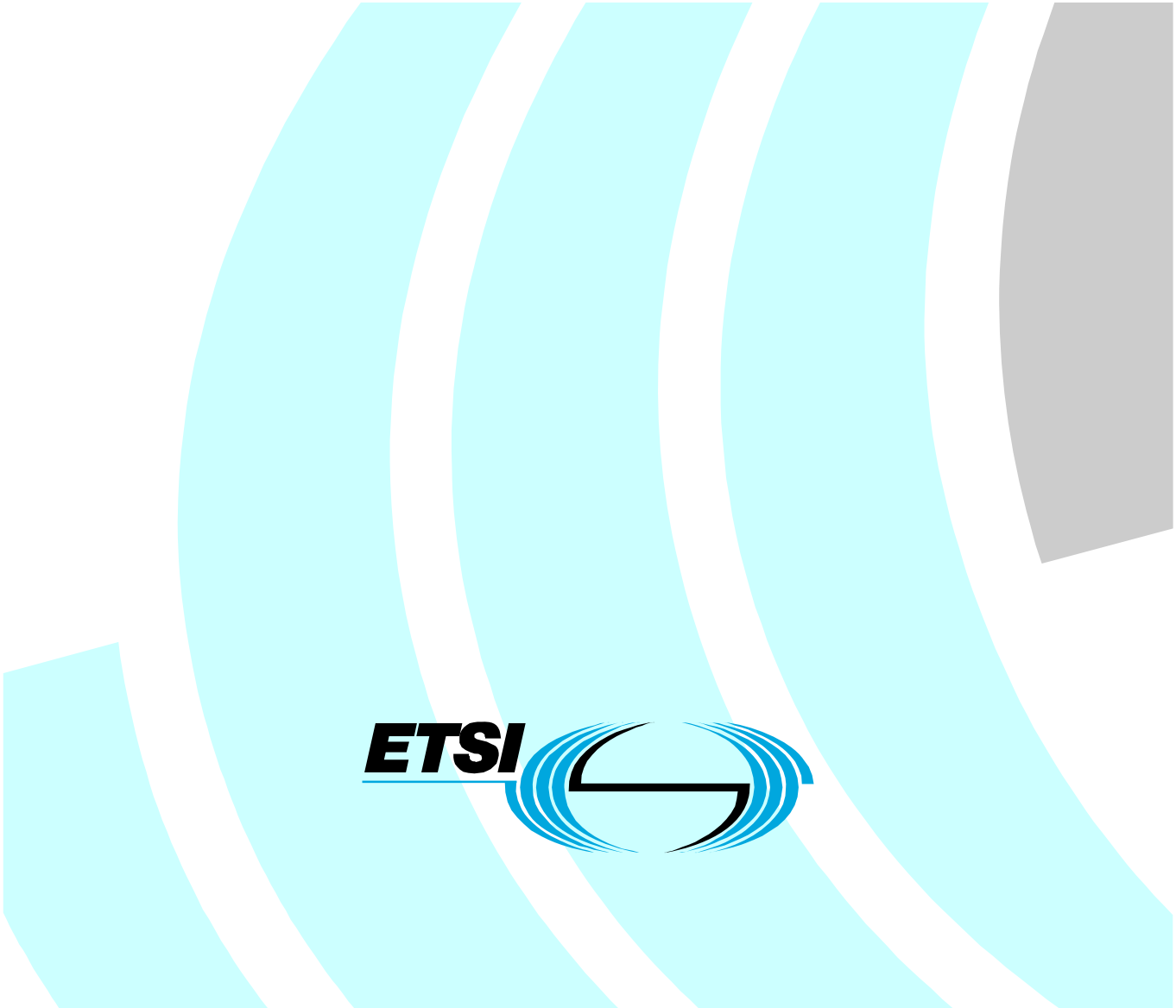
ETSI TS 186 009-3 V2.1.1 (2009-09)

Technical Specification

Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); Interworking between Session Initiation Protocol (SIP) and Bearer Independent Call Control Protocol (BICC) or ISDN User Part (ISUP); Part 3: Abstract Test Suite (ATS) and partial Protocol Implementation eXtra Information for Testing (PIXIT)



Reference
DTS/TISPAN-06025-3-NGN-R2

Keywords
ATS, BICC, IMS, interworking, ISUP, PIXIT, SIP, testing

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

Contents

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 (http://webapp.etsi.org/IPR/home.asp).

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 Technical Specification (TS) has been produced by ETSI Technical Committee Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN).

The present document is part 3 of a multi-part deliverable covering the Interworking between Session Initiation Protocol (SIP) and Bearer Independent Call Control Protocol (BICC) or ISDN User Part (ISUP), as identified below:

Part 1: "Protocol Implementation Conformance Statement (PICS)";

Part 2: "Test Suite Structure and Test Purposes (TSS&TP)";

Part 3: "Abstract Test Suite (ATS) and partial Protocol Implementation eXtra Information for Testing (PIXIT)";

1 Scope

The present document specifies the Abstract Test Suite (ATS) and partial Protocol Implementation eXtra Information for Testing (PIXIT) proforma based on the Testsuite Structure and Testpurposes defined in TS 186 009-2 [1].

The TSS&TP have been developed to test the interworking between Session Initiation Protocol (SIP) and Bearer Independent Call Control Protocol (BICC) or ISDN User Part, Profiles A and B. The ATS is sometimes referred to in the present document as "SIP-ISUP-Interworking ATS".

The test notation used in the ATS is TTCN-3 (ES 201 873-1 [8]).

The following test specification- and design considerations can be found in the body of the present document:

- the overall test suite structure;
- the testing architecture;
- the test methods and port definitions;
- the test configurations;
- the design principles, assumptions, and used interfaces to the TTCN3 tester (System Simulator);
- TTCN styles and conventions;
- the partial PIXIT proforma;
- the modules containing the TTCN-3 ATS.

Annex A provides the Partial Implementation Extra Information for Testing (IXIT) Proforma of the ATS.

Annex B provides the Testing and Test Control Notation (TTCN-3) part of the ATS.

2 References

References are either specific (identified by date of publication and/or edition number or version number) or non-specific.

- For a specific reference, subsequent revisions do not apply.
- Non-specific reference may be made only to a complete document or a part thereof and only in the following cases:
  - if it is accepted that it will be possible to use all future changes of the referenced document for the purposes of the referring document;
  - for informative references.

Referenced documents which are not found to be publicly available in the expected location might be found at http://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.

2.1 Normative references

The following referenced documents are indispensable for the application of the present document. For dated references, only the edition cited applies. For non-specific references, the latest edition of the referenced document (including any amendments) applies.

[1] ETSI TS 186 009-2 (Release 2): "Telecommunications and Internet Converged Services and Protocols for Advanced Networking (TISPAN); SIP-ISUP Interworking between the IP Multimedia (IM) Core Network (CN)subsystem and Circuit Switched (CS) networks; Part 2: Test Suite Structure and Test Purposes (TSS&TP) ".

NOTE: The latest version v2.y.z applies

[2] ETSI TS 102 351 (V2.1.1): "Methods for Testing and Specification (MTS); Internet Protocol Testing (IPT); IPv6 Testing: Methodology and Framework".

[3] ETSI TS 186 009-1 (Release 2): "Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); SIP-ISUP Interworking between the IP Multimedia (IM) Core Network (CN)subsystem and Circuit Switched (CS) networks; Part 1: Protocol Implementation Conformance Statement (PICS)".

NOTE: The latest version v2.y.z applies

[4] ETSI TS 129 163 (V7.12.0): "Digital cellular telecommunications system (Phase 2+) Universal Mobile Telecommunications System (UMTS) Interworking between the IP Multimedia (IM) Core Network (CN) subsystem and Circuit Switched (CS) networks (3GPP TS 29.163 Release 7).".

[5] ETSI TS 129 527: " Digital cellular telecommunications system (Phase 2+); Universal Mobile Telecommunications System (UMTS); TISPAN; Endorsement of the SIP-ISUP Interworking between the IP Multimedia (IM) Core Network (CN) subsystem and Circuit Switched (CS) networks [3GPP TS 29.163 (Release 7), modified] (3GPP TS 29.527 version 8.2.0 Release 8)".

[6] ITU-T Recommendation Q.2150.1 (2001): "Signalling Transport Converter on MTP3 and MTP3b".

[7] ETSI TS 102 027-3 (V3.1.1): "Methods for Testing and Specification (MTS); Conformance Test Specification for SIP (IETF RFC 3261); Part 3: Abstract Test Suite (ATS) and partial Protocol Implementation eXtra Information for Testing (PIXIT) proforma".

[8] ETSI ES 201 873-1 (V3.1.1): "Methods for Testing and Specification (MTS); The Testing and Test Control Notation version 3; Part 1: TTCN-3 Core Language".

[9] ETSI ES 201 873-5 (V3.1.1): "Methods for Testing and Specification (MTS); The Testing and Test Control Notation version 3; Part 5: TTCN-3 Runtime Interface (TRI)".

[10] ETSI ES 201 873-6 (V3.1.1): "Methods for Testing and Specification (MTS); The Testing and Test Control Notation version 3; Part 6: TTCN-3 Control Interface (TCI)".

[11] ISO/IEC 9646-1 (1992): "Information Technology - Open Systems Interconnection - Conformance Testing Methodology and Framework - Part 1: General concepts".

[12] ISO/IEC 9646-7 (1994): "Conformance testing methodology and framework - Part 7: Implementation Conformance Statement".

[13] ITU-T Recommendation Q.761 (2000): "Specifications of Signalling System No.7 ISDN User Part (ISUP)".

[14] ITU-T Recommendation Q.762 (2000): "Specifications of Signalling System No.7 ISDN User Part (ISUP)".

[15] ITU-T Recommendation Q.763 (2000): "Specifications of Signalling System No.7 ISDN User Part (ISUP); ISDN user part formats and codes".

[16] ITU-T Recommendation Q.764 (2000): "Specifications of Signalling System No.7 ISDN User Part (ISUP)".

[17] IETF RFC 3261 (2002): "SIP: Session Initiation Protocol".

[18] ITU-T Recommendation E.164: "The international public telecommunication numbering plan".

[19] ETSI EN 300 356-1 (V4.2.1): "Integrated Services Digital Network (ISDN); Signalling System No.7 (SS7); ISDN User Part (ISUP) version 4 for the international interface; Part 1: Basic services [ITU-T Recommendations Q.761 to Q.764 (1999) modified]".

[20] ITU-T Recommendation Q.931: "ISDN user-network interface layer 3 specification for basic call control".

[21] ETSI EN 300 097-1: "Integrated Services Digital Network (ISDN); Connected Line Identification Presentation (COLP) supplementary service; Digital Subscriber Signalling System No. one (DSS1) protocol; Part 1: Protocol specification ".

[22] IETF RFC 2617: "HTTP Authentication: Basic and Digest Access Authentication".

[23] IETF RFC 1321: "The MD5 Message-Digest Algorithm".

2.2 Informative references

The following referenced documents are not essential to the use of the present document but they assist the user with regard to a particular subject area. For non-specific references, the latest version of the referenced document (including any amendments) applies.

Not applicable.

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in:

- SIP/ISUP interworking reference specification is defined in TS 129 163 [4] and TS 129 527 [5];
- ISDN layer 3 reference specification is defined in EN 300 356-1 [19];
- ISDN User Part (ISUP) reference specification are defined in EN 300 356-1 [19];
- ISO/IEC 9646-1 [11] and ISO/IEC 9646-7 [12];
- ES 201 873-1 [8] (TTCN-3).

and the following apply:

Abstract Test Case (ATC): complete and independent specification of the actions required to achieve a specific test purpose, defined at the level of abstraction of a particular Abstract Test Method, starting in a stable testing state and ending in a stable testing state

Abstract Test Method (ATM): description of how an IUT is to be tested, given at an appropriate level of abstraction to make the description independent of any particular realization of a Means of Testing, but with enough detail to enable abstract test cases to be specified for this method

Abstract Test Suite (ATS): test suite composed of abstract test cases

Implementation Under Test (IUT): implementation of one or more OSI protocols in an adjacent user/provider relationship, being part of a real open system which is to be studied by testing

Means of Testing (MOT): combination of equipment and procedures that can perform the derivation, selection, parameterization and execution of test cases, in conformance with a reference standardized ATS, and can produce a conformance log

PICS proforma: document, in the form of a questionnaire, which when completed for an implementation or system becomes the PICS

PIXIT proforma: document, in the form of a questionnaire, which when completed for the IUT becomes the PIXIT

point of Control and Observation: point within a testing environment where the occurrence of test events is to be controlled and observed, as defined in an Abstract Test Method

pre-test condition: setting or state in the IUT which cannot be achieved by providing stimulus from the test environment

Protocol Implementation Conformance Statement (PICS): statement made by the supplier of a protocol claimed to conform to a given specification, stating which capabilities have been implemented

Protocol Implementation eXtra Information for Testing (PIXIT): statement made by a supplier or implementor of an IUT (protocol) which contains or references all of the information related to the IUT and its testing environment, which will enable the test laboratory to run an appropriate test suite against the IUT

SIP number: number conforming to the numbering and structure specified in ITU-T Recommendation E.164 [18]

System Under Test (SUT): real open system in which the IUT resides

3.2 Abbreviations

For the purposes of the present document, the abbreviations given in ITU-T Recommendation Q.762 [14] and the following apply:

ASP Abstract Service Primitive

NOTE: Exchanged between entities inside the TS or between the user of the ATS (operator) and the TS.

