



**Methods for Testing and Specification (MTS);
The Testing and Test Control Notation version 3;
Part 6: TTCN-3 Control Interface (TCI)**

iTeh STANDARD PREVIEW
(standards.iteh.ai)
Full standard/standards/etsi/es/201/873-6-v4.8.1-2016-07
https://standards.iteh.ai/catalog/standards/etsi/es/201/873-6-v4.8.1-2016-07
4c16-8a55-719bac780d00/etsi-es-201-873-6-v4.8.1-2016-07

Reference

RES/MTS-201873-6 T3ed481TCI

Keywordscontrol, interface, methodology, TCI, testing,
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 noticeThe 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 2016.
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.
GSM® and the GSM logo are Trade Marks registered and owned by the GSM Association.

Contents

Intellectual Property Rights	16
Foreword.....	16
Modal verbs terminology.....	16
1 Scope	17
2 References	17
2.1 Normative references	17
2.2 Informative references.....	18
3 Definitions and abbreviations.....	18
3.1 Definitions.....	18
3.2 Abbreviations	19
4 Introduction	20
5 Compliance.....	20
6 General structure of a TTCN-3 test system.....	20
6.1 Entities in a TTCN-3 test system.....	20
6.1.0 Types of entities.....	20
6.1.1 Test Management and Control (TMC).....	22
6.1.1.0 Test Management and Control Entities	22
6.1.1.1 Test Management (TM)	22
6.1.1.2 Coding and Decoding (CD)	22
6.1.1.3 Component Handling (CH).....	22
6.1.1.4 Test Logging (TL).....	24
6.1.2 TTCN-3 Executable (TE)	24
6.1.3 SUT Adaptor (SA).....	24
6.1.4 Platform Adaptor (PA).....	24
6.2 Execution requirements for a TTCN-3 test system.....	24
7 TTCN-3 control interface and operations.....	24
7.1 Overview of the TCI.....	24
7.1.0 TCI role in a TTCN-3 test system.....	24
7.1.1 Correlation between TTCN-3 and TCI operation invocations.....	25
7.1.1.0 Mapping of TTCN-3 operations to TCI operations.....	25
7.1.1.1 TTCN-3 operations with TCI operation equivalent	25
7.1.1.2 TTCN-3 operations with TCI operation pair equivalent	26
7.1.1.3 TTCN-3 operations without direct TCI operation equivalent	27
7.1.1.3.0 Mapping of TTCN-3 operations to series of TCI operations.....	27
7.1.1.3.1 Test case stop operation.....	27
7.2 TCI data.....	27
7.2.0 Abstract data types.....	27
7.2.1 General abstract data types	27
7.2.1.0 Use of general abstract data types	27
7.2.1.1 Management.....	27
7.2.1.2 Communication	28
7.2.2 Abstract TTCN-3 data types and values	29
7.2.2.0 Definition and scope of use.....	29
7.2.2.1 Abstract TTCN-3 data types	29
7.2.2.2 Abstract TTCN-3 values	30
7.2.2.2.0 Basic rules	30
7.2.2.2.1 The abstract data type Value	31
7.2.2.2.2 The abstract data type IntegerValue	33
7.2.2.2.3 The abstract data type FloatValue	33
7.2.2.2.4 The abstract data type BooleanValue	33
7.2.2.2.5 The abstract data type CharstringValue	34
7.2.2.2.6 The abstract data type UniversalCharstringValue.....	34

