ETSI TS 102 797-3 V1.1.1 (2012-08)

Intelligent Transport Systems (ITS);
Communications Access for Land Mobiles (CALM);
Test specifications for ITS station management (ISO 24102);
Part 3: Abstract Test Suite (ATS) and partial PIXIT proforma

Reference
DTS/ITS-0020030

Keywords
ATS, CALM, ITS, management, testing, TTCN

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://ipr.etsi.org).

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

Foreword

This Technical Specification (TS) has been produced by ETSI Technical Committee Intelligent Transport System (ITS).

The present document is part 3 of a multi-part deliverable covering "Abstract Test Suite" (ATS) and partial PIXIT proforma specifications for ITS station management protocols as identified below:

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

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

Part 3: "Abstract Test Suite (ATS) and partial PIXIT proforma".

1 Scope

The present document provides the Abstract Test Suite (ATS) and partial PIXIT proforma for the protocols specified in ISO/DIS 24102 [i.1], [1], [2] based on the related TSS&TP specification [4] and the PICS proforma [3] and in accordance with the relevant guidance given in ISO/IEC 9646-1 [5], ISO/IEC 9646-2 [6], ETS 300 406 [7] and EG 202 798 [i.5].

2 References

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

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 necessary for the application of the present document.

[1] ISO/DIS 24102-4: "Intelligent transport systems -- Communications access for land mobiles (CALM) -- ITS station management -- Part 4: Station-internal management communications".

NOTE: Available at http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=61565

[2] ISO/DIS 24102-5: "Intelligent transport systems -- Communications access for land mobiles (CALM) -- ITS station management -- Part 5: Fast service advertisement protocol (FSAP)".

NOTE: Available at http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=61566.

[3] ETSI TS 102 797-1: "Intelligent Transport Systems (ITS); Communications Access for Land Mobiles (CALM); Test specifications for ITS station management (ISO 24102); Part 1: Protocol Implementation Conformance Statement (PICS) specification".

[4] ETSI TS 102 985-2: "Intelligent Transport Systems (ITS); Communications Access for Land Mobiles (CALM); Test specifications for non-IP networking (ISO 29281); Part 2: Test Suite Structure and Test Purposes (TSS&TP)".

[5] ISO/IEC 9646-1 (1994): "Information technology -- Open Systems Interconnection -- Conformance testing methodology and framework -- Part 1: General concepts".

[6] ISO/IEC 9646-2 (1994): "Information technology -- Open Systems Interconnection -- Conformance testing methodology and framework -- Part 2: Abstract Test Suite specification".

[7] ETSI ETS 300 406 (1995): "Methods for testing and Specification (MTS); Protocol and profile conformance testing specifications; Standardization methodology".

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

[9] ETSI ES 201 873-7: "Methods for Testing and Specification (MTS); The Testing and Test Control Notation version 3; Part 7: Using ASN.1 with TTCN-3".

[10] ETSI ES 202 784: "Methods for Testing and Specification (MTS); The Testing and Test Control Notation version 3; TTCN-3 Language Extensions: Advanced Parameterization".

2.2 Informative references

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

[i.1] ISO/DIS 24102-1: "Intelligent transport systems -- Communications access for land mobiles (CALM) -- ITS station management -- Part 1: Local management".

[i.2] ISO/DIS 24102-3: "Intelligent transport systems -- Communications access for land mobiles (CALM) -- ITS station management -- Part 3: Service access points".

[i.3] ISO 21217: "Intelligent transport systems -- Communications access for land mobiles (CALM) -- Communications architecture".

[i.4] ISO 21218: "Intelligent transport systems -- Communications access for land mobiles (CALM) -- Medium service access point".

[i.5] ETSI EG 202 798: "Intelligent Transport Systems (ITS); Testing; Framework for conformance and interoperability testing".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in [1], [2], [3], [4], [5], [6], [7], [8], [9], [10], [i.1], [i.2], [i.3], [i.4] and [i.5] apply.

3.2 Abbreviations

For the purposes of the present document, the abbreviations given in [1], [2], [3], [4], [5], [6], [7], [8], [9], [10], [i.1], [i.2], [i.3], [i.4] and [i.5] apply.

4 Abstract protocol tester

In general, the conformance test system architecture as illustrated in the ITS testing framework [i.5], see figure 1, applies. For the present document, the IUT is given by protocols located in the ITS-S management entity, thus several types of IUTs need to be considered. The "Upper tester application" allows accessing the "upper side" of the IUT. Lower layer protocols indicated by the block "ITS lower layers" allow access to the IUT from the "lower side". "Upper side" and "lower side" are obvious terms in case of protocols residing in an OSI communication layer. For management protocols, it will be clearly specified in clauses 5 and 6 what "upper side" and "lower side" mean.

The test system simulates valid and invalid protocol behaviour and analyses the reaction of the IUT.

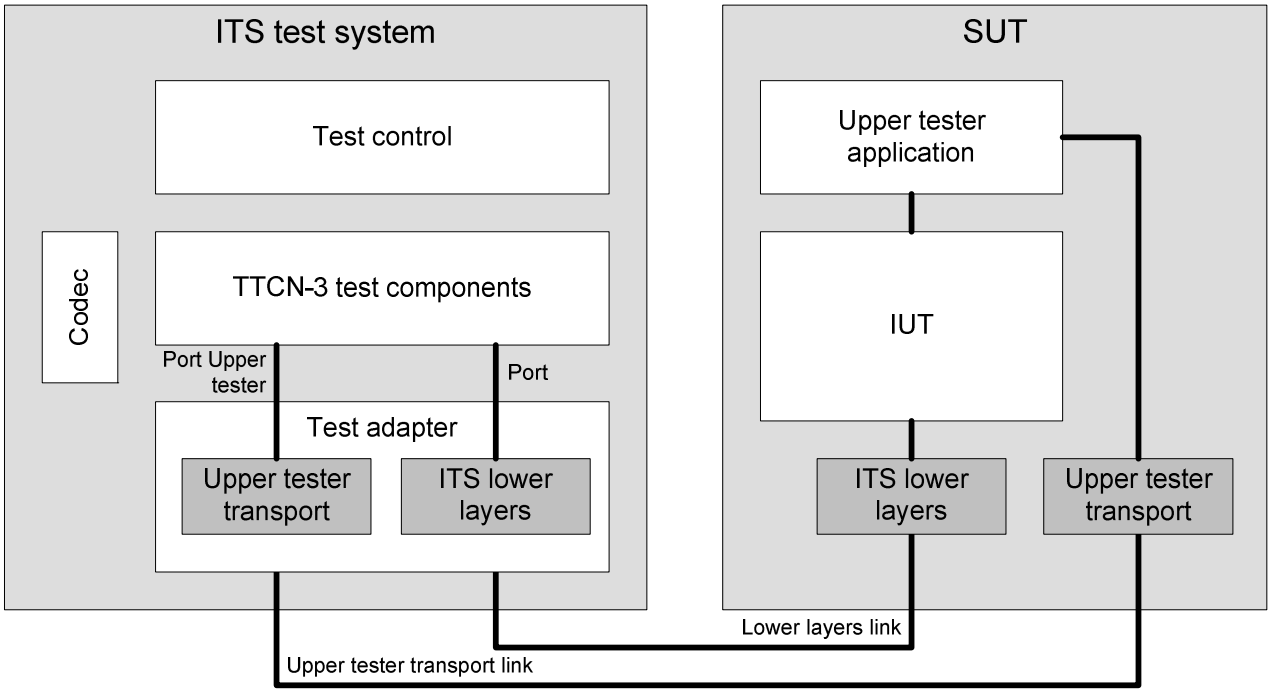

Figure 1: Abstract protocol tester - General approach

5 Abstract test method for FSAP

This clause describes the "Abstract Test Method" (ATM) used to test the "Fast Service Advertisement Protocol" (FSAP) [2].