ATC Abstract Test Case
ATM Abstract Test Method
ATM Asynchroneous Transfer Mode
ATS Abstract Test Suite
BCI Backward Call Indicators
BICC Bearer Independent Call Control
CIC Circuit Identification Code
DSS1 Digital Subscriber System No. 1
EDS Encoding/Decoding System
FCI Forward Call Indicators
G/W Type 1 GateWay Type 1
G/W Type 2 GateWay Type 1
IETF Internet Engineering Task Force
ISDN Integrated Services Digital Network
ISUP ISDN User Part
IUT Implementation Under Test
IWU InterWorking Unit
LT Lower Tester
MOT Means Of Testing
MTP Message Transfer Part
NCI Nature of Connection Indicators
NGN Next Generation Network
OCN Original Called Number
PA Platform Adapter
PICS Protocol Implementation Conformance Statement
PIXIT Protocol Implementation eXtra Information for Testing
PTC Parallel Test Component
RDN Redirecting Number
RNN Redirection Number
SA System Adapter
SDP Session Description Protocol
SIP Session Initiation Protocol

SN	Signalling Node
STC	Signalling Transport Converter

NOTE: According to ITU-T Recommendation Q.2150.1 [6].

SUT	System Under Test
TC	Test Case
TCI	TTCN-3 Control Interface
TCP	Test Coordination Procedures
TD	Test Description
TE	Test Equipment
TISPAN	Telecommunications and Internet converged Services and Protocols for Advanced Networking
TL	Test Logging
TM	Test Management
TMR	Transmission Medium Requirement
TP	Test Purpose
TS	Test System
TSS	Test Suite Structure
TSS&TP	Test Suite Structure and Test Purposes
TTCN	Tree and Tabular Combined Notation
TTCN-3	Testing and Test Control Notation edition 3

4 Abstract Test Method (ATM)

4.1 Network architecture

Figures 1 and 2 show the network architecture for SIP-ISUP/BICC Interworking Units.

Figure 1 shows the network architecture for SIP-ISUP Interworking.

Figure 1: Interworking between SIP and ISUP

Figure 2 shows the network architecture for SIP-BICC Interworking.

Figure 2: Interworking between SIP and BICC

NOTE: There are 3 profiles defined for IWU: Profile A, Profile B and Profile C (out of scope of the present document). Figures 1 and 2 in clause 5 of TS 186 009-2 [1] show the substructures of the IWU for Profiles A and B in terms of gateways and signalling nodes. In the ATS the SUT (IWU) represents either a G/W Type 1 (Profile A) or the combination of G/W Type 2 and SN (Profile B).

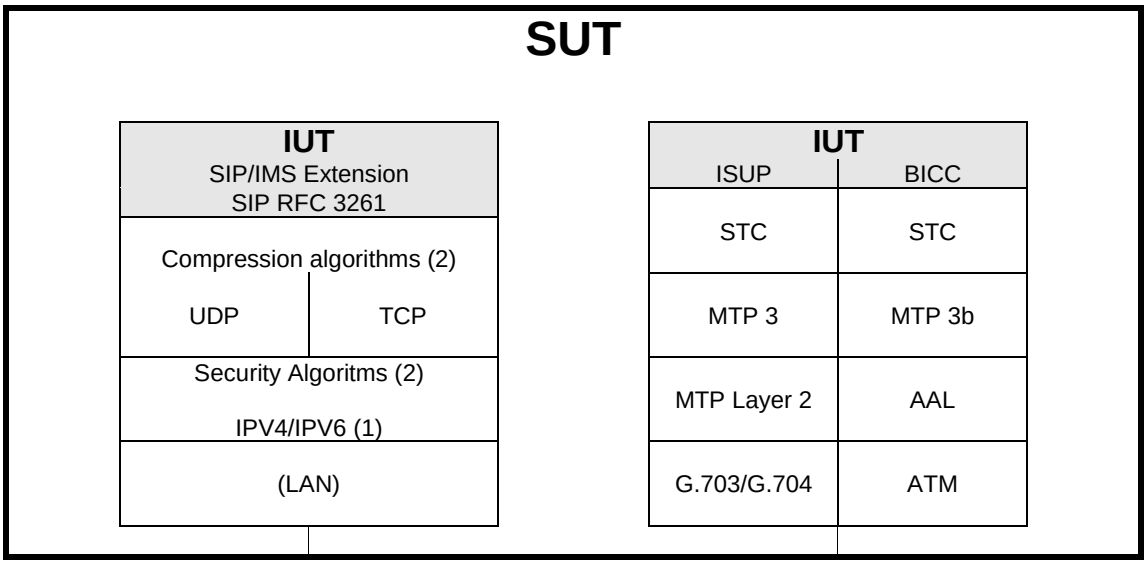
4.2 Protocol architecture

Figures 1 and 2 above show that there are 2 interfaces of the IWU (representing the SUT in the testing environment described in the present document): a SIP interface and an ISUP- or BICC interface.

Since the ISUP and BICC protocols are very similar (the latter one being derived from ISUP), they are treated here as one protocol.

NOTE: No signalling is used within the SIP-ISUP-Interworking ATS to control the ATM bearer in case of BICC (ASPs are used).

Figure **Error! Bookmark not defined.** shows the protocol architecture in 2 branches.



NOTE 1: Both IPV4 and IPV6 addressing should be supported.
NOTE 2: Optional security and compression algorithms should be supported.

Figure 3: Protocol architecture of the SIP-ISUP-Interworking ATS

4.3 Test architecture

4.3.1 Interconnection of TS and SUT

Figure 4 shows the interconnection of TS and SUT in terms of signalling message flows.

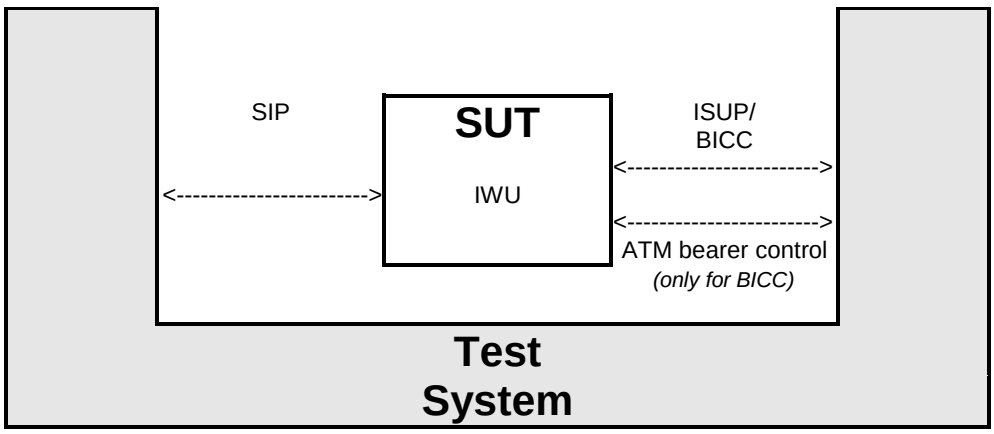


Figure 4: Interconnection of TS and SUT

4.3.2 Test system architecture

4.3.2.1 General

Test systems that implement this ATS shall conform to the requirements as defined in this clause.

4.3.2.2 Structure

An abstract architecture for a test system (TS) implementing a TTCN-3 ATS is displayed in figure 5 and also stated in ES 201 873-5 [9].

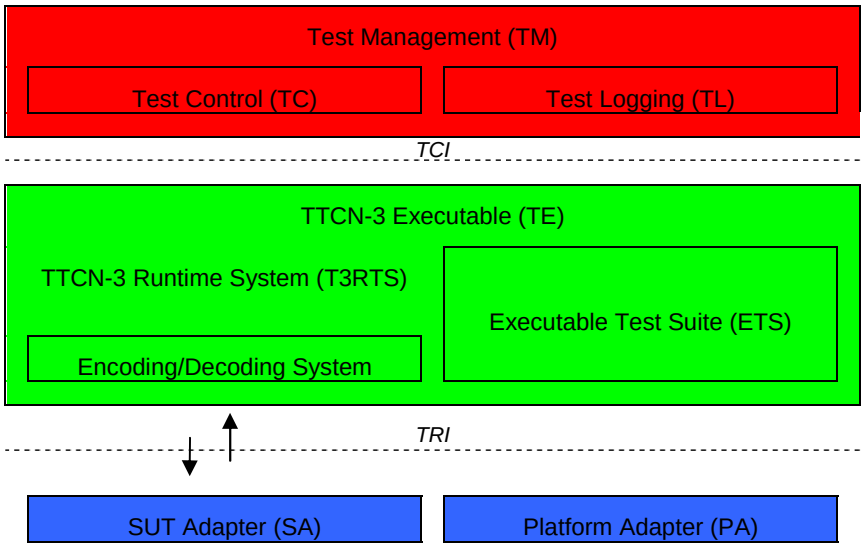


Figure 5: Abstract Test System Architecture

A TS has two interfaces, the TTCN-3 Control Interface (TCI) and the TTCN-3 Runtime Interface (TRI), which specify the interface between Test Management (TM) and TTCN-3 Executable (TE) entities, and TE, SUT Adapter (SA) and Platform Adapter (PA) entities, respectively. Out of these two interfaces the TRI has been standardized in ES 201 873-5 [9], whereas the specification and implementation of the TCI is in ES 201 873-6 [10].

The part of TS that deals with interpretation and execution of TTCN-3 modules, i.e. the Executable Test Suite (ETS), is shown as part of the TTCN-3 Executable (TE). This ETS corresponds either to the executable code produced by a TTCN-3 compiler or a TTCN-3 interpreter from the TTCN-3 ATS in a TS implementation. The remaining part of the TS, which deals with any aspects that cannot be concluded from information being present in the TTCN-3 ATS alone, can be decomposed into Test Management (TM), SUT Adapter (SA) and Platform Adapter (PA) entities. In general, these entities cover a TS user interface, test execution control, test event logging, communication of test data with the SUT, and timer implementation.

The part of SA used for SIP message transfer shall implement the TRI adaptation as well as the SIP transport protocol architecture described in clause 4.2.

The Encoding/Decoding System (EDS) entity, as far as applied to SIP messages, with the TE and Test Logging (TL) entity within the TM shall comply with the conventions defined in clause 4.3.2 of TS 102 027-3 [7].

The part of SA used for ISUP/BICC message transfer shall implement the *TRI* adaptation as well as the ISUP/BICC transport protocol architecture described in clause 4.2. For BICC, in addition, the ATM bearer control shall be implemented.

The Encoding/Decoding System (EDS) entity, as far as applied to ISUP/BICC messages, shall comply with the conventions and requirements defined in the following clauses.

4.3.2.3 Interaction between TTCN-3 Executable (TE) and SUT Adapter (SA)

4.3.2.3.1 Control of the SUT Adapter (SA) by using ASPs

Table 1 lists the ASPs used in the SIP-ISUP-Interworking ATS. Detailed descriptions of the ASPs together with their parameters follow.

Table 1: List of ASPs

ASP Name	Short description
InitializeIsupBicc_req	Initialize ISUP/BICC part of the test system.
InitializeIsupBicc_cnf	Answer whether all necessary ISUP/BICC test system initializations have been successfully performed.
ISUP_BICC_MSG_req	Used to send an ISUP/BICC message.
ISUP_BICC_MSG_ind	Used to receive an ISUP/BICC message.
BearerSetup_req	For BICC: request TS to setup the bearer connection between TS and SUT.
BearerSetup_acc	For BICC: answer to BearerSetup_req.
BearerSetup_ind	For BICC: indication that the bearer has been setup.
BearerRelease_req	For BICC: request to release established bearer connection.
BearerRelease_cnf	For BICC: confirmation that the requested bearer is released.
BearerRelease_ind	For BICC: indication that the bearer has been released (when no BearerRelease_req has been issued before).
s_IsupBicc_conversation	Check that conversation is possible on the bearer.
s_IsupBicc_ringing	Check that ringing occurs.

Tables 2 to 13 contain the descriptions of the ASPs used in the present document, including the ASP parameters (if any) and the types of values these may assume. No ASP parameter is optional.

