



## Methods for Testing and Specification (MTS); The Testing and Test Control Notation version 3; Part 5: TTCN-3 Runtime Interface (TRI)

**STANDARD PREVIEW**  
(standards.iteh.ai)  
Full standard available on  
<https://standards.iteh.ai/catalog/standards/sist/60115-2017-05-4715-a93f-004cf2b9de9/etsi-es-201-873-5-v4-8-1-2017-05>

---

Reference

RES/MTS-201873-5 T3ed481TRI

---

Keywords

interface, methodology, runtime, testing, TRI,  
TTCN-3

**ETSI**

650 Route des Lucioles  
F-06921 Sophia Antipolis Cedex - FRANCE

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

Siret N° 348 623 562 00017 - NAF 742 C  
Association à but non lucratif enregistrée à la  
Sous-Préfecture de Grasse (06) N° 7803/88

---

**Important notice**

The present document can be downloaded from:

<http://www.etsi.org/standards-search>

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the only prevailing document is the print of the Portable Document Format (PDF) version kept on a specific network drive within ETSI Secretariat.

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at

<https://portal.etsi.org/TB/ETSIDeliverableStatus.aspx>

If you find errors in the present document, please send your comment to one of the following services:

<https://portal.etsi.org/People/CommiteeSupportStaff.aspx>

---

**Copyright Notification**

No part may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm except as authorized by written permission of ETSI.

The content of the PDF version shall not be modified without the written authorization of ETSI.

The copyright and the foregoing restriction extend to reproduction in all media.

© European Telecommunications Standards Institute 2017.

All rights reserved.

DECT™, PLUGTESTS™, UMTS™ and the ETSI logo are Trade Marks of ETSI registered for the benefit of its Members.

3GPP™ and LTE™ are Trade Marks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners.

oneM2M logo is protected for the benefit of its Members

GSM® and the GSM logo are Trade Marks registered and owned by the GSM Association.

# Contents

Intellectual Property Rights .....	8
Foreword.....	8
Modal verbs terminology.....	8
Introduction .....	8
1 Scope .....	9
1.1 Scope of the present document.....	9
1.2 Compliance.....	9
2 References .....	9
2.1 Normative references .....	9
2.2 Informative references.....	10
3 Definitions and abbreviations.....	10
3.1 Definitions.....	10
3.2 Abbreviations .....	11
4 General Structure of a TTCN-3 Test System .....	12
4.1 Entities in a TTCN-3 test system.....	12
4.1.0 Types of entities.....	12
4.1.1 Test Management and Control (TMC).....	12
4.1.1.0 Test Management and Control Entities .....	12
4.1.1.1 Test Management (TM) .....	13
4.1.1.2 Test Logging (TL).....	13
4.1.1.3 Coding and Decoding (CD) .....	13
4.1.1.4 Component Handling (CH).....	13
4.1.2 TTCN-3 Executable (TE) .....	13
4.1.2.0 TTCN-3 Executable Entity.....	13
4.1.2.1 Executable Test Suite (ETS).....	13
4.1.2.2 TTCN-3 RunTime System (T3RTS).....	13
4.1.2.3 Encoding/Decoding System (EDS).....	14
4.1.2.4 Timers in the TTCN-3 Executable.....	14
4.1.3 SUT Adaptor (SA).....	14
4.1.4 Platform Adaptor (PA).....	14
4.2 Interfaces in a TTCN-3 Test System .....	15
4.3 Execution requirements for a TTCN-3 test system .....	15
5 TTCN-3 Runtime Interface and operations.....	15
5.1 Overview of the TRI.....	15
5.1.0 Sub-interfaces .....	15
5.1.1 The triCommunication Interface.....	16
5.1.2 The triPlatform Interface .....	16
5.1.3 Correlation between TTCN-3 and TRI Operation Invocations.....	16
5.2 Error handling .....	17
5.2.0 Basic rules.....	17
5.2.1 triSAErrorReq (SA → TE) .....	18
5.2.2 triPAErrorReq (PA → TE) .....	18
5.3 Data interface .....	18
5.3.0 Basic rules.....	18
5.3.1 Connection.....	18
5.3.2 Communication.....	19
5.3.3 Timer .....	20
5.3.4 Miscellaneous .....	20
5.4 Operation descriptions.....	20
5.5 Communication interface operations.....	21
5.5.1 triSAReset (TE → SA) .....	21
5.5.2 Connection handling operations .....	21
5.5.2.1 triExecuteTestCase (TE → SA) .....	21