5.1 Abstract protocol tester

SUTs which support the "ITS station-Internal management Communications Protocol" (IICP) [1] may benefit from the conformance test system architecture illustrated in figure 2, where the access to the IUT from top, i.e. in general via the upper tester application, is performed via management SAPs.

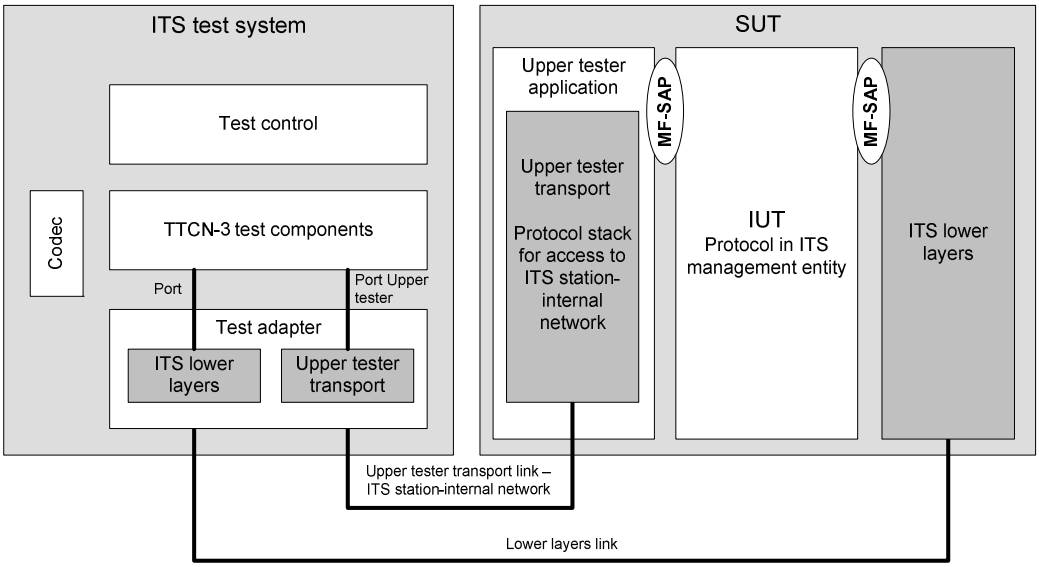

Figure 2: Abstract protocol tester for FSAP - IICP approach for upper tester

5.2 Test configurations

5.2.1 Roles of an ITS-SCU

The test suite for FSAP uses two test configurations in order to cover the different test scenarios. Distinction between the two configurations is given by the two possible implementation scenarios for an ITS station, i.e. a single-unit implementation, or an implementation with several "ITS station communication units" (ITS-SCU) which are interconnected via an ITS station-internal network [1], [2], [i.3]. These ITS-SCUs can take over the roles of an ITS-S host, or an ITS-S router, or the combined role of ITS-S host and ITS-S router. The two identified testing configurations are referred to as CF01, for the single unit implementation and CF02 for the multi-unit implementation and are described in clauses 5.2.2 and 5.2.3.

5.2.2 Test configuration CF01: No ITS station-internal network

In test configuration CF01, the roles of ITS-S host and ITS-S router are implemented in a single ITS-SCU. Consequently the whole supported functionality of FSAP is given in a single ITS-SCU and no station-internal forwarding between ITS-S host and ITS-S router is needed.

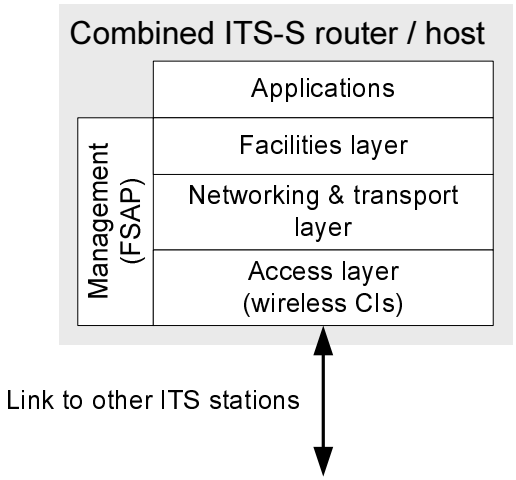

Figure 3: Test configuration CF01 architecture

This configuration is used in the cases listed below [4]:

- ITS-S station internat-network PICS (PICS_S_INW) is set to false.
- The roles PICS (PICS_ROLE_RH) is set to true.

5.2.3 Test configuration CF02: ITS station-internal network

In test configuration CF02, the roles of ITS-S host and ITS-S router are implemented in different ITS-SCUs. Consequently there is communications needed between the ITS-SCU with host functionality and the ITS-SCU with router functionality. This communication goes via the ITS station-internal network using the "ITS station-Internal management Communications Protocol" (IICP) [1].

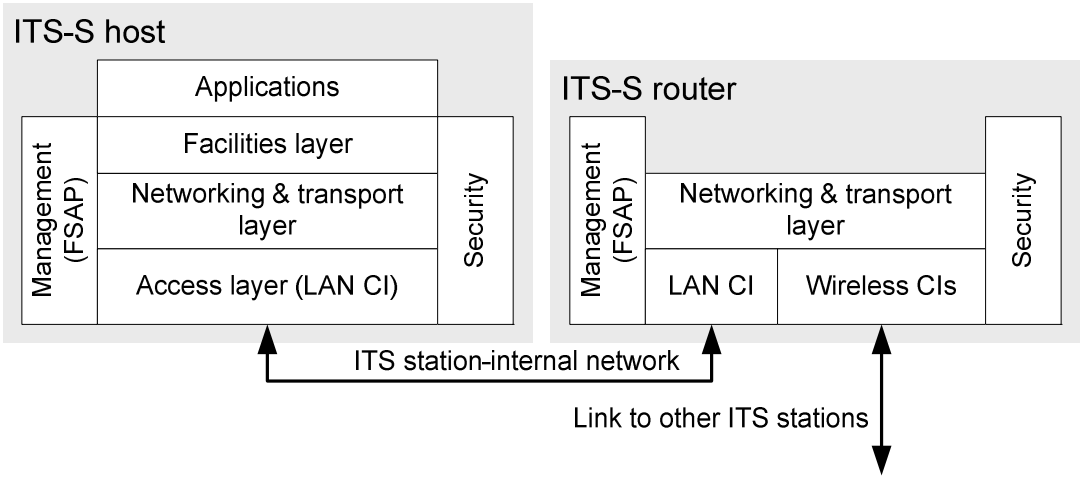

Figure 4: Test configuration CF02 architecture

This configuration is used in the cases listed below [4]:

- ITS-S station internat-network PICS (PICS_S_INW) is set to true.

