INTERNATIONAL STANDARD

ISO/IEC 8824-1

Third edition
2002-12-15

Information technology — Abstract Syntax Notation One (ASN.1): Specification of basic notation

Technologies de l'information — Notation de syntaxe abstraite numéro un (ASN.1): Spécification de la notation de base

Reference number
ISO/IEC 8824-1:2002(E)

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published by ISO in 2003

Published in Switzerland

CONTENTS

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.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. 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.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

ISO/IEC 8824-1 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 6, *Telecommunications and information exchange between systems*, in collaboration with ITU-T. The identical text is published as ITU-T Rec. X.680.

This third edition cancels and replaces the second edition (ISO/IEC 8824-1:1998), which has been technically revised. It also incorporates the Amendments ISO/IEC 8824-1:1998/Amd.1:2000 and ISO/IEC 8824-1:1998/Amd.2:2000 and the Technical Corrigenda ISO/IEC 8824-1:1998/Cor.1:1999, ISO/IEC 8824-1:1998/Cor.2:2002, ISO/IEC 8824-1:1998/Cor.3:2002 and ISO/IEC 8824-1:1998/Cor.4:2002.

ISO/IEC 8824 consists of the following parts, under the general title *Information technology — Abstract Syntax Notation One (ASN.1)*:

— *Part 1: Specification of basic notation*

— *Part 2: Information object specification*

— *Part 3: Constraint specification*

— *Part 4: Parameterization of ASN.1 specifications*

Introduction

This Recommendation | International Standard presents a standard notation for the definition of data types and values. A *data type* (or *type* for short) is a category of information (for example, numeric, textual, still image or video information). A *data value* (or *value* for short) is an instance of such a type. This Recommendation | International Standard defines several basic types and their corresponding values, and rules for combining them into more complex types and values.

In some protocol architectures, each message is specified as the binary value of a sequence of octets. However, standards-writers need to define quite complex data types to carry their messages, without concern for their binary representation. In order to specify these data types, they require a notation that does not necessarily determine the representation of each value. ASN.1 is such a notation. This notation is supplemented by the specification of one or more algorithms called *encoding rules* that determine the value of the octets that carry the application semantics (called the *transfer syntax*). ITU-T Rec. X.690 | ISO/IEC 8825-1, ITU-T Rec. X.691 | ISO/IEC 8825-2 and ITU-T Rec. X.693 | ISO/IEC 8825-4 specify three families of standardized encoding rules, called *Basic Encoding Rules (BER)*, *Packed Encoding Rules (PER)*, and *XML Encoding Rules (XER)*.

Some users wish to redefine their legacy protocols using ASN.1, but cannot use standardized encoding rules because they need to retain their existing binary representations. Other users wish to have more complete control over the exact layout of the bits on the wire (the transfer syntax). These requirements are addressed by ITU-T Rec. X.692 | ISO/IEC 8825-3 which specifies an *Encoding Control Notation (ECN)* for ASN.1. ECN enables designers to formally specify the abstract syntax of a protocol using ASN.1, but to then (if they so wish) take complete or partial control of the bits on the wire by writing an accompanying ECN specification (which may reference standardized Encoding Rules for some parts of the encoding).

A very general technique for defining a complicated type at the abstract level is to define a small number of *simple types* by defining all possible values of the simple types, then combining these simple types in various ways. Some of the ways of defining new types are as follows:

a) given an (ordered) list of existing types, a value can be formed as an (ordered) sequence of values, one from each of the existing types; the collection of all possible values obtained in this way is a new type (if the existing types in the list are all distinct, this mechanism can be extended to allow omission of some values from the list);

b) given an unordered set of (distinct) existing types, a value can be formed as an (unordered) set of values, one from each of the existing types; the collection of all possible unordered sets of values obtained in this way is a new type (the mechanism can again be extended to allow omission of some values);

c) given a single existing type, a value can be formed as an (ordered) list or (unordered) set of zero, one or more values of the existing type; the collection of all possible lists or sets of values obtained in this way is a new type;

d) given a list of (distinct) types, a value can be chosen from any one of them; the set of all possible values obtained in this way is a new type;

e) given a type, a new type can be formed as a subset of it by using some structure or order relationship among the values.