5.5.2.2	triMap (TE → SA) .....	21
5.5.2.3	triMapParam (TE → SA) .....	22
5.5.2.4	triUnmap (TE → SA) .....	22
5.5.2.5	triUnmapParam (TE → SA) .....	23
5.5.2.6	triEndTestCase (TE → SA) .....	23
5.5.3	Message based communication operations .....	24
5.5.3.1	triSend (TE → SA) .....	24
5.5.3.2	triSendBC (TE → SA) .....	24
5.5.3.3	triSendMC (TE → SA) .....	25
5.5.3.4	triEnqueueMsg (SA → TE) .....	25
5.5.4	Procedure based communication operations .....	26
5.5.4.1	triCall (TE → SA) .....	26
5.5.4.2	triCallBC (TE → SA) .....	27
5.5.4.3	triCallMC (TE → SA) .....	28
5.5.4.4	triReply (TE → SA) .....	29
5.5.4.5	triReplyBC (TE → SA) .....	30
5.5.4.6	triReplyMC (TE → SA) .....	31
5.5.4.7	triRaise (TE → SA) .....	31
5.5.4.8	triRaiseBC (TE → SA) .....	32
5.5.4.9	triRaiseMC (TE → SA) .....	32
5.5.4.10	triEnqueueCall (SA → TE) .....	33
5.5.4.11	triEnqueueReply (SA → TE) .....	33
5.5.4.12	triEnqueueException (SA → TE) .....	34
5.5.5	Miscellaneous operations .....	34
5.5.5.1	triSUTactionInformal (TE → SA) .....	34
5.6	Platform interface operations .....	34
5.6.1	triPAReset (TE → PA) .....	34
5.6.2	Timer operations .....	35
5.6.2.1	triStartTimer (TE → PA) .....	35
5.6.2.2	triStopTimer (TE → PA) .....	35
5.6.2.3	triReadTimer (TE → PA) .....	35
5.6.2.4	triTimerRunning (TE → PA) .....	36
5.6.2.5	triTimeout (PA → TE) .....	36
5.6.3	Miscellaneous operations .....	37
5.6.3.1	triExternalFunction (TE → PA) .....	37
5.6.3.2	triSelf (PA → TE) .....	37
5.6.3.3	triRnd (PA → TE) .....	37
6	Java™ language mapping .....	38
6.1	Introduction .....	38
6.2	Names and scopes .....	38
6.2.1	Names .....	38
6.2.2	Scopes .....	38
6.3	Type mapping .....	38
6.3.1	Basic type mapping .....	38
6.3.1.0	IDL type mapping .....	38
6.3.1.1	Boolean .....	39
6.3.1.2	String .....	39
6.3.2	Structured type mapping .....	39
6.3.2.0	Mapping rules .....	39
6.3.2.1	TriPortIdType .....	39
6.3.2.2	TriPortIdListType .....	40
6.3.2.3	TriComponentIdType .....	40
6.3.2.4	TriComponentIdListType .....	41
6.3.2.5	TriMessageType .....	41
6.3.2.6	TriAddressType .....	42
6.3.2.7	TriAddressListType .....	42
6.3.2.8	TriSignatureIdType .....	43
6.3.2.9	TriParameterType .....	43
6.3.2.10	TriParameterPassingModeType .....	44