5.3 Test architecture

This ITS test specification implements the general TTCN-3 test architecture described in EG 202 798 [i.5], clauses 6.3.2 and 8.3.1.

Figure 5 shows the TTCN-3 test architecture used for the FSAP ATS.

- The MTC is of type ItsFSAP and communicates with the SUT over fsapPort in order to exchange FSAP messages (SAM, CTX) between the FSAP test component and the FASP IUT. The "ITS lower layers transport" system adapter is used to enable usage of ITS lower layers in the SUT in case the MF-SAP is not directly accessible.
- The MTC communicates with the SUT over the utPort in order to trigger FSAP functionalities by simulating primitives from e.g. application entities. It is required to trigger the FSAP layer in the SUT to send FSAP messages, which are resulting from upper layer primitives. Furthermore, receiving FSAP messages may result in notifications to other entities. The "Upper tester transport" system adapter is used to adapt to the upper tester application implementation of the SUT.
- The MTC communicates with the SUT over the cfPort in order to perform settings in the SUT. The "Configuration transport" system adapter is used to adapt to theconfiguratoin access implementation of the SUT.

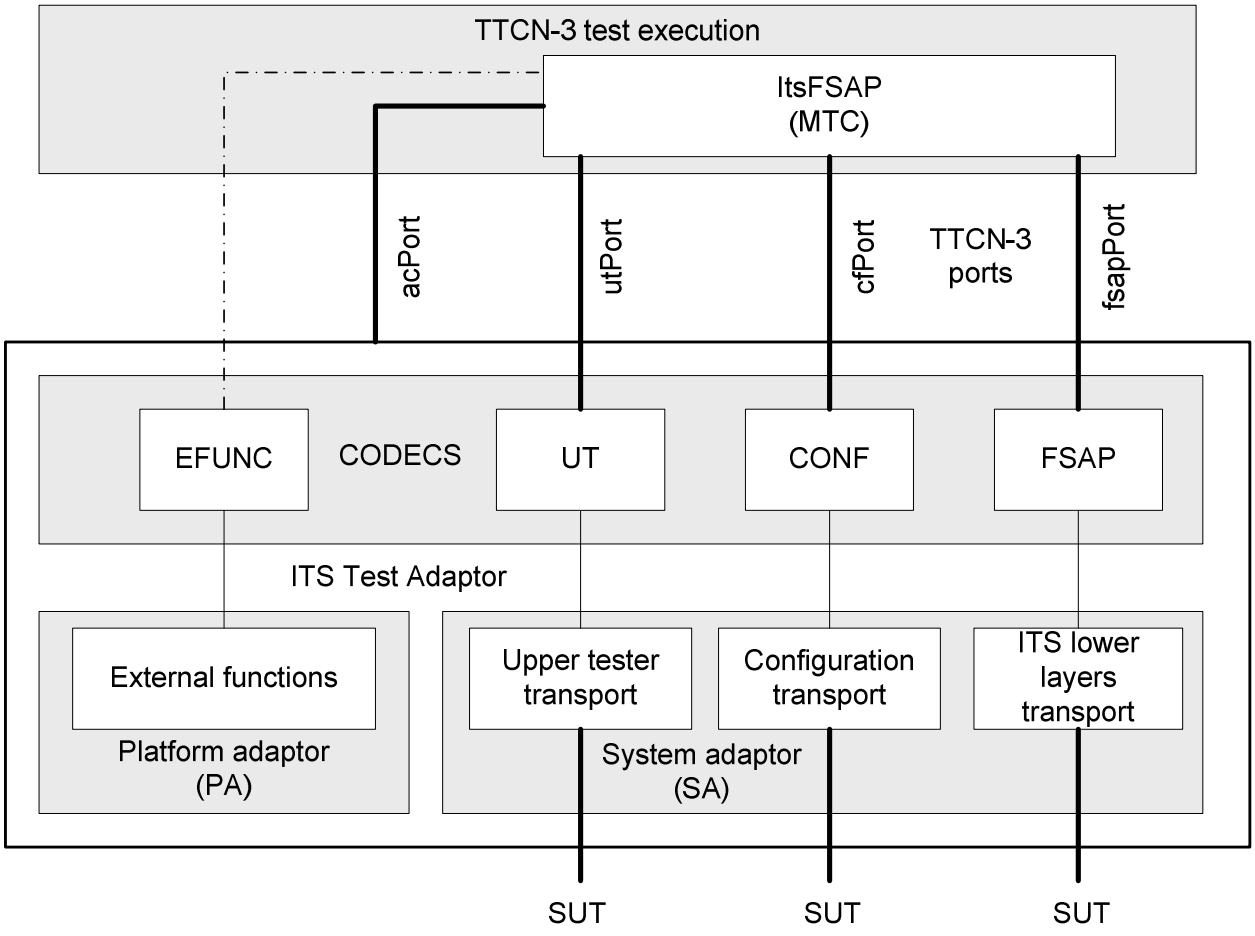

Figure 5: Test system architecture for FSAP

5.4 Ports and abstract service primitives

5.4.1 Overview

The following TTCN-3 ports are used by the FSAP ATS:

- fsapPort of type FsapPort is used to receive messages from and transmit messages to the IUT (via MF-SAP).
- utPort of type UpperTesterPort is used to receive service message from and transmit service messages to the IUT (via MF-SAP).

Every port provides "Abstract Service Primitives" (ASPs) as specified in clauses 5.4.2 and 5.4.3.

5.4.2 ASPs of the fsapPort

One ASP is used in the fsapPort:

- The FsapReq primitive used to receive messages of type MF-Command-Request or MF_COMMAND_Confirm sent by the IUT to the Groupcast Communication Manager.

These primitives use FSAP types, which are declared in the CALMfsap module, following the ASN.1 definition from the base standard [2].

5.4.3 ASPs of the utPort

The following ASPs are used in the utPort:

- The UtInitialize primitive is used to initialise IUT.
- The UtCommandRequest primitive is used to send NF-SAP service primitives to the IUT.
- The UtCommandConfirm primitive is used to received NF-SAP service primitives from the IUT.
- The UtCommandIndication primitive is used to received NF-SAP service primitives from the IUT.

