INTERNATIONAL STANDARD

ISO/IEC 8825

Second edition
1990-12-15

Information technology — Open Systems Interconnection — Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1)

Technologies de l'information — Interconnexion de systèmes ouverts — Spécification de règles de base pour coder la notation de syntaxe abstraite numéro UNE (ASN.1)

Reference number
ISO/IEC 8825 : 1990 (E)

Contents

Annexes

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

International Standard ISO/IEC 8825 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology.*

This second edition cancels and replaces the first edition (ISO 8825 : 1987), which has been technically revised.

Annexes A, B and C are for information only.

Introduction

ISO/IEC 8824 (Specification of Abstract Syntax Notation One) specifies a notation for the definition of abstract syntaxes, enabling application layer standards to define the types of information they need to transfer using the presentation service. It also specifies a notation for the specification of values of a defined type.

This International Standard defines a set of encoding rules that may be applied to values of types defined using the notation specified in ISO/IEC 8824. Application of these encoding rules produces a transfer syntax for such values. It is implicit in the specification of these encoding rules that they are also to be used for decoding.

There may be more than one set of encoding rules that can be applied to values of types that are defined using the notation of ISO/IEC 8824. This International Standard defines one set of encoding rules, called **basic encoding rules**.

This International Standard is technically aligned with CCITT Recommendation X.209(1988).

Annex A gives examples of the application of the encoding rules. It is not part of this International Standard.

Annex B summarizes the assignment of object identifier values made in this International Standard and is not part of this International Standard.

Annex C gives examples of applying the rules for encoding reals. It is not part of this International Standard.

This page intentionally left blank

Information technology — Open Systems Interconnection — Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1)

1 Scope

This International Standard specifies a set of basic encoding rules that may be used to derive the specification of a transfer syntax for values of types defined using the notation specified in ISO/IEC 8824. These basic encoding rules are also to be applied for decoding such a transfer syntax in order to identify the data values being transferred.

These basic encoding rules are used at the time of communication (by the presentation service provider when required by a presentation context).

2 Normative references

The following standards contain provisions which, through reference in this text, constitute provisions of this International Standard. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this International Standard are encouraged to investigate the possibility of applying the most recent editions of the standards indicated below. Members of IEC and ISO maintain registers of currently valid International Standards

ISO 2022: 1986, *Information processing - ISO 7-bit and 8-bit coded character sets - Code extension techniques.*

ISO 2375: 1985, *Data processing - Procedure for registration of escape sequences.*

ISO 6093: 1985, *Information processing - Representation of numerical values in character strings for information interchange.*

ISO 7498:1984, *Information processing systems - Open Systems Interconnection - Basic Reference Model.*

ISO 8823: 1988, *Information processing systems - Open Systems Interconnection - Connection-oriented presentation protocol specification.*

ISO/IEC 8824:1990, *Information technology — Open Systems Interconnection — Specification of Abstract Syntax Notation One (ASN.1).*

CCITT X.209 (1988), *Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1).*

3 Definitions

For the purposes of this International Standard the definitions of ISO 7498, of ISO/IEC 8824 and the following definitions apply.

3.1 dynamic conformance: A statement of the requirement for an implementation to adhere to the behaviour prescribed by this International Standard in an instance of communication.

3.2 static conformance: A statement of the requirement for support by an implementation of a valid set of features from among those defined by this International Standard.

3.3 data value: Information specified as the value of a type; the type and the value are defined using ASN.1.

3.4 encoding (of a data value): The complete sequence of octets used to represent the data value.

NOTE — Some CCITT Recommendations use the term "data element" for this sequence of octets, but the term is not used in this International Standard, as other International Standards use it to mean "data value".

3.5 identifier octets: Part of a data value encoding

which is used to identify the type of the value.

3.6 length octets: Part of a data value encoding following the identifier octets which is used to determine the end of the encoding.

3.7 end-of-contents octets: Part of a data value encoding, occurring at its end, which is used to determine the end of the encoding.

NOTE — Not all encodings require end-of-contents octets.

3.8 contents octets: That part of a data value encoding which represents a particular value, to distinguish it from other values of the same type.

3.9 primitive encoding: A data value encoding in which the contents octets directly represent the value.

3.10 constructed encoding: A data value encoding in which the contents octets are the complete encoding of one or more other data values.