6.3.2.11	TriParameterListType .....	44
6.3.2.12	TriExceptionType .....	45
6.3.2.13	TriTimerIdType .....	45
6.3.2.14	TriTimerDurationType.....	45
6.3.2.15	TriFunctionIdType .....	46
6.3.2.16	TriTestCaseIdType.....	46
6.3.2.17	TriActionTemplateType.....	46
6.3.2.18	TriStatusType .....	47
6.4	Constants .....	47
6.5	Mapping of interfaces.....	48
6.5.0	Basic rules.....	48
6.5.1	Out and InOut Parameter Passing Mode.....	48
6.5.2	triCommunication - Interface.....	48
6.5.2.0	Introduction .....	48
6.5.2.1	triCommunicationSA .....	48
6.5.2.2	triCommunicationTE.....	49
6.5.3	triPlatform - Interface .....	50
6.5.3.0	Introduction .....	50
6.5.3.1	TriPlatformPA.....	50
6.5.3.2	TriPlatformTE.....	51
6.6	Optional parameters .....	51
6.7	TRI initialization .....	51
7	ANSI C language mapping.....	51
7.1	Introduction .....	51
7.2	Names and scopes .....	51
7.2.0	Naming rules.....	51
7.2.1	Abstract type mapping.....	52
7.2.2	ANSI C type definitions .....	53
7.2.3	IDL type mapping.....	53
7.2.4	TRI operation mapping.....	53
7.3	Memory management.....	56
8	C++ language mapping .....	56
8.1	Introduction .....	56
8.2	Names and scopes .....	56
8.3	Memory management.....	56
8.4	Void.....	56
8.5	Type mapping.....	56
8.5.0	Basic rules.....	56
8.5.1	Encapsulated C++ types.....	56
8.5.2	Abstract data types.....	57
8.5.2.1	QualifiedName .....	57
8.5.2.2	TriAddress.....	57
8.5.2.3	TriAddressList .....	58
8.5.2.4	TriComponentId.....	59
8.5.2.5	TriComponentIdList.....	59
8.5.2.6	TriException.....	60
8.5.2.7	TriFunctionId .....	61
8.5.2.8	TriMessage.....	61
8.5.2.9	TriParameter.....	62
8.5.2.10	TriParameterList .....	62
8.5.2.11	TriParameterPassingMode .....	63
8.5.2.12	TriPortId.....	63
8.5.2.13	TriPortIdList.....	64
8.5.2.14	TriSignatureId .....	65
8.5.2.15	TriStatus .....	65
8.5.2.16	TriTestCaseId.....	65
8.5.2.17	TriTimerDuration.....	66
8.5.2.18	TriTimerId.....	66
8.6	Mapping of interfaces.....	67
8.6.1	TriCommunicationSA.....	67

