SLOVENSKI STANDARD
SIST-TP ETSI/ETR 060 E2:2005

01-april-2005

Signalizacijski protokoli in komutacija (SPS) - Navodila za uporabo zapisa abstraktne skladnje št. 1 (ASN.1) v telekomunikacijsko aplikacijskih protokolih

Signalling Protocols and Switching (SPS); Guidelines for using Abstract Syntax Notation One (ASN.1) in telecommunication application protocols

Ta slovenski standard je istoveten z: **ETR 060 Edition 2**

ICS:

33.040.30	Komutacijski in signalizacijski sistem	Switching and signalling systems

SIST-TP ETSI/ETR 060 E2:2005 **en**

ETSI TECHNICAL REPORT

ETR 060

September 1995

Second Edition

Source: ETSI TC-SPS

Reference: RTR/SPS-02015

ICS: 35.100.60

Key words: ASN.1

Signalling Protocols and Switching (SPS); Guidelines for using Abstract Syntax Notation One (ASN.1) in telecommunication application protocols

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

Contents

Foreword 5

1 Scope 7

2 References 8

3 Abbreviations 9

4 Overview of ASN.1 9

5 Specification of protocol data units 10
5.1 Modules 10
5.2 Tagging 11
5.3 Handling of optional and default elements 12
5.4 Subtyping 13
5.5 Importing and exporting data types 14
5.5.1 Exporting 14
5.5.2 Importing 14
5.6 Comments and user-defined constraints 15
5.7 Information elements dependencies 17
5.8 Miscellaneous 17
5.8.1 Elements and types 17
5.8.2 Order of elements 18
5.8.3 Specification of nested structures 19
5.8.4 Enumerated types 19
5.8.5 Specification of operations and errors 20

6 Leaving holes in specifications 20
6.1 General aspects 20
6.2 Embedding information 20
6.3 Defining generic types 22

7 Protocol modifications 23
7.1 Changes to abstract-syntaxes descriptions 23
7.1.1 Non compatible changes 23
7.1.2 Changes without impact on the abstract-syntax 24
7.1.3 Extension of an abstract syntax 24
7.1.4 Private extensions 26
7.2 Impact on the transfer-syntax 26
7.2.1 Non compatible changes 26
7.2.2 Changes without impact on transfer-syntaxes 26
7.2.3 Extension of a transfer-syntax 27

8 Compatibility issues 27
8.1 Backward compatibility 27
8.2 Forward compatibility 28

9 Changing names of information objects 29
9.1 Module names 29
9.2 Abstract syntax names 30
9.3 Application context names 30

Annex A: Migration from 1988/1990 notation to 1994 notation 31

Annex B: Specific guidance for users of the 1988/1990 notation 32

B.1 Use of identifiers 32

B.2 Choice and Any values 32

B.3 Tagging 33

B.4 Operations and Errors 36

History 38

Foreword

This ETSI Technical Report (ETR) has been produced by the Signalling Protocols and Switching (SPS) Technical Committee of the European Telecommunications Standards Institute (ETSI).

ETRs are informative documents resulting from ETSI studies which are not appropriate for European Telecommunication Standard (ETS) or Interim European Telecommunication Standard (I-ETS) status. An ETR may be used to publish material which is either of an informative nature, relating to the use or the application of ETSs or I-ETSs, or which is immature and not yet suitable for formal adoption as an ETS or an I-ETS.

This second edition of ETR 060 takes into account the further evolution of ASN.1 since the publication of the first edition in 1992.

Blank page

1 Scope

The purpose of this ETSI Technical Report (ETR) is to provide guidelines on the use of Abstract Syntax Notation One (ASN.1) for specifying telecommunication application protocols.

This ETR is based on ITU-T Recommendations X.680 [1], X.681 [2], X.682 [3] and X.683 [4] which specify the Abstract Syntax Notation One (ASN.1). In case of misalignment, these Recommendations shall be considered as the primary reference.