3.11 sender: An implementation encoding a data value for transfer.

3.12 receiver: An implementation decoding the octets produced by a sender, in order to identify the datavalue which was encoded.

4 Abbreviations and notation

4.1 Abbreviations

ASN.1 Abstract Syntax Notation One

4.2 Notation

4.2.1 This International Standard references the notation defined by ISO/IEC 8824.

4.2.2 This International Standard specifies the value of each octet in an encoding by use of the terms "most significant bit" and "least significant bit".

NOTE — Lower layer specifications use the same notation to define the order of bit transmission on a serial line, or the assignment of bits to parallel channels.

4.2.3 For the purposes of this International Standard only, the bits of an octet are numbered from 8 to 1, where bit 8 is the "most significant bit", and bit 1 is the "least significant bit".

5 Conformance

5.1 Dynamic conformance is specified by clause 6 to clause 23 inclusive.

5.2 Static conformance is specified by those standards which specify the application of these basic encoding rules.

5.3 Alternative encodings are permitted by this International Standard as a sender's option. Conforming receivers shall support all alternatives.

NOTE — Examples of such alternative encodings appear in 6.3.2 b) and table 2.

6 General rules for encoding

6.1 Structure of an encoding

6.1.1 The encoding of a data value shall consist of four components which shall appear in the following order:

a) identifier octets (see 6.2);

b) length octets (see 6.3);

c) contents octets (see 6.4);

d) end-of-contents octets (see 6.5).

6.1.2 The end-of-contents octets shall not be present unless the value of the length octets requires them to be present (see 6.3).

6.1.3 Figure 1 illustrates the structure of an encoding (primitive or constructed). Figure 2 illustrates an alternative constructed encoding.

6.2 Identifier octets

6.2.1 The identifier octets shall encode the ASN.1 tag (class and number) of the type of the data value.

6.2.2 For tags with a number ranging from zero to 30 (inclusive), the identifier octets shall comprise a single octet encoded as follows:

a) bits 8 and 7 shall be encoded to represent the class of the tag as specified in table 1.

b) bit 6 shall be a zero or a one according to the rules of 6.2.5;

c) bits 5 to 1 shall encode the number of the tag as a binary integer with bit 5 as the most significant bit.

6.2.3 Figure 3 illustrates the form of an identifier octet for a type with a tag whose number is in the range zero to 30 (inclusive).

6.2.4 For tags with a number greater than or equal to 31, the identifier shall comprise a leading octet followed by one or more subsequent octets.

6.2.4.1 The leading octet shall be encoded as follows:

a) bits 8 and 7 shall be encoded to represent the class of the tag as listed in table 1;

b) bit 6 shall be a zero or a one according to the rules of 6.2.5;

c) bits 5 to 1 shall be encoded as 11111_2

6.2.4.2 The subsequent octets shall encode the number of the tag as follows:

a) bit 8 of each octet shall be set to one unless it is the last octet of the identifier octets;

b) bits 7 to 1 of the first subsequent octet, followed by bits 7 to 1 of the second subsequent octet, followed in turn by bits 7 to 1 of each further octet, up to and including the last subsequent octet in the identifier octets shall be the encoding of an unsigned binary integer equal to the tag number, with bit 7 of the first subsequent octet as the most significant bit;

c) bits 7 to 1 of the first subsequent octet shall not all be zero.

6.2.4.3 Figure 4 illustrates the form of the identifier octets for a type with a tag whose number is greater than 30.

6.2.5 Bit 6 shall be set to zero if the encoding is primitive, and shall be set to one if the encoding is constructed.

NOTE — Subsequent clauses specify whether the encoding is primitive or constructed for each type.

6.2.6 ISO/IEC 8824 specifies that the tag of a type defined using the "CHOICE" keyword takes the value of the tag of the type from which the chosen data value is taken.

6.2.7 ISO/IEC 8824 specifies that the tag of a type defined using "ANY" is indeterminate. The "ANY" type is subsequently defined to be an ASN.1 type, and the complete encoding is then identical to that of a value of the assigned type (including the identifier octets).