6 Abstract Test Method for IICP

This clause describes the "Abstract Test Method" (ATM) used to test various protocols of the "ITS station management" [i.1], [i.2], [1] and [2].

6.1 Abstract protocol tester

SUTs which support the "ITS station-Internal management Communications Protocol" (IICP) [1] may benefit from the conformance test system architecture illustrated in figure 6, where the access to the IUT from top, i.e. in general via the upper tester application, is performed via management SAPs.

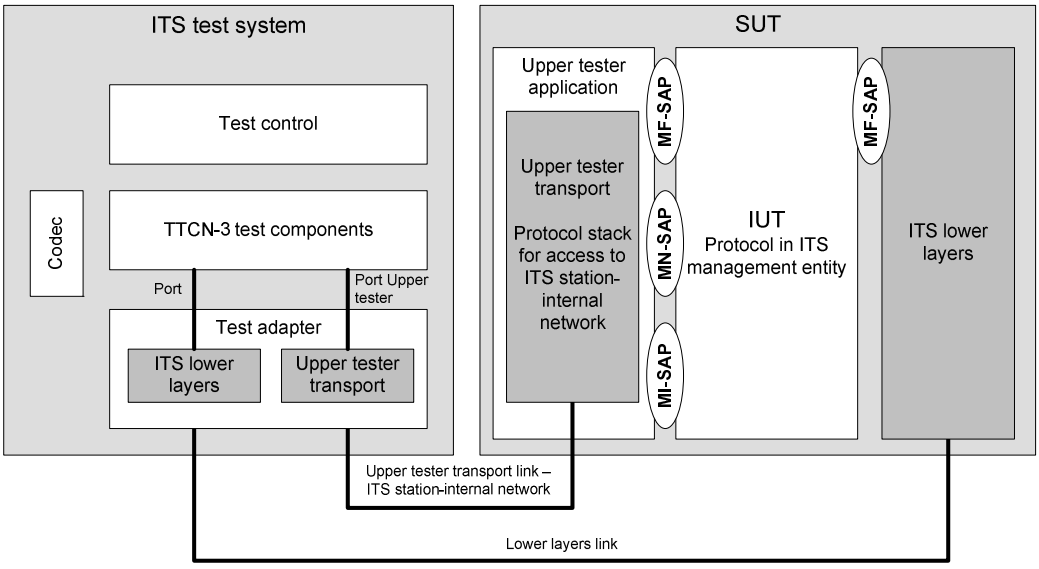

Figure 6: Abstract protocol tester for IICP

6.2 Test configurations

IICP becomes applicable once an ITS station-internal network is available. This results in the basic configuration illustrated in figure 7, where the SUT is an ITS-SCU which either has the role of an ITS-S router, or an ITS-S host, or a combined ITS-S host and router and where the ITS test system simulates such an ITS-SCU at the ITS station-internal network.

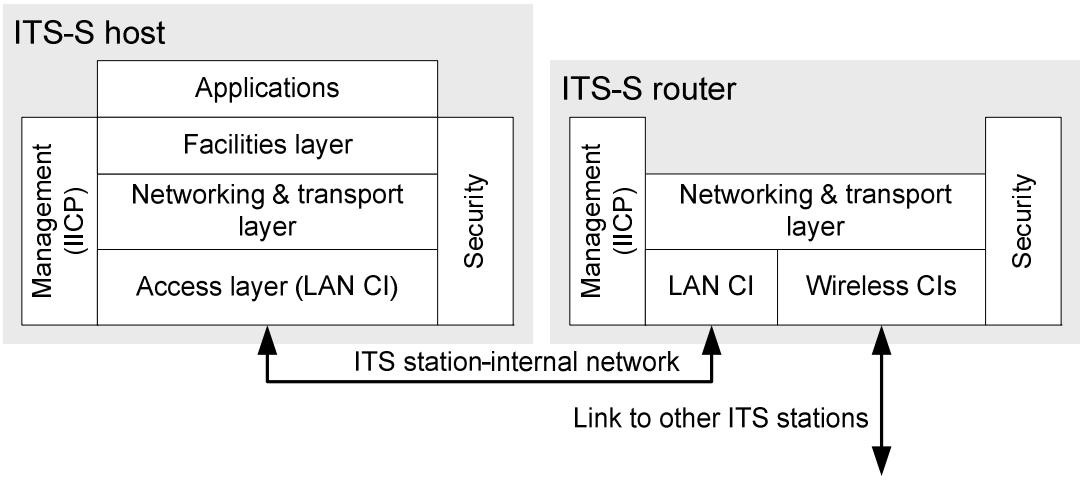

Figure 7: Test configuration CF01 architecture

This configuration is used in the cases listed below [4]:

- ITS-S station internal-network PICS (PICS_S_INW) is set to true.
- Either one of the roles PICS (PICS_ROLE_RH, PICS_ROLE_RONLY, PICS_ROLE_HONLY) is set to true.

6.3 Test architecture

This ITS test specification implements the general TTCN-3 test architecture described in EG 202 798 [i.5], clauses 6.3.2 and 8.3.1.

Figure 5 shows the TTCN-3 test architecture used for the IICP ATS.

- The MTC is of type ItsIICP and communicates with the SUT over iicpPort in order to exchange IICP PDUs between the IICP test component and the IICP IUT. The "ITS lower layers transport" system adapter is used to enable usage of ITS lower layers in the SUT in case the MF-SAP is not directly accessible.
- The MTC communicates with the SUT over the utPort in order to trigger IICP functionalities by simulating primitives from e.g. application entities. It is required to trigger the IICP layer in the SUT to send IICP PDUs, which are resulting from other entities. Furthermore, receiving IICP PDUs may result in notifications to other entities. The "Upper tester transport" system adapter is used to adapt to the upper tester application implementation of the SUT.
- The MTC communicates with the SUT over the cfPort in order to perform settings in the SUT. The "Configuration transport" system adapter is used to adapt to the configuration access implementation of the SUT.