7.2.2.2.7	The abstract data type BitstringValue.....	35
7.2.2.2.8	The abstract data type OctetstringValue.....	35
7.2.2.2.9	The abstract data type HexstringValue.....	36
7.2.2.2.10	The abstract data type RecordValue.....	37
7.2.2.2.11	The abstract data type RecordOfValue.....	38
7.2.2.2.12	The abstract data type UnionValue.....	39
7.2.2.2.13	The abstract data type EnumeratedValue.....	40
7.2.2.2.14	The abstract data type VerdictValue.....	40
7.2.2.2.15	The abstract data type AddressValue.....	40
7.2.2.3	Abstract TTCN-3 matching mechanisms.....	41
7.2.2.3.1	The abstract data type MatchingMechanism.....	41
7.2.2.3.2	The abstract data type MatchingList.....	41
7.2.2.3.3	The abstract data type ValueRange.....	42
7.2.2.3.4	The abstract data type CharacterPattern.....	42
7.2.2.3.5	The abstract data type MatchDecodedContent.....	42
7.2.2.4	Data types for complex TTCN-3 properties.....	43
7.2.2.4.0	Scope of use of TTCN-3 properties.....	43
7.2.2.4.1	The abstract data type LengthRestriction.....	43
7.2.2.4.2	The abstract data type Permutation.....	43
7.2.2.4.3	The abstract data type RangeBoundary.....	43
7.2.3	Abstract logging types.....	44
7.2.3.1	The abstract data type TciValueTemplate.....	44
7.2.3.2	The abstract data type TciNonValueTemplate.....	44
7.2.3.3	The Value List and Mismatch Types.....	44
7.2.3.4	The Status Types.....	45
7.3	TCI operations.....	45
7.3.0	The TCI interfaces.....	45
7.3.1	The TCI-TM interface.....	46
7.3.1.0	Scope of use.....	46
7.3.1.1	TCI-TM required.....	47
7.3.1.1.0	Scope of use.....	47
7.3.1.1.1	tciRootModule.....	47
7.3.1.1.2	tciGetImportedModules.....	47
7.3.1.1.3	tciGetModuleParameters.....	47
7.3.1.1.4	tciGetTestCases.....	47
7.3.1.1.5	tciGetTestCaseParameters.....	48
7.3.1.1.6	tciGetTestCaseTSI.....	48
7.3.1.1.7	tciStartTestCase.....	48
7.3.1.1.8	tciStopTestCase.....	48
7.3.1.1.9	tciStartControl.....	49
7.3.1.1.10	tciStopControl.....	49
7.3.1.2	TCI-TM provided.....	49
7.3.1.2.0	Scope of use.....	49
7.3.1.2.1	tciTestCaseStarted.....	49
7.3.1.2.2	tciTestCaseTerminated.....	49
7.3.1.2.3	tciControlTerminated.....	50
7.3.1.2.4	tciGetModulePar.....	50
7.3.1.2.5	tciLog.....	50
7.3.1.2.6	tciError.....	50
7.3.2	The TCI-CD interface.....	50
7.3.2.0	Scope of use.....	50
7.3.2.1	TCI-CD required.....	51
7.3.2.1.0	Scope of use.....	51
7.3.2.1.1	getTypeForName.....	51
7.3.2.1.2	getInteger.....	52
7.3.2.1.3	getFloat.....	52
7.3.2.1.4	getBoolean.....	52
7.3.2.1.5	Void.....	52
7.3.2.1.6	getCharstring.....	52
7.3.2.1.7	getUniversalCharstring.....	52