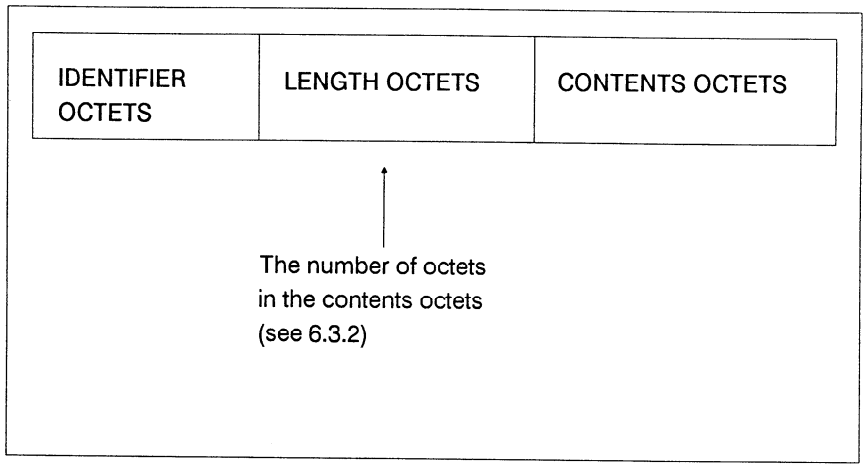

Figure 1 - Structure of an encoding

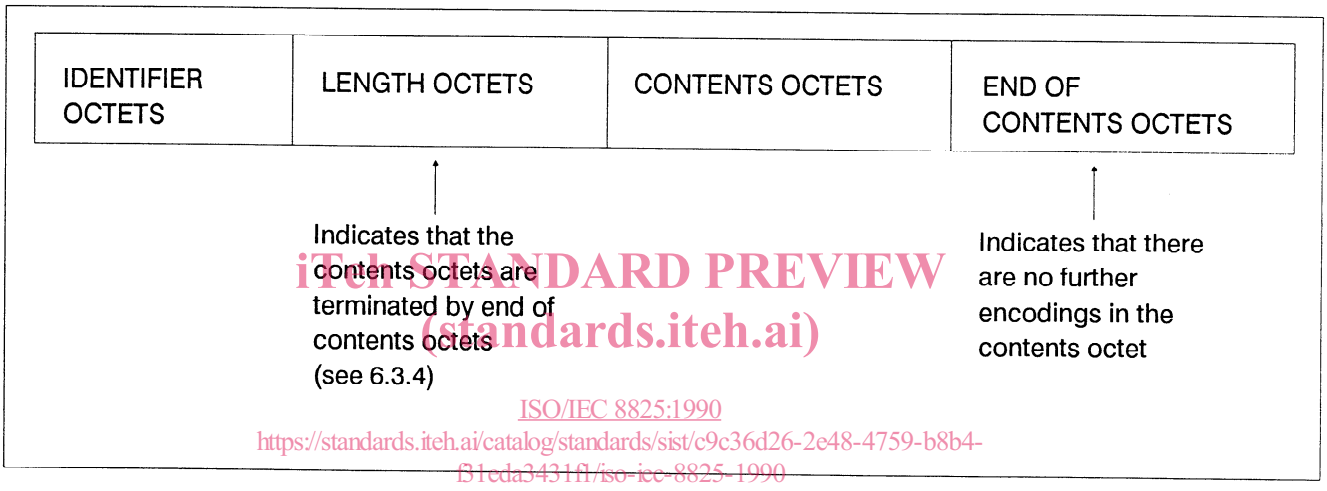

Figure 2 - An alternative constructed encoding

6.3 Length octets

6.3.1 Two forms of length octets are specified. These are

a) the definite form (see 6.3.3); and

b) the indefinite form (see 6.3.4).

6.3.2 A sender shall

a) use the definite form (6.3.3) if the encoding is primitive;

b) use either the definite form (6.3.3) or the indefinite form (6.3.4), a sender's option, if the encoding is constructed and all immediately available;

c) use the indefinite form (6.3.4) if the encoding is constructed and is not all immediately available.

6.3.3 For the definite form, the length octets shall consist of one or more octets, and shall represent the number of octets in the contents octets using either the short form (6.3.3.1) or the long form (6.3.3.2) as a sender's option.

NOTE – The short form can only be used if the number of octets in the contents octets is less than or equal to 127.

Table 1 - Encoding of class of tag

Class	Bit 8	Bit 7
Universal	0	0
Application	0	1
Context-specific	1	0
Private	1	1

6.3.3.1 In the short form, the length octets shall consist of a single octet in which bit 8 is zero and bits 7 to 1 encode the number of octets in the contents octets (which may be zero), as an unsigned binary integer with bit 7 as the most significant bit.

