



**E**UROPEAN  
**T**ELECOMMUNICATION  
**S**TANDARD

**DRAFT**  
pr **ETS 300 141-4**

November 1995

---

Source: ETSI TC SPS

Reference: DE/SPS-05061-S-4

ICS: 33.080

**Key words:** ISDN, DSS1, supplementary service, ATS, PIXIT, testing

**Integrated Services Digital Network (ISDN);  
Call Hold (HOLD) supplementary service;  
Digital Subscriber Signalling System No. one (DSS1) protocol;  
Part 4: Abstract Test Suite (ATS) and partial Protocol  
Implementation eXtra Information for Testing (PIXIT) proforma  
specification for the user**

**ETSI**

European Telecommunications Standards Institute

**ETSI Secretariat**

**Postal address:** F-06921 Sophia Antipolis CEDEX - FRANCE

**Office address:** 650 Route des Lucioles - Sophia Antipolis - Valbonne - FRANCE

**X.400:** c=fr, a=atlas, p=etsi, s=secretariat - **Internet:** secretariat@etsi.fr

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

---

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

© European Telecommunications Standards Institute 1995. All rights reserved.



## Contents

Foreword .....	5
1 Scope .....	7
2 Normative references .....	7
3 Definitions and abbreviations .....	8
3.1 Definitions .....	8
3.2 Abbreviations .....	8
4 Abstract Test Method (ATM) .....	9
5 Untestable test purposes .....	9
6 ATS conventions .....	9
6.1 Declarations part .....	9
6.1.1 Type definitions .....	9
6.1.1.1 Simple type definitions .....	9
6.1.1.2 Structured type definitions .....	10
6.1.1.2.1 TTCN structured type definitions .....	10
6.1.1.2.2 ASN.1 structured type definitions .....	10
6.1.1.3 ASP type definitions .....	11
6.1.1.3.1 TTCN ASP type definitions .....	11
6.1.1.3.2 ASN.1 ASP type definitions .....	12
6.1.1.4 PDU type definitions .....	12
6.1.1.4.1 TTCN PDU type definitions .....	12
6.1.1.4.2 ASN.1 PDU type definitions .....	12
6.1.2 Test suite constants .....	12
6.1.3 Test suite parameters .....	12
6.1.4 Variables .....	12
6.1.4.1 Test suite variables .....	12
6.1.4.2 Test case variables .....	12
6.1.5 Test suite operation definitions .....	13
6.2 Constraints part .....	13
6.2.1 Structured type constraint declaration .....	13
6.2.2 ASN.1 type constraint declaration .....	13
6.2.2.1 Specification of encoding rules .....	14
6.2.3 ASP type constraint declaration .....	15
6.2.3.1 ASN.1 ASP type constraint declaration .....	15
6.2.3.2 TTCN ASP type constraint declaration .....	15
6.2.4 PDU type constraint declaration .....	15
6.2.4.1 ASN.1 PDU type constraint declaration .....	15
6.2.4.2 TTCN PDU type constraint declaration .....	15
6.2.5 Chaining of constraints .....	15
6.2.5.1 Static chaining .....	15
6.2.5.2 Dynamic chaining .....	15
6.2.6 Derived constraint .....	16
6.2.7 Parameterized constraints .....	16
6.2.8 Value assignment .....	16
6.2.8.1 Specific values .....	16
6.2.8.2 Matching values .....	16
6.3 Dynamic part .....	16
6.3.1 Test cases .....	16
6.3.2 Test steps .....	16
6.3.3 Defaults .....	16
7 ATS to TP map .....	17

8	PCTR conformance .....	17
9	PIXIT conformance.....	17
10	ATS conformance.....	17
	Annex A (normative): Protocol Conformance Test Report (PCTR) proforma .....	18
A.1	Identification summary.....	18
A.1.1	Protocol conformance test report.....	18
A.1.2	IUT identification.....	18
A.1.3	Testing environment.....	18
A.1.4	Limits and reservations .....	19
A.1.5	Comments.....	19
A.2	IUT Conformance status.....	19
A.3	Static conformance summary.....	19
A.4	Dynamic conformance summary.....	19
A.5	Static conformance review report.....	20
A.6	Test campaign report.....	20
A.7	Observations.....	21
	Annex B (normative): Partial PIXIT proforma .....	22
B.1	Identification summary.....	22
B.2	Abstract test suite summary .....	22
B.3	Test laboratory.....	22
B.4	Client (of the test laboratory) .....	23
B.5	SUT.....	23
B.6	Protocol information.....	24
B.6.1	Protocol identification .....	24
B.6.2	Parameter values .....	24
B.6.3	Sending of messages by IUT .....	24
B.6.4	Configuration of IUT .....	27
B.6.5	Parameter values - information element codings.....	27
	Annex C (normative): Abstract Test Suite (ATS) .....	28
C.1	The TTCN Graphical form (TTCN.GR) .....	28
C.2	The TTCN Machine Processable form (TTCN.MP) .....	28
	Annex D (informative): General structure of ATS .....	29
	History .....	30

## Foreword

This draft European Telecommunication Standard (ETS) has been produced by the Signalling Protocols and Switching (SPS) Technical Committee of the European Telecommunications Standards Institute (ETSI), and is now submitted for the Public Enquiry phase of the ETSI standards approval procedure.

This ETS is part 4 of a multi-part standard covering the Digital Subscriber Signalling System No. one (DSS1) protocol specification for the Integrated Services Digital Network (ISDN) Call Hold (HOLD) supplementary service, as described below:

- Part 1: "Protocol specification";
- Part 2: "Protocol Implementation Conformance Statement (PICS) proforma specification";
- Part 3: "Test Suite Structure and Test Purposes (TSS&TP) specification for the user";
- Part 4: "Abstract Test Suite (ATS) and partial Protocol Implementation eXtra Information for Testing (PIXIT) proforma specification for the user";**
- Part 5: "TSS&TP specification for the network";
- Part 6: "ATS and partial PIXIT proforma specification for the network".

