ETSI TECHNICAL REPORT

ETR 346

December 1996

Source: ETSI TC-RES

Reference: DTR/RES-06013-1

ICS: 33.020

Key words: Testing, TTCN, abstract test suite, validation

Radio Equipment and Systems (RES); Trans-European Trunked Radio (TETRA); Air Interface (AI) layer 2 and layer 3 protocol validation; Part 1: Validation of test suites for Voice plus Data (V+D)

ETSI

European Telecommunications Standards Institute

ETSI Secretariat

Postal address: F-06921 Sophia Antipolis CEDEX - FRANCE
Office address: 650 Route des Lucioles - Sophia Antipolis - Valbonne - FRANCE
X.400: c=fr, a=atlas, p=etsi, s=secretariat - **Internet:** secretariat@etsi.fr

Tel.: +33 4 92 94 42 00 - Fax: +33 4 93 65 47 16

Copyright Notification: No part may be reproduced except as authorized by written permission. The copyright and the foregoing restriction extend to reproduction in all media.

© European Telecommunications Standards Institute 1996. All rights reserved.

Whilst every care has been taken in the preparation and publication of this document, errors in content, typographical or otherwise, may occur. If you have comments concerning its accuracy, please write to "ETSI Editing and Committee Support Dept." at the address shown on the title page.

Contents

Blank page

Foreword

This ETSI Technical Report (ETR) has been produced by the Radio Equipment and Systems (RES) Technical Committee of the European Telecommunications Standards Institute (ETSI).

ETRs are informative documents resulting from ETSI studies which are not appropriate for European Telecommunication Standard (ETS) or Interim European Telecommunication Standard (I-ETS) status. An ETR may be used to publish material which is either of an informative nature, relating to the use or the application of ETSs or I-ETSs, or which is immature and not yet suitable for formal adoption as an ETS or an I-ETS.

Blank page

1 Scope

This ETSI Technical Report (ETR) defines the methods, procedures, and validation purposes used for the formal validation of the Tree and Tabular Combined Notation (TTCN) conformance test suites for TETRA Voice and Data (V+D) Air Interface (AI) and documents the results of the validation.

2 References

For the purposes of this ETR, the following references apply:

[1] ETS 300 394-2-1: "Radio Equipment and Systems (RES); Trans-European Trunked Radio (TETRA); Conformance testing specification, Part 2: Protocol testing specification for Voice plus Data (V+D); Part 2-1: Test suites structure and test purposes".

[2] ETS 300 394-2-2: "Radio Equipment and Systems (RES); Trans-European Trunked Radio (TETRA); Conformance testing specification, Part 2: Protocol testing specification for Voice plus Data (V+D); Part 2-2: Abstract Test Suite (ATS) for Network (NWK) layer".

[3] ETS 300 394-2-3: "Radio Equipment and Systems (RES); Trans-European Trunked Radio (TETRA); Conformance testing specification, Part 2: Protocol testing specification for Voice plus Data (V+D); Part 2-3: Abstract Test Suite (ATS) for Logical Link Control (LLC)".

[4] ETS 300 394-2-4: "Radio Equipment and Systems (RES); Trans-European Trunked Radio (TETRA); Conformance testing specification, Part 2: Protocol testing specification for Voice plus Data (V+D); Part 2-4: Abstract Test Suite (ATS) for Medium Access Control (MAC)".

[5] ETS 300 392-2: "Radio Equipment and Systems (RES); Trans-European Trunked Radio (TETRA); Voice plus Data (V+D); Part 2: Air Interface (AI)".

[6] ETR 293-1: "Radio Equipment and Systems (RES); Trans-European Trunked Radio (TETRA); Air Interface (AI) layer 2 and 3 protocol validation; Part 1: Validation of SDL models for Voice plus Data (V+D)".

[7] ISO/IEC 9646-3 (1991): "Information technology - Open Systems Interconnection - Conformance testing methodology and framework - Part 3: The tree and tabular combined notation". (See also CCITT Recommendation X.292 (1992)).

[8] ITU-T Recommendation Z.105 (1995): "SDL combined with ASN.1".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of this ETR, the following definitions apply:

external validation: Validation of TTCN test suite properties except for those related to the TTCN language definition.

internal validation: Checking the correctness of a TTCN test suite according to the rules of TTCN as defined in ISO 9646, part 3 [7].

