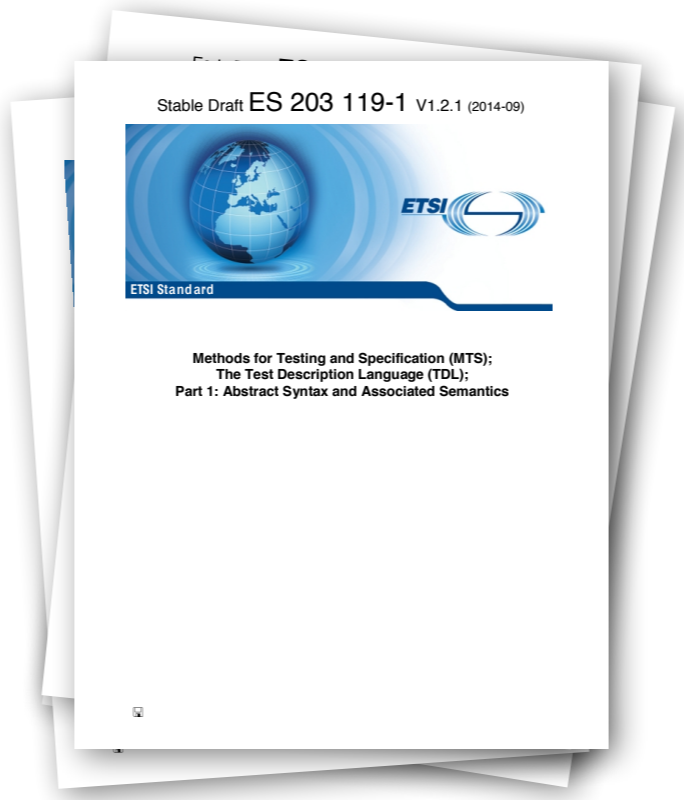
What is new in TDL?



What is new in TDL?







Where does TDL fit in?

Keyword-Driven Testing

MBT

Representation

Generation

Standards



Rail
Visualisation

Documentation

ITS

Interoperability

Conformance

Where does TDL fit in?



Documentation

Where does TDL fit in?



Documentation



-
- Context
 - Conformance and interoperability test descriptions
 - Standardised test specifications for various ETSI technologies
 - Typically protocol oriented, used in certification schemes
 - End-to-end interoperability of systems involving different equipment



-
- Stakeholders
 - High-level discussions at often large meetings (80 to 100 participants)
 - ETSI Technical Committees, 3GPP, other standards organisations, CTI Plugtests team and participants, industrial fora and equipment vendors
 - all need to be familiar with and fluent in the syntax being used.
 - different notions of “good” test
 - Better comprehension among developers with little or no testing expertise
 - bridge the gap between management, core specifications experts, testing experts



- Challenges

- Informal (Word, Excel) or semi-formal (TPLan) approaches
 - considered inadequate, often no test descriptions produced as a consequence
 - no single consistent approach, varying level of quality, detail, difficult maintenance
 - certification requires completeness and accuracy, test descriptions are the design stage before developing TTCN-3 test cases
- Acceptance for more rigorous approaches among Technical Committees
 - applicable to a wide range of technologies (protocols, services, applications)



-
- TDL
 - Standardised approach improves consistency
 - Tools offer faster development, higher quality, easier maintenance
 - Direct link to TTCN-3
 - Initial run within ITS, expand to other Technical Committees



From 3GPP TS 36.523-1 V10.2.0 (2012-09):

7.2.2.3 UM RLC / Reassembly / 5-bit SN / LI value > PDU size

7.2.2.3.1 Test Purpose (TP)

(1)

```
with { UE in E-UTRA RRC_CONNECTED state }
ensure that {
  when { UE receives a 5 bit SN configured RLC PDU with Length Indicator value larger than RLC PDU
size }
  then { UE discards the RLC PDU }
}
```

7.2.2.3.3.2 Test procedure sequence

Table 7.2.2.3.3.2-1: Main behaviour

| St | Procedure | Message Sequence | | TP | Verdict |
|----|--|------------------|------------------|----|---------|
| | | U - S | Message | | |
| - | EXCEPTION: the behaviour described in table 7.2.2.3.3.2-2 runs in parallel with steps 1 to 5 below. | - | - | - | - |
| 1 | The SS transmits UMD PDU#1 containing first segment of RLC SDU#1. | <-- | UMD PDU#1 (SN=0) | - | - |
| 2 | The SS transmits UMD PDU#2 containing last segment of RLC SDU#1 and first segment of RLC SDU#2. | <-- | UMD PDU#2 (SN=1) | - | - |
| 3 | The SS transmits UMD PDU#3 containing last segment of RLC SDU#2, first segment of RLC SDU#3 and with Length Indicator that points beyond the end of the UMD PDU#3. | <-- | UMD PDU#3 (SN=2) | - | - |
| 4 | The SS transmits UMD PDU#4 containing last segment of RLC SDU#3. | <-- | UMD PDU#4 (SN=3) | - | - |
| 5 | The SS transmits UMD PDU#5 containing RLC SDU#4. | <-- | UMD PDU#5 (SN=4) | - | - |

From 3GPP TS 36.523-1 V10.2.0 (2012-09):

7.2.2.3 UM RLC / Reassembly / 5-bit SN / LI value > PDU size

7.2.2.3.1 Test Purpose (TP)

(1)

```
with { UE in E-UTRA RRC_CONNECTED state }
ensure that {
  when { UE receives a 5 bit SN configured RLC PDU with Length Indicator value larger than RLC PDU
size }
  then { UE discards the RLC PDU }
}
```

7.2.2.3.3.2 Test procedure sequence

Table 7.2.2.3.3.2-1: Main behaviour

| St | Procedure | Message Sequence | | TP | Verdict |
|----|--|------------------|------------------|----|---------|
| | | U - S | Message | | |
| - | EXCEPTION: the behaviour described in table 7.2.2.3.3.2-2 runs in parallel with steps 1 to 5 below. | - | - | - | - |
| 1 | The SS transmits UMD PDU#1 containing first segment of RLC SDU#1. | <-- | UMD PDU#1 (SN=0) | - | - |
| 2 | The SS transmits UMD PDU#2 containing last segment of RLC SDU#1 and first segment of RLC SDU#2. | <-- | UMD PDU#2 (SN=1) | - | - |
| 3 | The SS transmits UMD PDU#3 containing last segment of RLC SDU#2, first segment of RLC SDU#3 and with Length Indicator that points beyond the end of the UMD PDU#3. | <-- | UMD PDU#3 (SN=2) | - | - |
| 4 | The SS transmits UMD PDU#4 containing last segment of RLC SDU#3. | <-- | UMD PDU#4 (SN=3) | - | - |
| 5 | The SS transmits UMD PDU#5 containing RLC SDU#4. | <-- | UMD PDU#5 (SN=4) | - | - |