<b>Proposed transposition dates</b>	
Date of latest announcement of this ETS (doa):	3 months after ETSI publication
Date of latest publication of new National Standard or endorsement of this ETS (dop/e):	6 months after doa
Date of withdrawal of any conflicting National Standard (dow):	6 months after doa

Blank page

## 1 Scope

This fourth part of ETS 300 141 specifies the Abstract Test Suite (ATS) and partial Protocol Implementation eXtra Information for Testing (PIXIT) proforma for the User side of the T reference point or coincident S and T reference point (as defined in ITU-T Recommendation I.411 [11]) of implementations conforming to the stage three standard for the Call Hold (HOLD) supplementary service for the pan-European Integrated Services Digital Network (ISDN) by means of the Digital Subscriber Signalling System No. one (DSS1) protocol, ETS 300 141-1 [2].

ETS 300 141-3 [4] specifies the Test Suite Structure and Test Purposes (TSS&TP) related to this ATS and partial PIXIT proforma. Other parts specify the TSS&TP and the ATS and partial PIXIT proforma for the Network side of the T reference point or coincident S and T reference point of implementations conforming to ETS 300 141-1 [2].

## 2 Normative references

This ETS incorporates by dated and undated reference, provisions from other publications. These normative references are cited at the appropriate places in the text and the publications are listed hereafter. For dated references, subsequent amendments to or revisions of any of these publications apply to this ETS only when incorporated in it by amendment or revision. For undated references the latest edition of the publication referred to applies.

[1] ETS 300 102-1: "Integrated Services Digital Network (ISDN); User-network interface layer 3; Specifications for basic call control".

[2] ETS 300 141-1 (1993): "Integrated Services Digital Network (ISDN); Call Hold (HOLD) supplementary service; Digital Subscriber Signalling System No. one (DSS1) protocol; Part 1: Protocol specification".

NOTE 1: ETS 300 141-1 (1993) was initially published as ETS 300 141 (1993).

[3] ETS 300 141-2 (1995): "Integrated Services Digital Network (ISDN); Call Hold (HOLD) supplementary service; Digital Subscriber Signalling System No. one (DSS1) protocol; Part 2: Protocol Implementation Conformance Statement (PICS) proforma specification".

[4] ETS 300 141-3: "Integrated Services Digital Network (ISDN); Call Hold (HOLD) supplementary service; Digital Subscriber Signalling System No. one (DSS1) protocol; Part 3: Test Suite Structure and Test Purposes (TSS&TP) specification for the user".

[5] ETS 300 196-1 (1993): "Integrated Services Digital Network (ISDN); Generic functional protocol for the support of supplementary services; Digital Subscriber Signalling System No. one (DSS1) protocol; Part 1: Protocol specification".

NOTE 2: ETS 300 196-1 (1993) was initially published as ETS 300 196 (1993).

[6] ISO/IEC 9646-1: "Information Technology - OSI Conformance Testing Methodology and Framework; Part 1: General Concepts".

[7] ISO/IEC 9646-2: "Information Technology - OSI Conformance Testing Methodology and Framework; Part 2: Abstract Test Suite Specification".

[8] ISO/IEC 9646-3: "Information Technology - OSI Conformance Testing Methodology and Framework; Part 3: The Tree and Tabular Combined Notation".

[9] ISO/IEC 9646-4: "Information Technology - OSI Conformance Testing Methodology and Framework; Part 4: Test realization".

- [10] ISO/IEC 9646-5: "Information Technology - OSI Conformance Testing Methodology and Framework; Part 5: Requirements on test laboratories and clients for the conformance assessment process".
- [11] ITU-T Recommendation I.411 (1993): "ISDN user-network interfaces - Reference configurations".
- [12] CCITT Recommendation X.209 (1988): "Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1)".

### 3 Definitions and abbreviations

#### 3.1 Definitions

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

**Abstract Test Suite (ATS):** See ISO/IEC 9646-1 [6].

**Implementation Under Test (IUT):** See ISO/IEC 9646-1 [6].

**System Under Test (SUT):** See ISO/IEC 9646-1 [6].

**Protocol Implementation Conformance Statement (PICS):** See ISO/IEC 9646-1 [6].

**PICS proforma:** See ISO/IEC 9646-1 [6].

**Protocol Implementation eXtra Information for Testing (PIXIT):** See ISO/IEC 9646-1 [6].

**PIXIT proforma:** See ISO/IEC 9646-1 [6].

**Lower Tester (LT):** See ISO/IEC 9646-1 [6].

**Upper Tester (UT):** See ISO/IEC 9646-1 [6].

**Point of Control and Observation (PCO):** See ISO/IEC 9646-1 [6].

#### 3.2 Abbreviations

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

ASP	Abstract Service Primitive
ATM	Abstract Test Method
ATS	Abstract Test Suite
BER	Basic Encoding Rules
ExTS	Executable Test Suite
HOLD	Call Hold
IE	Information Element
IUT	Implementation Under Test
LT	Lower Tester
MOT	Means Of Testing
PCO	Point of Control and Observation
PDU	Protocol Data Unit
PICS	Protocol Implementation Conformance Statement
PIXIT	Protocol Implementation eXtra Information for Testing
SUT	System Under Test
TCP	Test Co-ordination Procedures
TP	Test Purpose
TTCN	Tree and Tabular Combined Notation
UT	Upper Tester



## 4 Abstract Test Method (ATM)

The remote test method is applied for the 3PTY user ATS. The Point of Control and Observation (PCO) resides at the service access point between layers 2 and 3. This PCO is named "L" (for Lower). The L PCO is used to control and observe the behaviour of the Implementation Under Test (IUT) and test case verdicts are assigned depending on the behaviour observed at this PCO.