8.6.2	TriCommunicationTE.....	68
8.6.3	TriPlatformPA.....	69
8.6.4	TriPlatformTE.....	69
9	C# language mapping.....	70
9.1	Introduction.....	70
9.2	Names and scopes.....	70
9.2.1	Names.....	70
9.2.2	Scopes.....	70
9.3	Null value mapping.....	70
9.4	Type mapping.....	70
9.4.1	Basic type mapping.....	70
9.4.1.0	IDL type mapping.....	70
9.4.1.1	Boolean.....	71
9.4.1.2	String.....	71
9.4.2	Structured type mapping.....	71
9.4.2.0	Mapping rules.....	71
9.4.2.1	IQualifiedName.....	71
9.4.2.2	TriPortIdType.....	71
9.4.2.3	TriPortIdListType.....	72
9.4.2.4	TriComponentIdType.....	72
9.4.2.5	TriComponentIdListType.....	73
9.4.2.6	TriMessageType.....	73
9.4.2.7	TriAddressType.....	74
9.4.2.8	TriAddressListType.....	74
9.4.2.9	TriSignatureIdType.....	75
9.4.2.10	TriParameterPassingModeType.....	75
9.4.2.11	TriParameterType.....	75
9.4.2.12	TriParameterListType.....	75
9.4.2.13	TriExceptionType.....	76
9.4.2.14	TriTimerIdType.....	76
9.4.2.15	TriTimerDurationType.....	77
9.4.2.16	TriFunctionIdType.....	77
9.4.2.17	TriTestCaseIdType.....	77
9.4.2.18	TriStatusType.....	77
9.5	Mapping of interfaces.....	77
9.5.0	Basic rules.....	77
9.5.1	Out and inout parameter passing mode.....	78
9.5.2	triCommunication interface.....	78
9.5.2.0	Introduction.....	78
9.5.2.1	ITriCommunicationSA.....	78
9.5.2.2	ITriCommunicationTE.....	79
9.5.2.3	ITriPlatformPA.....	80
9.5.2.4	ITriPlatformTE.....	80
9.6	Optional parameters.....	80
<b>Annex A (normative):</b>	<b>IDL Summary.....</b>	<b>81</b>
<b>Annex B (informative):</b>	<b>Use scenarios.....</b>	<b>85</b>
B.0	Introduction.....	85
B.1	First scenario.....	86
B.1.0	Use case.....	86
B.1.1	TTCN-3 fragment.....	86
B.1.2	Message sequence chart.....	87
B.2	Second scenario.....	88
B.2.0	Use case.....	88
B.2.1	TTCN-3 fragment.....	88
B.2.2	Message sequence chart.....	89
B.3	Third scenario.....	90
B.3.0	Use case.....	90

B.3.1	TTCN-3 fragment.....	90
B.3.2	Message sequence chart .....	91
<b>Annex C (informative):</b>	<b>Bibliography.....</b>	<b>92</b>
History .....		93

**iTeh STANDARD PREVIEW**  
(standards.iteh.ai)

Full standard:  
<https://standards.iteh.ai/catalog/standards/sist/a6e24e5-c835-4715-a93f-004cf2b9de9/etsi-es-201-873-5-v4.8.1-2017-05>

---

## Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: "*Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards*", which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (<https://ipr.etsi.org/>).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

---

## Foreword

This ETSI Standard (ES) has been produced by ETSI Technical Committee Methods for Testing and Specification (MTS).

The present document is part 5 of a multi-part deliverable. Full details of the entire series can be found in part 1 [2].

---

## Modal verbs terminology

In the present document "**shall**", "**shall not**", "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the [ETSI Drafting Rules](#) (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

---

## Introduction

The present document consists of two distinct parts, the first part describing the structure of a TTCN-3 test system implementation and the second part presenting the TTCN-3 Runtime Interface specification.

The first part introduces the decomposition of a TTCN-3 test system into four main entities: Test Management (TM), TTCN-3 Executable (TE), SUT Adaptor (SA), and Platform Adaptor (PA). In addition, the interaction between these entities, i.e. the corresponding interfaces, is defined.

The second part of the present document specifies the TTCN-3 Runtime Interface (TRI). The interface is defined in terms of operations, which are implemented as part of one entity and called by other entities of the test system. For each operation, the interface specification defines associated data structures, the intended effect on the test system and any constraints on the usage of the operation. Note that this interface specification only defines interactions between the TSI and the SUT as well as timer operations.



---

# 1 Scope

## 1.1 Scope of the present document

The present document provides the specification of the runtime interface for TTCN-3 test system implementations. The TTCN-3 Runtime Interface provides a standardized adaptation for timing and communication of a test system to a particular processing platform and the system under test, respectively. The present document defines the interface as a set of operations independent of target language.

The interface is defined to be compatible with the TTCN-3 standard (see ETSI ES 201 873-1 [2]). The present document uses the CORBA Interface Definition Language (IDL) to specify the TRI completely. Clauses 6, 7 and 8 present language mappings for this abstract specification to the target languages Java™, ANSI C, and C++. A summary of the IDL based interface specification is provided in annex A.