Table 2: ISUP_BICC_MSG_req ASP structure

ASP Name:	ISUP_BICC_MSG_req	
Port:	sysPort	
Direction:	TE->SA	
Description:	ASP used to send an ISUP/BICC message.	
Parameter	**Type**	**Description**
isupBiccSelection	SelectIsupOrBicc	Selector used to distinguish between ISUP and BICC testing. "00000000"B means "ISUP" and any other value means "BICC".
serviceIndicatorOctet	ServiceIndicatorOctet	The contents of this ASP parameter is only evaluated in SA if ISUP has been selected in "isupBiccSelection".
routingLabel	RoutingLabel	The contents of this ASP parameter is only evaluated in SA if ISUP has been selected in "isupBiccSelection".
circuitIdentityCode	CircuitIdentityCode	The contents of this ASP parameter is only evaluated in SA if ISUP has been selected in "isupBiccSelection".
callInstanceCode	CallInstanceCode	The contents of this ASP parameter is only evaluated in SA if BICC has been selected in "isupBiccSelection".
iSUP_BICC_MSG	ISUP_BICC_MSG	ISUP_BICC_MSG is a union over all ISUP/BICC message bodie types, where a message body starts with the "message type" field. This body is common for ISUP and BICC messages. When using this ASP, a particular message(body) template is selected from the union for transmission.
Comments: The SA takes from the ASP, depending on the value of parameter "isupBiccSelection", either the ordered combination of "serviceIndicatorOctet", "routingLabel" and "circuitIdentityCode" (ISUP), or "callInstanceCode" (BICC"), puts it in front of encoded parameter "iSUP_BICC_MSG", and sends the so constructed message at the ISUP or BICC interface respectively.		

Table 3: ISUP_BICC_MSG_ind ASP structure

ASP Name: ISUP_BICC_MSG_ind **Port:** sysPort **Direction:** SA->TE **Description:** ASP used to receive an ISUP/BICC message.		
Parameter	**Type**	**Description**
isupBiccSelection	Bit8	Selector used to distinguish between ISUP and BICC testing. "00000000"B means "ISUP" and any other value means "BICC".
serviceIndicatorOctet	ServiceIndicatorOctet	The contents of this ASP parameter is only evaluated in TE if ISUP has been selected in "isupBiccSelection".
routingLabel	RoutingLabel	The contents of this ASP parameter is only evaluated in TE if ISUP has been selected in "isupBiccSelection".
circuitIdentityCode	CircuitIdentityCode	The contents of this ASP parameter is only evaluated in TE if ISUP has been selected in "isupBiccSelection".
callInstanceCode	CallInstanceCode	The contents of this ASP parameter is only evaluated in TE if BICC has been selected in "isupBiccSelection".
iSUP_BICC_MSG	ISUP_BICC_MSG	ISUP_BICC_MSG is a union over all ISUP/BICC message bodie types, where a message body starts with the "message type" field. This body is common for ISUP and BICC messages. When using this ASP, a particular message(body) template is selected from the union for receive matching.
Comments: The SA takes from the received message, depending on the value of parameter "isupBiccSelection", either the ordered combination of "serviceIndicatorOctet", "routingLabel" and "circuitIdentityCode" (ISUP), or "callInstanceCode" (BICC"), and puts it into the associated ASP parameters. The complementary ASP parameters "callInstanceCode" (ISUP) and combination of "serviceIndicatorOctet", "routingLabel" and "circuitIdentityCode" (BICC) are filled by the SA with "0"-bits according to the lengths of their types. The TE does not evaluate the contents of the complementary parameters (but needs the correct lengths to identify the start of "iSUP_BICC_MSG". The received message (body) is put by the SA into parameter "iSUP_BICC_MSG" and is matched in the ATS with an according receive template.		

Table 4: InitializeIsupBicc_req ASP structure

ASP Name: InitializeIsupBicc_req **Port:** IsupBiccPort **Direction:** TE->SA **Description:** Initialize ISUP/BICC part of the test system.		
Parameter	**Type**	**Description**
isupBiccSelection	Bit8	Selector used to distinguish between ISUP and BICC testing. "00000000"B means "ISUP" and any other value means "BICC".
ts_pointCode	Bit14	Signalling point code of the TS (ISUP).
sut_pointCode	Bit14	Signalling point code of the SUT (ISUP).
ts_address_sip	octetstring	Address (e.g. IP) of the TS (SIP side). The use of this address is to enable the TS to communicate with the SUT at the SIP side to establish and maintain the lower layer connections.
ts_address_isup_bicc	octetstring	Address (e.g. IP) of the TS (ISUP/BICC side). The use of this address is to enable the TS to communicate with the SUT at the ISUP/BICC side to establish and maintain the lower layer connections.
sut_address_sip	octetstring	Address (e.g. IP) of the SUT (SIP side). The use of this address is to enable the TS to communicate with the SUT at the SIP side to establish and maintain the lower layer connections.
sut_address_isup_bicc	octetstring	Address (e.g. IP) of the SUT (ISUP/BICC side). The use of this address is to enable the TS to communicate with the SUT at the ISUP/BICC side to establish and maintain the lower layer connections.
Comments: This ASP is used at the beginning of each test case to initiate the necessary initialization of the test system, particularly the interfaces to the SUT. If parameter isupBiccSelection indicates "bicc", the values of parameters "ts_pointCode" and "sut_pointCode" shall be ignored by the SA. If parameter isupBiccSelection indicates "isup", the values of parameters "ts_address_isup_bicc" and "sut_address_isup_bicc" may be ignored, if they are not necessary. Among the initializing actions there shall be: a) Verification that the ISUP/BICC link is operable between SUT and TS. b) Verification that the TS is ready to send and receive SIP messages. NOTE: It is a matter of TS implementation whether the TS, upon this request, sets up and initializes lower layer connections, if these are not setup. Other initialization actions may be TS-specific.		

Table 5: InitializeIsupBicc_cnf ASP STRUCTURE

ASP Name: InitializeIsupBicc_cnf **Port:** sysPort **Direction:** LT->TTCN **Description:** Answer whether all necessary ISUP/BICC test system initializations have been successfully performed. The result can be positive or negative. The result will be positive only if the TS is able to send and receive messages at the ISUP/BICC-interface of the SUT.		
Parameter	**Type**	**Description**
result	boolean	Indicating success or non-success of the whole initialization.
Comments:		

Table 6: BearerSetup_req ASP structure

ASP Name: BearerSetup_req **Port:** IsupBiccPort **Direction:** TE->SA **Description:** For BICC: request TS to setup the bearer connection between TS and SUT.		
Parameter	**Type**	**Description**
cic	CallInstanceCode	Call Instance Code identifying the bearer connection.
Comments:		

Table 7: BearerSetup_acc ASP structure

ASP Name: BearerSetup_acc **Port:** IsupBiccPort **Direction:** SA->TE **Description:** For BICC: answer to BearerSetup_req. The answer can be positive (bearer connection setup successful) or negative (bearer connection setup failed).		
Parameter	**Type**	**Description**
result	boolean	The answer is positive when the bearer connection setup was successful and negative when the bearer connection setup failed.
Comments:		

Table 8: BearerSetup_ind ASP structure

ASP Name: BearerSetup_ind **Port:** IsupBiccPort **Direction:** SA->TE **Description:** For BICC: indication that the bearer has been setup.		
Parameter	**Type**	**Description**
cic	CallInstanceCode	Call Instance Code identifying the bearer connection.
Comments:		

Table 9: BearerRelease_req ASP structure

ASP Name: BearerRelease_req **Port:** bcPort **Direction:** TE->SA **Description:** For BICC: request to release the established bearer connection.		
Parameter	**Type**	**Description**
cic	CIC	Circuit identity code identifying the bearer connection.
Comments:		

Table 10: BearerRelease_cnf ASP structure

ASP Name: BearerRelease_cnf **Port:** bcPort **Direction:** SA->TE **Description:** For BICC: confirmation that the requested bearer is released.		
Parameter	**Type**	**Description**
result	boolean	Indication of whether the bearer is successfully released.
Comments: At release collision the result is still "true".		

Table 11: BearerRelease_ind ASP structure

ASP Name:	BearerRelease_ind	
Port:	bcPort	
Direction:	SA->TE	
Description:	For BICC: indication that the bearer has been released (when no BearerRelease_req has been issued before).	
Parameter	**Type**	**Description**
cic	CIC	Circuit identity code identifying the bearer connection.
Comments:		

Table 12: s_IsupBicc_conversation ASP structure

ASP Name:	s_IsupBicc_conversation	
Port:	operatorPort_IsupBicc	
Direction:	SA-<>TE	
Description:	Check that conversation is possible on the through-connected bearer.	
Parameter	**Type**	**Description**
text	charstring	Request operator to check the conversation.
answer	boolean	Check result entered by the operator.
Comments: This ASP has been implemented as a signature. "text" is an "input" parameter and "answer" is an output parameter.		

Table 13: s_IsupBicc_ringing ASP structure

ASP Name:	s_IsupBicc_ringing	
Port:	operatorPort_IsupBicc	
Direction:	SA-<>TE	
Description:	Check that occurs on the through-connected bearer.	
Parameter	**Type**	**Description**
text	charstring	Request operator to check the ringing.
answer	boolean	Check result entered by the operator.
Comments: This ASP has been implemented as a signature. "text" is an "input" parameter and "answer" is an output parameter.		

4.3.2.3.2 Sending and receiving SIP and ISUP/BICC messages

4.3.2.3.2.1 General

Before starting a test case, the SA shall be prepared to provide the transport of SIP and ISUP/BICC messages by establishing appropriate connections on the lower layers (see figure **Error! Bookmark not defined.**).

4.3.2.3.2.2 Sending and receiving SIP/IMS messages

In order to forward messages received into the SA to the test suite and to send them to the SUT a clear and unique association between the TTCN-3 TSI ports and the real IP and port addresses used by the SUT is needed during test execution. The SA retrieves this information via values of TTCN-3 module parameters, i.e. PIXITs, and mappings to TSI ports, i.e. triMap operation invocations. TSI port names are the main source for the relating TSI ports with SUT IP addresses and ports.

The following table provides the relationships for TSI ports and SUT IP addresses and ports:

Table 14: TSI port mappings

TSI port	SUT (IP address, Port Id)	Test system (IP address, Port id)
IMSCN1	PX_IMS_SUT_IMGCF_IPADDR, PX_IMS_SUT_IMGCF_PORT	PX_IMS_TS_ICSCF_IPADDR, PX_IMS_TS_ICSCF_PORT
NOTE 1: TSI portnames are defined in SipIsup_TestSystem module as part of the ImsComponent type. Module parameters for the address information are defined in LibIms_PIXIT module (see clause 5.3.1 for complete list of modules). NOTE 2: For test configuration a TTCN-3 configuration functions has been implemented with the required mapping and unmapping statements (see clause 5.3.1 for complete list of modules), e.g. f_cf_imsIsupUp map one Ims related port of the test system to the SUT and one Isup port to the SUT IP/E1 module.		

4.3.2.3.2.3 Security and messages compression feature

Security transport layer, and signalling compression may be used transparently to the ATS.

4.3.2.3.2.4 Additional SA constraints

In order to execute this test suite the SA should support:

- communication channel handling (at least UDP and possibly also TCP)
- IPv4 transport.

4.3.2.3.3 Encoding/Decoding System requirements

4.3.2.3.3.1 Encoding/Decoding System requirements for basic SIP messages/headers

SIP is a text-based protocol that allows different syntactical presentations of the same information. In general, an implementation of this ATS should use a EDS to parse received encoded messages into TTCN-3 type structures and values, and encode structured TTCN-3 type structures and values into encoded messages. This EDS is not part of the ATS. Still all encoded messages, i.e. the messages as they are transmitted by the SA to or received by the SA from the SUT, shall be logged.

The following terms shall be used for the conventions defined below:

Syntactic delimiter	syntactic delimiters are characters like "=" or ";" that are used to separate encoded values.
LWS	linear white spaces as defined in RFC 3261 [17].
Parameter name	name of header parameters as defined in RFC 3261 [17].
Parameter value	the value of a parameter as defined in RFC 3261 [17].
Undefined method	an undefined method is a method other than: "INVITE", "ACK", "OPTIONS", "BYE", "CANCEL" and "REGISTER".
Undefined header	an undefined header is a header other than general-header, entity-header, request-header and response header as defined in RFC 3261 [17].
Unexpected header	an unexpected header is a header, which shall not be present in a specific request message. This definition complies to the definition of NOT APPLICABLE in RFC 3261 [17], section 20 for request messages.

Decoder requirements

TTCN-3 fields should not contain syntactic delimiters like white space, semicolon, equal characters etc. in fully decoded fields. Instead the information provided by a parser shall be used to build the decoded message in TTCN-3. Decoded messages shall use the TTCN-3 enumeration types where ever appropriate, e.g. for the method and the header field name.

For **charstring** fields the following decoding rules shall be applied by the EDS:

1) Subsequent LWS shall compress to a single space character " ".

2) Decoded parameter names shall use only lower case letters.

3) Parameter values containing an integer value shall be decoded to a TTCN-3 integer value where a TTCN-3 **integer** type is used for a SIP parameter value.

The following decoding rules shall be applied by the EDS to each received message in the following order:

1) In case a request message indicating an undefined method is received by the test system, the message shall not be passed in the TE to the ETS. However the message is subject to logging as defined in clause 4.3.2.3.3 ("Logging conventions").

2) In case an undefined header has been received the header field shall be decoded as **UndefinedHeader** field.