EXAMPLE

L = 38 can be encoded as 00100110_2

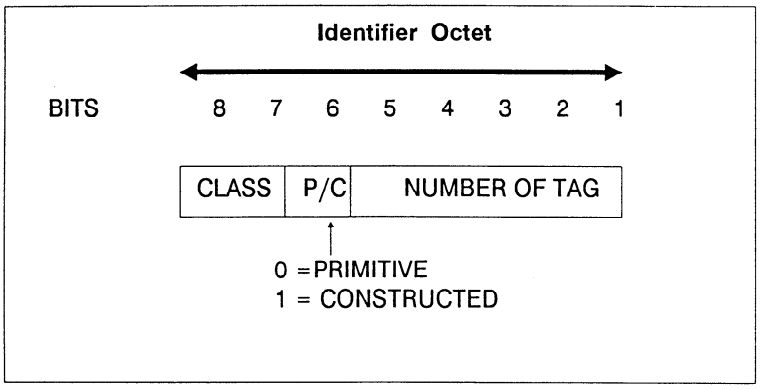

Figure 3 - Identifier octet (low tag number)

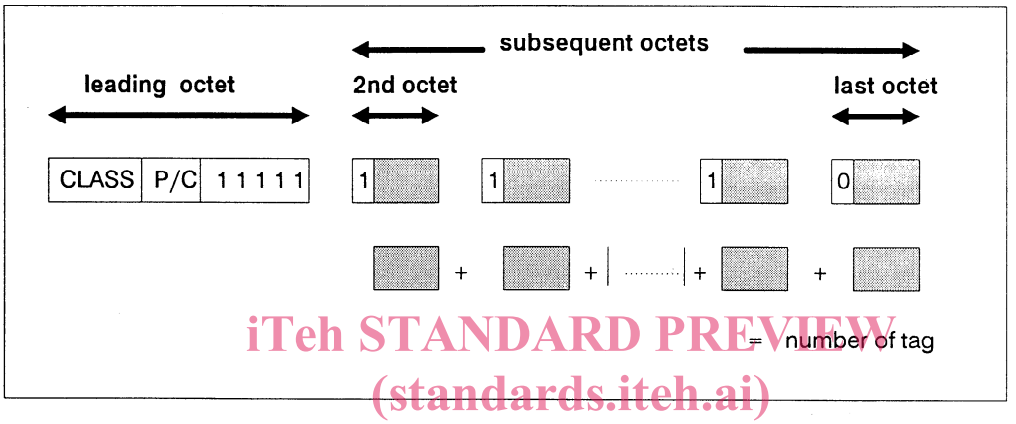

Figure 4 - Identifier octets (high tag number)

6.3.3.2 In the long form, the length octets shall consist of an initial octet and one or more subsequent octets. The initial octet shall be encoded as follows:

a) bit 8 shall be one;

b) bits 7 to 1 shall encode the number of subsequent octets in the length octets, as an unsigned binary integer with bit 7 as the most significant bit;

c) the value 11111111_2 shall not be used

NOTE – This restriction is introduced for possible future extension.

Bits 8 to 1 of the first subsequent octet, followed by bits 8 to 1 of the second subsequent octet, followed in turn by bits 8 to 1 of each further octet up to and including the last subsequent octet, shall be the encoding of an unsigned binary integer equal to the number of octets in the contents octets, with bit 8 of the first subsequent octet as the most significant bit.

EXAMPLE

L = 201 can be encoded as:
10000001_2
11001001_2

NOTE – In the long form, it is a sender's option whether to use more length octets than the minimum necessary.

6.3.4 For the indefinite form, the length octets indicate that the contents octets are terminated by end-of-contents octets (see 6.5), and shall consist of a single octet.

6.3.4.1 The single octet shall have bit 8 set to one, and bits 7 to 1 set to zero.

6.3.4.2 If this form of length is used, then end-of-contents octets (see 6.5) shall be present in the encoding following the contents octets.

6.4 Contents octets

The contents octets shall consist of zero, one or more octets, and shall encode the data value as specified in subsequent clauses.

NOTE – The contents octets depend on the type of the data value; subsequent clauses follow the same sequence as the definition of types in ASN.1.