Unless explicitly indicated, all references to encoding and decoding functions assume the use of the Basic Encoding Rules (BER) or any of their variants as they are specified in ITU-T Recommendation X.690 [5][1].

This ETR is not a tutorial on ASN.1. Tutorial matter exists on this subject, e.g. "A tutorial on Abstract Syntax Notation One" [17], "ASN.1 MACRO Facility" [18], "ASN.1 and ROS" [19], "An overview of ASN.1" [20]. More specific tutorial information on the latest extensions to ASN.1 can be found in "An introduction to the ASN.1 MACRO replacement notation" [21] and "Efficient encoding rules for ASN.1-based protocols" [22].

Annex F of ITU-T Recommendation X.680 [1] also provides a set of general guidelines for use of the notation.

Throughout this ETR, the term "user" denotes a person who employs ASN.1 for protocol design. The term *1988/90 notation* is used to refer to that ASN.1 notation specified in CCITT Recommendation X.208 (1988) | ISO/IEC 8824:1990 [9]. The term *current notation* is used to refer to that specified in ITU-T Recommendation X.680 [1].

Unless explicitly stated, all the guidelines contained in the body of this ETR are also applicable to users of the 1988/90 notation.

Annex A provides guidance for the migration from the 1988/90 notation to the current notation.

Annex B provides specific guidance which only applies to superseded features of the 1988/90 notation.

Terms between quotation marks refer directly to items or productions defined by the ASN.1 standard (e.g. "typereference", "Symbol").

The main objectives of the recommendations made in this ETR are:

a) allow the re-use of data types from one domain to another;

b) ease protocol evolution, taking into account compatibility issues;

c) ease the maintainability of the specifications;

d) ease automated implementation of encoding and decoding functions;

e) ease the production of test specifications, especially when specified using the Tree and Tabular Combined Notation (see ITU-T Recommendation X.292 [11]) which makes a direct use of the ASN.1 type definitions of the protocol to be tested.

---

1 ITU-T Recommendation X.690 supersedes CCITT Recommendation X.209 [10].

2 References

This ETR incorporates by dated and undated reference, provisions from other publications. These 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 ETR only when incorporated in it by amendment or revision. For undated references the latest edition of the publication referred to applies.

[1] ITU-T Recommendation X.680 (1994): "Specification of abstract syntax notation one (ASN.1): Specification of the basic notation" (also published as ISO/IEC 8824-1).

[2] ITU-T Recommendation X.681 (1994): "Abstract Syntax Notation One (ASN.1): Information Object Specification" (also published as ISO/IEC 8824-2).

[3] ITU-T Recommendation X.682 (1994): "Abstract Syntax Notation One (ASN.1): Constraint Specification" (also published as ISO/IEC 8824-3).

[4] ITU-T Recommendation X.683 (1994): "Abstract Syntax Notation One (ASN.1): Parameterisation of ASN.1 specifications" (also published as ISO/IEC 8824-4).

[5] ITU-T Recommendation X.690 (1994): "Specification of ASN.1 encoding rules: basic encoding rules" (also published as ISO/IEC 8825-1).

[6] ITU-T Recommendation X.691 (1994): "Abstract Syntax Notation One (ASN.1): Packed Encoding Rules" (also published as ISO/IEC 8825-2).

[7] ITU-T Recommendation X.680 (1994): "Specification of abstract syntax notation one (ASN.1): Specification of the basic notation - Amendment 1: Rules for extensibility".

[8] ITU-T Recommendation X.681 (1994): "Abstract Syntax Notation One (ASN.1): Information Object Specification - Amendment 1: Rules for extensibility".

[9] CCITT Recommendation X.208 (1988): "Specification of abstract syntax notation one (ASN.1)" (also published as ISO/IEC 8824:1990).

[10] CCITT Recommendation X.209 (1988): "Specification of basic encoding rules for abstract syntax notation one (ASN.1)".

[11] ITU-T Recommendation X.292 (1993): "OSI Conformance Testing Methodology and Framework: Tree and Tabular Combined Notation (TTCN)" (also published as ISO/IEC 9646-3).