3.2 Abbreviations

For the purposes of this ETR, the following abbreviations apply:

ASN.1	Abstract Syntax Notation number One
ATS	Abstract Test Suite
CMCE	Circuit Mode Control Entity
ExTS	Executable Test Suite
LLC	Logical Link Control
MAC	Medium Access Control
MLE	Mobile Link Entity
MM	Mobility Management
NWK	Network
PDU	Protocol Data Unit
SCLNP	Specific Connectionless Network Protocol
SDL	Specification and Description Language
TTCN	Tree and Tabular Combined Notation

4 Introduction

This ETR documents the validation of the TETRA V+D protocol conformance test suites for Network (NWK) layer, ETS 300 394-2-2 [2], Logical Link Control (LLC), ETS 300 394-2-3 [3], and the Medium Access Control (MAC), ETS 300 394-2-4 [4].

For the test purposes and test suites structure refer to ETS 300 394-2-1 [1] and for the protocol specification to ETS 300 392-2 [5].

The purpose of the test suite validation is to ensure the quality of the Abstract Test Suites (ATS) which forms the basis for test suite implementations.

This test suite validation has been focused on the use of formal validation methods that are supported by the tools available at ETSI. The ETR also provides a brief description of other aspects of test suite validation, which were considered as potential validation methods, but were not applied in the validation documented in this ETR.

5 General

5.1 Test suite validation principles

The Abstract Test Suites (ATSs) for the protocols in TETRA are specifications in the process of defining the conformance criterion for TETRA products. An ATS specification serves both as an operational definition of the test purposes and as a basis for development of Executable Test Suites (ExTSs). The checking of these different properties of an ATS is the purpose of test suite validation.

In figure 1 validation properties for an ATS are shown. The term "Protocol requirements" used in figure 1 refers to all requirements defined in either of the protocol description, protocol model or protocol test purpose documents. Each of the highlighted validation properties are further explained in subclause 5.2.

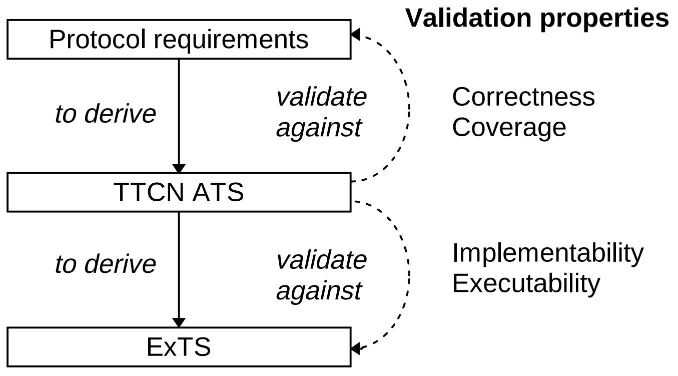

Figure 1: Test suite validation properties

In addition to the validation properties shown in figure 1, a TTCN test suite should also be checked against the rules of TTCN, defined in ISO 9646, part 3 [7]. This type of validation is denoted internal validation, while the validation of all other properties is denoted external validation.

Methods for TTCN test suite validation can be distinguished on whether they are based on formal models and formally defined relations between such models or on informal methods. Informal validation methods often are based on inspection techniques, which can be automated and supported by tools only to a limited extent.

Validation methods based on formal models are a prerequisite for tool supported and automatic check of validation properties. Formal methods, in principle, allow exhaustive checking of the validation properties.

For the TETRA protocol test suites both informal and formal validation methods in principle could be applied as indicated in figure 2.