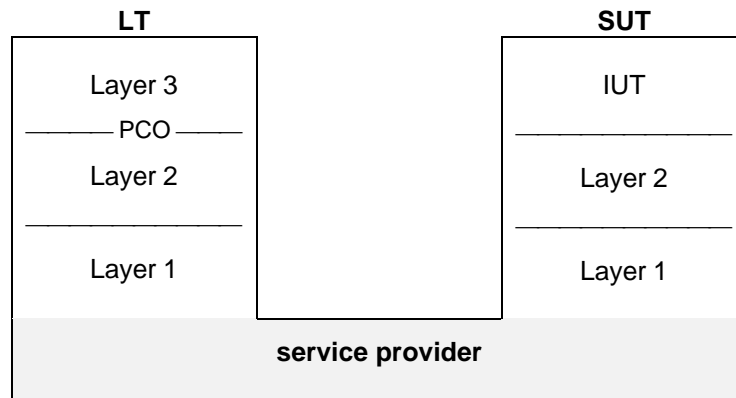


Figure 1: Remote test method

ISO/IEC 9646-2 [7] allows the informal expression of Test Co-ordination Procedures (TCP) between the System Under Test (SUT) upper layer(s) and the Lower Tester (LT). In the ATS contained in annex C, TCP is achieved by use of a second "informal" PCO, called "O" (for Operator). This PCO is used to specify control but not observation above the IUT and consequently, events at this PCO are never used to generate test case verdicts. The use of this O PCO is regarded as a preferred alternative to the use of the implicit send event, in that it allows the ATS to specify in a clear and meaningful way what actions are required to be performed on the IUT.

## 5 Untestable test purposes

There are no untestable test purposes associated with this ATS.

## 6 ATS conventions

This clause is structured similarly to the structure of a TTCN ATS. However, the names of the subclauses are arranged in a way more suitable to this ETS.

### 6.1 Declarations part

#### 6.1.1 Type definitions

##### 6.1.1.1 Simple type definitions

Where appropriate, simple types have a length, a value list or a range restriction attached.

Simple types defined as being of some string type (e.g. BITSTRING, OCTETSTRING), have a length restriction or a value list attached.

Simple types, defined as being of INTEGER type, have a value list or a range restriction attached.

**6.1.1.2 Structured type definitions**

**6.1.1.2.1 TTCN structured type definitions**

All structured type definitions are provided with a full name.

All elements in every structured type definition, defined as being of some string type (e.g. BITSTRING, OCTETSTRING), have a length restriction attached.

If an element in a structured type definition is defined as being of a referenced type, the (possible) restriction is defined in that referenced type.

For information elements the identifier, which is unique for each element, has its type defined as a simple type where the value list is restricted to the single value which is the identifier itself. This has the advantage that it allows a test system derived from this ATS to easily identify information elements embedded in messages. An ATS where information element identifiers are represented as unrestricted types can present difficulties for a derived test system in the case where it needs to find one information element embedded in a number of others and the constraints for the other elements have the any-or-omit value. In such a case the test system cannot easily find the beginning of each information element.

**6.1.1.2.2 ASN.1 structured type definitions**

ASN.1 has been used for three major reasons. First, types defined in ASN.1 can model problems that "pure" TTCN cannot. For instance, data structures modelling ordered or unordered sequences of data are preferably defined in ASN.1. Second, ASN.1 provides a better restriction mechanism for type definitions by using sub-type definitions. Third, it is necessary to use ASN.1 to reproduce the type definitions for remote operation components as specified in the base standards.

The fact that ASN.1 provides a better restriction mechanism for type definitions is used for the purpose of achieving type-compatibility.

In table 1, the ASN.1 type BIT7OR15 is defined as being of type BIT STRING with a size constraint attached to it. The size is determined by the value of CR\_LENGTH, a test suite parameter. It can have the value of either 7 or 15. The type BIT7OR15 is used in the structured type CR, field cr\_r allowing this type to represent a Basic Access or a Primary Rate Access call reference. By using this type definition the field cr\_r is always type compatible with values of type BIT STRING (SIZE(7)) and BIT STRING (SIZE(15)). Another approach to solve this type problem would be to define the type BIT7OR15 as BIT STRING(SIZE(7 | 15)). This type has a small disadvantage compared with the previous one. It is impossible, in run-time, to determine the actual length of any instance of this type.

**Table 1: ASN.1 type definition BIT7OR15**

ASN.1 Type Definition	
Type Name :	BIT7OR15
Comments :	
Type Definition	
BIT STRING(SIZE(CR_LENGTH))	

Table 2 shows a typical use of ASN.1. The CHI element will have two different type definitions depending on whether it represents basic or primary rate access. In TTCN, this needs to be defined as two different types. In ASN.1 this can be done in one, the type being a choice of either BASIC\_CHI or PRIMARY\_CHI. These two types are then (locally) defined in the same table and according to the standard.

Table 2: ASN.1 type definition CHI

ASN.1 Type Definition	
<b>Type Name</b> :	CHI
<b>Comments</b> :	Info Element Channel Identification ETS 300 102-1 clause 4.5.13
Type Definition	
<pre> CHOICE {   basic    BASIC_CHI,   primary  PRIMARY_CHI }  -- Local type definitions --  BASIC_CHI ::= SEQUENCE {   chi_i    CHI_I,           -- Identifier   chi_l    BIT STRING(SIZE(8)), -- Length   chi_e3_cs BIT STRING(SIZE(8)) -- Channel selection }  PRIMARY_CHI ::= SEQUENCE {   chi_i    CHI_I,           -- Identifier   chi_l    BIT STRING(SIZE(8)), -- Length   chi_e3_p1 BIT STRING(SIZE(4)), -- First nibble of Channel selection   chi_e3_pe BIT STRING(SIZE(1)), -- Preferred/Exclusive Bit   chi_e3_p3 BIT STRING(SIZE(3)), -- Last three bits of Channel selection   chi_e4    BIT STRING(SIZE(8)), -- Channel type   chi_e5_ch1 BIT STRING(SIZE(1)),   chi_e5_ch2 BIT STRING(SIZE(7)) -- Channel number } </pre>	

Table 3 shows an example of how ASN.1 can be used to model unordered sequences.

Table 3: ASN.1 type definition FIES

ASN.1 Type Definition	
<b>Type Name</b> :	FIES
<b>Comments</b> :	
Type Definition	
SET OF FIE	

The possibility to use TTCN and ASN.1 in combination is used, i.e. referring to an ASN.1 type from a TTCN type.