RFC 3261 [17] allows for multiple header field values of the same kind to either arrive in one or multiple occurrences of the corresponding header field. The SIP ATS has been written assuming only the first format. Therefore, should the EDS receive multiple header fields of the same kind in a SIP message, e.g. of a Via header field, it shall convert them into the equivalent single header field with multiple values. This can be achieved by adding the value of, e.g. the second received Via header field as the last value to the value(s) of the first Via header field.

Encoder requirements

Encoders shall follow all encoding rules that are defined in RFC 3261 [17] when encoding structured values received from templates. This applies in particular to but it is not restricted to section 7.3.1 of RFC 3261 [17].

Values of type **Raw** shall be send to the SUT without any modification.

4.3.2.3.3.2 Encoding/Decoding System requirements for ISUP/BICC

4.3.2.3.3.2.1 General

ISUP/BICC messages are sent and received in the test suite by embedding them in ASPs ISUP_BICC_MSG_req and ISUP_BICC_MSG_ind respectively.

The ASPs contain all information to route the ISUP/BICC messages to/from the SUT.

ISUP messages and parameters are structured by using tables (see ITU-T Recommendation Q.763 [15]).

NOTE 1: The term "parameter" is used as defined in the ISUP protocol context. It corresponds e.g. to the term "Information Element" in other protocols.

All structure elements are bitstrings, hexstrings or octetstrings.

For ISUP message/parameter elements a specific way is defined to extend bitstring- or hexstring elements over octet boundaries. This is known as "LowToHigh encoding", as shown in the following example:

EXAMPLE 1:

Coding of element "Circuit Identity Code" (CIC), consisting of 12 bits.

Octet #	**Bit 8**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**
Octet 1	CIC (LSB)							
Octet 2	spare				CIC (MSB)			

Figure 6: Bit field structure of the "CIC" parameter

The 8 least significant bits of the CIC value fill octet 1 (the least significant bit of CIC is assigned to bit 1 of octet 1), and the 4 most significant bits of the CIC value fill the lower 4 bits of octet 2.

NOTE 2: When a bitstring (hexstring) is presented as a sequence of bits (semi-octets) from left to right, the leftmost bit (semi-octet) is the most significant and the rightmost bit (semi-octet) is the least significant.

EXAMPLE 2:

Adress digits

Several ISUP parameters have an element "Adress digits", where the individual digits are BCD-encoded (i.e. e.g. digit "0" is encoded as "0000"B, digit "9" is encoded as "1001"B.

When an address string is given as a sequence of ASCII digits, as a user would type them in, e.g. "0123456789", the encoded value is as shown on figure 7.

Octet #	**Bits 8 7 6 5**	**Bits 4 3 2 1**
Octet 1	0001	0000
Octet 2	0011	0010
Octet 3	0101	0100
Octet 4	0111	0110
Octet 5	1001	1000

Figure 7: Hex (BCD) field structure of an "address digits" element

This also corresponds to a "LowToHigh" encoding. In this particular case however, for the sake of ATS user convenience, a conversion function is used in the ATS in the following way:

- All module parameters containing address digits have type "charstring" (resp. IA5String), which means that the user enters digits as ASCII characters "1", "2" and so on.
- Inside the address parameter templates the conversion function converts the ASCII string into a BCD-coded octetstring, taking also care of:
  - "sending complete" digit (only applicable to the Called Party Number);
  - filler (final semi-octet, if the number of coded digits is odd.

The encoding of octetstrings however is not LowToHigh, as shown in the following example:

EXAMPLE 3:

octetstring value

The octetstring value "01234ABCDE"O is encoded as shown on figure 8.

Octet #	**Bits 8 7 6 5**	**Bits 4 3 2 1**
Octet 1	0000	0001
Octet 2	0010	0011
Octet 3	0100	1010
Octet 4	1011	1100
Octet 5	1101	1110

Figure 8: Octetstring field encoding

4.3.2.3.3.2.2 Decoding of parameters containing strings of variable length

Typical fields addressed here are e.g. the "adress digits" field in the "Called Party Number" parameter, or the "diagnostics" field in the "Cause Indicators" parameter.

The above mentioned strings of variable length are the last elements of the related parameter, which has a preceding length field. A "real" decoder deduces the length (and thereby the value) of such fields from the value of the "length" field of the parameter and the position of the decoder where the field starts.

The decoder of the test system shall also be able to decode such fields when the value of the template is "?" or "*".

In order to support this encoding the relevant types have a trailing "with { encode ..." statement, like in the following example (Called Party Number):

EXAMPLE 4:

```
....
 with { encode (paramLen) "tag=""CDN_paramLen"";";
       encode (addressSignals) "length=valueOf(getTag(""CDN_paramLen"")).toInt()-2;"; }
End
```

4.3.2.3.3.2.3 Decoding of parameters containing extension bits

Some parameters transport IEs from the DSS1 protocol (ITU-T Recommendation Q.931 [20]), such as the Bearer Capability IE:

- IEs of this kind contain extension bits specifying the presence of succeeding octets.
- The decoder shall be able to evaluate the extension bits to deduce the presence of optional octets in case wildcards "?" or "*" are specified in templates of such IEs.

4.3.2.3.3.2.4 Receipt of unknown ISUP/BICC messages

Unknown messages in this context are messages not defined in the dated version of ITU-T Recommendation Q.763 [15] referred to in the present document.

Unknown messages shall not be passed to TE by the test system.

4.3.2.3.3.2.5 Receipt of unknown ISUP/BICC parameters

Unknown parameters in this context are parameters not defined in the dated version of ITU-T Recommendation Q.763 [15] referred to in the present document, or defined parameters not being assigned in ITU-T Recommendation Q.763 [15] to the particular received message carrying this parameter.

Unknown parameters shall not be passed to TE by the test system (i.e. they shall be removed from the carrying known message before passing this message to TE).

4.3.2.3.3.2.6 Ordering of optional ISUP/BICC parameters and multiple occurrence of parameters

According to ITU-T Recommendation Q.763 [15] optional parameters may occur in any order in a message, and some (few) parameters may occur more than once.

For the controlled test environment specified in this ATS the following assumption has been made:

- Parameters that may occur more than once appear at most two times in a message.

For each message that may contain optional parameters the list of parameters has been specified in the ATS as a **set**.

The decoder shall be able to decode the parameters of a received message correctly, even if they appear in an odder different from the one specified in the message template (and type).

4.3.2.3.3.2.7 Platform adaptation requirements

For the execution of this test suite implementations of the following external functions have to be provided (cp. module LibSip_Steps):

1) *rndStr() return charstring;*
   returns a random charstring;

2) *putInLowercase(charstring par_string) return charstring;*
   returns the equivalent string in lower case:

3) *getIpAddr(charstring host_name) return charstring;*
   resolves a domain name to its equivalent IPv4 address;

4) *calculateDigestResponse(charstring nonce, charstring cnonce, charstring user, charstring realm, charstring passwd, charstring alg, charstring nonceCount, charstring method, charstring qop, charstring URI, charstring HEntity) return charstring*;
generates a digest response according to RFC 2617 [22] (HTTP Authentication: Basic and Digest Access Authentication), and RFC 1321 [23] The MD5 Message-Digest Algorithm. (See RFC 2617 [22], section 5 Sample implementation, for example usage, as the signature of calculateDigestResponse is according to the example given in the RFC.).

4.3.2.3.3.3 Logging conventions

As the ATS defines on an abstract level the message exchange between TS and SUT the messages encoded messages send and received shall be logged. The TM entity in the TS shall provide access to this log.

5 The ATS development process

5.1 Requirements and Test Purposes

For each test purpose there is a table defined in clause 6 of TS 186 009-2 [1]. The requirements applicable to this TP are given by a reference to RFC 3261 [17] (SIP) and TS 129 163 [4] or TS 129 527 [5] (ISUP). There are no explicit formulations of requirements.

NOTE: During the ATS development comments have been made on TS 186 009-2 [1] (TSS&TP) and TS 186 009-1 [3] (PICS). These are not referred to in detail in the present document. Part of the comments related to inconsistent namings of the TP tables in TS 186 009-2 [1]. Re-naming of the TP tables was agreed by TISPAN. Annex C contains a list showing the pairings of original TP identifiers in TS 186 009-2 [1] and the naming used in the ATS.

5.2 ATS structure

5.2.1 Test case grouping

The ATS structure defined in table 15 is based on the structuring of Test Purposes in clause 5 of TS 186 009-2 [1]. The group names in columns 1 to 3 of table 15 are those assigned in the ATS; they are based on the names provided in clause 5 of TS 186 009-2 [1], but use the naming conventions defined for the ATS (see clause 5.3.2.2).

Table 15: ATS structure

Group	Subgroup	Sub-Subgroup	Group Index
Basic call			
	SIP-ISUP		1
		Sending of the Initial address message (IAM)	101
		Sending of the Subsequent address message (SAM)	102
		Sending of COT	103
		Receipt of the Address complete message (ACM)	104
		Receipt of the Call progress message (CPG)	105
		Receipt of the answer message (ANM)	106
		Receipt of the Connect message (CON)	107
		Receipt of the Release message (REL)	108
		Autonomous release at I-MGCF	1081
		Receipt of the BYE, CANCEL message / sending of a REL message	109
		Receipt of Reset circuit message (RSC), Circuit group reset message (GRS) or Circuit group blocking message (CGB) with the indication hardware failure oriented	110
		Receipt of the SUSPEND Message (SUS)	111
		Receipt of the RESUME Message (RES)	112

Group	Subgroup	Sub-Subgroup	Group Index
	ISUP-SIP		3
		Sending of the INVITE message	301
		Receipt of the Subsequent address message (SAM)	302
		Sending of the Address complete message (ACM)	303
		Sending of the Call progress message (CPG)	304
		Sending of the answer message (ANM)	305
		Sending of the Connect message (CON)	306
		Receipt of the Release message (REL)	307
		Sending of the Release Message (REL)	308
		Autonomous release	309
		Receipt of Reset circuit message (RSC)	310
		Receipt of Circuit group reset message (GRS)	311
		Receipt of Circuit group blocking message (CGB) with the indication hardware failure oriented	312
Supplementary Services			
	SIP-ISUP		5
		Calling Line Identification (CLI)	501
		Call Hold (HOLD)	502
		Terminal Portability (TP)	503
		Conference Calling (CONF)	504
		Three-Party (3PTY)	505
		Connected Line Identification (COL)	506
		Malicious call identification (MCID)	507
		Subaddressing (SUB)	508
		Call Diversion (CDIV)	509
		Call Waiting (CW)	510
		User to User Signalling (UUS)	511
		Explicit Call transfer (ECT)	512
		Completion of Call to Busy Subscriber (CCBS)	513
		Completion of Calls on No reply (CCNR)	514
		Anonymous Call Rejection (ACR)	515
		Closed user group (CUG)	516
	ISUP-SIP		6
		Calling Line Identification (CLI)	601
		Call Hold (HOLD)	602
		Terminal Portability (TP)	603
		Conference Calling (CONF)	604
		Three-Party (3PTY)	605
		Connected Line Identification (COL)	606
		Subaddressing (SUB)	607
		Closed User Group (CUG)	608
		Call Diversion (CDIV)	609
		User to User Signalling (UUS)	610
		Explicit Call transfer (ECT)	611
		Anonymous Call Rejection (ACR)	612
		Call waiting (CW)	613
		Malicious call identification (MCID)	614
NOTE: All subgroups except for "Autonomous release at I-IWU"/1081 use 3 digits to number test cases inside this subgroup. For "Autonomous release at I-IWU"/1081 only 2 digits are available.			

5.2.2 Test case identifiers

The test case names are built up according to the following scheme:

<"TC">"_"<Group index>"_"<TC number>

where:

a) double quotes (") are used to enclose literal strings;

b) <Group path index> is the 3-digit number in column 4 of table 15 (which uniquely identifies the path of groups/subgroups);

c) <TC number> is a running 3-digit decimal number, starting in each subgroup path with "001".

NOTE 1: See note in table 15 for the one exception from this rule and its reason.

EXAMPLE:

TC_101_001:

i) the identifier has Group index "101", i.e. it is in the subgroup having complete path: BasicCall/SIP-ISUP/Sending of the Initial address message (IAM)/

ii) the identifier is the first test case of this group/subgroup.

NOTE 2: This naming scheme provides a 1-1 correspondence of TP identifiers as defined in TS 186 009-2 [1] and test case names.
The TP identifier of TC_101_001 is TP101001. See however annex C for the list of re-named test purposes.

5.3 ATS specification framework

5.3.1 ATS Library

For this interworking ATS there are 2 applicable base protocols:

a) SIP protocol (RFC 3261 [17]); and

b) ISUP protocol (ITU-T Recommendation Q.76n series [13] to [16], plus associated standards for supplementary services, etc.).

Since e.g. the data structures of these 2 base protocols are independent, and other objects like test cases are common, the TTCN-3 library modules are basically organized as:

- ATSCommon modules (generated for the present ATS);
- LibIms modules;
- LibSip modules;
- ISUP modules;
- LibCommon modules (taken from an improved version of TS 102 351 [2]).

Table 16 shows the organization of the ATS as library of modules.