7.3.2.1.8	getHexstring	52
7.3.2.1.9	getBitstring	52
7.3.2.1.10	getOctetstring	53
7.3.2.1.11	getVerdict	53
7.3.2.1.12	tciErrorReq	53
7.3.2.2	TCI-CD provided	53
7.3.2.2.0	Scope of use.....	53
7.3.2.2.1	decode.....	53
7.3.2.2.2	encode.....	53
7.3.2.2.3	decodeValue	54
7.3.2.2.4	encodeValue	54
7.3.3	The TCI-CH interface	54
7.3.3.0	Scope of use	54
7.3.3.1	TCI-CH required	55
7.3.3.1.0	Scope of use.....	55
7.3.3.1.1	tciEnqueueMsgConnected	56
7.3.3.1.2	tciEnqueueCallConnected	56
7.3.3.1.3	tciEnqueueReplyConnected.....	56
7.3.3.1.4	tciEnqueueRaiseConnected	57
7.3.3.1.5	tciCreateTestComponent	57
7.3.3.1.6	tciStartTestComponent	57
7.3.3.1.7	tciStopTestComponent	58
7.3.3.1.8	tciConnect.....	58
7.3.3.1.9	tciDisconnect	58
7.3.3.1.10	tciMap.....	58
7.3.3.1.11	tciMapParam	58
7.3.3.1.12	tciUnmap	59
7.3.3.1.13	tciUnmapParam	59
7.3.3.1.14	tciTestComponentTerminated	59
7.3.3.1.15	tciTestComponentRunning	59
7.3.3.1.16	tciTestComponentDone	59
7.3.3.1.17	tciGetMTC.....	60
7.3.3.1.18	tciExecuteTestCase.....	60
7.3.3.1.19	tciReset	60
7.3.3.1.20	tciKillTestComponent	60
7.3.3.1.21	tciTestComponentAlive.....	61
7.3.3.1.22	tciTestComponentKilled.....	61
7.3.3.2	TCI-CH provided	61
7.3.3.2.0	Scope of use.....	61
7.3.3.2.1	tciSendConnected.....	61
7.3.3.2.2	tciSendConnectedBC.....	61
7.3.3.2.3	tciSendConnectedMC.....	62
7.3.3.2.4	tciCallConnected	62
7.3.3.2.5	tciCallConnectedBC	62
7.3.3.2.6	tciCallConnectedMC	63
7.3.3.2.7	tciReplyConnected	63
7.3.3.2.8	tciReplyConnectedBC	64
7.3.3.2.9	tciReplyConnectedMC	64
7.3.3.2.10	tciRaiseConnected	65
7.3.3.2.11	tciRaiseConnectedBC.....	65
7.3.3.2.12	tciRaiseConnectedMC.....	65
7.3.3.2.13	tciCreateTestComponentReq.....	66
7.3.3.2.14	tciStartTestComponentReq.....	66
7.3.3.2.15	tciStopTestComponentReq.....	66
7.3.3.2.16	tciConnectReq	66
7.3.3.2.17	tciDisconnectReq.....	67
7.3.3.2.18	tciMapReq	67
7.3.3.2.19	tciMapParamReq	67
7.3.3.2.20	tciUnmapReq	67
7.3.3.2.21	tciUnmapParamReq.....	68
7.3.3.2.22	tciTestComponentTerminatedReq.....	68
7.3.3.2.23	tciTestComponentRunningReq	68

7.3.3.2.24	tciTestComponentDoneReq.....	68
7.3.3.2.25	tciGetMTCReq.....	68
7.3.3.2.26	tciExecuteTestCaseReq.....	69
7.3.3.2.27	tciResetReq.....	69
7.3.3.2.28	tciKillTestComponentReq.....	69
7.3.3.2.29	tciTestComponentAliveReq.....	69
7.3.3.2.30	tciTestComponentKilledReq.....	69
7.3.4	The TCI-TL interface.....	70
7.3.4.0	Scope of use.....	70
7.3.4.1	TCI-TL provided.....	70
7.3.4.1.0	Scope of use.....	70
7.3.4.1.1	tliTcExecute.....	70
7.3.4.1.2	tliTcStart.....	71
7.3.4.1.3	tliTcStop.....	71
7.3.4.1.4	tliTcStarted.....	71
7.3.4.1.5	tliTcTerminated.....	72
7.3.4.1.6	tliCtrlStart.....	72
7.3.4.1.7	tliCtrlStop.....	72
7.3.4.1.8	tliCtrlTerminated.....	73
7.3.4.1.9	tliMSend_m.....	73
7.3.4.1.10	tliMSend_m_BC.....	74
7.3.4.1.11	tliMSend_m_MC.....	74
7.3.4.1.12	tliMSend_c.....	75
7.3.4.1.13	tliMSend_c_BC.....	75
7.3.4.1.14	tliMSend_c_MC.....	75
7.3.4.1.15	tliMDetected_m.....	76
7.3.4.1.16	tliMDetected_c.....	76
7.3.4.1.17	tliMMismatch_m.....	76
7.3.4.1.18	tliMMismatch_c.....	77
7.3.4.1.19	tliMReceive_m.....	77
7.3.4.1.20	tliMReceive_c.....	78
7.3.4.1.21	tliPrCall_m.....	78
7.3.4.1.22	tliPrCall_m_BC.....	79
7.3.4.1.23	tliPrCall_m_MC.....	79
7.3.4.1.24	tliPrCall_c.....	80
7.3.4.1.25	tliPrCall_c_BC.....	80
7.3.4.1.26	tliPrCall_c_MC.....	80
7.3.4.1.27	tliPrGetCallDetected_m.....	81
7.3.4.1.28	tliPrGetCallDetected_c.....	81
7.3.4.1.29	tliPrGetCallMismatch_m.....	82
7.3.4.1.30	tliPrGetCallMismatch_c.....	82
7.3.4.1.31	tliPrGetCall_m.....	83
7.3.4.1.32	tliPrGetCall_c.....	83
7.3.4.1.33	tliPrReply_m.....	84
7.3.4.1.34	tliPrReply_m_BC.....	84
7.3.4.1.35	tliPrReply_m_MC.....	85
7.3.4.1.36	tliPrReply_c.....	85
7.3.4.1.37	tliPrReply_c_BC.....	86
7.3.4.1.38	tliPrReply_c_MC.....	86
7.3.4.1.39	tliPrGetReplyDetected_m.....	87
7.3.4.1.40	tliPrGetReplyDetected_c.....	87
7.3.4.1.41	tliPrGetReplyMismatch_m.....	88
7.3.4.1.42	tliPrGetReplyMismatch_c.....	88
7.3.4.1.43	tliPrGetReply_m.....	89
7.3.4.1.44	tliPrGetReply_c.....	89
7.3.4.1.45	tliPrRaise_m.....	90
7.3.4.1.46	tliPrRaise_m_BC.....	90
7.3.4.1.47	tliPrRaise_m_MC.....	91
7.3.4.1.48	tliPrRaise_c.....	91
7.3.4.1.49	tliPrRaise_c_BC.....	92
7.3.4.1.50	tliPrRaise_c_MC.....	92
7.3.4.1.51	tliPrCatchDetected_m.....	93