**NOTE:** Java™ is the trade name of a programming language developed by Oracle Corporation. This information is given for the convenience of users of the present document and does not constitute an endorsement by ETSI of the programming language named. Equivalent programming languages may be used if they can be shown to lead to the same results.

## 1.2 Compliance

The requirement for a TTCN-3 test system to be TRI compliant is to adhere to the interface specification stated in the present document as well as to one of the target language mappings included.

**EXAMPLE:** If a vendor supports Java™, the TRI operation calls and implementations, which are part of the TTCN-3 executable, have to comply with the IDL to Java™ mapping specified in the present document.

---

# 2 References

## 2.1 Normative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

Referenced documents which are not found to be publicly available in the expected location might be found at <https://docbox.etsi.org/Reference/>.

**NOTE:** While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are necessary for the application of the present document.

- [1] Recommendation ITU-T X.290: "OSI conformance testing methodology and framework for protocol Recommendations for ITU-T applications - General concepts".

**NOTE:** The corresponding ISO/IEC standard is ISO/IEC 9646-1: "Information technology -- Open Systems Interconnection -- Conformance testing methodology and framework; Part 1: General concepts".

- [2] ETSI ES 201 873-1: "Methods for Testing and Specification (MTS); The Testing and Test Control Notation version 3; Part 1: TTCN-3 Core Language".
- [3] ETSI ES 201 873-4: "Methods for Testing and Specification (MTS); The Testing and Test Control Notation version 3; Part 4: TTCN-3 Operational Semantics".
- [4] CORBA 3.0: "The Common Object Request Broker: Architecture and Specification", OMG Formal Document (specifies IDL).

[5] Sun Microsystems: "Java™ Language Specification".

NOTE: See at [http://java.sun.com/docs/books/jls/third\\_edition/html/j3TOC.html](http://java.sun.com/docs/books/jls/third_edition/html/j3TOC.html).

[6] ISO/IEC 9899: "Information technology -- Programming Languages -- C".

[7] ISO/IEC 14882: "Information technology -- Programming Languages -- C++".

[8] ECMA-334: "C# Language Specification".

NOTE: See at <http://www.ecma-international.org/publications/standards/Ecma-334.htm>.

## 2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

Not applicable.

---

## 3 Definitions and abbreviations

### 3.1 Definitions

For the purposes of the present document, the terms and definitions given in ETSI ES 201 873-1 [2] and the following apply:

**Abstract Test Suite (ATS):** See Recommendation ITU-T X.290 [1].

**communication port:** abstract mechanism facilitating communication between test components

NOTE: A communication port is modelled as a FIFO queue in the receiving direction. Ports can be message-based, procedure-based or a mixture of the two.

**Executable Test Suite (ETS):** See Recommendation ITU-T X.290 [1].

**explicit timer:** timer that is declared in a TTCN-3 ATS and that can be accessed through TTCN-3 timer operations

**Implementation eXtra Information for Testing (IXIT):** See Recommendation ITU-T X.290 [1].

**implicit timer:** system timer that is created by the TTCN-3 Executable to guard a TTCN-3 call or execute operation

NOTE: Implicit timers are not accessible to the TTCN-3 user.

**Platform Adaptor (PA):** entity that adapts the TTCN-3 Executable to a particular execution platform

NOTE: The Platform Adaptor creates a single notion of time for a TTCN-3 test system, and implements external functions as well as explicit and implicit timers.

**SUT Adaptor (SA):** entity that adapts the TTCN-3 communication operations with the SUT based on an abstract test system interface and implements the real test system interface

**System Under Test (SUT):** See Recommendation ITU-T X.290 [1].

NOTE: All types are known at compile time, i.e. are statically bound.