An important aspect of combining types in this way is that encoding rules should recognize the combining constructs, providing unambiguous encodings of the collection of values of the basic types. Thus, every basic type defined using the notation specified in this Recommendation | International Standard is assigned a *tag* to aid in the unambiguous encoding of values.

Tags are mainly intended for machine use, and are not essential for the human notation defined in this Recommendation | International Standard. Where, however, it is necessary to require that certain types be distinct, this is expressed by requiring that they have distinct tags. The allocation of tags is therefore an important part of the use of this notation, but (since 1994) it is possible to specify the automatic allocation of tags.

NOTE 1 – Within this Recommendation | International Standard, tag values are assigned to all simple types and construction mechanisms. The restrictions placed on the use of the notation ensure that tags can be used in transfer for unambiguous identification of values.

An ASN.1 specification will initially be produced with a set of fully defined ASN.1 types. At a later stage, however, it may be necessary to change those types (usually by the addition of extra components in a sequence or set type). If this is to be possible in such a way that implementations using the old type definitions can interwork with implementations using the new type definitions in a defined way, encoding rules need to provide appropriate support. The ASN.1 notation supports the inclusion of an *extension marker* on a number of types. This signals to encoding rules the intention of the designer that this type is one of a series of related types (i.e., versions of the same initial type) called an *extension series*, and that the encoding rules are required to enable information transfer between implementations using different types that are related by being part of the same extension series.

Clauses 10 to 31 (inclusive) define the simple types supported by ASN.1, and specify the notation to be used for referencing simple types and for defining new types using them. Clauses 10 to 31 also specify notations to be used for specifying values of types defined using ASN.1. Two value notations are provided. The first is called the basic ASN.1 value notation, and has been part of the ASN.1 notation since its first introduction. The second is called the XML ASN.1 Value Notation, and provides a value notation using Extensible Markup Language (XML).

NOTE 2 – The XML Value Notation provides a means of representing ASN.1 values using XML. Thus, an ASN.1 type definition also specifies the structure and content of an XML element. This makes ASN.1 a simple schema language for XML.

Clauses 33 to 34 (inclusive) define the types supported by ASN.1 for carrying within them the complete encoding of ASN.1 types.

Clauses 35 to 40 (inclusive) define the character string types.

Clauses 41 to 44 (inclusive) define certain types which are considered to be of general utility, but which require no additional encoding rules.

Clauses 45 to 47 (inclusive) define a notation which enables subtypes to be defined from the values of a parent type.

Clause 48 defines a notation which allows ASN.1 types specified in a "version 1" specification to be identified as likely to be extended in "version 2", and for additions made in subsequent versions to be separately listed and identified with their version number.

Clause 49 defines a notation which allows ASN.1 type definitions to contain an indication of the intended error handling if encodings are received for values which lie outside those specified in the current standardized definition.

Annex A forms an integral part of this Recommendation | International Standard, and specifies ASN.1 regular expressions.

Annex B forms an integral part of this Recommendation | International Standard, and specifies rules for type and value compatibility.

Annex C forms an integral part of this Recommendation | International Standard, and records object identifier and object descriptor values assigned in the ASN.1 series of Recommendations | International Standards.

Annex D does not form an integral part of this Recommendation | International Standard, and describes the top-level arcs of the registration tree for object identifiers.

Annex E does not form an integral part of this Recommendation | International Standard, and provides examples and hints on the use of the ASN.1 notation.

Annex F does not form an integral part of this Recommendation | International Standard, and provides a tutorial on ASN.1 character strings.

Annex G does not form an integral part of this Recommendation | International Standard, and provides a tutorial on the ASN.1 model of type extension.

Annex H does not form an integral part of this Recommendation | International Standard, and provides a summary of ASN.1 using the notation of clause 5.

INTERNATIONAL STANDARD

ITU-T RECOMMENDATION

Information technology – Abstract Syntax Notation One (ASN.1): Specification of basic notation

1 Scope

This Recommendation | International Standard provides a standard notation called Abstract Syntax Notation One (ASN.1) that is used for the definition of data types, values, and constraints on data types.