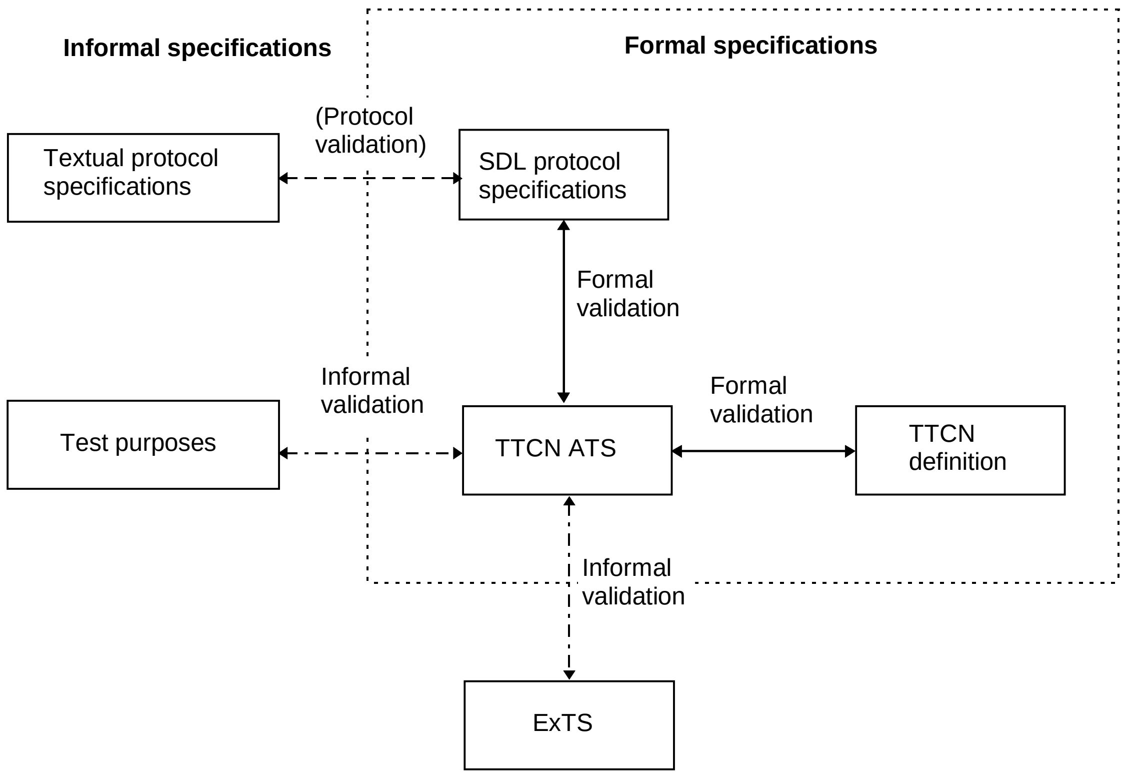

Figure 2: Formal and informal validation possibilities for TETRA protocol ATSs

Protocol validation is indicated in figure 2 but is not part of test suite validation, rather it can be seen as a prerequisite to ensure the value of the test suite validation. Using protocol validation it is possible to check that a formal model consistently specifies the requirements of the textual protocol specifications. In the case of TETRA, a validated model of the V+D protocols exists. The model is specified using formal Specification and Description Language (SDL). For further information on the protocol validation refer to ETR 293-1 [6].

In figure 2 it is also indicated that the validation of the ATS with respect to the ExTS can only be performed using informal validation methods. In principle this can be done using formal validation as the ETSI tools for TTCN test case specification includes a TTCN compiler. However, as the implementation of an ATS is outside the scope of standardization, only informal validation methods are supposed to be applied for validation properties like implementation and execution.

5.2 Validation properties

As the validation properties are open for different interpretations they shall be further refined to be supported by systematic methods. The internal TTCN test suite validation property generally covers validation of an ATS with respect to the rules of the TTCN definition in ISO 9646, part 3 [7].

The internal validation property comprises of:

- checking the syntactical correctness of a TTCN test suite according to the TTCN syntax rules;

- checking that the test suite comply with the static semantics rules for TTCN, e.g. that all test suite and test case variables used have been declared and that Protocol Data Units (PDUs) and constraints are of complying types;

- checking the dynamic behaviour of the test cases by simulation against e.g. an executable validation model or a prototype implementation.

The syntax and static semantics analysis of an ATS is fully supported by the TTCN tool available at ETSI.

The simulation of TTCN test cases is also supported by the TTCN tool, but the actual check of the simulation results should be done by inspection.

Another way to ensure the correctness of the dynamic behaviour of the test cases is to use a totally opposite approach to simulation. Using this method, the dynamic behaviour of the test cases are semi-automatically derived from a formal protocol model that is assumed to correctly reflect the protocol requirements, i.e. has been validated. This method is also possible within the tools available at ETSI.

All other validation properties belong to the category of external validation properties. Two main external validation properties can be identified: correctness property and coverage property.

Basically, the correctness validation checks that all test cases define a test for a requirement specified in the protocol specification. The correctness property can be checked using the ETSI tools which allow a TTCN test case to be simulated against an SDL-protocol model. Using the simulation, it is a prerequisite that all the requirements that are tested in the test cases are also present in the SDL-model used in the simulation.

The coverage property validates that the ATS includes test cases for the selected requirements of the protocol description. For a complete conformance test suite this means to check that test cases are specified for all requirements in the protocol description. In principle this validation can be performed using an SDL protocol model and an ATS alone. For all conformance test suites the test purpose document is needed, as even complete conformance test suites cover only a subset of the protocol requirements. So coverage validation is performed taking the test purposes into account. Even if the validation of the test cases against an SDL protocol model can be supported by the ETSI tools, only informal methods can be used to check that test cases are specified for all essential requirements.