**test case:** See Recommendation ITU-T X.290 [1].

**test event:** either sent or received test data (message or procedure call) on a communication port that is part of the test system interface

**Test Management (TM):** entity that provides a user interface and administers the TTCN-3 test system

**test system:** See Recommendation ITU-T X.290 [1].

**Test System Interface (TSI):** test component that provides a mapping of the ports available in the (abstract) TTCN-3 test system to those offered by a real test system

**Timer Identification (TID):** unique identification for explicit or implicit timer instances that is generated by the TTCN-3 Executable

**TTCN-3 Control Interface (TCI):** four interfaces that define the interaction of the TTCN-3 Executable with the test management, the coding and decoding, the test component handling, and the logging in a test system

**TTCN-3 Executable (TE):** part of a test system that deals with interpretation or execution of a TTCN-3 ETS

**TTCN-3 Runtime Interface (TRI):** two interfaces that define the interaction of the TTCN-3 Executable between the SUT and the Platform Adaptor (PA) and the System Adaptor (SA) in a test system

## 3.2 Abbreviations

For the purposes of the present document, the following abbreviations apply:

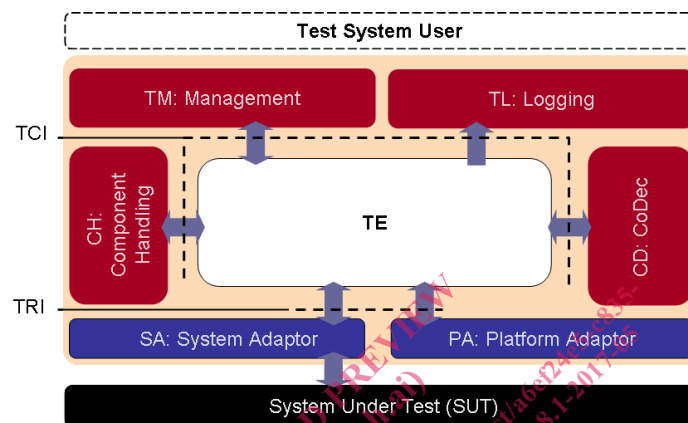
ADT	Abstract Data Type
ANSI	American National Standards Institute
ASN.1	Abstract Syntax Notation One
ATS	Abstract Test Suite
CD	(External) Coding/Decoding
CH	Component Handling
CORBA	Common Object Request Broker Architecture
EDS	(Internal) Encoding/Decoding System
ETS	Executable Test Suite
FIFO	First-In-First-Out (Scheduling Discipline)
IDL	Interface Definition Language
IXIT	Implementation eXtra Information for Testing
MSC	Message Sequence Chart
MTC	Main Test Component
OMG	Object Management Group
PA	Platform Adaptor
SA	SUT Adaptor
STL	Standard Template Library of C++
SUT	System Under Test
T3RTS	TTCN-3 RunTime System
TCI	TTCN-3 Control Interface
TE	TTCN-3 Executable
TID	Timer IDentification
TL	Test Logging
TM	Test Management
TMC	Test Management and Control
TRI	TTCN-3 Runtime Interface
TSI	Test System Interface
TTCN	Testing and Test Control Notation
TTCN-3	Tree and Tabular Combined Notation version 3

## 4 General Structure of a TTCN-3 Test System

### 4.1 Entities in a TTCN-3 test system

#### 4.1.0 Types of entities

A TTCN-3 test system can be thought of conceptually as a set of interacting entities where each entity corresponds to a particular aspect of functionality in a test system implementation. These entities manage test execution, interpreting or executing compiled TTCN-3 code, realize proper communication with the SUT, implement external functions, and handle timer operations.



**Figure 1: General Structure of a TTCN-3 Test System**

The structure of a TTCN-3 test system implementation is illustrated in figure 1. It should be noted that the further refinement of TM into smaller entities, as shown in figure 1 and used in the following clauses of the present document, is purely an aid to define TTCN-3 test system interfaces.

The part of the test system that deals with interpretation and execution of TTCN-3 modules, i.e. the Executable Test Suite (ETS), is part of the TTCN-3 Executable (TE). This corresponds either to the executable code produced by a TTCN-3 compiler or a TTCN-3 interpreter in a test system implementation. It is assumed that a test system implementation includes the ETS as derived from a TTCN-3 ATS.