Table 7.2.2.3.3.2-2: Parallel behaviour

| St | Procedure | Message Sequence | | TP | Verdict |
|----|--|------------------|-------------|----|---------|
| | | U - S | Message | | |
| 1 | The UE transmits RLC SDU#1. | --> | (RLC SDU#1) | - | - |
| 2 | Check: Does the UE transmit RLC SDU#2? | --> | (RLC SDU#2) | 1 | F |
| 3 | Check: Does the UE transmit RLC SDU#3? | --> | (RLC SDU#3) | 1 | F |
| 4 | The UE transmits RLC SDU#4. | --> | (RLC SDU#4) | - | - |

4.5.1 General Capabilities

4.5.1.1 SIP messages longer than 1 500 bytes

| Interoperability Test Description | | |
|-----------------------------------|--|---|
| Identifier: | TD_IMS_MESS_0001 | |
| Summary: | IMS network shall support SIP messages greater than 1 500 bytes | |
| Configuration: | CF_INT_CALL | |
| SUT | IMS_B | |
| References | Test Purpose | Specification Reference |
| | TP_IMS_4002_1 | TS 124 229 [1], clause 4.2A ¶1 |
| Use Case ref.: | UC_05_1 | |
| Pre-test conditions: | <ul style="list-style-type: none"> • HSS of IMS_A and of IMS B is configured according to table 1 • UE_A and UE_B have IP bearers established to their respective IMS networks as per clause 4.2.1 • UE_A and IMS_A configured to use TCP for transport • UE_A is registered in IMS_A using any user identity • UE_B is registered user of IMS_B using any user identity • MESSAGE request and response has to be supported at II-NNI (TS 129 165 [16] see tables 6.1 and 6.3) | |
| Test Sequence: | Step | |
| | 1 | User A sends message to User B with at least 1 500 characters |
| | 2 | Verify that user B receives message from user A |
| Conformance Criteria: | Check | |
| | 1 | TP_IMS_4002_01 in CFW step 4 (MESSAGE) <i>ensure that { when { UE_A sends a MESSAGE to UE_B containing a Message_Body greater than 1 300 bytes } then { IMS_B receives the MESSAGE containing the Message_Body greater than 1 300 bytes } }</i> |

| Step | Direction | | | | | | | | | Message | Comment |
|------|-----------------------|-------------|------------------|-----------------------|-----------------------|------------------|-------------|-----------------------|--|---------|---|
| | U s e r A | U E A | I M S A | I B C F A | I B C F B | I M S B | U E B | U s e r B | | | |
| 1 | | → | | | | | | | | | User A sends an instant message to user B |
| 2 | | | → | | | | | | | MESSAGE | UE_A sends MESSAGE to IMS_A |
| 3 | | | | → | | | | | | MESSAGE | IMS_A sends MESSAGE to IBCE_A |

4.5.1 General Capabilities

4.5.1.1 SIP messages longer than 1 500 bytes

| Interoperability Test Description | | |
|-----------------------------------|---|--------------------------------|
| Identifier: | TD_IMS_MESS_0001 | |
| Summary: | IMS network shall support SIP messages greater than 1 500 bytes | |
| Configuration: | CF_INT_CALL | |
| SUT | IMS_B | |
| References | Test Purpose | Specification Reference |
| | TP_IMS_4002_1 | TS 124 229 [1], clause 4.2A ¶1 |

| Step | Direction | | | | | | | | Message | Comment |
|------|-----------|------|-------|--------|--------|-------|------|--------|---------|--|
| | User A | UE A | IMS A | IBCF A | IBCF B | IMS B | UE B | User B | | |
| 1 | | → | | | | | | | | User A sends an instant message to user B |
| 2 | | | → | | | | | | MESSAGE | UE_A sends MESSAGE to IMS_A |
| 3 | | | | → | | | | | MESSAGE | IMS_A sends MESSAGE to IBCF_A |
| 4 | | | | | → | | | | MESSAGE | IBCF_A sends MESSAGE to IBCF_B |
| 5 | | | | | | → | | | MESSAGE | IBCF_B sends MESSAGE to IMS_B with via header indicating TCP |
| 6 | | | | | | | → | | MESSAGE | IMS_B sends MESSAGE to UE_B |
| 7 | | | | | | | | → | | User B is informed about the instant message |
| 8 | | | | | | | ← | | 200 OK | UE_B sends 200 OK to IMS_B |
| 9 | | | | | | | ← | | 200 OK | IMS_B sends 200 OK to IBCF_B |
| 10 | | | | | ← | | | | 200 OK | IBCF_B sends 200 OK to IBCF_A |
| 11 | | | | ← | | | | | 200 OK | IBCF_A sends 200 OK to IMS_A |
| 12 | | | ← | | | | | | 200 OK | IMS_A sends 200 OK to UE_A |
| 13 | | ← | | | | | | | | Optional: User A is presented a delivery report |



From ETSI TS 102 868-2 V1.1.1 (2011-03):

16

ETSI TS 102 868-2 V1.1.1 (2011-03)

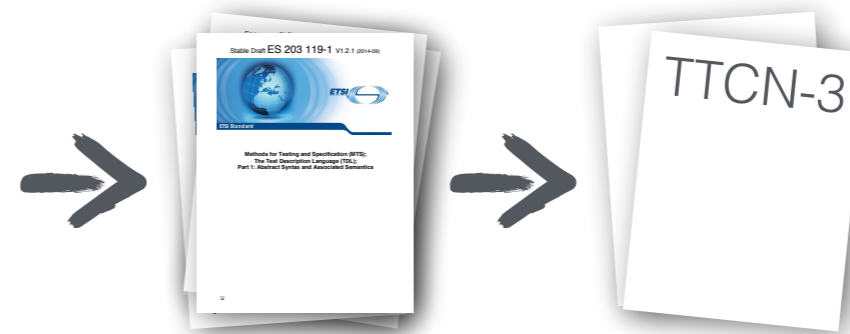
| | |
|---|--|
| TP Id | TP/CAM/INA/DOP/BV/02 |
| Test objective | Checks that CAM message includes DoorOpen information 30s after closed |
| Reference | TS 102 637-2 [1], clauses 7.1 and 7.2 |
| PICS Selection | PICS_PUBTRANSVEH |
| Initial conditions | |
| with { the IUT being in the "initial state" and the IUT having sent a valid CAM message containing DoorOpen TaggedValue } | |
| Expected behaviour | |
| ensure that { when { the door is closed } then { the IUT sends CAM messages containing DoorOpen TaggedValue during the 30s following the door closing event } } | |