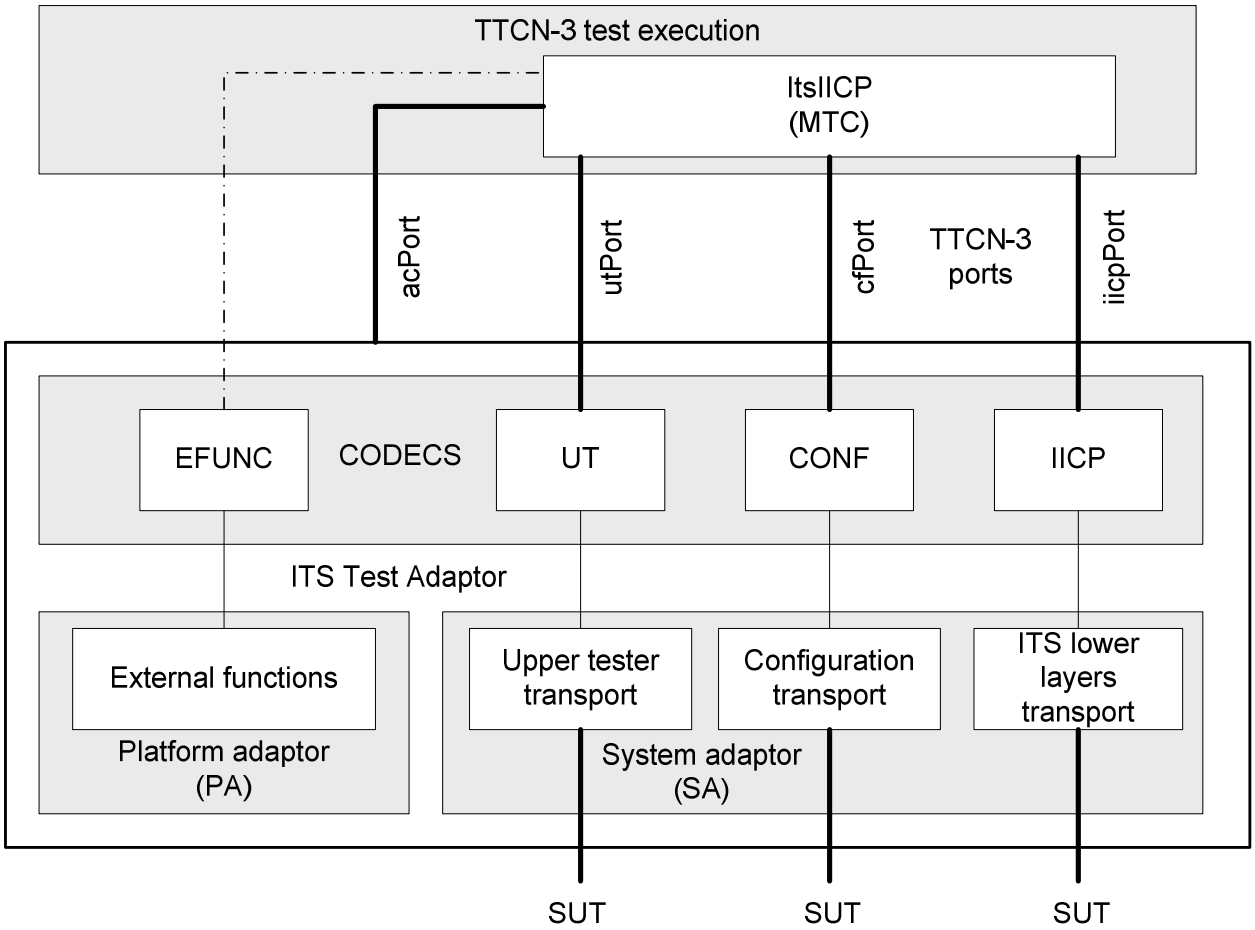

Figure 8: Test system architecture for IICP

6.4 Ports and abstract service primitives

6.4.1 Overview

The following TTCN-3 ports are used by the IICP ATS:

- iicpPort of type IicpPort is used to receive messages from and transmit messages to the IUT (via MF-SAP).
- utPort of type UpperTesterPort is used to receive service message from and transmit service messages to the IUT.
- cfPort of type CfPort is used to configure the IICP (via MF-SAP).

Each of the above ports provides "Abstract Service Primitives" (ASPs) as specified in clauses 6.4.2, 6.4.3 and 6.4.4.

6.4.2 ASPs of the iicpPort

Two ASPs are used in the iicpPort:

- The IicpReq primitive used to receive messages of type MF-Command-Request or MF_COMMAND_Confirmwith IICrequestTX/IICresponseTX sent to the IICA by the IUT.
- The IicpResp primitive used to send messages of type MF-Request.request received from the IUT.

These primitives use IICP types, which are declared in the CALMiitssci module, following the ASN.1 definition from the base standard [1].

6.4.3 ASPs of the utPort

The following ASPs are used in the utPort:

- The UtInitialize primitive is used to initialise IUT.

6.4.4 ASPs of the cfPort

This port is used to simulate the behaviour of the IIC Agent. The following ASPs are used in the cfPort:

- The mgtMFSapRequestReq primitive is used to send and receive messages of types MgtMNSapRequestReq for MF-REQUEST.request service primitives.

7 ATS conventions

The ATS conventions are intended to give a better understanding of the ATS but they also describe the conventions made for the development of the ATS. These conventions shall be considered during any later maintenance or further development of the ATS.

The ATS conventions contain two clauses, the testing conventions and the naming conventions. The testing conventions describe the functional structure of the ATS. The naming conventions describe the structure of the naming of all ATS elements.

To define the ATS, the guidelines of the document ETS 300 406 [7] was considered.

7.1 Testing conventions

7.1.1 Testing states

7.1.1.1 Initial state

All test cases start with the function f_prInitialState. This function brings the IUT in an "initialized" state by invoking the upper tester primitive UtInitialize.

7.1.1.2 Final state

All test cases end with the function f_poDefault. This function brings the IUT back in an "idle" state. As no specific actions are required for the idle state in the base standard, the function f_ poDefault does not invoke any action.

As necessary, further actions may be included in the f_poDefault function.

7.1.2 Message types - ASN.1 definitions

Message types are defined in ASN.1. ASN.1 definitions from the base standard are directly imported in TTCN-3 using the ASN.1 import method specified in ES 201 873-7 [9].

The following example shows the TTCN-3 import statement used to import ASN.1 definitions in the TTCN-3 modules:

```
import from CAMfsap language "ASN.1:1997" all;
```

7.2 Naming conventions

This test suite follows the naming convention guidelines provided in EG 202 798 [i.5].

7.2.1 General guidelines

The naming convention is based on the following underlying principles:

- Identifiers should be prefixed with a short alphabetic string (specified in table 1) indicating the type of TTCN-3 element it represents.
- Suffixes should not be used except in those specific cases identified in table 7.
- Prefixes and suffixes should be separated from the body of the identifier with an underscore ("_").

EXAMPLE 1: c_sixteen, t_wait.

- Only 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.

- The start of second and subsequent words in an identifier should be indicated by capitalizing the first character. Underscores should not be used for this purpose.

EXAMPLE 2: f_initialState.

Table 1 specifies the naming guidelines for each element of the TTCN-3 language indicating the recommended prefix, suffixes (if any) and capitalization.