The remaining part of the TTCN-3 test system, which deals with any aspects that cannot be concluded from information being present in the original ATS alone, can be decomposed into Test Management (TM), SUT Adaptor (SA), and Platform Adaptor (PA) entities. In general, these entities cover a test system user interface, test execution control, test event logging, as well as communication with the SUT and timer implementation.

#### 4.1.1 Test Management and Control (TMC)

##### 4.1.1.0 Test Management and Control Entities

The TMC entity includes functionality related to management of:

- test execution;
- components;
- encoding and decoding; and
- logging.

#### 4.1.1.1 Test Management (TM)

The TM entity is responsible for overall management of the test system. After the test system has been initialized, test execution starts within the TM entity. The entity is responsible for the proper invocation of TTCN-3 modules, i.e. propagating module parameters and/or IXIT information to the TE if necessary. Typically, this entity would also implement a test system user interface.

#### 4.1.1.2 Test Logging (TL)

The TL entity is responsible for maintaining the test log. It is explicitly notified to log test events by the TE. The TL entity has a unidirectional interface where any entity part of the TE may post a logging request to the TL entity. A TM internal interface may also be used to record test management information generated by the TE.

#### 4.1.1.3 Coding and Decoding (CD)

The CD entity is optionally responsible for the external encoding and decoding data associated with message based or procedure based communication within the TE. The external codecs can be used in parallel with, or instead of, the built-in codecs associated with the TE. Unlike the built-in codecs the external codecs have a standardized interface which makes them portable between different TTCN-3 systems and tools.

#### 4.1.1.4 Component Handling (CH)

The CH entity is responsible for distributing parallel test components. This distribution might be across one or many physical systems. The CH entity allows the test management to create and control distributed test systems in a manner which is transparent and independent from the TE.

### 4.1.2 TTCN-3 Executable (TE)

#### 4.1.2.0 TTCN-3 Executable Entity

The TE entity is responsible for the interpretation or execution of the TTCN-3 ATS. Conceptually, the TE can be decomposed into three interacting entities: an ETS, TTCN-3 RunTime System (T3RTS), and an optional internal Encoding/Decoding System (EDS) entity. Note that this refinement of the TE into smaller entities is purely a conceptual aid to define TTCN-3 test system interfaces - there is no requirement for this distinction to be reflected in TRI implementations.

The following clauses define the responsibilities of each entity and also discuss the handling of timers in the TRI.

#### 4.1.2.1 Executable Test Suite (ETS)

The ETS entity handles the execution or interpretation of test cases, the sequencing and matching of test events, as defined in the corresponding TTCN-3 modules ETSI ES 201 873-1 [2]. It interacts with the T3RTS entity to send, attempt to receive (or match), and log test events during test case execution, to create and remove TTCN-3 test components, as well as to handle external function calls, action operations, and timers. Note that the ETS entity does not directly interact with the SA via the TRI.

#### 4.1.2.2 TTCN-3 RunTime System (T3RTS)

The T3RTS entity interacts with the TM, SA and PA entities via TCI and TRI, and manages ETS and EDS entities. The T3RTS initializes adaptors as well as ETS and EDS entities. This entity performs all the actions necessary to properly start the execution of a test case or function with parameters in the ETS entity. It queries the TM entity for module parameter values required by the ETS and sends logging information to it. It also collects and resolves associated verdicts returned by the ETS entity as defined in ETSI ES 201 873-1 [2].

The T3RTS entity implements the creation and removal of TTCN-3 test components, as well as the TTCN-3 semantics of message and procedure based communication, external function calls, action operations and timers. This includes notifying the SUT Adaptor (SA) which message or procedure call is to be sent to the SUT, or the Platform Adaptor (PA) which external function is to be executed or which timers are to be started, stopped, queried, or read. Similarly, the T3RTS notifies the ETS entity of incoming messages or procedure calls from the SUT as well as timeout events.