### 6.1.1.3 ASP type definitions

#### 6.1.1.3.1 TTCN ASP type definitions

TTCN ASP type definitions only contain one PDU or no PDU at all. The relationship between an ASP type and a PDU type is one-to-one. That is, there exists one ASP type definition for each PDU type definition (if that ASP type contains a PDU).

All TTCN ASP type definitions are provided with a full identifier.

Some ASPs are not parameterized as shown in the example in table 4. Such ASPs are only used for requesting or receiving service from the lower layer.

Table 4: TTCN ASP type definition DL\_REL\_IN

TTCN ASP Type Definition		
<b>ASP NAME</b> : DL_REL_IN (DL_RELEASE_INDICATION)		
<b>PCO Type</b> : SAP		
<b>Comments</b> :		
Parameter Name	Parameter Type	Comments
<b>Detailed Comments</b> :		

Table 5 shows an example of a parameterized ASP. All ASPs containing PDUs contain only that PDU and no other parameters.

**Table 5: TTCN ASP type definition DL\_DATA\_RQ\_ALERT**

TTCN ASP Type Definition		
ASP NAME : DL_DATA_RQ_ALERT (DL_DATA_REQUEST)		
PCO Type : SAP		
Comments :		
Parameter Name	Parameter Type	Comments
mun (MessageUnit)	ALERT_PDU	
Detailed Comments :		

**6.1.1.3.2 ASN.1 ASP type definitions**

There are no ASN.1 ASP type definitions in the ATS.

**6.1.1.4 PDU type definitions**

**6.1.1.4.1 TTCN PDU type definitions**

The TTCN PDU type reflects the actual data being transferred or received. All PDUs are embedded in ASPs.

If a specific PDU type definition contains elements defined in terms of a pre-defined type, that element has a restriction attached to it.

**6.1.1.4.2 ASN.1 PDU type definitions**

There are no ASN.1 PDU type definitions in the ATS.

**6.1.2 Test suite constants**

No test suite constants are used or defined in this ATS.

**6.1.3 Test suite parameters**

Each test suite parameter is defined in terms of a predefined type or a referenced type. A referenced type is used when it is necessary to attach restrictions to these type definitions (it is not allowed to include restrictions directly in the test suite parameter table). The referenced type can have a length or value restriction attached to it in its declaration table.

**6.1.4 Variables**

**6.1.4.1 Test suite variables**

No test suite variables are used or defined in this ATS.

**6.1.4.2 Test case variables**

Each test case variable is defined in terms of a predefined type or a referenced type. A referenced type is used when it is necessary to attach restrictions to these type definitions (it is not allowed to include restrictions directly in the test case variable table). The referenced type can have a length or value restriction attached to it in its declaration table.

Where test case variables are used in constraints, they are passed as formal parameters.

### 6.1.5 Test suite operation definitions

The description part of a test suite operation definition uses either natural language or meta C.

**Table 6: Test suite operation definition ASSIGN\_CHI**

Test Suite Operation Definition	
<b>Operation Name</b>	: ASSIGN_CHI(basic, primary : CHI; basic_flag : BOOLEAN)
<b>Result Type</b>	: CHI
<b>Comments</b>	: This operation is used to assign a correct Channel identification information element to PDUs dependent on the type of access that is tested.
Description	
<pre>{ if(basic_flag) return basic; else return primary }</pre>	
<b>Detailed comments</b>	:

The test suite operation definition shown in table 6 is used in the constraints part when assigning an element of type CHI a value. As previously described, the CHI type can be defined in two ways depending on whether the ATS is testing basic or primary rate access. To avoid duplicate types and thereby duplicate test cases the CHI type is defined in ASN.1. This operation is used to assign a value to an element of CHI type. It takes three parameters:

```
primary:          a constraint of type CHI valid for primary rate access;
basic:            a constraint of type CHI valid for basic access;
basic_flag:      a boolean value: TRUE if basic access is applicable, FALSE otherwise.
```

This operation returns the correct constraint according to the boolean flag basic\_flag. That constraint will then be assigned to the specific element of type CHI.

## 6.2 Constraints part

### 6.2.1 Structured type constraint declaration

For every structured type definition there exists one or more structured type constraint.

### 6.2.2 ASN.1 type constraint declaration

Constraints of this type are used to assign the corresponding type a specific value. These constraints are used for the purpose of modelling unordered data or specific types that cannot be expressed in TTCN.

A value assigned to an element of type SET OF differs depending on whether it is a send or receive constraint.

**Table 7: ASN.1 type constraint declaration fIEs (send constraint)**

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: fIEs(comp : Component)
<b>ASN.1 Type</b>	: FIE
<b>Derivation Path</b>	:
<b>Comments</b>	: Send fie which will contain one component "comp".
Description	
<pre>{ informationElementIdentifier  '00011100'B, length                       CALC_FIE_LENGTH(comp), extBit                        '1'B, spareBits                     '00'B, protocolProfile               '10001'B, components                    {comp} }</pre>	
<b>Detailed comments</b>	:

**NOTE:** The last element in the constraint, *components*, is of type *SET OF Component* where *Component* is structured data of some type.

If the constraint is a send constraint (see table 7) the value for the component element is stated as "{comp}" where comp is an argument received as a parameter. The "{" and "}" turns the value into a SET OF value which is correct according to that element's type definition.

**Table 8: ASN.1 type constraint declaration fIEr (receive constraint)**

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: fIEr(comp : Component)
<b>ASN.1 Type</b>	: FIE
<b>Derivation Path</b>	:
<b>Comments</b>	: A received FIE which can contain several components, but which contains at least "comp".
Description	
{	
informationElementIdentifier	'00011100'B,
length	'????????'B,
extBit	'1'B,
spareBits	'00'B,
protocolProfile	'10001'B,
components	SUPERSET({comp})
}	
<b>Detailed comments</b>	:

NOTE: The last element in the constraint, named *components*, is of type *SET OF Component* where *Component* is structured data of some type.