[12] CCITT Recommendation X.219 (1988): "Remote operations; model, notation and service definition".

[13] ITU-T Recommendations Q.771 to Q.775 (1993): "Specifications of Signalling System No 7: Transaction Capabilities (TC)".

[14] CCITT Recommendations Q.771 to Q.775 (1988): "Specifications of Signalling System No. 7, Transaction Capabilities Application Part (TCAP)".

[15] ETS 300 351 (1994): "ETSI object identifier tree; Rules and registration procedures".

[16] ITU-T Recommendation X.880 (1995): "Remote Operations: concepts, model and notation".

[17] "A tutorial on Abstract Syntax Notation One" - David Chappel, Cray Research Inc. - OMNICOM Open System Data Transfer, Trans #25, December 1986.

[18] "ASN.1 MACRO Facility" - Jim Reinstedler, Ungermann-Bass Inc. - OMNICOM Open System Data Transfer, Trans #33, April 1988.

[19] "ASN.1 and ROS: The impact of X400 on OSI" - James E. White - IEEE Journal on Selected Areas In Communications, Vol.7, No.7 - September 1989.

[20] "An overview of ASN.1" - Gerald Neufeld, Son Vuang - Computers and ISDN Systems - No.23 (1992).

[21] "An introduction to the ASN.1 MACRO replacement notation" - Nilo Mitra - AT&T Technical Journal - Vol.73 - No.3 - May/June 1994.

[22] "Efficient encoding rules for ASN.1-based protocols"- Nilo Mitra - AT&T Technical Journal - Vol.73 - No.3 - May/June 1994.

3 Abbreviations

For the purposes of this ETR, the following abbreviations apply

AC	Application Context
APDU	Application Protocol Data Unit
ASE	Application Service Element
ASN.1	Abstract Syntax Notation One
BER	Basic Encoding Rules
DSS1	Digital Subscriber Signalling Number one
MAP	Mobile Application Part
PDU	Protocol Data Unit
PICS	Protocol Implementation Conformance Statement
PER	Packed Encoding Rules
ROSE	Remote Operation Service Element
SS7	Signalling System No.7
TC	Transaction Capabilities
TTCN	Tree and Tabular Combined Notation

4 Overview of ASN.1

Signalling messages are often described using a tabular notation; their format and binary representation is specified using tables whose entries are the information elements from which they are built. This method is rather convenient when the message structure is simple and when there is no need to consider different encoding schemes to represent the same information.

The ITU-T Recommendations covering Signalling System No.7 (SS7) and Digital Subscriber Signalling System No. one (DSS1), currently describe most of the Application Protocol Data Units (APDU) in this manner (e.g. Telephone User Part messages, DSS1 "layer 3" messages, etc.). This is also the approach taken in OSI to describe the protocol data units up to layer 5.

However, as far as the signalling information to be exchanged between telecommunication systems becomes more and more complex, the limits of this tabular notation become clear; difficulties for representing structured elements, duplication of definitions due to the mixture between the syntax of an information and the way it is encoded, etc.

For the above reasons, it then becomes necessary to change the description technique of signalling messages. This is achieved using the Abstract Syntax Notation One (ASN.1).

ASN.1 provides a means to describe data types as well as value of these types in an abstract manner. It does this without determining the way instances of these data types are to be represented during transmission.

Since a signalling message, as any protocol data unit, can be represented by a data type (generally a structured one) ASN.1 fulfils very well the requirements for describing complex messages.

Beside the abstraction and the formalism of data descriptions, one of the objectives of ASN.1 is to facilitate the encoding and decoding of values of the types defined using the notation. This is why, unlike the data declaration portion of programming languages, it provides inherently a means for associating identification tags with data types.

ITU-T Recommendation X.680 [1] specifies a number of simple and structured built-in types which allows the user of the notation to define more complex types and associated data values by combining these built-in types. In addition this notation also provides a set of subtype constructors (e.g. value range, size constraint) to define types whose values are only a subset of the values of some other type (the parent type).