NDI assignments Semi-Persistent either
presence during TTI
configured deliver DL-SCH
relevant each provide UE process
first toggled UE's indicated value
Persistent CRNTI Semi monitors particular same



- Context
 - TDL in MBT: Keyword driven UI testing
 - Create behavioural model of the SUT using symbolic action descriptions
 - define keywords once
 - map abstract keyword definitions to keyword implementations in execution language
 - Generate abstract test sequences by means of MBT
 - Convert abstract test sequences to a test execution language

NDI assignments Semi-Persistent either
presence during TTI
configured deliver DL-SCH
relevant each provide UE process
first toggled UE's indicated value
Persistent CRNTI Semi monitors particular same



- Challenges

- Generated test sequences

- proprietary format - not accessible, tool-specific integrations to requirements management, test planning
- straight to executable code - loss of meta-data, difficult parameterisation

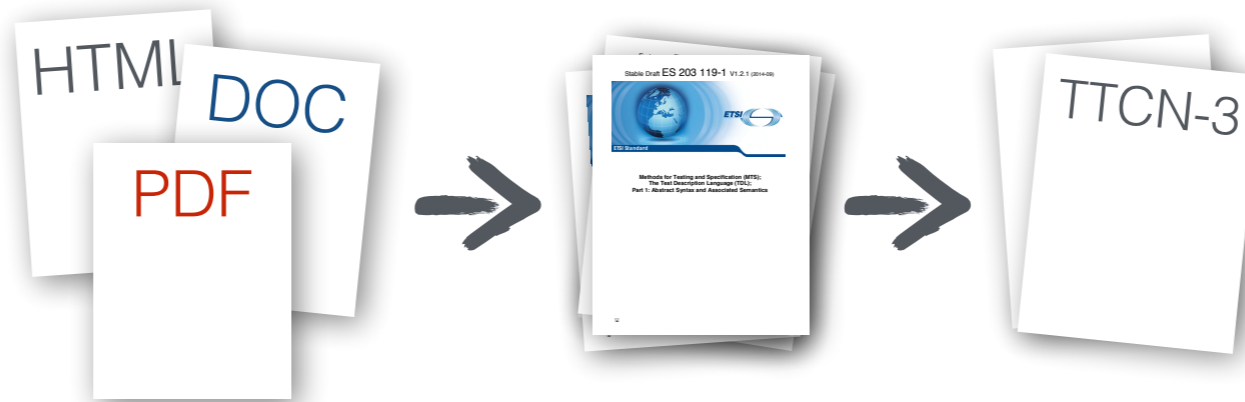
- Mapping between abstract (symbolic) and real test system interface

- implicit - error-prone
- implemented in test execution language - additional overhead, language limitations

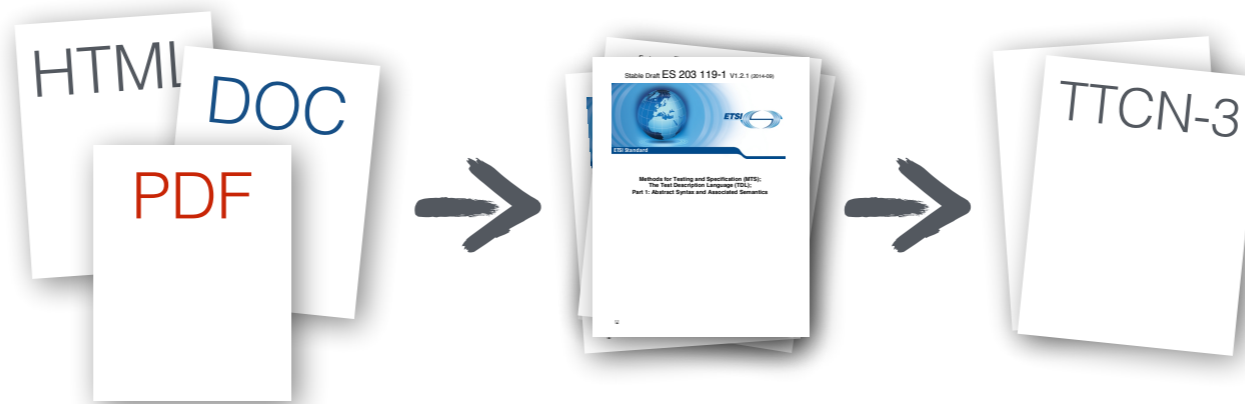
NDI assignments Semi-Persistent either
presence during TTI
configured deliver DL-SCH
relevant each provide UE process
first toggled UE's indicated value
Persistent CRNTI Semi monitors particular same



-
- TDL
 - Interoperability with requirements management by explicit test objectives
 - Parameterisation of test descriptions and symbolic data representations
 - Explicit data mapping to underlying data system of execution language
 - Advantages over alternatives
 - Less ambiguity, testing specific (e.g. break, stop, default concepts)
 - Use of variables and data in expressions and interactions