If the constraint is a receive constraint (as in table 8) the corresponding matching value is assigned by using SUPERSET. The key-word SUPERSET has an argument that is type compatible with the type definition of that field. In table 8, the element named *components* is defined as "SET OF Component" and this implies that the argument to SUPERSET should be of type SET OF Component. This is achieved the same way as for send constraints, enclosing the value in curly brackets.

The semantic of SUPERSET is stated in ISO/IEC 9646-3 [8], subclause 11.6.4.7. In short it defines the semantic as follows: "A value that uses SUPERSET matches the incoming value if, and only if, the incoming value contains **at least all of the elements** defined within the SUPERSET, and **may contain more elements**." This is exactly the semantic definition used in this ATS.

**6.2.2.1 Specification of encoding rules**

At the time of specifying this ATS the mechanisms related to encoding of ASN.1 types, specified in DAM-2 of ISO/IEC 9646-3 [8], were not yet stable. Nevertheless as there is a variation in the encoding rules as applied to ASN.1 types and constraints specified in this ATS, a mechanism is used to differentiate the different encoding rules. Given the non-finalized status of DAM-2, a solution which is broadly in the spirit of DAM-2 has been created. Comment fields have been used as a means of including the encoding rules.

For ASN.1 used in this ATS, two variations of encoding rules are used. One is the commonly known Basic Encoding Rules (BER) as specified in CCITT Recommendation X.209 [12]. In the second case the encoding is according to ISDN, i.e. the ASN.1 data types are a representation of structures contained within the ISDN specification (basic call, Generic functional protocol or individual supplementary service). For example, if octets of an information element are specified in ASN.1 as a SEQUENCE then this should be encoded in an Executable Test Suite (ExTS) as any other ISDN information element specified using tabular TTCN. This ISDN encoding variation is the default encoding rule for this ATS. This means that all ASN.1 constraint tables are encoded using ISDN (non-BER) encoding unless stated otherwise. BER encoding should never be applied to an ASN.1 constraint where BER encoding has not been specified.

For BER encoding, an indication is given in the comments field of the table header. For this ATS such indications appear in the ASN.1 type constraint declaration tables only. In the first line of the table header comment field, the notation "ASN1\_Encoding: BER" is used.

Note that within BER, there are a number of variations for the encoding of lengths of fields. According to ETS 300 196-1 [5], an IUT should be able to interpret all length forms within BER for received PDUs. When sending PDUs containing BER encoding, ETS 300 196-1 [5] gives guidelines but makes no restrictions on the length forms within BER which an IUT may apply.

In relation to components sent by the tester to the IUT, implementors of this ATS shall use a variety of length forms such that at least one of each of the length forms is sent to the IUT during a test campaign. The variations of length forms to be used are indefinite, short definite and long definite.

In this particular ATS all ASN.1 type constraints which are of type "Component" are to be encoded using BER.

**Table 9: ASN.1 type constraint declaration showing use of encoding variation**

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b> :	Beg3PTYinv
<b>ASN.1 Type</b> :	Component
<b>Derivation Path</b> :	
<b>Comments</b> :	ASN1_Encoding: BER Receive component: Begin3PTY invoke component
Description	
begin3PTY_Components begin3PYT_InvokeComp { invokeID           ? , operation_value    localValue   4 }	
<b>Detailed comments</b> :	

### 6.2.3 ASP type constraint declaration

#### 6.2.3.1 ASN.1 ASP type constraint declaration

No ASN.1 ASP type constraint declaration exist in this ATS.

#### 6.2.3.2 TTCN ASP type constraint declaration

For TTCN ASP constraint declarations there is a one-to-one relationship between its type and the constraint. That is, there is only one constraint for each TTCN ASP Type Declaration. The reason for this is that the ASPs are used only for carrying a specific PDU value. The many ASP constraints (and types) could have been avoided by using the meta type **PDU**, but that was not suitable as values inside a specific PDU have to be referenced. To reference elements inside a value of meta type **PDU** is not allowed according to ISO/IEC 9646-3 [8], so each ASP has to be defined as having a parameter of a specific PDU type.

In all ASP constraints the embedded PDU constraint is either chained static or "semi-dynamic". That is, the PDU constraint is always fixed to a specific ASP constraint but it (the PDU) may be parameterized.

All ASP constraints have a specific value for its parameter. No matching symbols are used in ASPs.

### 6.2.4 PDU type constraint declaration

#### 6.2.4.1 ASN.1 PDU type constraint declaration

No ASN.1 PDU type constraint declaration exists in this ATS.

#### 6.2.4.2 TTCN PDU type constraint declaration

PDU constraints are used for assigning values or patterns to the data being sent or received.

### 6.2.5 Chaining of constraints

#### 6.2.5.1 Static chaining

Static chaining, that is a fixed reference to a specific constraint, is used in this ATS. The static chaining is used for static binding of both variables and sub-structures.

#### 6.2.5.2 Dynamic chaining

Dynamic chaining is achieved when having a reference to a value which is unknown. The only thing known (before run-time) is the type of that reference. The reference is passed as a parameter. Strict dynamic chaining is not used in this ATS. What is used is something that is called "semi-dynamic chaining". The definition of semi-dynamic chaining is that the fixed reference is parameterized with an unknown value. That value is received as a parameter.

**Table 10: TTCN ASP constraint declaration A\_RST1**

TTCN ASP Constraint Declaration		
Constraint Name : A_RST1 (FLAG : INTEGER)		
ASN.1 Type : DL_DAT_IN_RESTARTr		
Derivation Path :		
Comments :		
Parameter Name	Parameter Value	Comments
mun	RST1 (FLAG)	RST1 (FLAG)
Detailed comments :		

Table 10 is an example of semi-dynamic chaining. The TTCN ASP constraint is parameterized with an INTEGER value named FLAG. That value is passed further down in the structure as a parameter to a static named PDU constraint reference.

**6.2.6 Derived constraint**

No derivation of any constraints is used. All constraints are considered to be base constraints.

**6.2.7 Parameterized constraints**

Parameterized constraints are used in this ATS.

**6.2.8 Value assignment**

**6.2.8.1 Specific values**

For specific value assignment both explicit values and references to explicit values are used.