7.3.4.1.52	tliPrCatchDetected_c	93
7.3.4.1.53	tliPrCatchMismatch_m	94
7.3.4.1.54	tliPrCatchMismatch_c	94
7.3.4.1.55	tliPrCatch_m	95
7.3.4.1.56	tliPrCatch_c	95
7.3.4.1.57	tliPrCatchTimeoutDetected	96
7.3.4.1.58	tliPrCatchTimeout	96
7.3.4.1.59	tliCCreate	96
7.3.4.1.60	tliCStart	97
7.3.4.1.61	tliCRunning	97
7.3.4.1.62	tliCAlive	97
7.3.4.1.63	tliCStop	98
7.3.4.1.64	tliCKill	98
7.3.4.1.65	tliCDoneMismatch	98
7.3.4.1.66	tliCDone	99
7.3.4.1.67	tliCKilledMismatch	99
7.3.4.1.68	tliCKilled	99
7.3.4.1.69	tliCTerminated	100
7.3.4.1.70	tliPConnect	100
7.3.4.1.71	tliPDisconnect	100
7.3.4.1.72	tliPMap	101
7.3.4.1.73	tliPMapParam	101
7.3.4.1.74	tliPUnmap	101
7.3.4.1.75	tliPUnmapParam	102
7.3.4.1.76	tliPClear	102
7.3.4.1.77	tliPStart	102
7.3.4.1.78	tliPStop	103
7.3.4.1.79	tliPHalt	103
7.3.4.1.80	tliEncode	103
7.3.4.1.81	tliDecode	104
7.3.4.1.82	tliTTimeoutDetected	104
7.3.4.1.83	tliTTimeoutMismatch	104
7.3.4.1.84	tliTTimeout	105
7.3.4.1.85	tliTStart	105
7.3.4.1.86	tliTStop	105
7.3.4.1.87	tliTRead	106
7.3.4.1.88	tliTRunning	106
7.3.4.1.89	tliSEnter	106
7.3.4.1.90	tliSLeave	107
7.3.4.1.91	tliVar	107
7.3.4.1.92	tliModulePar	107
7.3.4.1.93	tliGetVerdict	108
7.3.4.1.94	tliSetVerdict	108
7.3.4.1.95	tliLog	108
7.3.4.1.96	tliAEnter	109
7.3.4.1.97	tliALeave	109
7.3.4.1.98	tliANomatch	109
7.3.4.1.99	tliARepeat	109
7.3.4.1.100	tliADefaults	110
7.3.4.1.101	tliAActivate	110
7.3.4.1.102	tliADeactivate	110
7.3.4.1.103	tliAWait	111
7.3.4.1.104	tliAction	111
7.3.4.1.105	tliMatch	111
7.3.4.1.106	tliMatchMismatch	111
7.3.4.1.107	tliInfo	112
7.3.4.1.108	tliMChecked_m	112
7.3.4.1.109	tliMChecked_c	112
7.3.4.1.110	tliPrGetCallChecked_m	113
7.3.4.1.111	tliPrGetCallChecked_c	113
7.3.4.1.112	tliPrGetReplyChecked_m	114
7.3.4.1.113	tliPrGetReplyChecked_c	114