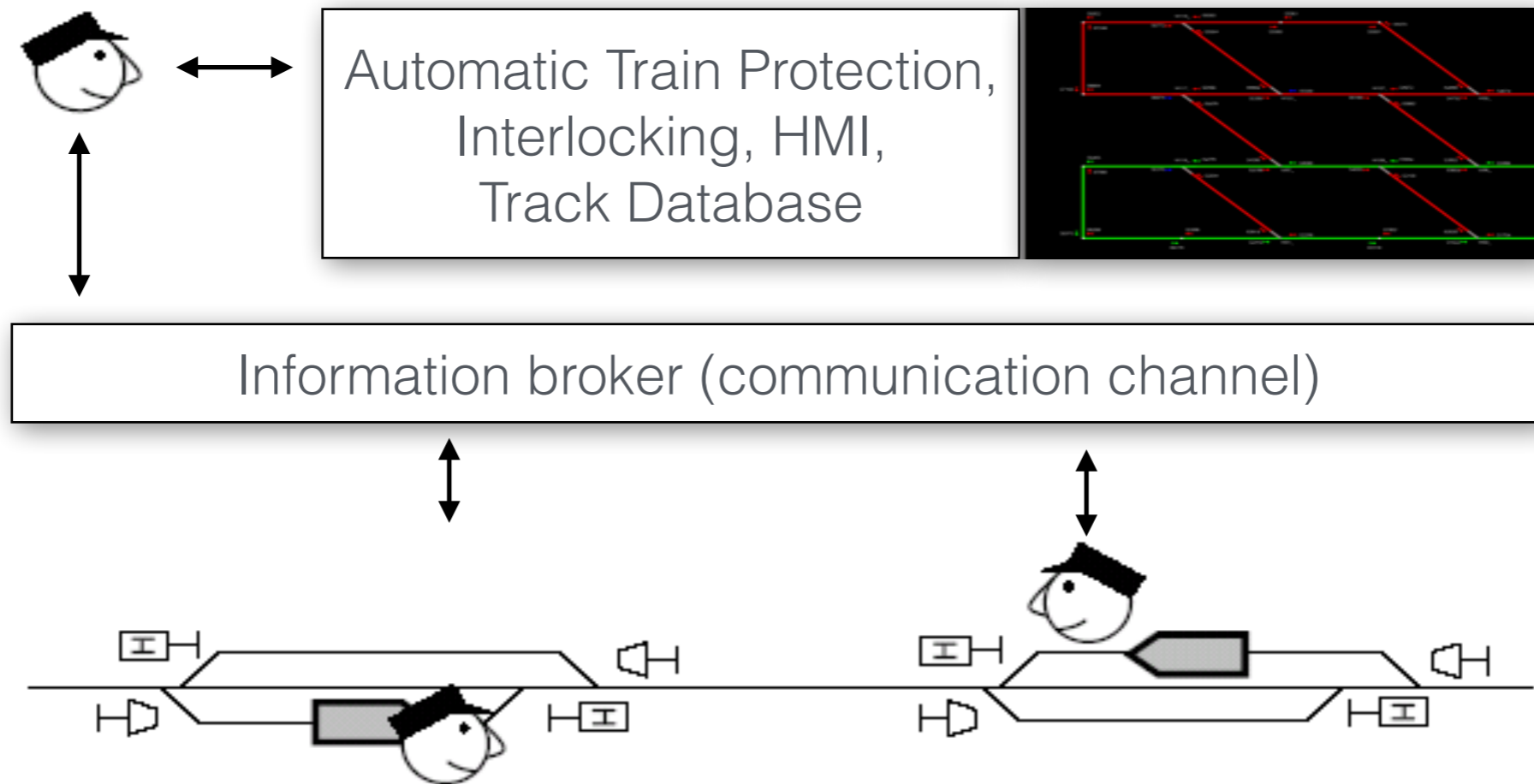
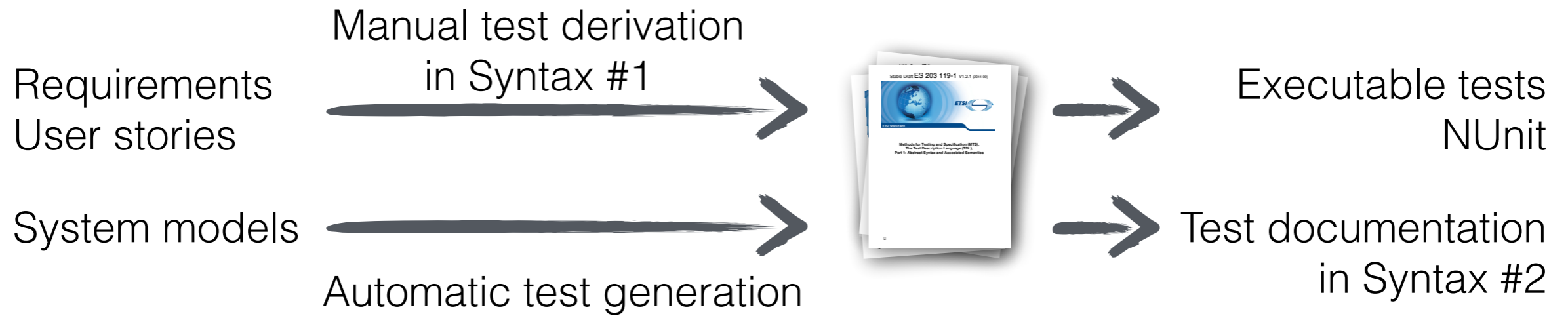


-
- Context
 - Test automation tools for performance and load tests
 - Challenges
 - Textual test specifications with sequence diagram-like examples (or using a different graphical notation)
 - Manual derivation of TTCN-3 code and configuration settings
 - Too wide a gap between input and output!



-
- TDL
 - Raises the abstraction level of the test description
 - multiple levels of test specification (from system to implementation), iterative and agile development
 - Concentrate on the problems themselves rather than programming details
 - Application
 - Visualisation of test case behaviour
 - Automatic generation of TTCN-3 code from TDL test descriptions

Where does TDL fit in?





- Context
 - Testing communication between independent rail sub-systems
- Challenges
 - High-level concurrency and non-determinism
 - Multiple aspects over the whole system - safety, real-time, functionality
 - Different development techniques for different components