**6.2.8.2 Matching values**

As matching values the following mechanisms are used:

- Instead of Value:
- AnyOrOmit "\*"
  - AnyValue "?"
  - SuperSet SUPERSET
  - Omit "-"
- Inside value:
- AnyOne "?"
  - AnyOrNone "\*"

**6.3 Dynamic part**

**6.3.1 Test cases**

Each test case contains the test purpose text from ETS 300 141-3 [4]. To be able to read and understand the test case dynamic behaviour it is recommended that the test steps are understood first.

**6.3.2 Test steps**

Much use has been made of test steps to avoid needless repetition of dynamic behaviour. Many test steps are based on those used for the ISDN basic call ATS.

**6.3.3 Defaults**

Note the use of the RETURN statement which is defined in DAM1 of ISO/IEC 9646-3 [8]. This allows valid background behaviour to be handled in the default tree with a possibility to return to the original set of alternatives in the test case.



## 7 ATS to TP map

The identifiers used for the TPs are reused as test case names. Thus there is a straightforward one-to-one mapping.

## 8 PCTR conformance

A test laboratory, when requested by a client to produce a PCTR, is required, as specified in ISO/IEC 9646-5 [10], to produce a PCTR conformant with the PCTR template given in annex B of ISO/IEC 9646-5 [10].

Furthermore, a test laboratory, offering testing for the ATS specification contained in annex C, when requested by a client to produce a PCTR, is required to produce a PCTR conformant with the PCTR proforma contained in annex A of this ETS.

A PCTR which conforms to this PCTR proforma specification shall preserve the content and ordering of the clauses contained in annex A. Clause A.6 of the PCTR may contain additional columns. If included, these shall be placed to the right of the existing columns. Text in italics may be retained by the test laboratory.

## 9 PIXIT conformance

A test realizer, producing an executable test suite for the ATS specification contained in annex C, is required, as specified in ISO/IEC 9646-4 [9], to produce an augmented partial PIXIT proforma conformant with this partial PIXIT proforma specification.

An augmented partial PIXIT proforma which conforms to this partial PIXIT proforma specification shall, as a minimum, have contents which are technically equivalent to annex B. The augmented partial PIXIT proforma may contain additional questions that need to be answered in order to prepare the Means Of Testing (MOT) for a particular IUT.

A test laboratory, offering testing for the ATS specification contained in annex C, is required, as specified in ISO/IEC 9646-5 [10], to further augment the augmented partial PIXIT proforma to produce a PIXIT proforma conformant with this partial PIXIT proforma specification.

A PIXIT proforma which conforms to this partial PIXIT proforma specification shall, as a minimum, have contents which are technically equivalent to annex B. The PIXIT proforma may contain additional questions that need to be answered in order to prepare the test laboratory for a particular IUT.

## 10 ATS conformance

The test realizer, producing MOT and ExTS for this ATS specification, shall comply with the requirements of ISO/IEC 9646-4 [9]. In particular, these concern the realization of an ExTS based on each ATS. The test realizer shall provide a statement of conformance of the MOT to this ATS specification.

An ExTS which conforms to this ATS specification shall contain test groups and test cases which are technically equivalent to those contained in the ATS in annex C. All sequences of test events comprising an abstract test case shall be capable of being realized in the executable test case. Any further checking which the test system might be capable of performing is outside the scope of this ATS specification and shall not contribute to the verdict assignment for each test case.

Test laboratories running conformance test services using this ATS shall comply with ISO/IEC 9646-5 [10].

A test laboratory which claims to conform to this ATS specification shall use an MOT which conforms to this ATS.

## Annex A (normative): Protocol Conformance Test Report (PCTR) proforma

Notwithstanding the provisions of the copyright clause related to the text of this ETS, ETSI grants that users of this ETS may freely reproduce the PCTR proforma in this annex so that it can be used for its intended purposes and may further publish the completed PCTR.

### A.1 Identification summary

#### A.1.1 Protocol conformance test report

PCTR number:	
PCTR date:	
Corresponding SCTR number:	
Corresponding SCTR date:	
Test laboratory identification:	
Test laboratory manager:	
Signature:	

#### A.1.2 IUT identification

Name:	
Version:	
Protocol specification:	ETS 300 141-1
PICS:	
Previous PCTRs (if any)	

#### A.1.3 Testing environment

PIXIT Reference number:	
ATS Specification:	ETS 300 141-4
Abstract Test Method:	Remote test method (see ISO/IEC 9646-2)
Means of Testing identification:	
Dates of testing:	
Conformance log reference(s):	
Retention date for log reference(s):	

**A.1.4 Limits and reservations**

*Additional information relevant to the technical contents or further use of the test report, or to the rights and obligations of the test laboratory and the client, may be given here. Such information may include restriction on the publication of the report.*

.....  
.....  
.....  
.....

**A.1.5 Comments**

*Additional comments may be given by either the client or the test laboratory on any of the contents of the PCTR, for example, to note disagreement between the two parties.*

.....  
.....  
.....  
.....

**A.2 IUT Conformance status**

This IUT has / has not been shown by conformance assessment to be non-conforming to the specified protocol specification.

*Strike the appropriate words in this sentence. If the PICS for this IUT is consistent with the static conformance requirements (as specified in clause A.3 of this report) and there are no "FAIL" verdicts to be recorded (in clause A.6) strike the words "has or", otherwise strike the words "or has not".*

**A.3 Static conformance summary**

The PICS for this IUT is / is not consistent with the static conformance requirements in the specified protocol.

*Strike the appropriate words in this sentence.*

**A.4 Dynamic conformance summary**

The test campaign did / did not reveal errors in the IUT.

*Strike the appropriate words in this sentence. If there are no "FAIL" verdicts to be recorded (in clause A.6 of this report) strike the word "did", otherwise strike the words "did not".*

Summary of the results of groups of tests:

.....  
.....  
.....  
.....  
.....

### A.5 Static conformance review report

*If clause A.3 indicates non-conformance, this subclause itemizes the mismatches between the PICS and the static conformance requirements of the specified protocol specification.*

.....

.....

.....

.....

.....

.....