7.3.4.1.114	tliPrCatchChecked_m.....	115
7.3.4.1.115	tliPrCatchChecked_c	115
7.3.4.1.116	tliCheckedAny_m.....	116
7.3.4.1.117	tliCheckedAny_c	116
7.3.4.1.118	tliCheckAnyMismatch_m.....	116
7.3.4.1.119	tliCheckAnyMismatch_c	117
7.3.4.1.120	tliRnd	117
7.3.4.1.121	tliEvaluate.....	117
8	Java™ language mapping.....	118
8.1	Introduction	118
8.2	Names and scopes	118
8.2.1	Names	118
8.2.2	Scopes	118
8.3	Type mapping.....	118
8.3.1	Basic type mapping.....	118
8.3.2	Structured type mapping	119
8.3.2.0	General principles	119
8.3.2.1	TciParameterType	119
8.3.2.2	TciParameterPassingModeType.....	120
8.3.2.3	TciParameterListType.....	120
8.3.2.4	TciTypeClassType	120
8.3.2.5	TciTestComponentKindType.....	121
8.3.2.6	TciBehaviourIdType	121
8.3.2.7	TciTestCaseIdType	121
8.3.2.8	TciModuleIdType	121
8.3.2.9	TciModuleParameterIdType	121
8.3.2.10	TciModuleParameterListType	122
8.3.2.11	TciModuleParameterType.....	122
8.3.2.12	TciParameterTypeListType.....	122
8.3.2.13	TciParameterTypeType.....	123
8.3.2.14	TciModuleIdListType	123
8.3.2.15	TciTestCaseIdListType.....	123
8.3.2.16	TciDecodingResult.....	124
8.3.2.17	TciMatchingTypeType.....	124
8.3.2.18	LengthRestriction.....	124
8.3.2.19	Permutation	125
8.3.2.20	RangeBoundary.....	125
8.3.3	Abstract type mapping.....	126
8.3.3.0	General principles	126
8.3.3.1	Type	126
8.3.4	Abstract value mapping	127
8.3.4.0	General principles	127
8.3.4.1	Value	127
8.3.4.2	IntegerValue.....	128
8.3.4.3	FloatValue	128
8.3.4.4	BooleanValue.....	129
8.3.4.5	CharstringValue	129
8.3.4.6	BitstringValue	129
8.3.4.7	OctetstringValue	130
8.3.4.8	UniversalCharstringValue	131
8.3.4.9	HexstringValue	132
8.3.4.10	RecordValue.....	132
8.3.4.11	RecordOfValue	133
8.3.4.12	UnionValue	134
8.3.4.13	EnumeratedValue.....	135
8.3.4.14	VerdictValue	135
8.3.4.15	AddressValue	136
8.3.5	Abstract template mapping	136
8.3.5.0	General principles	136
8.3.5.1	MatchingMechanism.....	136
8.3.5.2	MatchingList	136