Requirements
User stories



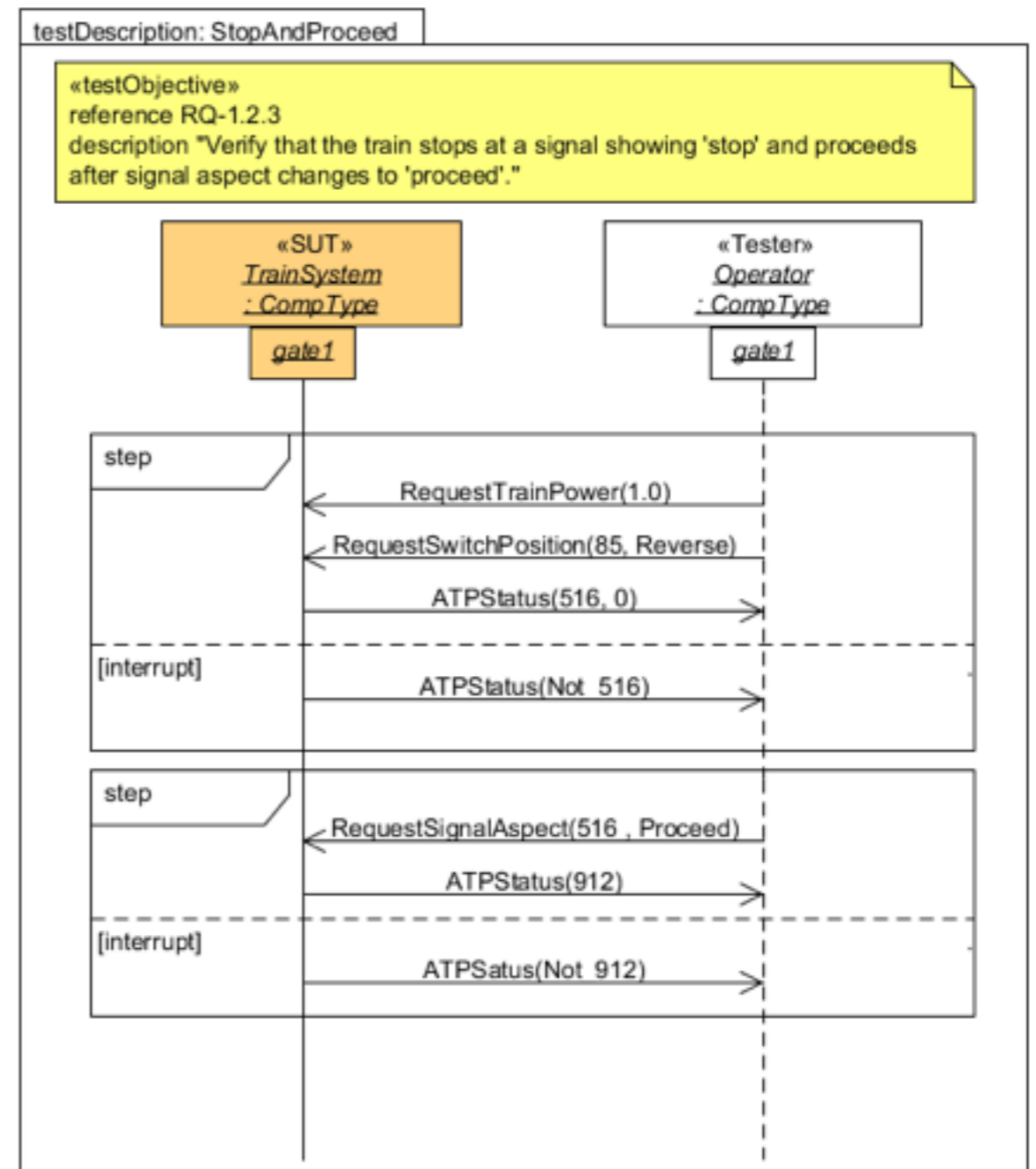
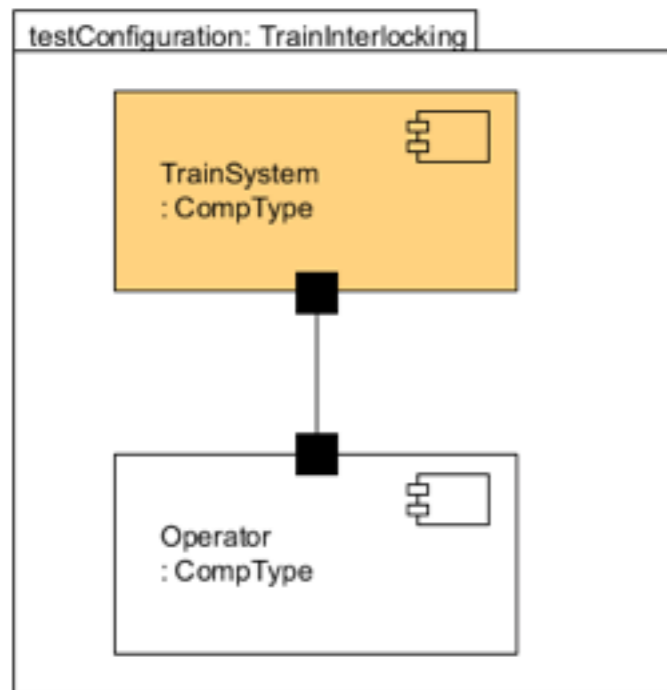
System models