Examples of simple built-in types are: boolean type, enumerated type, integer type, octet string type, while examples of built-in structured types are: sequence type, set type, choice type, etc.

Beyond the specification of data units, the latest version of ASN.1 also provides tools for describing other kind of information object classes, relationships between components of a PDU or other kind of constraints, and for parameterizing a specification (see ITU-T Recommendations X.681 [2], X.682 [3], X.683 [4]). Most of these features are intended to serve as a replacement for the MACRO notation and the ANY type defined as part of the 1988/90 notation

Although the term "ASN.1" is still often used to refer both to this notation and the Basic Encoding Rules, new Standards and Recommendations defining signalling protocols should made very clear that the two aspects are distinct (i.e. other encoding rules may be applied to the defined abstract syntax).

While message description is mainly based on the ASN.1 type notation, the ASN.1 value notation is a basis for some implementations and for specifying constraints for test cases written using the TTCN notation (see ITU-T Recommendation X.292 [11]). It is therefore of high importance to define data types in such a way that it is ensured that the resulting value notation is not ambiguous.

5 Specification of protocol data units

5.1 Modules

The following guidelines are appropriate when considering modules:

a) The set of ASN.1 productions which forms a protocol specification shall be organized into one or several ASN.1 modules.

   The criteria for organizing the modules are up to the protocol designer (functional domain, PDU type, etc.). However for maintainability purposes, the number of inter-module dependencies (i.e. number of modules seen from one module, number of symbols exported and imported) shall be limited.

   NOTE: The number of ASN.1 modules involved in the definition of data units of a particular protocol is independent from the number of ASEs in terms of which this protocol is structured. It has also no impact on the number of abstract syntaxes used to represent instances of these data units.

b) Attention should be paid to avoid cross references between modules which make the parsing of complete protocol data units unnecessarily more complex.

c) As stated in ITU-T Recommendation X.680 [1], each ASN.1 module should be given a module identifier. This is used as a formal reference when exporting or importing definitions between modules or when using external references.

   This identifier shall be composed of a "modulereference" (i.e. a name starting with an upper case letter) and optionally a value of type object identifier. Unlike an application-context-name or an abstract-syntax-name, this value is never exchanged between peer protocol machines. However, it is recommended that an object identifier value be always allocated to modules defined in ETSI standards.

   For further guidance on the use of object identifiers see "An overview of ASN.1" [20]. Rules for assigning object identifier values within the scope of ETSI are described in ETS 300 351 [15].

d) It is suggested that modules defined for signalling applications be allocated a "modulereference" of the following form:

<Protocol-Name>-<Qualifier>

e.g., MAP-Operations

Where <Protocol-Name> identifies a set of related application layer signalling protocols (e.g. MAP) and <Qualifier> is a suitable acronym for the contents of this module (e.g. Operations, CommonTypes, etc.).

5.2 Tagging

The following guidelines are appropriate when considering tagging:

a) the AUTOMATIC TAGS construct should always be used when defining a new module;

NOTE: The AUTOMATIC TAGS construct is not available to users of the 1988/90 notation.

EXAMPLE:

```
My-Module DEFINITIONS
AUTOMATIC TAGS
::=

BEGIN

My-Type ::= SEQUENCE {
        a       INTEGER,
        b       INTEGER OPTIONAL,
        c       BOOLEAN OPTIONAL
}

END
```


b) protocol designers which need to modify a module defined using the 1988/90 notation should follow the guidelines provided in annex B. They should not add the AUTOMATIC TAGS construct in the module header;

c) protocol designers should avoid to add new definitions in modules where the AUTOMATIC TAGS construct was not used. They should preferably create a new module for that purpose;

d) the protocol designer shall be aware that automatic tagging places restrictions on the possible modifications to a type definition when backward compatibility need to be ensured. In addition to those provided in clause 7, users of automatic tagging shall apply the following rules:

- the order of elements in an existing set- or choice- type shall not be modified in a new version of the specification;

- new elements in a set-, sequence- or choice- type shall be added after existing elements.