8.3.5.3	ValueRange	137
8.3.5.4	CharacterPattern	137
8.3.5.5	MatchDecodedContent	137
8.3.6	Abstract logging types mapping	138
8.3.6.0	General principles	138
8.3.6.1	TciValueTemplate	138
8.3.6.2	TciNonValueTemplate	138
8.3.6.3	TciValueList	138
8.3.6.4	TciValueDifference	139
8.3.6.5	TciValueDifferenceList	139
8.3.6.6	ComponentStatus	139
8.3.6.7	TimerStatus	139
8.3.6.8	TciStatus	140
8.4	Constants	140
8.5	Mapping of interfaces	141
8.5.0	Calling rules	141
8.5.1	The TCI-TM interface	142
8.5.1.1	TCI-TM provided	142
8.5.1.2	TCI-TM required	142
8.5.2	The TCI-CD interface	142
8.5.2.1	TCI-CD provided	142
8.5.2.2	TCI-CD required	143
8.5.3	The TCI-CH interface	143
8.5.3.1	TCI-CH provided	143
8.5.3.2	TCI-CH required	144
8.5.4	The TCI-TL interface	145
8.5.4.1	TCI-TL provided	145
8.6	Optional parameters	149
8.7	TCI initialization	150
8.8	Error handling	150
9	ANSI C language mapping	150
9.1	Introduction	150
9.2	Value interfaces	150
9.3	Logging interface	155
9.4	Operation interfaces	156
9.4.1	The TCI-TM interface	156
9.4.1.1	TCI-TM provided	156
9.4.1.2	TCI-TM required	156
9.4.2	The TCI-CD interface	156
9.4.2.1	TCI-CD provided	156
9.4.2.2	TCI-CD required	157
9.4.3	The TCI-CH interface	157
9.4.3.1	TCI-CH provided	157
9.4.3.2	TCI-CH required	158
9.4.4	The TCI-TL interface	158
9.4.4.1	TCI-TL provided	158
9.5	Data	163
9.6	Miscellaneous	165
9.7	Optional parameters	165
10	C++ language mapping	166
10.1	Introduction	166
10.2	Names and scopes	166
10.3	Memory management	166
10.4	Error handling	166
10.5	Type mapping	166
10.5.0	Basic concepts	166
10.5.1	Encapsulated C++ types	166
10.5.2	General abstract data types	167
10.5.2.1	TciBehaviourId	167
10.5.2.2	TciModuleId	167

10.5.2.3	TciModuleParameterId	167
10.5.2.4	TciTestCaseId	168
10.5.2.5	TciModuleIdList	168
10.5.2.6	TciModuleParameter	169
10.5.2.7	TciModuleParameterList	169
10.5.2.8	TciParameterPassingMode	170
10.5.2.9	TciParameter	170
10.5.2.10	TciParameterList	170
10.5.2.11	TciParameterType	171
10.5.2.12	TciParameterTypeList	171
10.5.2.13	TciTestComponentKind	172
10.5.2.14	TciTypeClass	172
10.5.2.15	TciTestCaseIdList	173
10.5.2.16	TciMatchingTypeType	173
10.5.2.17	LengthRestriction	173
10.5.2.18	Permutation	174
10.5.2.19	RangeBoundary	175
10.5.3	Abstract TTCN-3 data types and values	175
10.5.3.1	TciType	175
10.5.3.2	TciValue	176
10.5.3.3	IntegerValue	177
10.5.3.4	FloatValue	177
10.5.3.5	BooleanValue	178
10.5.3.6	CharstringValue	178
10.5.3.7	UniversalCharstringValue	179
10.5.3.8	BitstringValue	179
10.5.3.9	OctetstringValue	180
10.5.3.10	HexstringValue	181
10.5.3.11	RecordValue	182
10.5.3.12	RecordOfValue	182
10.5.3.13	UnionValue	183
10.5.3.14	EnumeratedValue	183
10.5.3.15	VerdictValue	184
10.5.3.16	VerdictValueEnum	184
10.5.3.17	AddressValue	185
10.5.3.18	MatchingMechanism	185
10.5.3.19	MatchingList	185
10.5.3.20	ValueRange	186
10.5.3.21	CharacterPattern	186
10.5.3.22	MatchDecodedContent	187
10.5.4	Abstract logging types	187
10.5.4.1	TciValueTemplate	187
10.5.4.2	TciNonValueTemplate	188
10.5.4.3	TciValueList	188
10.5.4.4	TciValueDifference	189
10.5.4.5	TciValueDifferenceList	189
10.5.4.6	ComponentStatus	190
10.5.4.7	TimerStatus	190
10.5.4.8	TciStatus	190
10.6	Operations mapping	191
10.6.1	TCI-TM	191
10.6.1.1	TciTmRequired	191
10.6.1.2	TciTmProvided	191
10.6.2	TCI-CD	192
10.6.2.1	TciCdRequired	192
10.6.2.2	TciCdProvided	192
10.6.3	TCI-CH	192
10.6.3.1	TciChRequired	192
10.6.3.2	TciChProvided	194
10.6.4	TCI-TL	195
10.6.4.1	TciTIProvided	195