.....

.....

### A.6 Test campaign report

ATS reference	Selected? (Y/N)	Run? (Y/N)	Verdict	Observations
HOLD_U01_001				
HOLD_U01_002				
HOLD_U01_003				
HOLD_U01_004				
HOLD_U01_005				
HOLD_U01_006				
HOLD_U01_007				
HOLD_U01_008				
HOLD_U01_009				
HOLD_U01_010				
HOLD_U01_011				
HOLD_U01_012				
HOLD_U02_001				
HOLD_U02_002				
HOLD_U02_003				
HOLD_U02_004				
HOLD_U02_005				
HOLD_U02_006				
HOLD_U02_007				
HOLD_U02_008				
HOLD_U02_009				
HOLD_U02_010				
HOLD_U02_011				
HOLD_U02_012				
HOLD_U03_001				
HOLD_U03_002				
HOLD_U03_003				
HOLD_U03_004				
HOLD_U04_001				
HOLD_U04_002				
HOLD_U04_003				
HOLD_U04_004				

(continued)

ATS reference	Selected? (Y/N)	Run? (Y/N)	Verdict	Observations
HOLD_U05_001				
HOLD_U05_002				
HOLD_U05_003				
HOLD_U05_004				
HOLD_U05_005				
HOLD_U05_006				
HOLD_U05_007				
HOLD_U05_008				
HOLD_U05_009				
HOLD_U05_010				
HOLD_U05_011				
HOLD_U05_012				
HOLD_U06_001				
HOLD_U06_002				
HOLD_U06_003				
HOLD_U06_004				
HOLD_U06_005				
HOLD_U06_006				
HOLD_U06_007				
HOLD_U06_008				
HOLD_U06_009				
HOLD_U06_010				
HOLD_U07_001				
HOLD_U07_002				
HOLD_U08_001				
HOLD_U08_002				
HOLD_U08_003				
HOLD_U08_004				
HOLD_U08_005				
HOLD_U08_006				
HOLD_U08_007				
HOLD_U08_008				
HOLD_U08_009				
HOLD_U08_010				
HOLD_U09_001				
HOLD_U09_002				
HOLD_U09_003				
HOLD_U09_004				

## A.7 Observations

*Additional information relevant to the technical content of the PCTR are given here.*

.....

.....

.....

.....

.....

.....

.....

.....

**Annex B (normative):      Partial PIXIT proforma**

Notwithstanding the provisions of the copyright clause related to the text of this ETS, ETSI grants that users of this ETS may freely reproduce the partial PIXIT proforma in this annex so that it can be used for its intended purposes and may further publish the completed PIXIT.

**B.1 Identification summary**

PIXIT number:

.....

Test laboratory name:

.....

Date of issue:

.....

Issued to:

.....

**B.2 Abstract test suite summary**

Protocol specification:                      ETS 300 141-1

ATS specification:                            ETS 300 141-4

Abstract test method:                      Remote test method (see ISO/IEC 9646-2)

**B.3 Test laboratory**

Test laboratory identification:

.....

Accreditation status of the test service:

.....

Accreditation reference:

.....

Test laboratory manager:

.....

Test laboratory contact:

.....

Means of Testing:

.....

Test laboratory instructions for completion:

.....

**B.4 Client (of the test laboratory)**

Client identification:

.....

Client test manager:

.....

Client contact:

.....

Test facilities required:

.....

**B.5 SUT**

Name:

.....

Version:

.....

SCS reference:

.....

Machine configuration:

.....

Operating system identification:

.....

IUT identification:

.....

PICS (all layers):

.....

.....

Limitations of the SUT:

.....

Environmental conditions:

.....

## B.6 Protocol information

### B.6.1 Protocol identification

Specification reference: ETS 300 141-1

Protocol version:

PICS reference:

NOTE: The PICS reference should reference a completed PICS which is conformant with the PICS proforma contained in ETS 300 141-2.

### B.6.2 Parameter values

Table B.1: Parameter values

Item	Question	Supported? (Y/N)	Allowed values	Value
1.1	Does the IUT support Basic Access?		N/A	N/A
1.2	What length of Call Reference is used?		1, 2	

### B.6.3 Sending of messages by IUT

Table B.2: Actions required to stimulate IUT to send messages

Item	Action: What actions, if possible, have to be taken to cause the IUT to send a..	Supported? (Y/N)	Stimulus (action taken)
2.1	HOLD message?		
2.2	RETRIEVE message?		

Table B.3: Actions required to stimulate IUT to send messages for explicit reservation control

Item	Action: What actions, if possible, have to be taken to cause the IUT, in state ...; to send i) message ... with (component ...); followed by <back to i) if other messages>	Supported? (Y/N)	Stimulus (action taken)
2.3	State U04; HOLD, FACILITY or INFORMATION (ERCC_I)		
2.4	State U10; HOLD, FACILITY or INFORMATION (ERCC_I)		
2.5	State U04; HOLD (no ECR); RETRIEVE, FACILITY or INFORMATION (ERCC_I)		
2.6	State U10; HOLD (no ECR); RETRIEVE, FACILITY or INFORMATION (ERCC_I)		
2.7	State U04; HOLD (ERCC_I,RI); RETRIEVE (ERM_I,RI)		

(continued)