This Recommendation | International Standard:

- defines a number of simple types, with their tags, and specifies a notation for referencing these types and for specifying values of these types;
- defines mechanisms for constructing new types from more basic types, and specifies a notation for defining such types and assigning them tags, and for specifying values of these types;
- defines character sets (by reference to other Recommendations and/or International Standards) for use within ASN.1.

The ASN.1 notation can be applied whenever it is necessary to define the abstract syntax of information.

The ASN.1 notation is referenced by other standards which define encoding rules for the ASN.1 types.

2 Normative references

The following Recommendations and International Standards contain provisions which, through reference in this text, constitute provisions of this Recommendation | International Standard. At the time of publication, the editions indicated were valid. All Recommendations and Standards are subject to revision, and parties to agreements based on this Recommendation | International Standard are encouraged to investigate the possibility of applying the most recent edition of the Recommendations and Standards listed below. Members of IEC and ISO maintain registers of currently valid International Standards. The Telecommunication Standardization Bureau of the ITU maintains a list of currently valid ITU-T Recommendations.

2.1 Identical Recommendations | International Standards

- CCITT Recommendation X.660 (1992) | ISO/IEC 9834-1:1993, *Information technology – Open Systems Interconnection – Procedures for the operation of OSI Registration Authorities: General procedures: (plus amendments)*.
- ITU-T Recommendation X.681 (2002) | ISO/IEC 8824-2:2002, *Information technology – Abstract Syntax Notation One (ASN.1): Information object specification*.
- ITU-T Recommendation X.682 (2002) | ISO/IEC 8824-3:2002, *Information technology – Abstract Syntax Notation One (ASN.1): Constraint specification*.
- ITU-T Recommendation X.683 (2002) | ISO/IEC 8824-4:2002, *Information technology – Abstract Syntax Notation One (ASN.1): Parameterization of ASN.1 specifications*.
- ITU-T Recommendation X.690 (2002) | ISO/IEC 8825-1:2002, *Information technology – ASN.1 encoding Rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)*.
- ITU-T Recommendation X.691 (2002) | ISO/IEC 8825-2:2002, *Information technology – ASN.1 encoding rules: Specification of Packed Encoding Rules (PER)*.
- ITU-T Recommendation X.692 (2002) | ISO/IEC 8825-3:2002, *Information technology – ASN.1 encoding rules: Specification of Encoding Control Notation (ECN)*.

– ITU-T Recommendation X.693 (2001) | ISO/IEC 8825-4:2002, *Information technology – ASN.1 encoding rules: XML Encoding Rules (XER)*.

2.2 Additional references

– ITU-R Recommendation TF.460-5 (1997), *Standard-frequency and time-signal emissions.*

– CCITT Recommendation T.100 (1988), *International information exchange for interactive videotex.*

– ITU-T Recommendation T.101 (1994), *International interworking for videotex services.*

– ISO *International Register of Coded Character Sets to be used with Escape Sequences*.

– ISO/IEC 646:1991, *Information technology – ISO 7-bit coded character set for information interchange*.

– ISO/IEC 2022:1994, *Information technology – Character code structure and extension techniques*.

– ISO/IEC 6523:1998, *Data interchange – Structures for the identification of organizations*.

– ISO/IEC 7350:1991, *Information technology – Registration of repertoires of graphic characters from ISO/IEC 10367*.

– ISO 8601:2000, *Data elements and interchange formats – Information interchange – Representation of dates and times*.

– ISO/IEC 10646-1:2000, *Information technology – Universal Multiple-Octet Coded Character Set (UCS) – Part 1: Architecture and Basic Multilingual Plane*.

– The Unicode Standard, Version 3.2.0:2002. The Unicode Consortium. (Reading, MA, Addison-Wesley)

NOTE 1 – The above reference is included because it provides names for control characters.

– W3C XML 1.0:2000, *Extensible Markup Language (XML) 1.0 (Second Edition), W3C Recommendation, Copyright © [6 October 2000] World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University), http://www.w3.org/TR/2000/REC-xml-20001006*.