Table 1: ETSI TTCN-3 generic naming conventions

Language element	Naming convention	Prefix	Example identifier
Module	Use upper-case initial letter	none	FsapTemplates
Group within a module	Use lower-case initial letter	none	transmittingPackets
Data type	Use upper-case initial letter	none	SetupContents
Message template	Use lower-case initial letter	m_	m_setupInit
Message template with wildcard or matching expression	Use lower-case initial letters	mw_	mw_anyUserReply
Modifying message template	Use lower-case initial letter	md_	md_setupInit
Modifying message template with wildcard or matching expression	Use lower-case initial letters	mdw_	mdw_anyUserReply
Signature template	Use lower-case initial letter	s_	s_callSignature
Port instance	Use lower-case initial letter	none	fsapPort
Test component instance	Use lower-case initial letter	none	userTerminal
Constant	Use lower-case initial letter	c_	c_portExt
Constant (defined within component type)	Use lower-case initial letter	cc_	cc_minDuration
External constant	Use lower-case initial letter	cx_	cx_macId
Function	Use lower-case initial letter	f_	f_cf01Up()
External function	Use lower-case initial letter	fx_	fx_calculateLength()
Altstep (incl. Default)	Use lower-case initial letter	a_	a_cf01Down()
Test case	Use ETSI numbering	TC_	TC_FSAP_SP_HR_BV_01
Variable (local)	Use lower-case initial letter	v_	v_pduCounter
Variable (defined within a component type)	Use lower-case initial letters	vc_	vc_pduCounter
Timer (local)	Use lower-case initial letter	t_	t_wait
Timer (defined within a component)	Use lower-case initial letters	tc_	tc_ac
Module parameters for PICS	Use all upper case letters	PICS_	PICS_ROLE_RH
Module parameters for other parameters	Use all upper case letters	PX_	PX_LINK_ID
Formal Parameters	Use lower-case initial letter	p_	p_commRef
Enumerated Values	Use lower-case initial letter	e_	e_success

7.2.2 ITS specific TTCN-3 naming conventions

In addition to such general naming conventions, table 2 shows specific naming conventions that apply to the ITS TTCN-3 test suite.

Table 2: ITS specific TTCN-3 naming conventions

Language element	Naming convention	Prefix	Example identifier
ITS Module	Use upper-case initial letter	Its"IUTname"_	ItsFsap_
Module containing types and values	Use upper-case initial letter	Its"IUTname"_TypesAndValues	ItsFsap_TypesAndValues
Module containing Templates	Use upper-case initial letter	Its"IUTname"_Templates	ItsFsap_Templates
Module containing test cases	Use upper-case initial letter	Its"IUTname"_TestCases	ItsFsap_TestCases
Module containing functions	Use upper-case initial letter	Its"IUTname"_Functions	ItsFsap_Functions
Module containing external functions	Use upper-case initial letter	Its"IUTname"_ExternalFunctions	ItsFsap_ExternalFunctions
Module containing components, ports and message definitions	Use upper-case initial letter	Its"IUTname"_Interface	ItsFsap_Interface
Module containing main component definitions	Use upper-case initial letter	Its"IUTname"_TestSystem	ItsFsap_TestSystem
Module containing the control part	Use upper-case initial letter	Its"IUTname"_TestControl	ItsFsap_TestControl

7.2.3 Usage of Log statements

All TTCN-3 log statements use the following format:

- Three asterisks followed by a blank character.
- The TTCN-3 test case or function identifier in which the log statement is defined followed by a colon and a blank character.
- One of the log categories: INFO, WARNING, ERROR, PASS, FAIL, INCONC, TIMEOUT followed by a colon and a blank character.
- Free text.
- A blank character followed by three asterisks.

EXAMPLE 1:

log("*** TC_FSAP_SP_HR_BV_01: INFO: Preamble: IUT was setup properly ***");

Furthermore, the following rules are applied for the Fsap ATS:

- Log statements are used in the body of the functions, so that invocation of functions are visible in the test logs.
- All TTCN-3 setverdict statements are combined (as defined in TTCN-3 v3.4.1) with a log statement following the same above rules (see example 2).

EXAMPLE 2:

setverdict(pass, "*** TC_FSAP_SP_HR _BV_01: PASS: SAM transmitted at prescribed periodicity ***");

7.2.4 Test Case (TC) identifier

Table 3 shows the test case naming convention for FSAP [2], which follows the test purposes [4] naming convention.

Table 3: FSAP TC naming conventions

TC_<root>_<gr>_<sgr>_<x>_<nn>		
<root> = root	FSAP	Fast Service Advertisement Protocol
<gr> = group	SP	Service provider
	SU	Service user
<sgr> = sub-group	HR	Combined ITS-S host and ITS-S router
	HO	ITS-S host only
	RO	ITS-S router only
<x> = type of testing	BV	Valid Behaviour tests
	BI	Invalid Syntax or Behaviour Tests
<nn> = sequential number		01 to 99

EXAMPLE 1: TP identifier: TP/FSAP/SP/HR/BV/01
TC identifier: TC_FSAP_SP_HR_BV_01

Table 4 shows the test case naming convention for IICP [1], which follows the test purposes [4] naming convention.

Table 4: IICP TC naming conventions

TC_<root>_<gr>_<x>_<nn>		
<root> = root	IICP	Inter-ITS-SCU communication Protocol
<gr> = group	MGM	Management
	COM	Communication
<x> = type of testing	BV	Valid Behaviour tests
	BI	Invalid Syntax or Behaviour Tests
<nn> = sequential number		01 to 99

EXAMPLE 2: TP identifier: TP/IICP/COM/BV/01
TC identifier: TC_IICP_COM_BV_01

Annex A (normative): Partial PIXIT proforma for FSAP

A.1 Identification summary

Table A.1

PIXIT Number:	
Test Laboratory Name:	
Date of Issue:	
Issued to:	

A.2 ATS summary

Table A.2: Summary

Protocol Specification:	ISO/DIS 24102-5
Protocol to be tested:	FSAP
ATS Specification:	TS 102 797-3
Abstract Test Method:	Clause 5

A.3 Test laboratory

Table A.3: Test laboratory

Test Laboratory Identification:	
Test Laboratory Manager:	
Means of Testing:	
SAP Address:	

A.4 Client identification

Table A.4: Client identification

Client Identification:	
Client Test manager:	
Test Facilities required:	

A.5 SUT

Table A.5: SUT identification

Name:	
Version:	
SCS Number:	
Machine configuration:	
Operating System Identification:	
IUT Identification:	
PICS Reference for IUT:	
Limitations of the SUT:	
Environmental Conditions:	