The methods are also applied in a certain order to ensure correct validation results. A feasible order of application is illustrated in figure 3.

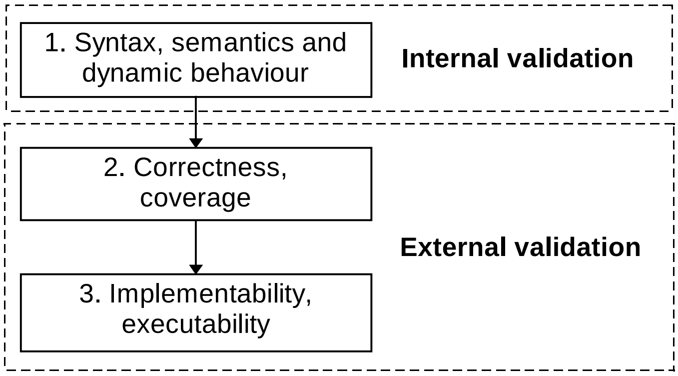

Figure 3: Order of different validation methods

The presented order helps to trace any errors to their origin. Until the internal validation has successfully been performed, it might not be possible at all to perform formal correctness or coverage validation or any other kind of external validation.

5.3 Different validation approaches

Different approaches can be defined to check properties that may covered by formal test suite validation methods. With respect to the available tools for the validation, the selection of the approach applied for the TETRA V+D protocol test suites was made among the following alternatives:

5.3.1 Approach 1

The validation is performed in the following steps:

1) For each test purpose a corresponding TTCN test case is created using the ETSI SDL-tool to derive the TTCN test case from the SDL specification.

2) Minor and systematic modifications are made to the derived TTCN declarations and constraints in order to make them comply with the ASN.1 rules.

3) The TTCN test cases are simulated against the same SDL-model from which they were initially derived from.

The major disadvantage of this approach is that the functional context of the test cases is the same as the one of the SDL model, presuming that the tools work correctly. Hence no functional errors can be found in the test cases performing this validation. The approach will only identify errors in the tools used, as inconsistencies discovered during simulation will be related either to errors in the TTCN test case derivation tool or the TTCN-SDL simulation tool. It will, however, be possible to check if the modifications made to TTCN data types and constraints in the TTCN test suites are consistent with those of the SDL model of the protocol. If the test cases were manually created then this simulation approach was valuable to the validation process.

In addition there would be a significant amount of work in creating the virtual test environment before the simulator could be used, since the mapping between the executable validation model and the ExTS will require some manual C-code writing. It may not even be possible to perform without modifying the tools and upgrading the ETSI licenses of the tools which may not be feasible.

Nevertheless, there are two significant advantages of this approach:

1) the TTCN validation can easily be repeated for the finalized test cases in case of modifications;

2) the value of the validation is better, as it is performed to the final TTCN test cases, instead of during the test case writing, as described in the second approach.

5.3.2 Approach 2

In this approach the TTCN test cases are created using two different methods depending on the suitability of the TTCN test case derivation tool for the creation of each test case.

Method 1:

- Some test cases are derived from the SDL specification by using the TTCN test case derivation tool. For these test cases validation can be seen as being performed while the test cases are created as the test cases are derived from already validated SDL model. Again, this validation is based on the assumption that the tool correctly derives the TTCN test cases from the SDL specification.

Method 2:

- For those test purposes where the derivation tool is not suitable the corresponding test cases are defined manually. These TTCN test cases are executed, i.e. validated, against the SDL-model with the TTCN-SDL simulator.

When the test cases are made the following validation is performed:

- the necessary changes are made to the TTCN declarations and constraints as automatically as possible e.g. using scripts on the TTCN.MP file to do the changes; and

- the updated TTCN test cases are then checked with the syntax and static semantics checker.

However after this last step it is not feasible to formally validate the test case dynamic behaviour against the SDL model as already noted in approach 1.

5.3.3 Approach 3

This approach emphasizes validation of the implementation and execution aspects of the TTCN test suites. The ATSs are given to one or more test laboratories before publication, which then are supposed to implement and execute the test suites against real protocol implementations to check if the test cases are correct and executable. For ETSI this approach may be the most efficient way to perform comprehensive test suite validation.

There are anyhow some significant problems using this approach:

- The test cases should still be syntactically and semantically correct in order to reduce the amount of work for the test laboratories. Then such errors in the test cases should be removed by ETSI, before delivered to any test laboratory for implementation on a real test system.

- This kind of validation can not be performed before both a test system and a test equipment are available, which could lead to an unnecessary delay in products implementing the standards to reach the market.