Executable tests
NUnit

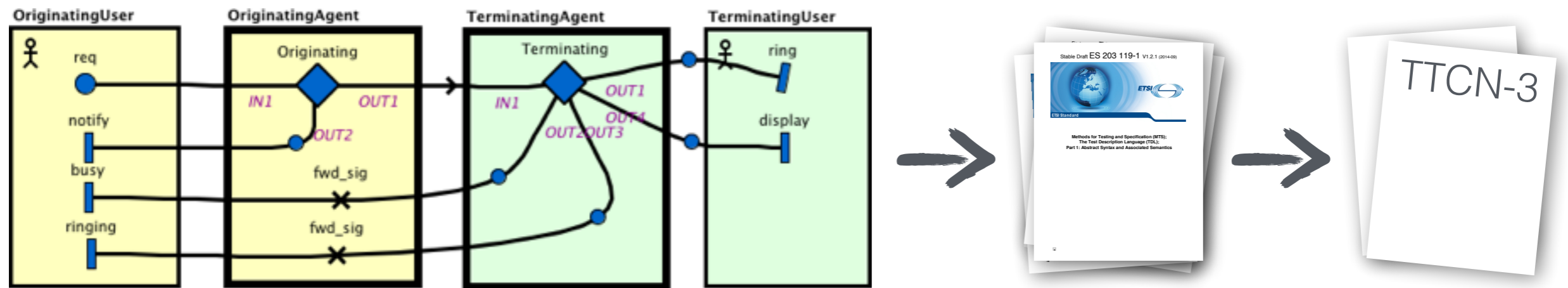


Test documentation
in Syntax #2

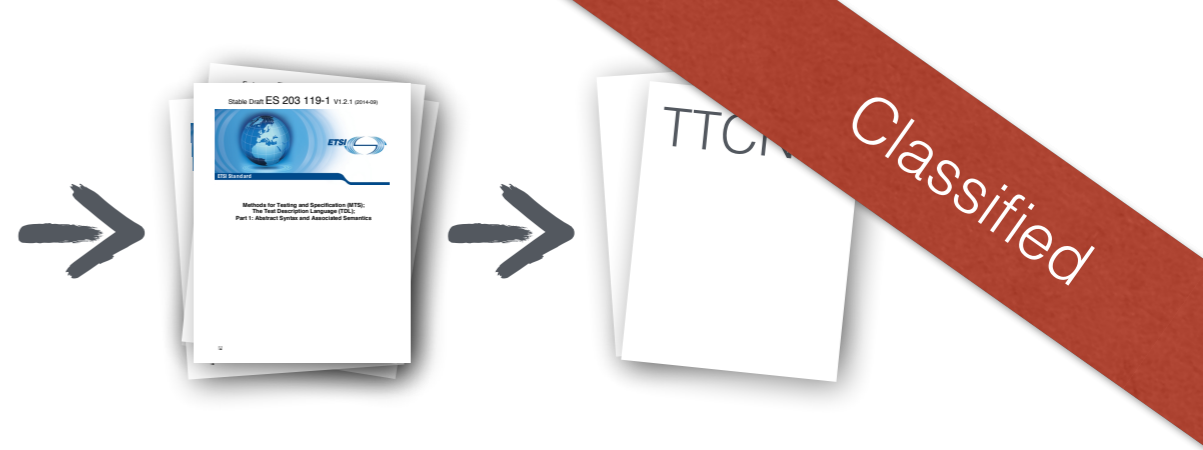
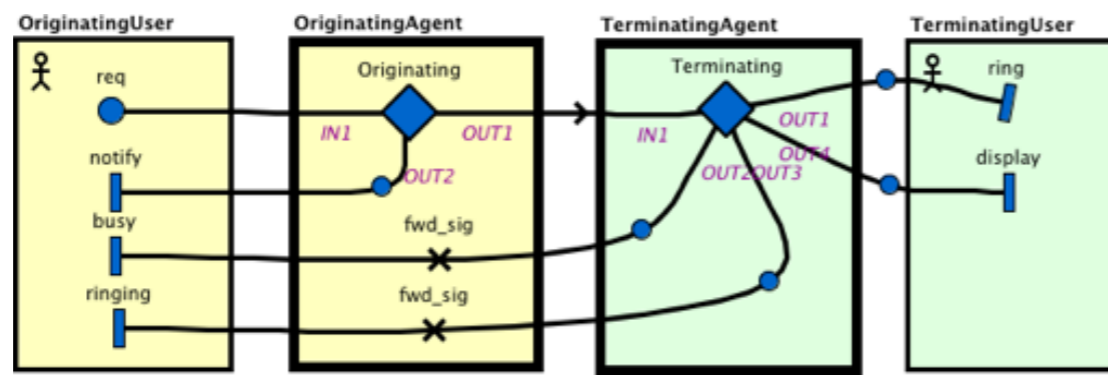


- More on this tomorrow morning!

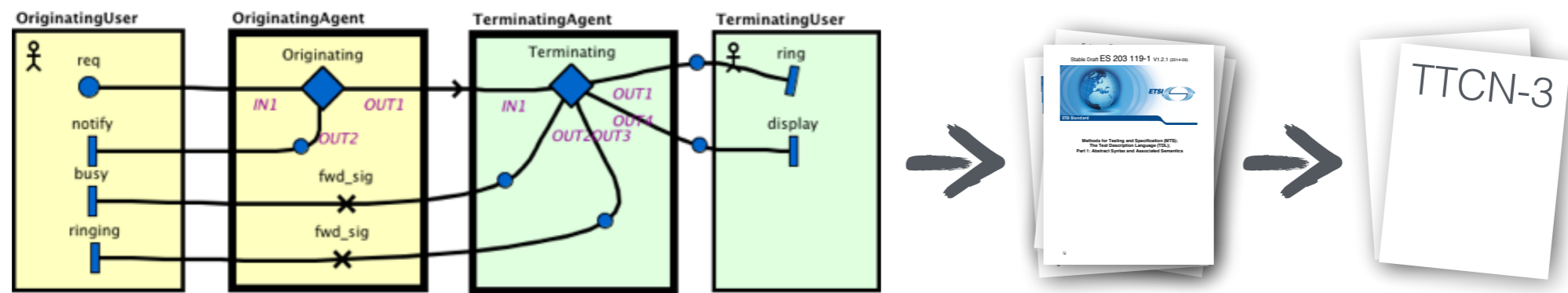
Where does TDL fit in?



- User Requirements Notation (URN)
 - Elicitation, analysis, specification, and validation of requirements
 - Complementary views - goals (GRL) and **scenarios** (UCM)
 - ITU-T Recommendation Z.151 (10/12)



- Context
 - Test systems for cockpit systems and avionics solutions
 - Alternative means for
 - standards-based and model-based test generation and test automation
 - replace proprietary solutions
 - Transformation from high-level requirements and scenarios in UCM to TDL
 - Transformation from TDL to TTCN-3

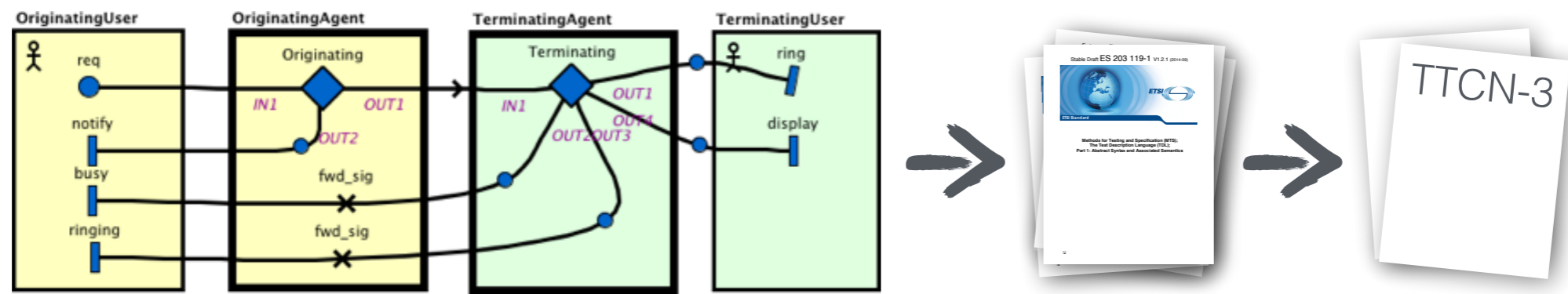


- Goals

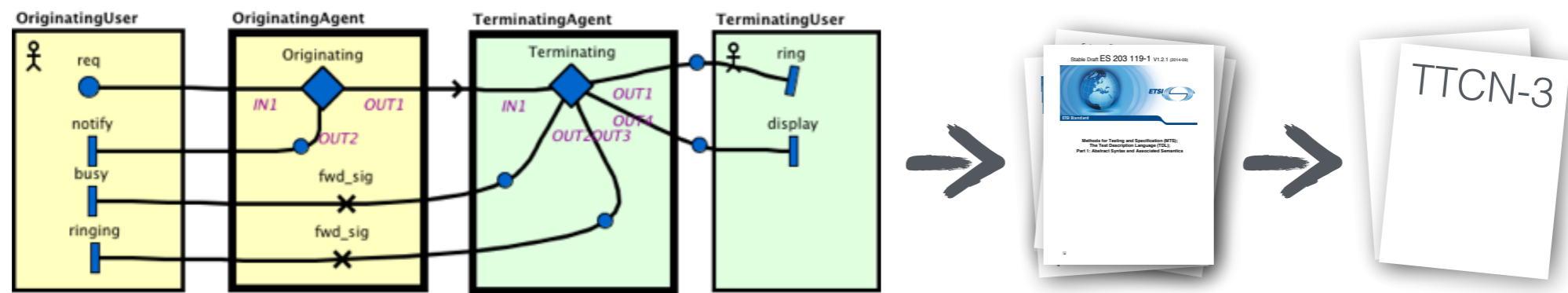
- URN/UCM suitable starting point for modelling requirements?
- TDL appropriate intermediate representation or even starting point?
- TTCN-3 viable technology in the avionics industry?