A.6 Protocol layer information

A.6.1 Protocol identification

Table A.6: Protocol identification

Name:	ISO/DIS 24102-5
Version:	
PICS References:	TS 102 797-1

A.6.2 IUT information

Table A.7: Fsap Pixits

Identifier	Description	
PX_WL_LOCAL_CIID	**Comment**	Identifies the CI on ITS-S host
	Type	EUI64
	Def. value	'03000AFFFFFF0000'O
PX_SRC_REMOTE_CIID	**Comment**	Identifies the VCI on ITS-S host
	Type	EUI64
	Def. value	'FF000AFFFFFFFFFF'O
PX_SERVER_ID	**Comment**	The service provider ITS station
	Type	StationID
	Def. value	'11111111'O
PX_CLIENT_ID	**Comment**	The service client ITS station
	Type	StationID
	Def. value	'22222222'O
PX_ITS_AID	**Comment**	The globally unique ITS-AID of the ITS-S application
	Type	ITSaid
	Def. value	{ content := 8 }
PX_UNKNOWN_ITS_AID	**Comment**	An unknown ITS-AID of the ITS-S application
	Type	ITSaid
	Def. value	{ content := 126 }
PX_SESSION_PORT	**Comment**	A session port value
	Type	PortNumber
	Def. value	{ portLong := 1234 }
PX_NO_SESSION_PORT	**Comment**	An unspecified session port value
	Type	PortNumber
	Def. value	{ portLong := c_portNon }
PX_UNKNOWN_SESSION_PORT	**Comment**	An unnknown session port value
	Type	PortNumber
	Def. value	{ portLong := 1234 }
PX_APPLICATION_ID	**Comment**	ITS application object ID (ITS-AID) for GCregServer MF-REQUEST
	Type	ApplicationID

	Def. value	{
		hostITS_scuId := 8,
		seqNumber := 1
		}

PX_CLIENT_APPLICATION_ID	**Comment**	ITS application object ID (ITS-AID) for GCregClient MF-REQUEST
	Type	ApplicationID

	Def. value	{
		hostITS_scuId := 2,
		seqNumber := 2
		}

Identifier	Description	
PX_GSCHED_ACCESS_TECH_NONIP	**Comment**	Scheduling information for registration / deregistration request in order to select the proper VCI in the ITS-station for communication 'medium' field indicates a request of specific access technology
	Type	GCsched

	Def. value	{
		medium := 128,
		directivity := {
		mode := 0,
		dirPredef := 0,
		fill := '0000000'B,
		dirVar := { }
		}, // End of field 'directivity'
		gcInterval := 1
		}

PX_GSCHED_ACCESS_UNKNOWN_TECH_NONIP	**Comment**	Scheduling information for registration / deregistration request in order to select the proper VCI in the ITS-station for communication 'medium' field indicates a request of an unknown access technology
	Type	GCsched

	Def. value	{
		medium := 254,
		directivity := {
		mode := 0,
		dirPredef := 0,
		fill := '0000000'B,
		dirVar := { }
		}, // End of field 'directivity'
		gcInterval := 1
		}

PX_GSCHED_NONIP	**Comment**	Scheduling information for registration / deregistration request in order to select the proper VCI in the ITS-station for communication 'medium' field indicates no request of specific access technology
	Type	GCsched

	Def. value	{
		medium := 1,
		directivity := {
		mode := 0,
		dirPredef := 0,
		fill := '0000000'B,
		dirVar := { }
		}, // End of field 'directivity'
		gcInterval := 1
		}

PX_SERVICE_DATA_REG_WITH_NO_SESSION_PHASE	**Comment**	Receive template for advertisement details with no session phase
	Type	ServiceDataReg

	Def. value	{
		fill := '0000000'B,
		datareg := {
		nonipData := {
		serviceID := PX_ITS_AID,
		timeout_ := 100,
		serviceData := 'O,
		providerPort :=
		PX_NO_SESSION_PORT
		}
		}

Identifier	Description	
PX_SERVICE_DATA_REG_WITH_SESSION_PHASE_AND_CHANNEL_CHANGE	**Comment**	Receive template for advertisement details with session phase
	Type	ServiceDataReg

	Def. value	{
		fill := '0000000'B,
		datareg := {
		nonipData := {
		serviceID := PX_ITS_AID,
		timeout_ := 100,
		serviceData := 'O,
		providerPort :=
		PX_SESSION_PORT
		}
		}

PX_NO_IP_SERVICE_WITH_NO_SESSION_AND_NO_CHANGE_CHANNEL	**Comment**	Non-IP information on services offered, with no session phase and no channel change requested
	Type	ServiceDataReg

	Def. value	{
		serviceID := PX_ITS_AID,
		timeout_ := 100,
		serviceData := 'O,
		providerPort :=
		PX_NO_SESSION_PORT
		}

PX_NO_IP_SERVICE_WITH_SESSION_AND_NO_CHANGE_CHANNEL	**Comment**	Non-IP information on services offered, with session phase and no channel change requested
	Type	NonipService

	Def. value	{
		serviceID := PX_ITS_AID,
		timeout_ := 100,
		serviceData := 'O,
		providerPort :=
		PX_SESSION_PORT,
		sessionChannel := 0
		}

PX_NO_IP_SERVICE_WITH_SESSION_AND_CHANNEL_CHANGE	**Comment**	Non-IP information on services offered, with no session phase and channel change requested
	Type	NonipService

	Def. value	{
		serviceID := PX_ITS_AID,
		timeout_ := 100,
		serviceData := 'O,
		providerPort :=
		PX_SESSION_PORT,
		sessionChannel := 1
		}

PX_NO_IP_SERVICE_WITH_UNKNOWN_SERVICE_ID	**Comment**	Non-IP information on an unknown services offered, with session phase and no channel change requested
	Type	NonipService

	Def. value	{
		serviceID :=
		PX_UNKNOWN_ITS_AID,
		timeout_ := 100,
		serviceData := 'O,
		providerPort :=
		PX_SESSION_PORT,
		sessionChannel := 1
		}

Identifier	Description	
PX_NO_IP_SERVICE_WITH_UNKNOWN_CHANNEL	**Comment**	Non-IP information on services offered, with session phase and channel change requested on an unknown channel
	Type	NonipService
	Def. value	{ serviceID := PX_ITS_AID, timeout_ := 100, serviceData := "O, providerPort := PX_UNKNOWN_SESSION_PORT, sessionChannel := 1 }
PX_FMTID_SAM	**Comment**	SAM tag identifier
	Type	FmtID
	Def. value	0
PX_FMTID_CTX	**Comment**	CTX tag identifier
	Type	FmtID
	Def. value	1
PX_VERSION_FSAP	**Comment**	FSAP version number
	Type	VersionFSAP
	Def. value	0

Table A.8: Configuration Pixits (cfPort)

Identifier	Description	
PX_SRC_ITS_SCU_ID	**Comment**	ITS-SCU-ID of the source ITS-SCU which produces the request
	Type	ITS_scuId
	Def. value	8
PX_DST_ITS_SCU_ID	**Comment**	ITS-SCU-ID of the destination ITS-SCU which shall evaluate the request
	Type	ITS_scuId
	Def. value	9
PX_HOST_SCU_ID	**Comment**	ITS-SCU-ID of the host ITS-SCU
	Type	ITS_scuId
	Def. value	7
PX_HOST_CIID	**Comment**	Host CI identifier
	Type	CIID
	Def. value	0
PX_REMOTE_PORT	**Comment**	Indicate the remote port number
	Type	PortNumber
	Def. value	0
PX_USER_PRIORITY	**Comment**	The user priority as specified in ISO 21218
	Type	UserPriority
	Def. value	0