Table 16: Library of modules

Module Class	Module Id	Description
AtsCommon	SipIsup_PICS	Module Parameter declarations associated with PICS.
	SipIsup_PIXITS	SIP-ISUP common Module Parameter declarations associated with PIXIT.
	SipIsup_Testcases	Test case defintions
	SipIsup_TestConfiguration	Functions which implement the configuration of the SUT adapter and mapping of test components for establishing and tearing down different test configurations.
	SipIsup_TestExecution	Module control: execute test cases depending on selection conditions; repeat parameterized test cases based on the "Variant-tables" defined in the test prose.
	SipIsup_TestSystem	Common functions, components, ASPs controlling the test system.
	SipIsup_IMS_TCFunctions	Test case functions
LibIms	LibIms_PIXITS	IMS specific common Module Parameter (e.g. addresses related to SUT components and TS) declarations associated with PIXIT.
	LibIms_Interface	IMS component
	LibIms_SIPTypesAndValues	IMS specific user and interface specific profile data (see note 3)
	LibIms_Templates	Modified templates with IMS specific header fields
	LibIms_Steps	functions using IMS specific types
LIbSip	LibSip_PIXITS	SIP general common Module Parameter (e.g. SDP/SIP procedure options) declarations associated with PIXIT.
	LibSip_Interface	SIP component
	LibSip_SIPTypesAndValues	SIP message types and constants, simple user profiles (see note 3)
	LibSip_SDPTypes	SDP types and constants
	LibSip_Templates	Basic and modified templates with SIP specific header fields
	LibSip_Steps	SIP specific behaviour function library
	LibSip_XMLTypes	XML types for SIP tests
	XSDAUX	Basic types used in XML
IsupAts	SipIsup_ISUP_Constants	Constant declarations, mostly corresponding to field values of ISUP messages/parameters.
	SipIsup_ISUP_ModuleParams	Module parameters (all associated with PIXIT).
	SipIsup_ISUP_ParamTypes	ISUP data types (parameter types according to ITU-T Recommendation Q.763 [15] and types required for ASPs).
	SipIsup_ISUP_MsgTypes	ISUP data types (message types according to ITU-T Recommendation Q.763 [15] and ASP type declarations).
	SipIsup_ISUP_ParamTemplates	Templates for ISUP message parameters.
	SipIsup_ISUP_MsgTemplates	Templates for ISUP messages.
	SipIsup_ISUP_Steps	Test step declarations, including preambles, postambles and default.
	SipIsup_ISUP_TCFunctions	Test case functions running on the Isup/Bicc component.
LibCommon	LibCommon_AbstractData	Generic data types for a stack and its operations.
	LibCommon_BasicTypesAndValues	Basic type and value definitions (integer and Boolean).
	LibCommon_DataStrings	Bit and Octet string types.
	LibCommon_Sync	Co-ordination/synchronization of test components.
	LibCommon_TextStrings	Basic character and string types with fixed length.
	LibCommon_Time	Time handling functions and moduleparameter.
	LibCommon_VerdictControl	Basic functions for setting of test component verdicts.

5.3.2 Use of TTCN-3

5.3.2.1 General

TTCN-3 as defined in ES 201 873-1 [8] is used as ATS specification language.

A number of requirements have been identified for the development and production of the TTCN-3 specification for the SIP/ISUP Interworking ATS:

- Top-down design.
- A uniquely defined testing architecture and test method.

- Uniform TTCN-3 style and naming conventions.
- TTCN-3 is human-readability.
- TTCN-3 specification is feasible, implementable, compilable and maintainable.
- Test cases shall be designed in a way to be easily adaptable, upwards compatible with the evolution of the base protocol and protocol interworking of future releases.
- The test declarations, data structures and data values shall be largely reusable.
- Modularity and modular working method.
- Minimizing the requirements of intelligence on the emulators of the lower testers.
- Giving enough design freedom to the test equipment manufacturers.

Fullfilling these requirements should ensure the investment of the test equipment manufacturers and users of the ATS having stable testing means for a relatively long period.

5.3.2.2 TTCN-3 naming conventions

Like in other software projects using a programming language, the use of naming conventions supports or increases:

a) the readability;

b) the detection of semantic errors;

c) the shared work of several developers;

d) the maintainability.

The naming conventions applied to the SIP/ISUP Interworking ATS are based on the following underlying principles:

- when constructing meaningful identifiers, the general guidelines specified for naming in clause 9 of [2] should be followed;
- for the SIP ATS part, which is based on a subset of TS 102 027-3 [7], with extensions, the naming conventions defined in TS 102 027-3 [7] should be followed;
- the names of TTCN-3 objects being associated with standardized data types (e.g. in the base protocols) should reflect the names of these data types as close as possible (of course not conflicting with syntactical requirements or other conventions being explicitly stated);
- the subfield names of TTCN-3 objects being associated with standardized data type should also be similar to corresponding element names in the base standards (be recognizable in the local context);
- in most other cases, identifiers should be prefixed with a short alphabetic string (specified in table 3) indicating the type of TTCN-3 element it represents;
- prefixes should be separated from the body of the identifier with an underscore ("_");
- only test case names, module names, data type names and module parameters should begin with an upper-case letter. All other names (i.e. the part of the identifier following the prefix) should begin with a lower-case letter.

Table 17 specifies the naming guidelines for each element of the TTCN-3 language indicating the recommended prefix and capitalization.

Table 17: TTCN-3 naming conventions

Language element	Naming convention	Prefix	Example	Notes
Module	Use upper-case initial letter	none	IPv6Templates	
TSS grouping	Use all upper-case letters as specified in clause 7.1.2.1.1	none	TP_RT_PS_TR	
Item group within a module	Use lower-case initial letter	none	messageGroup	
ISUP message type	Use upper-case initial letter and message name abbreviations as defined in [14].	none	IAM	
ISUP parameter type	Use upper-case initial letter and parameter name abbreviations taken from [15].	none	CalledPartyNumber	
SIP message type	Use upper-case initial letter	none	Request, Response	note 4
SIP header type	Use upper-case initial letter	none	MaxForwards	note 4
Basic common data types (e.g. bit string types of fixed length)	Use upper-case initial letter	none	Take from common module	
Other Data types	Use upper-case initial letter	none	SetupContents	
Template	None	m_	m_IAM_Basic	note 1 note 5
Message template with wildcard or matching expression	None	mw_	mw_AnyUserReply	note 2 note 5
Signature template	Use lower-case initial letter	s_	s_callSignature	
Port instance	Use lower-case initial letter	none	signallingPort	
Test component ref	Use lower-case initial letter	none	userTerminal	
Constant	Use lower-case initial letter	c_	c_maxRetransmission	
External constant	Use lower-case initial letter	cx_	cx_macId	
Function	Use lower-case initial letter	f_	f_authentication()	
External function	Use lower-case initial letter	fx_	fx_calculateLength()	
Altstep (incl. Default)	Use lower-case initial letter	a_	a_receiveSetup()	
Test case	Use naming as specified in clause 5.2.2	TC_	TC_101_001	
Variable (local)	Use lower-case initial letter	v_	v_macId	
Variable (defined within a component)	Use lower-case initial letters	vc_	vc_systemName	
Timer (local)	Use lower-case initial letter	t_	t_wait	
Timer (defined within a component)	Use lower-case initial letters	tc_	tc_authMin	
Module parameter	Use initial upper case letters	PX	PX_MAC_ID	note 3
Parameterization	Use lower-case initial letter	p_	p_macId	
Enumerated Value	Use lower-case initial letter	e_	e_syncOk	

NOTE 1: This prefix must be used for all template definitions which do not assign or refer to templates with wildcards or matching expressions, e.g. templates specifying a constant value, parameterized templates without matching expressions, etc.
NOTE 2: This prefix must be used in identifiers for templates which either assign a wildcard or matching expression (e.g. ?, *, value list, if present, pattern, etc.) or reference another template which assigns a wildcard or matching expression
NOTE 3: In this case it is acceptable to use underscore as a word delimiter.
NOTE 4: This convention has been used in TS 102 027-3 [7] (SIP ATS).
NOTE 5: Names of ISUP messages and parameters (IEs) start with a syllable being composed of capital letters only, like IAM e.g. This is different for SIP. Naming conventions concerning the first letter of a template (after prefix "m_" or "mw_", may be handled differently for ISUP/BICC and SIP respectively.

5.3.2.3 Additional TTCN-3 IMS/SIP and ISUP naming convention

In addition to the general TTCN-3 naming conventions listed in the previous section the following rules have been applied to templates.

Table 18: TTCN-3 naming conventions

Language element	Naming convention	Prefix	Example	Notes
Message template	Use lower-case initial letter, followed by message type in upper-case letters (for requests) or "Response" keyword	m_	m_BYE_Request_UE	
Message template with wildcard or matching expression	Use lower-case initial letters	mw_	mw_SUBSCRIBE_Request_IMS	

SIP Templates have been defined in a 3-step approach. First, a dummy template is defined for every message type and direction, e.g. m_ACK_Dummy and mw_ACK_Dummy. Secondly, for each message type and direction a base template has been defined that modifies respective dummy templates and includes all mandatory header fields. Template identifiers of this modifications include the keyword "Base", e.g. m_ACK_Request_Base, mw_ACK_Request_Base. More specific templates are then derived on the basis of these base templates and modify fields that need to be restricted for a very speicifc purpose, e.g. m_ACK_Request_route, etc.

5.3.2.4 Additional concepts and conventions

IMS procedures and tests requires the inclusion of user identification and network address information in SIP messages. Since this information depends on the specific SUT at hand it is defined using module parameters. Due to the big amount of such parameters a profile concept have been introduced for particular parameter collections (records) that are related to IMS users and interfaces.

The so-called user profile information (cp. module LibSip_SIPTypesAndValue) contains the following elements: userprofile identifier, current IP port and address to exchange SIP messages, IP port and address for further contact, IP address used by the TS to exchange media streams, public identity (home domain, username), quality-of-protection parameters, authentication parameters (RFC 2617 [22], section 3.2.2). A list of user profile identifiers (module LibIMS_SIPTypesAndValue) introduces available settings for UE with different locations and homes: e.g. c_userProfile_UE1atSUThome should be used in case where UE1 is a registered user of SUT and currently not visiting another IMS. User profiles are construced from module parameters (cp. module LibIMS_Steps).

Additionally some interface information is needed to indicate or validate IMS component addresses to be used in SIP header fields like Via, Route, etc.. They are defined in a similar way as user profiles (cp. LibIms_SIPTypeAndValues) and contain IP address, port and domain information. For example c_interfaceProfile_IMS_SUT_IBCF1 defines an IBCF access point at the SUT. Interface profiles are also constructed based on module parameters (cp. module LibIMS_Steps).

5.3.2.5 PICS information

No TTCN-3 control part has been defined for this test suite. If applicable PICS information is evaluated at the begining of each test case definition using an "if" statement. Log information is provided in case that a test has not been executed due to PICS setting violation.

5.3.2.6 TTCN-3 comment tags

Any TTCN-3 definition in the Test Suite Repository or Library should contain embedded comment tags. These comment tags can be used by tools to extract information from the TTCN-3 code to create, for example, a HTML-based reference documentation.

Comment tags which cover one or more lines should be specified using block comments, as illustrated:

```
/* -----------------------
 * @desc This line of text is now identified as a description
 *       which covers multiple lines
 * -----------------------*/
```

Comments tags specified within a single line may be specified using line comments, as illustrated:

```
// @author John Doe
```

or:

```
/* @author John Doe */
```

Table 19 lists the tags that can be used in ETSI TTCN-3 test specifications with a short description of the intended use of each tag. Tools may support other, non standard tags. Such tags should not be used in TTCN-3 modules standardized by ETSI.

NOTE: Tools may also extract other information from the TTCN-3 code based, for example, on TTCN- 3 keywords. The definition of that extraction is beyond the scope of the present document.

Table 19: TTCN-3 Comment Tags

Tag	Description
@author	This tag should be used to specify the names of the authors or an authoring organization which either has created or is maintaining a particular piece of TTCN-3 code.
@desc	This is probably the most import of all the tags. It should be used to describe the purpose of a particular piece of TTCN-3 code. The description should be concise yet informative and describe the function and use of the construct.
@remark	This tag may be used to add additional information, such as highlighting a particular feature or aspect not covered in the description.
@img	This tag may be used to associate images with a particular piece of TTCN-3 code.
@see	This tag may be used to refer to other TTCN-3 definitions in the same or another module.
@url	This tag should be used to associate references to external files or web pages with a particular piece of TTCN-3 code, e.g. a protocol specification or standard.
@return	This tag should only be used with functions. It is used to provide additional information on the value returned by the given function.
@param	This tag is used to document the parameters of parameterized TTCN-3 definitions.
@version	This tag is used to state the version of a particular piece of TTCN-3 code.

The following provides some basic guidelines on the usage of tags for specific TTCN-3 definitions:

- each TTCN-3 module should use the *@author*, *@version* and *@desc* tags;
- the *@desc* tag should be used with all TTCN-3 definitions. However, this should not be taken to the extreme. For example, it is probably not useful to tag literally every single constant or template declaration. It is left to the discretion of the writer to find the right level of use. At least all major constructs such as test cases and functions should have a comprehensive description:
  - when a TTCN-3 definition uses module parameters, it is also recommended to mention this explicitly in the description;
  - descriptions for behavioural constructs should mention if they set the test component verdict and also all known limitations of the construct;
  - descriptions for type definitions, e.g. component types, should mention if the type has been designed to be type compatible to another type or vice versa to be used as a basis for other type definitions.