- The test laboratories might only constructively be involved in the test suite validation if ETSI performs some parts of the validation before starting the implementation and execution validation in co-operation with test laboratories.

This is the most comprehensive kind of validation, but for the reasons mentioned and because it is not within the scope of the validation described in this report, it is not further considered.

However, the idea to release the TTCN test suites to interested test laboratories and companies, as soon as possible after the ETSI tool-supported validation, should still be followed. This will allow the TTCN test suites to become stable and useful conformance testing standards more rapidly.

6 Validation performed

6.1 Validation process

The second approach was selected to be used for the validation of the TTCN test suites. This selection was based on the scope defined for the validation, evaluation of the tools available at ETSI and the available resources.

6.2 Validation results

All ATSs were internally validated using the ITEX syntax and static semantics checker.

For all the three ATSs the SDL models have been used as input to generate the ASN.1 data-types using the SDT-TTCN Link tool as the derivation tool.

The SDL protocol models have also been used for the derivation of the TTCN test cases for the Mobility Management (MM) and Circuit Mode Control Entity (CMCE) using the SDT-TTCN Link tool. For the other two protocols of the NWK test suite, Specific Connectionless Network Protocol (SCLNP) and Mobile Link Entity (MLE), the TTCN test cases have been manually derived from the test purposes. Due to the size and complexity of the SDL protocol model needed for the MLE to semi-automatically derive the test cases, it was not feasible to use the SDT-TTCN Link tool.

For the LLC and MAC test suites, the SDT-TTCN Link tool was used only to derive the ASN.1 data type definitions, which were afterwards slightly modified. For the LLC, it was not possible to use the tool to generate the TTCN test cases due to state space size problems similar to the MLE protocol SDL model. For the MAC test suite, it was considered not feasible to adapt the existing SDL-model for the use with the SDT-TTCN Link tool due to reasonably small amount of the test cases to be specified.

The possibilities of using the ITEX simulator in checking the dynamic behaviour of the manually created test cases against the SDL model were studied more comprehensively during the progress of the validation with unfortunately negative results.

During the validation process an intensive liaison was achieved with the tool manufacturer, and all the problems encountered were reported, which will hopefully help to further improve the tools used.

Even if the validation appeared not to be comprehensive regarding all the properties set for the validation using the selected approach, the advantage of using the syntax and static semantics checking, automatic derivation of TTCN declarations, and partial use of the TTCN semi-automatic test case generation is expected to have significantly improved the quality of the conformance test suites for TETRA V+D.

7 Summary

Using the SDT version 3.02 and ITEX version 3.02 the following formal validation was performed for the TTCN test suites:

- internal validation of the TTCN test suites has been performed using the TTCN syntax and static semantic analyser;

- all ASN.1 data declarations were generated using the SDT-TTCN Link tool;

- for MM and CMCE, the TTCN test case dynamic behaviour parts were derived using the SDT-TTCN Link tool.

During validation the following main problems with the tool-supported methods were encountered:

- The SDL data types used are not always possible to be directly converted into ASN.1 data types, e.g. naming conventions in SDL and ASN.1 are different. This should considered already during SDL modelling. Using ASN.1 according to the ITU-T Recommendation Z.105 [8] for the data type modelling in SDL could have solved this problem. The combined use of SDL and ASN.1, however, was not supported by the version of the SDL tool used for the SDL modelling, SDT version 3.02.

- TTCN test case derivation using the SDT-TTCN Link tool was not initially possible using the SDL models due to state space explosion. So the SDL models had to be modified by excluding all parts that were not directly involved in the test cases currently being derived. The complexity of adapted MLE and LLC SDL models exceeded the capabilities of the SDT-TTCN Link tool. For MAC it was not feasible to perform the needed modifications to allow the SDL model to be used for tool supported test case specification.

Based on the experience gained from the test suite specification and validation, the following recommendations are made:

- SDL with ASN.1 data types should be used if possible. If not, then during the specification of the SDL model, also the naming conventions of ASN.1 should be taken into consideration.

- The SDT-TTCN Link tool in the current version only allows rather limited SDL specifications to be used as a basis for TTCN test case specification. This could be taken into account in the SDL modelling phase if the model does not have to serve other purposes as well, e.g. as a protocol validation model.

- Operational guidelines for validation of the test suites should be defined. The guidelines should especially take into account the complexity and size of the test suites and protocol models. This means that exhaustive validation methods are not likely to be applicable to complex protocol models, even if the methods are supported by the tools.

History

Document history	
December 1996	First Edition

ISBN 2-7437-1216-3
Dépôt légal : Décembre 1996