6.5 End-of-contents octets

The end-of-contents octets shall be present if the length is encoded as specified in 6.3.4, otherwise they shall not be present.

The end-of-contents octets shall consist of two zero octets.

NOTE – The end-of-contents octets can be considered as the encoding of a value whose tag is universal class, whose form is primi-

tive, whose number of the tag is zero, and whose contents is absent, thus:

End-of-contents	Length	Contents
00_{16}	00_{16}	Absent

7 Encoding of a boolean value

7.1 The encoding of a boolean value shall be primitive. The contents octets shall consist of a single octet.

7.2 If the boolean value is

FALSE

the octet shall be zero.

If the boolean value is

TRUE

the octet shall have any non-zero value, as a sender's option.

EXAMPLE - If of type BOOLEAN, the value TRUE can be encoded as:

Boolean	Length	Contents
01_{16}	01_{16}	FF_{16}

8 Encoding of an integer value

8.1 The encoding of an integer value shall be primitive. The contents octets shall consist of one or more octets.

8.2 If the contents octets of an integer value encoding consist of more than one octet, then the bits of the first octet and bit 8 of the second octet

a) shall not all be ones; and

b) shall not all be zero.

NOTE – These rules ensure that an integer value is always encoded in the smallest possible number of octets.

8.3 The contents octets shall be a two's complement binary number equal to the integer value, and consisting of bits 8 to 1 of the first octet, followed by bits 8 to 1 of the second octet, followed by bits 8 to 1 of each octet in turn up to and including the last octet of the contents octets.

NOTE – The value of a two's complement binary number is derived by numbering the bits in the contents octets, starting with bit 1 of the last octet as bit zero and ending the numbering with bit 8 of the first octet. Each bit is assigned a numerical value of 2^N, where N is its position in the above numbering sequence. The value of the two's complement binary number is obtained by summing the numerical values assigned to each bit for those bits which are set to one, excluding bit 8 of the first octet, and then reducing this value by the numerical value assigned to bit 8 of the first octet if that bit is set to one.

9 Encoding of an enumerated value

The encoding of an enumerated value shall be that of the integer value with which it is associated.

NOTE – It is primitive.

10 Encoding of a real value

10.1 The encoding of a real value shall be primitive.

10.2 If the real value is the value zero, there shall be no contents octets in the encoding.

10.3 If the real value is non-zero, then the base used for the encoding shall be B', chosen by the sender. If B' is 2, 8 or 16, a binary encoding, specified in 10.5, shall be used. If B' is 10, a character encoding, specified in 10.6, shall be used.

NOTE – The form of storage, generation, or processing by senders and receivers, and the form used in the ASN.1 value notation are all independent of the base used for transfer.

10.4 Bit 8 of the first contents octet shall be set as follows:

a) if bit 8 = 1, then the binary encoding specified in 10.5 applies;

b) if bit 8 = 0 and bit 7 = 0, then the decimal encoding specified in 10.6 applies;

c) if bit 8 = 0 and bit 7 = 1, then a "SpecialRealValue" (see ISO 8824) is encoded as specified in 10.7.

10.5 When binary encoding is used (bit 8 = 1), then if the mantissa, M is non-zero, it shall be represented by a sign S, a non- negative integer value N and a binary scaling factor F, such that

$$M = S \times N \times 2^F,$$

$$0 \leq F < 4,$$

$$S = +1 \text{ or } -1$$

NOTE – This freedom to choose F is provided to enable easier generation of the transfer format by eliminating the need to align the implied decimal point of the mantissa with an octet boundary (see annex C). The existence of F does not noticeably complicate the task of receivers.

10.5.1 Bit 7 of the first contents octets shall be 1 if S is -1 and 0 otherwise.

10.5.2 Bits 6 to 5 of the first contents octets shall encode the value of the base B' as follows:

Bits 6 to 5	Base
00	base 2
01	base 8
10	base 16
11	Reserved for further editions of this International Standard.

10.5.3 Bits 4 to 3 of the first contents octet shall encode the value of the binary scaling factor F as an unsigned binary integer.

10.5.4 Bits 2 to 1 of the first contents octet shall encode the format of the exponent as follows:

a) if bits 2 to 1 are 00, then the second contents octet encodes the value of the exponent as a two's complement binary number;