11	W3C XML mapping.....	204
11.1	Introduction.....	204
11.2	Scopes.....	204
11.3	Type mapping.....	204
11.3.1	Mapping of simple types.....	204
11.3.1.1	TBoolean.....	204
11.3.1.2	TString.....	204
11.3.1.3	TInteger.....	204
11.3.1.4	TriTimerDurationType.....	204
11.3.1.5	TciParameterPassingModeType.....	204
11.3.1.6	TriStatusType.....	205
11.3.1.7	TciStatusType.....	205
11.3.1.8	ComponentStatusType.....	205
11.3.1.9	TimerStatusType.....	205
11.3.1.10	PortStatusType.....	205
11.3.2	Complex type mapping.....	205
11.3.2.1	TriPortIdType.....	205
11.3.2.2	TriComponentIdType.....	205
11.3.2.3	TriComponentIdListType.....	206
11.3.2.4	Port.....	206
11.3.2.5	Id.....	206
11.3.2.6	TriMessageType.....	207
11.3.2.7	TriParameterType.....	207
11.3.2.8	TriParameterListType.....	208
11.3.2.9	TriAddressType.....	208
11.3.2.10	TriAddressListType.....	208
11.3.2.11	TriExceptionType.....	208
11.3.2.12	TriSignatureIdType.....	209
11.3.2.13	TriTimerIdType.....	209
11.3.2.14	TriTimerDurationType.....	209
11.3.2.15	QualifiedName.....	209
11.3.2.16	TciBehaviourIdType.....	210
11.3.2.17	TciTestCaseIdType.....	210
11.3.2.18	TciParameterType.....	210
11.3.2.19	TciParameterListType.....	211
11.3.2.20	TriPortIdListType.....	211
11.3.3	Abstract value mapping.....	211
11.3.3.1	Value.....	211
11.3.3.2	IntegerValue.....	213
11.3.3.3	FloatValue.....	213
11.3.3.4	BooleanValue.....	214
11.3.3.5	Void.....	214
11.3.3.6	VerdictValue.....	214
11.3.3.7	BitstringValue.....	215
11.3.3.8	HexstringValue.....	215
11.3.3.9	OctetstringValue.....	215
11.3.3.10	CharstringValue.....	216
11.3.3.11	UniversalCharstringValue.....	216
11.3.3.12	RecordValue.....	217
11.3.3.13	RecordOfValue.....	218
11.3.3.14	ArrayValue.....	220
11.3.3.15	SetValue.....	221
11.3.3.16	SetOfValue.....	223
11.3.3.17	EnumeratedValue.....	224
11.3.3.18	UnionValue.....	225
11.3.3.19	AnytypeValue.....	226
11.3.3.20	AddressValue.....	227
11.3.3.21	ComponentValue.....	228
11.3.3.22	PortValue.....	229
11.3.3.23	DefaultValue.....	229
11.3.3.24	TimerValue.....	230
11.3.4	Abstract logging types mapping.....	230