- Stakeholders

- Research, industry, agencies
- Test engineers, test developers, test managers, analysts and modellers

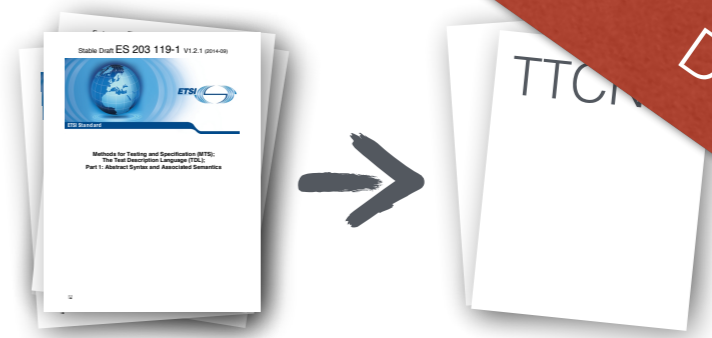
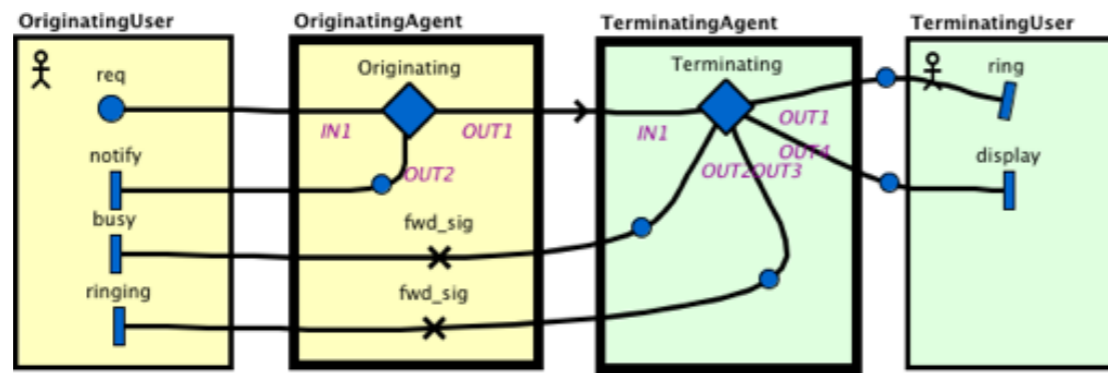


- Motivation
 - Tree-like structure of tests
 - TDL/TTCN-3 reflect this, existing transformations from UCM to e.g. MSC/UML do not
 - UCMs do not include much data information
 - appropriate stage to add data for executable test cases (UCM/TDL/TTCN-3/other)?
 - Peculiarities of the domain
 - support testing in an environment where an unknown number of sensors can send alarms (over unreliable channels) and messages in parallel



- TDL
 - Close enough to UCM for test generation
 - Close enough to TTCN-3 for generating executable test cases and test configurations
- Prototype
 - Part of jUCMNav (v6.0.0), developed at EECS (University of Ottawa)
 - Support for sequence and concurrent events (no alternatives yet)

<http://jucmnav.softwareengineering.ca/ucm/bin/view/ProjetSEG/ExportTdlUserGuide>



Demo Time

The screenshot shows the Eclipse IDE with the following components:

- UML Diagram:** A sequence diagram showing interactions between OriginatingUser, OriginatingAgent, TerminatingAgent, and TerminatingUser.
- Types and Instances:** A tree view showing the class hierarchy and instances of the system components.
- Test Description Code:** A snippet of TTCN code defining component types and a test description for the system.
- Sequence Diagram:** A detailed sequence diagram showing the execution of the test case, including messages like start, initFeatures, checkOCS, and checkTime.

<http://jucmnav.softwareengineering.ca/ucm/bin/view/ProjetSEG/ExportTdlUserGuide>

Concluding remarks

- New technology, growing rapidly
- Commercial tool support not yet available
- Custom tools can be put together in a matter of hours
 - basic yet capable
 - make early adoption easier
- Advanced solutions still require additional effort
 - not immediately necessary to get started with using TDL

Summary

What is TDL?

- Test Description Language
 - Design, documentation, and representation of formal test descriptions
 - Scenario-based approach
- Standardised at ETSI by TC MTS
- Initial work within STF 454
- Current efforts within STF 476

© ETSI 2014. All rights reserved

ETSI ES 203 119 V1.1.1 (2014-04)

Where does TDL fit in?

Keyword-Driven Testing
MBT Representation
Generation Standards Rail
Documentation Visualisation
Interoperability ITS
Conformance

© ETSI 2014. All rights reserved

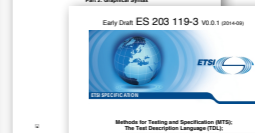
33

What is new in TDL?