- the @see tag should be used to make dependencies between TTCN-3 definitions which are described by a @desc tag more explicit in the documentation, e.g. if some TTCN-3 definition uses a module parameter then its TTCN-3 definition should be referenced to using a @see tag;

- where applicable, parameterized constructions such as functions, altsteps and templates should use the *@param* and *@return* tags. The @param tags should first list the parameter name and then a brief description of how this parameter is used by the construct;

- the *@url* tag should be used to refer to the specification from which the TTCN-3 definition was derived from, e.g. a type definition could refer to a particular RFC IETF page. In some cases it may be necessary to use the *@desc* tag instead for this purpose as documents often are hard to access internally, i.e., it may only be possible to specify a reference to a complete document but impossible to point to a very specific clause in the present document;

- the *@url* and *@img* tag may be used to link to relevant documentation such as Test Purposes or original requirements or even drawings of test configurations. Generally, the corresponding Test Purpose (in the TSS&TP) and to the corresponding Requirement (in the Requirements Catalogue) should be linked from the relevant TTCN-3 test case definition;

- the *@remark* tag may be used with any TTCN-3 definition. It should be used sparingly, e.g. possibly to indicate how a TTCN-3 definition should not be used.

5.4 ATS archive

Annex B contains the ATS archive (.zip file expanding to text files with TTCN-3 code).

Annex A (normative): Partial PIXIT proforma

A.1 Introduction

This partial PIXIT proforma contained in the present document is provided for completion, when the related Abstract Test Suite is to be used against the Implementation Under Test (IUT).

The completed partial PIXIT will normally be used in conjunction with the completed PICS, as it adds precision to the information provided by the PICS.

A.2 PIXIT items

According to the interworking type of ATS defined in the present document, the PIXIT are divided in common, SIP-related PIXIT and ISUP/BICC-related PIXIT.

A.2.1 Common PIXIT related to SIP and ISUP/BICC

The PIXIT items of table A.1 apply for SIP and ISUP/BICC and contain values that are used on both sides of the interworking function.

Table A.1: Common PIXIT items related to SIP and ISUP/BICC

Item	Module Parameter	Description	Type	Value
1.1	PX_TC_VA	Number of test case variant according to table entry in table to test purpose description, if present	integer	
1.2	PX_SIP_MESSAGE_VA	Number of SIP message variant according to table entry in table to test purpose description, if present	integer	
1.3	PX_BearerCapabilityInformation TransferCapability	Bearer Capability Information Transfer Capability used for mapping between ISUP: Bearer Capability information element within USI parameter and SIP: SDP offer or PSTN XML BearerCapability Used in TC_301_014, TC_301_015 and TC_301_023	bitstring(5)	
1.4	PX_BearerCapabilityInformation TransferCapability2	Second Bearer Capability Information Transfer Capability used for mapping between ISUP: Bearer Capability information element within ATP parameter and SIP: PSTN XML BearerCapability Used in TC_304_008, TC_304_009, TC_304_010, TC_305_005, TC_305_006, TC_305_008, TC_306_006, TC_306_007 and TC_306_009	bitstring(5)	

Item	Module Parameter	Description	Type	Value
1.5	PX_HighLayerCharacteristicsIdentification	High layer characteristics identification used for mapping between ISUP: High layer compatibility information element within ATP or UTSI parameter and SIP: PSTN XML HighLayerCompatibility Used in TC_105_012, TC_105_013, TC_106_006, TC_106_007, TC_107_008, TC_107_009, TC_301_031, TC_301_032 TC_301_033, TC_304_011, TC_304_012, TC_305_004, TC_305_007, TC_306_005 and TC_306_008	bitstring(7)	
1.6	PX_HighLayerCharacteristicsIdentification2	Second High layer characteristics identification used for mapping between ISUP: High layer compatibility information element within ATP parameter and SIP: PSTN XML HighLayerCompatibility Used in TC104_015, TC_104_016, TC_301_033, TC_305_004, TC_305_007, TC_306_005 and TC_306_008	bitstring(7)	
1.7	PX_LowLayerInformationTransferCapability	Low layer Information Transfer Capability used for mapping between ISUP: Low layer compatibility information element within ATP parameter and SIP: PSTN XML LowLayerCompatibility Used in TC_104_018, TC_104_019, TC_106_008, TC_107_010, TC_301_030, TC_305_003 and TC_306_004	bitstring(5)	
1.8	PX_ProgressIndicator	Progress description used for mapping between ISUP: Progress indicator information element within ATP parameter and SIP: PSTN XML ProgressIndicator Used in TC_104_008, TC_104_020, TC_105_006, TC_107_004, TC_301_029, TC_305_002 and TC_306_003	bitstring(7)	
1.9	PX_CUG_NetworkIndicator	NetworkIndicator description used for mapping between ISUP: Networkindicator information element within CUG parameter and SIP: CUG XML NetworkIndicator Used in TC_516_003, TC_516_004, TC_608_003 and TC_608_004	hexstring(1)	
1.10	PX_CUG_InterlockBinaryCode	InterlockBinaryCode description used for mapping between ISUP: InterlockBinaryCode information element within CUG parameter and SIP: CUG XML InterlockBinaryCode Used in TC_516_003, TC_516_004, TC_608_003 and TC_608_004	hexstring(2)	
1.11	PX_CauseValue	Cause value used for mapping between ISUP: Cause value within CAUI parameter and SIP: Q.850 cause value in Reason header Used in TC_110_001, TC_110_002, TC_307_003, TC_308_002, TC_308_004 and TC_308_005	integer	
1.12	PX_Timeout_Tiw1	Nominal timeout value of ISUP/SIP interworking protocol timer TOIW1.	float	
1.13	PX_Timeout_Tiw3	Nominal timeout value of ISUP/SIP interworking protocol timer TOIW3.	float	

Item	Module Parameter	Description	Type	Value
1.14	PX_SIP_privacy	Privacy value used for TC606006-606008	PrivacyValue	
1.15	PX_SIP_privacy_VA	Value used for preselected privacy values (0=id, 1=user, 2=header)	integer	
1.16	PX_SIP_NameAddr_From	NameAddr default value for From field	NameAddr	
1.17	PX_SIP_NameAddr_UserB	Default value for diverted user field Used in group 509	NameAddr	
1.18	PX_SIP_NameAddr_UserC	Default value for diverted user field Used in group 509	NameAddr	
1.19	PX_SIP_NameAddr_UserD	Default value for diverted user field Used in group 509	NameAddr	
1.20	PX_SIP_NameAddr_UserE	Default value for diverted user field Used in group 509	NameAddr	
1.21	PX_SIP_NameAddr_ChangedFrom	Default value for CHANGED From field Used in TC_606_008	NameAddr	
1.22	PX_SIP_NameAddr_PAsserted	NameAddr default value for PAsserted (with sip scheme) field Used in group 501	NameAddr	
1.23	PX_SIP_NameAddrTel_PAsserted	NameAddr default value for PAsserted (with tel scheme) field Used in group 501	NameAddr	
1.24	PX_SIP_NameAddrTel_PAsserted_otherCC	Default value for PAsserted (with tel scheme) field Used in groups 501 and 606	NameAddr	
1.25	PX_SIP_DummyUser_userInfo	Default value for user info (dummy user number) Used in group 609	charstring	
1.26	PX_SIP_User2userIData	Default value for User2userInfoData Used in group 610	charstring	
1.27	PX_SIP_XML_Conference_AS_URI	Default value for conference application server uri Used in groups 504 and505	charstring	
1.28	PX_SIP_XML_Conference_ISUP_userInfo	Default value for ISUP user number Used in groups 504 and 505	charstring	
1.29	PX_SIP_XML_Conference_DummyUser_userInfo	Default value for dummy user number Used in groups 504 and 505	charstring	
1.30	PX_SIP_XML_Conference_ReferedBy_userInfo	Default value for referedBy field Used in TC_504_013	NameAddr	

A.2.2 SIP/IMS -related PIXIT

For the SIP side of the ATS the PIXIT defined in TS 102 351 [2] apply. In addition the SIP-related PIXIT of table A.2 apply, which have been provided for the particular purposes of this ATS. Each PIXIT item corresponds to a Module Parameter of the ATS.

Table A.2: Additional SIP-related PIXIT items

Item	Module Parameter	Description	Type	Value
2.1	PX_SIP_SDP_dyn	SDP dynamic port.	charstring	
2.2	PX_SIP_SDP_b_modifier	SDP bandwidth modifier.	charstring	
2.3	PX_SIP_SDP_b_bandwidth	SDP bandwidth value.	integer	
2.4	PX_SIP_SDP_encoding	SDP media attribute encoding supported by the IUT.	charstring	
2.5	PX_SIP_SDP_encoding_unavail	SDPmedia attribute encoding unavailable by the IUT.	charstring	
2.6	PX_SIP_SDP_encoding_unsup	SDP media attribute encoding unsupported by the IUT.	charstring	
2.7	PX_SIP_SDP_transport	SDP media T.	charstring	
2.8	PX_SIP_ISUP_LANGUAGE	Used CPC language.	charstring	
2.9	PX_SIP_ISUP_CPC_VALUE	Used CPC language.	charstring	

Item	Module Parameter	Description	Type	Value
2.10	PX_SIP_100rel	True if 100rel mechanism is supported in SIP.	boolean	
2.11	PX_SIP_precondition	True if precondition mechanism is supported in SIP.	boolean	
2.12	PX_SIP_UDP	True if UDP Transport is used by the IUT to run campaign.	boolean	
2.13	PX_SIP_TRANSPORT	Used Transport in upper case "UDP"/"TCP".	charstring	
2.14	PX_SIP_BYE_CAUSE	Release cause to be used in BYE and in Failure messages.	integer	
2.15	PX_SIP_CheckConversation	True, if conversation check is implemented.	boolean	
2.16	PX_SIP_CheckDTMF	True, if DTMF check is implemented.	boolean	
2.17	PX_SIP_SendAnnouncement	True, if Announcement sending is implemented.	boolean	
2. 18	PX_SIP_CheckRinging	True, if ringing check is implemented.	boolean	
2.19	PX_SIP_T1	T1 RTT estimate (500 ms).	float	
2.20	PX_T2	T2 Maximum retransmit interval for non-INVITE requests and INVITE response (4 000 ms).	float	
2.21	PX_T4	T4 Maximum duration a message will remain in the network.	float	
2.22	PX_SIP_TF	TDELAY default value for timeout on outgoing SIP request (ie 64*T1).	float	
2.23	PX_SIP_TWAIT	TWait default value for waiting an operator action.	float	
2.24	PX_SIP_TACK	TAck default value for waiting an acknowledgement.	float	
2.25	PX_SIP_TRESP	TResp default value for waiting for a response from the IUT.	float	
2.26	PX_SIP_TNOACT	TNoAct default value for waiting no message from the IUT Value given for PX_TNOACT should be less than value of SHORT_REGISTRATION constant (which is currently "3" (seconds)).	float	
2.27	PX_SIP_TSYNC	TSYNC default value to synchronise ptc.	float	
2.28	PX_SIP_TGUARD	TGUARD default value for an extra long timer to limit test execution.	float	
2.29	PX_TRespRetention	TRespRetention minimum time that a Proxy will wait before sending a final response.	float	
2.30	PX_IMS_TS_ICSCF_IPADDR	TS/I-CSCF IP address to exchange SIP messages.	charstring	
2.31	PX_IMS_TS_ICSCF_PORT	IUT/I-CSCF port number to exchange SIP messages.	integer	
2.32	PX_IMS_TS_ICSCF_HOME_DOMAIN	TS/I-CSCF domain.	charstring	
2.33	PX_IMS_SUT_IMGCF_IPADDR	SUT/I-MGCF IP address to exchange SIP messages.	charstring	
2.34	PX_IMS_SUT_IMGCF_PORT	SUT/I-MGCF port number to exchange SIP messages.	integer	
2.35	PX_IMS_SUT_IMGCF_HOME_DOMAIN	SUT/I-MGCFdomain.	charstring	

A.2.3 ISUP/BICC-related PIXIT

Tables A.3 to A.6 list the ISUP/BICC-related PIXIT items associated with the ATS. Each PIXIT item corresponds to a Module Parameter of the ATS. Default values are not provided.