11.3.4.1	TciValueTemplate	230
11.3.4.2	TciNonValueTemplate	231
11.3.4.3	TciValueList	232
11.3.4.4	TciValueDifference	232
11.3.4.5	TciValueDifferenceList	232
11.4	Mapping of the operations on the logging interface	233
11.4.0	Mapping rules	233
11.4.1	Event	233
11.4.2	The TCI-TL interface	233
11.4.2.1	TCI-TL provided	233
12	C# mapping	255
12.1	Introduction	255
12.2	Names and scopes	255
12.2.1	Names	255
12.2.2	Scopes	255
12.3	Null value mapping	255
12.4	Type mapping	255
12.4.1	Basic type mapping	255
12.4.1.0	Mapped types	255
12.4.1.1	TciVerdict	256
12.4.2	Structured type mapping	256
12.4.2.0	Mapping rules	256
12.4.2.1	TciParameterPassingModeType	256
12.4.2.2	TciParameterType	257
12.4.2.3	TciParameterListType	257
12.4.2.4	TciTypeClassType	257
12.4.2.5	TciTestComponentKindType	258
12.4.2.6	TciBehaviourIdType	258
12.4.2.7	TciTestCaseIdType	258
12.4.2.8	TciTestCaseIdListType	258
12.4.2.9	TciModuleIdType	259
12.4.2.10	TciModuleIdListType	259
12.4.2.11	TciModuleParameterIdType	259
12.4.2.12	TciModuleParameterType	259
12.4.2.13	TciModuleParameterListType	259
12.4.2.14	TciParameterTypeType	260
12.4.2.15	TciParameterTypeListType	260
12.4.2.16	TciMatchingTypeType	261
12.4.2.17	LengthRestriction	261
12.4.2.18	Permutation	261
12.4.2.19	RangeBoundary	261
12.4.3	Abstract type mapping	262
12.4.3.0	Mapping rules	262
12.4.3.1	Type	262
12.4.4	Abstract value mapping	263
12.4.4.0	Mapping rules	263
12.4.4.1	Value	263
12.4.4.2	IntegerValue	264
12.4.4.3	FloatValue	264
12.4.4.4	BooleanValue	265
12.4.4.5	CharstringValue	265
12.4.4.6	BitstringValue	265
12.4.4.7	OctetstringValue	266
12.4.4.8	UniversalCharstringValue	267
12.4.4.9	HexstringValue	267
12.4.4.10	RecordValue	268
12.4.4.11	RecordOfValue	268
12.4.4.12	UnionValue	269
12.4.4.13	EnumeratedValue	270
12.4.4.14	VerdictValue	270
12.4.4.15	AddressValue	270

12.4.5	Abstract template mapping	271
12.4.5.0	Mapping rules	271
12.4.5.1	MatchingMechanism.....	271
12.4.5.2	MatchingList	271
12.4.5.3	ValueRange	271
12.4.5.4	CharacterPattern.....	272
12.4.5.5	MatchDecodedContent.....	272
12.4.6	Abstract logging types mapping	272
12.4.6.0	Mapping rules	272
12.4.6.1	TciValueTemplate.....	272
12.4.6.2	TciNonValueTemplate.....	273
12.4.6.3	TciValueList.....	273
12.4.6.4	TciValueDifference.....	273
12.4.6.5	TciValueDifferenceList.....	274
12.4.6.6	TciStatusType	274
12.4.6.7	ComponentStatusType.....	274
12.4.6.8	TimerStatusType.....	274
12.5	Mapping of interfaces.....	275
12.5.0	Calling rules.....	275
12.5.1	TCI-TM interface.....	275
12.5.1.1	TCI-TM provided.....	275
12.5.1.2	TCI-TM required.....	275
12.5.2	TCI-CD interface	276
12.5.2.1	TCI-CD provided	276
12.5.2.2	TCI-CD required	276
12.5.3	TCI-CH interface	276
12.5.3.1	TCI-CH provided.....	276
12.5.3.2	TCI-CH required.....	277
12.5.4	TCI-TL interface.....	278
12.5.4.1	TCI-TL provided.....	278
12.6	Optional parameters	284
12.7	Error Handling.....	284
Annex A (normative):	IDL Specification of TCI.....	285
Annex B (normative):	XML Mapping for TCI TL Provided.....	303
B.0	Introduction	303
B.1	TCI-TL XML Schema for Simple Types.....	303
B.2	TCI-TL XML Schema for Types	304
B.3	TCI-TL XML Schema for Values	306
B.4	TCI-TL XML Schema for Templates	313
B.5	TCI-TL XML Schema for Events	320
B.6	TCI-TL XML Schema for a Log.....	342
Annex C (informative):	Use scenarios	345
C.0	Introduction	345
C.1	Initialization, collecting information, logging.....	345
C.1.1	Use scenario: initialization	345
C.1.1.0	Scenario description.....	345
C.1.1.1	Sequence diagram.....	346
C.1.1.2	TTCN-3 fragment	346
C.1.2	Use scenario: requesting module parameters	346
C.1.2.0	Scenario description.....	346
C.1.2.1	Sequence diagram.....	347
C.1.2.2	TTCN-3 fragment	347
C.1.3	Use scenario: logging.....	347