**Table B.3 (continued): Actions required to stimulate IUT to send messages for explicit reservation control**

Item	<b>Action:</b> <b>What actions, if possible, have to be taken to cause the IUT, in state ...; to send i) message ... with (component ...); followed by &lt;back to i) if other messages&gt;</b>	<b>Supported?</b> <b>(Y/N)</b>	<b>Stimulus (action taken)</b>
2.8	State U04; HOLD (ERCC_I,no RI); RETRIEVE (ERM_I,no RI)		
2.9	State U10; HOLD (ERCC_I,RI); RETRIEVE (ERM_I,RI)		
2.10	State U10; HOLD (ERCC_I,no RI); RETRIEVE (ERM_I,no RI)		
2.11	State U04; HOLD (ERCC_I, RI); RETRIEVE (no ECR); FACILITY or INFORMATION (ERC_I, RI)		
2.12	State U04; HOLD (ERCC_I, no RI); RETRIEVE (no ECR); FACILITY or INFORMATION (ERC_I, no RI)		
2.13	State U10; HOLD (ERCC_I, RI); RETRIEVE (no ECR); FACILITY or INFORMATION (ERC_I, RI)		
2.14	State U10; HOLD (ERCC_I, no RI); RETRIEVE (no ECR); FACILITY or INFORMATION (ERC_I, no RI)		
2.15	State U04; HOLD (ERCC_I, RI); RETRIEVE (no ECR); HOLD (ERC_I, RI)		
2.16	State U04; HOLD (ERCC_I, no RI); RETRIEVE (no ECR); HOLD (ERC_I, no RI)		
2.17	State U10; HOLD (ERCC_I, RI); RETRIEVE (no ECR); HOLD (ERC_I, RI)		
2.18	State U10; HOLD (ERCC_I, no RI); RETRIEVE (no ECR); HOLD (ERC_I, no RI)		
2.19	State U04; HOLD (ERCC_I, RI); RETRIEVE, FACILITY or INFORMATION (ERC_I,RI)		
2.20	State U04; HOLD (ERCC_I, no RI); RETRIEVE, FACILITY or INFORMATION (ERC_I, no RI)		

(continued)

**Table B.3 (concluded): Actions required to stimulate IUT to send messages for explicit reservation control**

Item	Action: What actions, if possible, have to be taken to cause the IUT, in state ...; to send i) message ... with (component ...); followed by <back to i) if other messages>	Supported? (Y/N)	Stimulus (action taken)
2.21	State U10; HOLD (ERCC_I, RI); RETRIEVE, FACILITY or INFORMATION (ERC_I,RI)		
2.22	State U10; HOLD (ERCC_I, no RI); RETRIEVE, FACILITY or INFORMATION (ERC_I, no RI)		
2.23	State U04; HOLD (ERCC_I)		
2.24	State U10; HOLD (ERCC_I)		
2.25	State U04; HOLD; RETRIEVE (ERCC_I)		
2.26	State U10; HOLD; RETRIEVE (ERCC_I)		
2.27	State U04; HOLD (ERCC_I); RETRIEVE (ERM_I)		
2.28	State U10; HOLD (ERCC_I); RETRIEVE (ERM_I)		
2.29	State U04; HOLD (ERCC_I); RETRIEVE (no ECR); HOLD (ERC_I)		
2.30	State U10; HOLD (ERCC_I); RETRIEVE (no ECR); HOLD (ERC_I)		
2.31	State U04; HOLD (ERCC_I); RETRIEVE (ERC_I)		
2.32	State U10; HOLD (ERCC_I); RETRIEVE (ERC_I)		
Abbreviations: ERCC_I     ExplicitReservationCreationControl Invoke component ERM_I     ExplicitReservationManagement Invoke component ERC_I     ExplicitReservationCancel Invoke component RI         Reservation Indicator requested (for ERCC_I) or present no ECR     No Explicit Reservation component			

**B.6.4 Configuration of IUT**

**Table B.4: Actions required to configure the IUT**

<b>Item</b>	<b>Action: What actions, if possible, have to be taken to configure the IUT for..</b>	<b>Supported? (Y/N)</b>	<b>Stimulus (action taken)</b>
3.1	Explicit Channel Reservation using reservation indicator?		
3.2	Explicit Channel Reservation without using reservation indicator?		

**B.6.5 Parameter values - information element codings**

**Table B.5: Codings of information elements**

<b>Item</b>	<b>Information element: provide, if possible, ...</b>	<b>Supported? (Y/N)</b>	<b>Value</b>
4.1	coding of Facility IE which does not cause any response from IUT.		

## **Annex C (normative):      Abstract Test Suite (ATS)**

This ATS has been produced using the Tree and Tabular Combined Notation (TTCN) according to ISO/IEC 9646-3 [8].

The ATS was developed on a separate TTCN software tool and therefore the TTCN tables are not completely referenced in the contents table. The ATS itself contains a test suite overview part which provides additional information and references (see also annex D).

### **C.1    The TTCN Graphical form (TTCN.GR)**

The TTCN.GR representation of this ATS is contained in a Postscript file (DEP01414.PS) which can be found on the diskette which is attached to the last page of this ETS.

### **C.2    The TTCN Machine Processable form (TTCN.MP)**

The TTCN.MP representation corresponding to this ATS is contained in an ASCII file (DEP01414.MP) which can be found on the diskette which is attached to the last page of this ETS.

NOTE:      According to ISO/IEC 9646-3 [8], in case of a conflict in interpretation of the operational semantics of TTCN.GR and TTCN.MP, the operational semantics of the TTCN.GR representation takes precedence.

## **Annex D (informative): General structure of ATS**

This annex gives a simple listing of the order of types of tables which appear in a typical supplementary service ATS. This is intended as an aid in helping readers find particular sections quickly.

### **Test Suite Overview**

Test Suite Structure

Test Case Index

Test Step Index

Default Index

### **Declarations Part**

Simple Type Definitions

Structured Type Definitions

ASN.1 Type Definitions

Test Suite Operation Definitions

Test Suite Parameter Declarations

Test Case Selection Expression Definitions

Test Suite Constant Declarations

Test Case Variable Declarations

PCO Declarations

Co-ordination Point Declarations

Timer Declarations

Test Component Declarations

Test Components Configuration Declarations

TTCN ASP Type Definition

TTCN PDU Type Definition

TTCN CM Type Definition

Alias Definitions

### **Constraints Part**

Structured Type Constraint Declarations

ASN.1 Type Constraint Declarations

TTCN ASP Constraint Declarations

TTCN PDU Constraint Declarations

TTCN CM Constraint Declarations

### **Dynamic Part**

Test Case Dynamic Behaviour

Test Step Dynamic Behaviour

Default Dynamic Behaviour

## History

Document history	
November 1995	Public Enquiry PE 95: 1995-11-06 to 1996-03-01
June 1996	Converted into Adobe Acrobat Portable Document Format (PDF)