Annex B (normative): Partial PIXIT proforma for IICP

B.1 Identification summary

Table B.1

PIXIT Number:	
Test Laboratory Name:	
Date of Issue:	
Issued to:	

B.2 ATS summary

Table B.2: Summary

Protocol Specification:	ISO/DIS 24102-4
Protocol to be tested:	IICP
ATS Specification:	TS 102 797-3
Abstract Test Method:	Clause 6

B.3 Test laboratory

Table B.3: Test laboratory

Test Laboratory Identification:	
Test Laboratory Manager:	
Means of Testing:	
SAP Address:	

B.4 Client identification

Table B.4: Client identification

Client Identification:	
Client Test manager:	
Test Facilities required:	

B.5 SUT

Table B.5: SUT identification

Name:	
Version:	
SCS Number:	
Machine configuration:	
Operating System Identification:	
IUT Identification:	
PICS Reference for IUT:	
Limitations of the SUT:	
Environmental Conditions:	

B.6 Protocol layer information

B.6.1 Protocol identification

Table B.6: Protocol identification

Name:	ISO/DIS 24102-4
Version:	
PICS References:	TS 102 797-1

B.6.2 IUT information

Table B.7: IICP Pixits

Identifier	Description	
PX_ACTIVE_VCI_LINK_ID	**Comment**	Defines the active CI link identifier
	Type	Link_ID

	Def. value	{
		remoteCIID := '0000000000000000'O,
		localCIID := '0000000000000000'O
		}

PX_PDU_REQUEST_FILL_FIELD_VALUE	**Comment**	Defines the value to set to fill field for PduRequest field
	Type	Bit4
	Def. value	'0000'B
PX_SRC_ITS_SCU_ID	**Comment**	ITS-SCU-ID of the source ITS-SCU which produces the request
	Type	ITS_scuId
	Def. value	5
PX_DST_ITS_SCU_ID	**Comment**	ITS_SCUtype of the source ITS-SCU which produces the request
	Type	ITS_SCUtype
	Def. value	0
PX_LOCAL_ITS_SCU_ID	**Comment**	The own ITS scuID
	Type	ITS_scuId
	Def. value	8
PX_LOCAL_ITS_TYPE	**Comment**	The type ITS scuID
	Type	ITS_SCUtype
	Def. value	1
PX_HOST_SCU_ID	**Comment**	Host ITS-SCU-ID
	Type	ITS_scuId
	Def. value	0
PX_TALIVE	**Comment**	Alive timer
	Type	Talive
	Def. value	100
PX_MI_RCMD_STATECINOTIFY	**Comment**	MI-Command value used for IICP/COM/xx TPs
	Type	MI_Command

	Def. value	{
		fill := PX_PDU_REQUEST_FILL_FIELD_VALUE,
		miCmd := {
		wakeUp := 10
		}
		}

PX_MN_RCMD_STATECINOTIFY	**Comment**	MN-Command value used for IICP/COM/xx TPs
	Type	MN_Command

	Def. value	{
		fill := '00000'B,
		mnCmd := {
		fWTdelete := {
		fill := '0000000'B,
		delete := {
		fntp := {
		reference := 10
		}
		}
		}
		}
		}

Identifier	Description	
PX_MN_RCMD_FWYSETNOTIFY	**Comment**	MN-Request value used for IICP/COM/xx TPs
	Type	MN_Request
	Def. value	{ fill := '00000'B, mnReq := { fWTsetNot := { fill := '0000000'B, setNot := { fast := { reference := 0, remotePort := { portShort := 0 }, linkID := { remoteCIID := '0000000000000000'O, localCIID := '0000000000000000'O }, ciStatus := 0, linkPort := { portShort := 0 }, serviceInfo := { servicePort := { portShort := 0 }, hostITSscu := 0, servicePriority := 0 }, priority := 0, timeout_ := 0 } } } } }
PX_MI_RCMD_REGTYPE	**Comment**	MI-Request value used for IICP/COM/xx TPs
	Type	MI_Request
	Def. value	{ fill := '0000'B, miReq := { regReq := { medType := MedType_iso17515_ } } }
PX_MF_RCMD_STATECINOTIFY	**Comment**	MF-Command value used for IICP/COM/xx TPs
	Type	MF_Command
	Def. value	{ fill := PX_PDU_REQUEST_FILL_FIELD_VALUE, mfCmd := { stateCInotify := { linkId := { remoteCIID := '0000000000000000'O, localCIID := '0000000000000000'O }, cIstatus := 8 } } }

Identifier	Description	
PX_MF_RCMD_LDM_REGISTER	**Comment**	MF-Command value used for IICP/COM/xx TPs
	Type	MF_Request

	Def. value	{
		fill :=
		PX_PDU_REQUEST_FILL_FIELD_VALUE,
		mfReq := {
		lDMregister := {
		iTS_scuId := 0,
		reference := "O
		}
		}
		}

PX_MI_IPARAMNOLIST	**Comment**	List of reference number of parameter to be monitored
	Type	IParamNoList

	Def. value	{
		0, // AuxiliaryChannel
		1 // ControlChannel
		}

PX_MI_IPARAMLIST	**Comment**	List of error status for each parameter to be monitored
	Type	IParamList

	Def. value	{
		{
		fill := '00'B,
		param_ := {
		errors := {
		{
		paramNo := 0,
		fill := '0000000'B,
		med := { },
		errStatus := 0
		}
		}
		}
		}
		}

PX_MI_ERRORSLIST	**Comment**	List of errors
	Type	ErrorsList

	Def. value	{
		{
		{
		paramNo := 0,
		fill := '0000000'B,
		med := { },
		errStatus := 0
		}
		}
		}

PX_IIC_RESPONSE	**Comment**	Error status in response of MF-REQUEST.request service primitive
	Type	MF_Request_confirm

	Def. value	{
		commandRef := 1,
		reqConfirm := {
		fill := '0000'B,
		mfReqConf := {
		lDMregister := 0
		}
		},
		errStatus := 0
		}

Annex C (normative): TTCN-3 library modules

This ATS has been produced using the Testing and Test Control Notation (TTCN) according to ES 201 873-1 [8]. The ATS was developed on a separate TTCN software tool and therefore the TTCN tables are not completely referenced in the table of contents. The ATS itself contains a test suite overview part which provides additional information and references.

This test suite has been compiled error-free using three different commercial TTCN-3 compilers.

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

The TTCN-3 library modules, which form parts of the present technical standard, are contained in archive ts_10279703v010101p0.zip which accompanies the present document.

History

Document history		
V1.1.1	August 2012	Publication