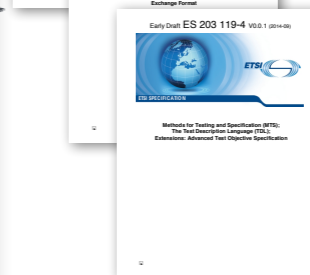
Work in Progress



Part 2:
Graphical
Syntax



Part 3:
Exchange
Format



Part 4:
Advanced
Test Objective
Specification

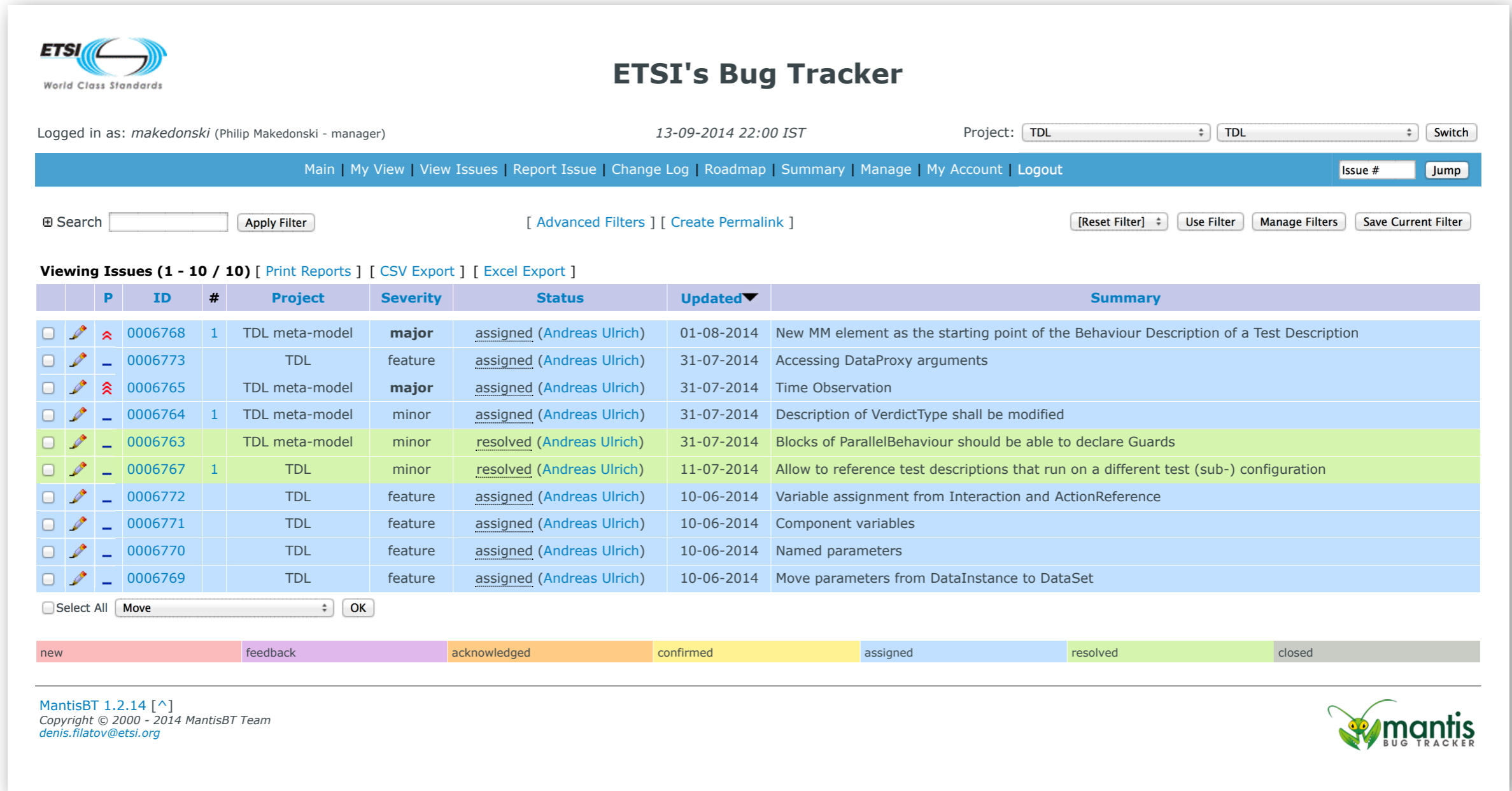
rights reserved

25

Where would you consider using TDL?

STF 476 - http://portal.etsi.org/STFs/STF_HomePages/STF476/STF476.asp

What would you want to see in TDL?



ETSI's Bug Tracker

Logged in as: *makedonski* (Philip Makedonski - manager) 13-09-2014 22:00 IST Project: TDL TDL

Main | My View | View Issues | Report Issue | Change Log | Roadmap | Summary | Manage | My Account | Logout Issue #

Search [[Advanced Filters](#)] [[Create Permalink](#)]


Viewing Issues (1 - 10 / 10) [[Print Reports](#)] [[CSV Export](#)] [[Excel Export](#)]

| | P | ID | # | Project | Severity | Status | Updated | Summary |
|--------------------------|---|---------|---|----------------|----------|---------------------------|------------|---|
| <input type="checkbox"/> | | 0006768 | 1 | TDL meta-model | major | assigned (Andreas Ulrich) | 01-08-2014 | New MM element as the starting point of the Behaviour Description of a Test Description |
| <input type="checkbox"/> | | 0006773 | | TDL | feature | assigned (Andreas Ulrich) | 31-07-2014 | Accessing DataProxy arguments |
| <input type="checkbox"/> | | 0006765 | | TDL meta-model | major | assigned (Andreas Ulrich) | 31-07-2014 | Time Observation |
| <input type="checkbox"/> | | 0006764 | 1 | TDL meta-model | minor | assigned (Andreas Ulrich) | 31-07-2014 | Description of VerdictType shall be modified |
| <input type="checkbox"/> | | 0006763 | | TDL meta-model | minor | resolved (Andreas Ulrich) | 31-07-2014 | Blocks of ParallelBehaviour should be able to declare Guards |
| <input type="checkbox"/> | | 0006767 | 1 | TDL | minor | resolved (Andreas Ulrich) | 11-07-2014 | Allow to reference test descriptions that run on a different test (sub-) configuration |
| <input type="checkbox"/> | | 0006772 | | TDL | feature | assigned (Andreas Ulrich) | 10-06-2014 | Variable assignment from Interaction and ActionReference |
| <input type="checkbox"/> | | 0006771 | | TDL | feature | assigned (Andreas Ulrich) | 10-06-2014 | Component variables |
| <input type="checkbox"/> | | 0006770 | | TDL | feature | assigned (Andreas Ulrich) | 10-06-2014 | Named parameters |
| <input type="checkbox"/> | | 0006769 | | TDL | feature | assigned (Andreas Ulrich) | 10-06-2014 | Move parameters from DataInstance to DataSet |

Select All Move

new feedback acknowledged confirmed assigned resolved closed

MantisBT 1.2.14 [^]
Copyright © 2000 - 2014 MantisBT Team
denis.filatov@etsi.org



STF 476 - http://portal.etsi.org/STFs/STF_HomePages/STF476/STF476.asp

Bringing TDL to Users: A Hands-on Tutorial

Philip Makedonski, Gusztav Adamis, Martti Käärik,
Andreas Ulrich, Marc-Florian Wendland, Anthony Wiles

STF 476 - http://portal.etsi.org/STFs/STF_HomePages/STF476/STF476.asp