Table A.3: General SS/SUT-related ISUP/BICC PIXIT items

Item	IModule Parameter	Description	Type	Value
3.1	PX_ISUP_Isup	Select whether ISUP (true) or BICC (false) testing is done (depending on whether the SUT implements ISUP or BICC on the outgoing circuits under test).	boolean	
3.2	PX_ISUP_NW_IND	Network indicator inside the Service Indicator octet (SIO).	bitstring(2)	
3.3	PX_ISUP_SLS	Signalling Link Selection (SLS) value of the ISUP link between TS and SUT.	bitstring(4)	
3.4	PX_ISUP_PC_SUT	Point code of the SUT (ISUP interface).	bitstring(14)	
3.5	PX_ISUP_PC_TS	Point code of the TS (ISUP interface).	bitstring(14)	
3.6	PX_SUT_ADRESS_ISUP_BICC	Address (e.g. IP) of the SUT (ISUP/BICC side). The use of this address is to enable the TS to communicate with the SUT at the ISUP/BICC side to establish and maintain the lower layer connections.	charstring	
3.7	PX_TS_ADRESS_ISUP_BICC	Address (e.g. IP) of the TS (ISUP/BICC side). The use of this address is to enable the TS to communicate with the SUT at the ISUP/BICC side to establish and maintain the lower layer connections.	octetstring	
3.8	PX_ISUP_TX_CIC_cicv1	Default Circuit Identity Code value for signalling connection 1).	bitstring(12)	
3.9	PX_ISUP_TX_CIC_cicv2	Default Circuit Identity Code value for signalling connection 2).	bitstring(12)	
3.10	PX_ISUP_TX_CIC_caicv1	Default Call Instance Code value for signalling connection 1).	octetstring(4)	
3.11	PX_ISUP_TX_CIC_caicv2	Default Call Instance Code value for signalling connection 2).	octetstring(4)	

Table A.4: Timer-related ISUP/BICC PIXIT items

Item	IModule Parameter	Description	Type	Value
4.31	PX_ISUP_TAC	Time to control the reception of a message.	float	
4.32	PX_ISUP_TNOAC	Time to control that IUT sends nothing.	float	
4.33	PX_ISUP_TSYNC	Time to control synchronization.	float	
4.34	PX_ISUP_TSYNC_TIME_LIMIT	Time to control synchronization.	float	
4.35	PX_ISUP_TDONE	Time to control PTC.stop.	float	
4.36	PX_ISUP_TWAIT	Time to control that IUT reacts prior to Upper Tester action.	float	
4.37	PX_TDelay	Time to delay messages before sending.	float	
4.38	PX_Timeout_T7	Nominal timeout value of ISUP protocol timer T7.	float	
4.39	PX_Timeout_T8	Nominal timeout value of ISUP protocol timer T8.	float	
4.40	PX_Timeout_T9	Nominal timeout value of ISUP protocol timer T9.	float	
4.41	PX_Timeout_T39	Nominal timeout value of ISUP protocol timer T39.	float	

Table A.5: Operator-check-related ISUP/BICC PIXIT items

Item	IModule Parameter	Description	Type	Value
5.1	PX_IsupBicc_CheckConversation	True if conversation check is implemented and used. Otherwise false (see note 1).	boolean	
5.2	PX_IsupBicc_CheckRinging	True if ringing check is implemented and used. Otherwise false (see note 2).	boolean	
NOTE 1: If true, test execution will stop at positions where the TP indicates "conversation" until the operator enters the check result. NOTE 2: If true, test execution will stop at positions where the TP indicates "ringing" until the operator enters the check result.				

Table A.6: ISUP/BICC PIXIT items associated with message fields

Item	Module Parameter	Description	Type	Value
Called party number - sending				
6.4.1.1	PX_ISUP_IAM_CLD_digits_txDef	Default "address digits" value sent in the "Called party number" parameter in the IAM message, containing the complete address and "sending complete". See ITU-T T Rec. Q.763 [15], 3.9.	IA5String	
6.4.1.2	PX_ISUP_TX_CLD_natAddr_txDef	Default "nature of address" value sent in the "Called party number" parameter in the IAM message, containing the complete address and "sending complete". See ITU-T T Rec. Q.763 [15], 3.9.	bitstring(7)	
6.4.1.3	PX_ISUP_IAM_CLD_digits_txDef_inat	Default "complete address digits" value sent in the "Called party number" parameter in the IAM message, when the nature of address is specified as "international number". See ITU-T T Rec. Q.763 [15], 3.9.	IA5String	
6.4.1.4	PX_ISUP_IAM_CLD_digits_txDef_nat	Default "complete address digits" value sent in the "Called party number" parameter in the IAM message, when the nature of address is specified as "national (sign.) number". See ITU-T T Rec. Q.763 [15], 3.9.	IA5String	
6.1.2.1	PX_ISUP_IAM_CLD_digits_analysis	"address digits" value sent in the "Called party number" parameter in the IAM message, when "sending complete" is not sent, not the maximum number of digits are sent, the number is complete and completeness is determined by analysis of the number. See ITU-T T Rec. Q.763 [15], 3.9.	IA5String	
6.1.2.2	PX_ISUP_TX_CLD_natAddr_analysis	"nature of address" value sent in the "Called party number" parameter in the IAM message, when "sending complete" is not sent, not the maximum number of digits are sent, the number is complete and completeness is determined by analysis of the number. See ITU-T T Rec. Q.763 [15], 3.9.	bitstring(7)	

Item	Module Parameter	Description	Type	Value
6.1.3.1	PX_ISUP_IAM_CLD_digits_timeout	"address digits" value sent in the "Called party number" parameter in the IAM message, when "sending complete" is not sent, not the maximum number of digits are sent, the number is complete and completeness is determined by timeout. See ITU-T T Rec. Q.763 [15], 3.9.	IA5String	
6.1.3.2	PX_ISUP_TX_CLD_natAddr_timeout	"nature of address" value sent in the "Called party number" parameter in the IAM message, when "sending complete" is not sent, not the maximum number of digits are sent, the number is complete and completeness is determined by timeout. See ITU-T T Rec. Q.763 [15], 3.9.	bitstring(7)	
6.1.4.1	PX_ISUP_IAM_CLD_digits_max	"address digits" value sent in the "Called party number" parameter in the IAM message, containing the maximum number of digits according to the national numbering plan, and no "sending complete". See ITU-T T Rec. Q.763 [15], 3.9.	IA5String	
6.1.4.2	PX_ISUP_TX_CLD_natAddr_max	"nature of address" value sent in the "Called party number" parameter in the IAM message, containing the maximum number of digits according to the national numbering plan, and no "sending complete". See ITU-T T Rec. Q.763 [15], 3.9.	bitstring(7)	
6.1.5.1	PX_ISUP_IAM_CLD_digits_less	"address digits" value (less than minimum number digits to route the call) sent in the "Called party number" parameter in the IAM message.	IA5String	
6.1.5.2	PX_ISUP_IAM_CLD_natAddr_less	"nature of address" value (number of digits less than minimum number digits to route the call) sent in the "Calling party number" parameter in the IAM message. See ITU-T T Rec. Q.763 [15], 3.9.	bitstring(7)	
6.1.6.1	PX_ISUP_IAM_CLD_digits_min	"address digits" value sent in the "Called party number" parameter in the IAM message, containing the minimum number of digits required for routing, and no "sending complete". See ITU-T T Rec. Q.763 [15], 3.9.	IA5String	
6.1.6.2	PX_ISUP_TX_CLD_natAddr_min	"nature of address" value sent in the "Called party number" parameter in the IAM message, containing the minimum number of digits required for routing, and no "sending complete". See ITU-T T Rec. Q.763 [15], 3.9.	bitstring(7)	
Calling party number - receiving				
6.2.1	PX_ISUP_IAM_CLI_digits_rxInat	Default "address digits" value received in the "Calling party number" parameter in the IAM message, when the Called party number is "international". See ITU-T T Rec. Q.763 [15], 3.10.	IA5String	
6.2.2	PX_ISUP_IAM_CLI_digits_rxNat	Default "address digits" value received in the "Calling party number" parameter in the IAM message, when the Called party number is "national (sign.) number". See ITU-T T Rec. Q.763 [15], 3.10.	IA5String	

Item	Module Parameter	Description	Type	Value
Calling party number - sending				
6.3.1	PX_ISUP_IAM_CLI_digits_txInat	Default "address digits" value sent in the "Calling party number" parameter in the IAM message, when the Called party number is "international". See ITU-T T Rec. Q.763 [15], 3.10.	IA5String	
6.3.2	PX_ISUP_IAM_CLI_digits_txNat	Default "address digits" value sent in the "Calling party number" parameter in the IAM message, when the Called party number is "national (sign.) number". See ITU-T T Rec. Q.763 [15], 3.10.	IA5String	
Generic number - receiving				
6.4.1	PX_ISUP_IAM_GEN_digits_rxInat	"address digits" value received in the "Generic number" parameter in the IAM message, when the Nature of Address is "international number". See ITU-T T Rec. Q.763 [15], 3.26.	IA5String	
6.4.2	PX_ISUP_IAM_GEN_digits_rxNat	"address digits" value received in the "Generic number" parameter in the IAM message, when the Nature of Address is "national (sign.) number". See ITU-T T Rec. Q.763 [15], 3.26.	IA5String	
6.4.3	PX_ISUP_ANM_GEN_digits_rxInat	"address digits" value received in the "Generic number" parameter in the ANM message, when the Nature of Address is "international number". See ITU-T T Rec. Q.763 [15], 3.26.	IA5String	
6.4.4	PX_ISUP_ANM_GEN_digits_rxNat	"address digits" value received in the "Generic number" parameter in the ANM message, when the Nature of Address is "national (sign.) number". See ITU-T T Rec. Q.763 [15], 3.26.	IA5String	
Generic number - sending				
6.5.1	PX_ISUP_IAM_GEN_digits_txInat	"address digits" value sent in the "Generic number" parameter in the IAM message, when the Nature of Address is "international number". See ITU-T T Rec. Q.763 [15], 3.26.	IA5String	
6.5.2	PX_ISUP_IAM_GEN_digits_txNat	"address digits" value sent in the "Generic number" parameter in the IAM message, when the Nature of Address is "national (sign.) number". See ITU-T T Rec. Q.763 [15], 3.26.	IA5String	
6.5.3	PX_ISUP_ANM_GEN_digits_txInat	"address digits" value sent in the "Generic number" parameter in the ANM message, when the Nature of Address is "international number". See ITU-T T Rec. Q.763 [15], 3.26.	IA5String	
6.5.4	PX_ISUP_ANM_GEN_digits_txNat	"address digits" value sent in the "Generic number" parameter in the ANM message, when the Nature of Address is "national (sign.) number". See ITU-T T Rec. Q.763 [15], 3.26.	IA5String	
Conected number - receiving				
6.6.1	PX_ISUP_ANM_CPN_digits_rxInat	"address digits" value received in the "Connected number" parameter in the ANM message, when the Nature of Address is "international number". See ITU-T T Rec. Q.763 [15], 3.28.	IA5String	
6.6.2	PX_ISUP_ANM_CPN_digits_rxNat	"address digits" value received in the "Connected number" parameter in the ANM message, when the Nature of Address is "national (sign.) number". See ITU-T T Rec. Q.763 [15], 3.28.	IA5String	

Item	Module Parameter	Description	Type	Value
Conected number - receiving				
6.7.1	PX_ISUP_ANM_CPN_digits_txInat	"address digits" value sent in the "Connected number" parameter in the ANM message, when the Nature of Address is "international number". See ITU-T T Rec. Q.763 [15], 3.28.	IA5String	
6.7.2	PX_ISUP_ANM_CPN_digits_txNat	"address digits" value sent in the "Connected number" parameter in the ANM message, when the Nature of Address is "national (sign.) number". See ITU-T T Rec. Q.763 [15], 3.28.	IA5String	
Original called number - receiving				
6.8.1	PX_ISUP_IAM_OCN_digits_rxInat	"address digits" value received in the "Original called number" parameter in the IAM message, when the Nature of Address is "international number". See ITU-T T Rec. Q.763 [15], 3.39.	IA5String	
6.8.2	PX_ISUP_IAM_OCN_digits_rxNat	"address digits" value received in the "Original called number" parameter in the IAM message, when the Nature of Address is "national (sign.) number". See ITU-T T Rec. Q.763 [15], 3.39.	IA5String	
Original called number - sending				
6.9.1	PX_ISUP_TX_OCN_natOfAddressInd	Default value for element natureOfAddressIndicator inside Original called number parameter (OCN); Optional(O) format (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.763 [15], 3.39.	bitstring(7)	
6.9.2	PX_ISUP_TX_OCN_addrSignals	Default value for element addressSignals inside Original called number parameter (OCN); Optional(O) format (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.763 [15], 3.39.	IA5String	
Redirecting number - receiving				
6.10.1	PX_ISUP_IAM_RDN_digits_rxInat	"address digits" value received in the "Redirecting number" parameter in the IAM message, when the Nature of Address is "international number". See ITU-T T Rec. Q.763 [15], 3.44.	IA5String	
6.10.2	PX_ISUP_IAM_RDN_digits_rxNat	"address digits" value received in the "Redirecting number" parameter in the IAM message, when the Nature of Address is "national (sign.) number". See ITU-T T Rec. Q.763 [15], 3.44.	IA5String	
Redirecting number - sending				
6.11.1	PX_ISUP_TX_RDN_natOfAddressInd	Default value for element natureOfAddressIndicator inside Redirecting number parameter (RDN); Optional(O) format (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.763 [15], 3.44.	bitstring(7)	
6.11.2	PX_ISUP_TX_RDN_addrSignals	Default value for element addressSignals inside Redirecting number parameter (RDN); Optional(O) format (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.763 [15], 3.44.	IA5String	

Item	Module Parameter	Description	Type	Value
Redirection number - receiving				
6.12	PX_ISUP_RX_RNN_addrSignals	Default value for element addressSignals inside Redirection number parameter (RNN); Optional(O) format (to be received in ACM or CPG messages). See ITU-T T Rec. Q.763 [15], 3.46.	IA5String	
Redirection number - receiving				
6.13.1	PX_ISUP_TX_RNN_natOfAddressInd	Default value for element natureOfAddressIndicator inside Redirection number parameter (RNN); Optional(O) format (to be sent when the TP does not specify a specific value for that field in ANM or CPG messages). See ITU-T T Rec. Q.763 [15], 3.46.	bitstring(7)	
6.13.2	PX_ISUP_TX_RNN_addrSignals	Default value for element addressSignals inside Redirection number parameter (RNN); Optional(O) format (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.763 [15], 3.46.	IA5String	
Subsequent number				
6.14.1	PX_ISUP_SAM_SQN_digits_txLess_AllowRoute	"address digits" value sent in the "Subsequent number" parameter in a SAM message, containing enough number digits to allow the routing to the SIP side, where the IAM contained less than the minimum digits to route the call through to the SIP side. See ITU-T T Rec. Q.763 [15], 3.51.		
6.14.2	PX_ISUP_SAM_SQN_digits_tx_2nd	"address digits" value sent in the "Subsequent number" parameter in the SAM message, containing the second part of the number, where the IAM contained already enough digits to route the call through to the SIP side. See ITU-T T Rec. Q.763 [15], 3.51.		
6.14.3	PX_ISUP_SAM_SQN_digits_tx_3rd	"address digits" value sent in the "Subsequent number" parameter in the second SAM message, containing the third part of the number, where the IAM contained already enough digits to route the call through to the SIP side. See ITU-T T Rec. Q.763 [15], 3.51.		
6.14.4	PX_ISUP_SAM_SQN_digits_tx_4th	"address digits" value sent in the "Subsequent number" parameter in the SAM message, containing the fourth and final part of the number, where the IAM contained already enough digits to route the call through to the SIP side. See ITU-T T Rec. Q.763 [15], 3.51.		
6.14.5	PX_ISUP_SAM_SQN_digits_tx_4th_max	"address digits" value sent in the "Subsequent number" parameter in the third SAM message, containing the fourth and final part of the number with the amount of digits leading to the overall maximum of digits allowed according to the numbering plan, where the IAM contained already enough digits to route the call through to the SIP side. See ITU-T T Rec. Q.763 [15], 3.51.		

Item	Module Parameter	Description	Type	Value
Backward call indicators				
6.15.1	PX_ISUP_TX_BCI_v_chargeInd	Default value for element chargeIndicator inside Backward call indicators parameter (BCI); Fixed(F) format (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.763 [15], 3.5.	bitstring(2)	
6.15.2	PX_ISUP_TX_BCI_v_cldPStatInd	Default value for element calledPartysStatusIndicator inside Backward call indicators parameter (BCI); Fixed(F) format (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.763 [15], 3.5.	bitstring(2)	
6.15.3	PX_ISUP_TX_BCI_v_cldPCatInd	Default value for element calledPartysCategoryIndicator inside Backward call indicators parameter (BCI); Fixed(F) format (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.763 [15], 3.5.	bitstring(2)	
6.15.4	PX_ISUP_TX_BCI_v_eTOeMethodInd	Default value for element end_to_endMethodIndicator inside Backward call indicators parameter (BCI); Fixed(F) format (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.763 [15], 3.5.	bitstring(2)	
6.15.5	PX_ISUP_TX_BCI_v_interwInd	Default value for element interworkingIndicator inside Backward call indicators parameter (BCI); Fixed(F) format (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.763 [15], 3.5.	bitstring(1)	
6.15.6	PX_ISUP_TX_BCI_v_eTOeInfoInd	Default value for element end_to_endInformationIndicator inside Backward call indicators parameter (BCI); Fixed(F) format (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.763 [15], 3.5.	bitstring(1)	
6.15.7	PX_ISUP_TX_BCI_v_iSDNUserPartInd	Default value for element iSDNUserPartIndicator inside Backward call indicators parameter (BCI); Fixed(F) format (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.763 [15], 3.5.	bitstring(1)	
6.15.8	PX_ISUP_TX_BCI_v_holdingInd	Default value for element holdingIndicator inside Backward call indicators parameter (BCI); Fixed(F) format (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.763 [15], 3.5.	bitstring(1)	
6.15.9	PX_ISUP_TX_BCI_v_iSDNAccessInd	Default value for element iSDNAccessIndicator inside Backward call indicators parameter (BCI); Fixed(F) format (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.763 [15], 3.5.	bitstring(1)	

Item	Module Parameter	Description	Type	Value
6.15.10	PX_ISUP_TX_BCI_v_echoContrDevInd	Default value for element echoControlDeviceIndicator inside Backward call indicators parameter (BCI); Fixed(F) format (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.763 [15], 3.5.	bitstring(1)	
6.15.11	PX_ISUP_TX_BCI_v_sCCPMethodInd	Default value for element sCCPMethodIndicator inside Backward call indicators parameter (BCI); Fixed(F) format (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.763 [15], 3.5.	bitstring(2)	
Calling party category				
6.16	PX_ISUP_TX_CGC_cliPCategory	Default value for element callingPartysCategory inside Calling party"s category parameter (CGC); Optional(O) format (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.763 [15], 3.11.	bitstring(8)	
Forward call indicators				
6.17.1	PX_ISUP_TX_FCI_natInternatCallInd	Default value for element natInternatCallIndicator inside Forward call indicators parameter (FCI); Fixed(F) format (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.763 [15], 3.23.	bitstring(1)	
6.17.2	PX_ISUP_TX_FCI_endToEndMethodInd	Default value for element endToEndMethodIndicator inside Forward call indicators parameter (FCI); Fixed(F) format (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.763 [15], 3.23.	bitstring(2)	
6.17.3	PX_ISUP_TX_FCI_interwInd	Default value for element interworkingIndicator inside Forward call indicators parameter (FCI); Fixed(F) format (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.763 [15], 3.23.	bitstring(1)	
6.17.4	PX_ISUP_TX_FCI_eTOeInfoIndic	Default value for element endToEndInfoIndicator inside Forward call indicators parameter (FCI); Fixed(F) format (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.763 [15], 3.23.	bitstring(1)	
6.17.5	PX_ISUP_TX_FCI_iSDNUserPartInd	Default value for element iSDNUserPartIndicator inside Forward call indicators parameter (FCI); Fixed(F) format (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.763 [15], 3.23.	bitstring(1)	
6.17.6	PX_ISUP_TX_FCI_iSDNUserPartPrefInd	Default value for element iSDNUserPartPrefIndicator inside Forward call indicators parameter (FCI); Fixed(F) format (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.763 [15], 3.23.	bitstring(2)	

Item	Module Parameter	Description	Type	Value
6.17.7	PX_ISUP_TX_FCI_iSDNAccessInd	Default value for element iSDNAccessIndicator inside Forward call indicators parameter (FCI); Fixed(F) format (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.763 [15], 3.23.	bitstring(1)	
6.17.8	PX_ISUP_TX_FCI_sCCPMethodInd	Default value for element sCCPMethodIndicator inside Forward call indicators parameter (FCI); Fixed(F) format (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.763 [15], 3.23.	bitstring(2)	
6.17.9	PX_ISUP_TX_FCI_reserved	Default value for element reserved inside Forward call indicators parameter (FCI); Fixed(F) format (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.763 [15], 3.23.	bitstring(4)	
Nature of connection indicators				
6.18.1	PX_ISUP_TX_NCI_satelliteInd	Default value for element satelliteIndicator inside Nature of connection indicators parameter (NCI); Fixed(F) format (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.763 [15], 3.35.	bitstring(2)	
6.18.2	PX_ISUP_TX_NCI_contCheckInd	Default value for element continuityCheckIndicator inside Nature of connection indicators parameter (NCI); Fixed(F) format (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.763 [15], 3.35.	bitstring(2)	
6.18.3	PX_ISUP_TX_NCI_echoContrDevInd	Default value for element echoControlDeviceIndicator inside Nature of connection indicators parameter (NCI); Fixed(F) format (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.763 [15], 3.35.	bitstring(1)	
Range and status				
6.19.1	PX_ISUP_TX_RAS_range	Default value for element range inside Range and status parameter (RAS); Variable(V) format (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.763 [15], 3.43.	bitstring(8)	
6.19.2	PX_ISUP_TX_RAS_status	Default value for element status inside Range and status parameter (RAS); Variable(V) format (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.763 [15], 3.43.	octetstring	
Redirection number restriction				
6.21	PX_ISUP_TX_RNS_presRestrInd	Default value for element presRestrIndicator inside Redirection number restriction parameter (RNS); Optional(O) format (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.763 [15], 3.46.	bitstring(2)	

Item	Module Parameter	Description	Type	Value
Transmission medium required				
6.20	PX_ISUP_TX_TMR_transmMedReq	Default value for element transmissionMediumRequirement inside Transmission medium requirement parameter (TMR); Optional(O) format (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.763 [15], 3.54.	bitstring(8)	
Hop counter				
6.21	PX_ISUP_TX_HPC_hopCounter	Default value for element hopCounter inside Hop counter parameter (HPC); Optional(O) format (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.763 [15], 3.80.	bitstring(5)	
User-to-user information				
6.22.1	PX_ISUP_UUI_userInfo_rx	Default "user-to-user information" value received in the "User-to-user information" parameter. See ITU-T T Rec. Q.763 [15], 3.61.	octetstring	
6.22.2	PX_ISUP_UUI_userInfo_tx	Default "user-to-user information" value sent in the "User-to-user information" parameter. See ITU-T T Rec. Q.763 [15], 3.61.	octetstring	
Cause indicators				
6.23	PX_ISUP_CAU_location	"Location" value sent in the "Cause indicators" parameter.	bitstring(4)	
Unknown parameter/message identifier				
6.24.1	PX_ISUP_TX_unknown_parameter_type	Default value for an unknown parameter type (to be sent when the TP does not specify a specific value for that field).	bitstring(8)	
6.24.2	PX_ISUP_TX_unknown_message_type	Default value for an unknown message type (to be sent when the TP does not specify a specific value for that field).	bitstring(8)	
Bearer capability				
6.25	PX_userInfoLayer1	Default value for bit field element "User Information Layer 1 Protocol Indicator" in IE Bearer Capability encapsulated in "User service information" or "Access transport" parameter (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.931 [20], 4.5.5.	bitstring(5)	
Called party subaddress				
6.26.1	PX_ISUP_RX_cdps_information	Called party subaddress information value received in the "Calling party subaddress" in the ATP parameter in the IAM message. See ITU-T T Rec. Q.931 [20], 4.5.8.	octetstring	
6.26.2	PX_ISUP_TX_cdps_information	Default value for called party subaddress information (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.931 [20], 4.5.8.	octetstring	
6.26.3	PX_ISUP_TX_cdps_odd_even_indicator	Default value for called party subaddress odd even indicator (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.931 [20], 4.5.8.	bitstring(1)	

Item	Module Parameter	Description	Type	Value
Calling party subaddress				
6.27.1	PX_ISUP_RX_cgps_information	Calling party subaddress information value received in the "Calling party subaddress" in the ATP parameter in the IAM message. See ITU-T T Rec. Q.931 [20], 4.5.11.	octetstring	
6.27.2	PX_ISUP_TX_cgps_information	Default value for calling party subaddress information (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.931 [20], 4.5.11.	octetstring	
6.27.3	PX_ISUP_TX_cgps_odd_even_indicator	Default value for calling party subaddress odd even indicator (to be sent when the TP does not specify a specific value for that field). See ITU-T T Rec. Q.931 [20], 4.5.11.	bitstring(1)	
Connected subaddress				
6.26.1	PX_ISUP_RX_cons_information	Connected subaddress information value received in the "Calling party subaddress" in the ATP parameter in the ANM message. See EN 300 097-1 [21], 7.2.	octetstring	
6.26.2	PX_ISUP_TX_cons_information	Default value for connected subaddress information (to be sent when the TP does not specify a specific value for that field). See EN 300 097-1 [21], 7.2.	octetstring	
6.26.3	PX_ISUP_TX_cons_odd_even_indicator	Default value for connected subaddress odd even indicator (to be sent when the TP does not specify a specific value for that field). See EN 300 097-1 [21], 7.2.	bitstring(1)	
NOTE: For Module Parameters containing address digits the following requirement applies: each digit is represented either as one of the IA5 characters "0" to "9", or as one of the special IA5 characters "*", "#", "a", "b" or "c".				

Annex B (informative): TTCN-3 library modules

B.1 Electronic annex, zip file with TTCN-3 code

The TTCN-3 library modules are contained in archive ts_18600903v020101p0.zip which accompanies the present document.

History

Document history		
V2.1.1	September 2009	Publication