NOTE 2 – The reference to a document within this Recommendation | International Standard does not give it, as a stand-alone document, the status of a Recommendation or International Standard.

3 Definitions

For the purposes of this Recommendation | International Standard, the following definitions apply.

3.1 Information object specification

This Recommendation | International Standard uses the following terms defined in ITU-T Rec. X.681 | ISO/IEC 8824-2:

a) information object;

b) information object class;

c) information object set;

d) instance-of type;

e) object class field type.

3.2 Constraint specification

This Recommendation | International Standard uses the following terms defined in ITU-T Rec. X.682 | ISO/IEC 8824-3:

a) component relation constraint;

b) table constraint.

3.3 Parameterization of ASN.1 specification

This Recommendation | International Standard uses the following terms defined in ITU-T Rec. X.683 | ISO/IEC 8824-4:

a) parameterized type;

b) parameterized value.

3.4 Structure for identification of organizations

This Recommendation | International Standard uses the following terms defined in ISO/IEC 6523:

a) issuing organization;
b) organization code;
c) International Code Designator.

3.5 Universal Multiple-Octet Coded Character Set (UCS)

This Recommendation | International Standard uses the following terms defined in ISO/IEC 10646-1:

a) Basic Multilingual Plane (BMP);
b) cell;
c) combining character;
d) graphic symbol;
e) group;
f) limited subset;
g) plane;
h) row;
i) selected subset.

3.6 Additional definitions

3.6.1 abstract character: An abstract value which is used for the organization, control or representation of textual data.

NOTE – Annex F provides a more complete description of the term abstract character.

3.6.2 abstract value: A value whose definition is based only on the type used to carry some semantics, independently of how it is represented in any encoding.

NOTE – Examples of abstract values are the values of the integer type, the boolean type, a character string type, or of a type which is a sequence (or a choice) of an integer and a boolean.

3.6.3 ASN.1 character set: The set of characters, specified in clause 10, used in the ASN.1 notation.

3.6.4 ASN.1 specification: A collection of one or more ASN.1 modules.

3.6.5 associated type: A type which is used only for defining the value and subtype notation for a type.

NOTE – Associated types are defined in this Recommendation | International Standard when it is necessary to make it clear that there may be a significant difference between how the type is defined in ASN.1 and how it is encoded. Associated types do not appear in user specifications.

3.6.6 bitstring type: A simple type whose distinguished values are an ordered sequence of zero, one or more bits.

NOTE – Where there is a need to carry embedded encodings of an abstract value, the use of a bitstring (or an octetstring) type without a contents constraint (see ITU-T Rec. X.682 | ISO/IEC 8824-3, clause 11) is deprecated. Otherwise, the use of the embedded-pdv type (see clause 33) provides a more flexible mechanism, allowing the announcement of the abstract syntax and of the encoding of the abstract value that is embedded.

3.6.7 boolean type: A simple type with two distinguished values.

3.6.8 character property: The set of information associated with a cell in a table defining a character repertoire.

NOTE – The information will normally include some or all of the following items:

a) a graphic symbol;
b) a character name;
c) the definition of functions associated with the character when used in particular environments;
d) whether it represents a digit;
e) an associated character differing only in (upper/lower) case.

3.6.9 character abstract syntax: Any abstract syntax whose values are specified as the set of character strings of zero, one or more characters from some specified collection of characters.

3.6.10 character repertoire: The characters in a character set without any implication on how such characters are encoded.

3.6.11 character string types: Simple types whose values are strings of characters from some defined character set.

3.6.12 character transfer syntax: Any transfer syntax for a character abstract syntax.

NOTE – ASN.1 does not support character transfer syntaxes which do not encode all character strings as an integral multiple of 8 bits.

3.6.13 choice types: Types defined by referencing a list of distinct types; each value of the choice type is derived from the value of one of the component types.

3.6.14 component type: One of the types referenced when defining a **CHOICE**, **SET**, **SEQUENCE**, **SET OF**, or **SEQUENCE OF**.

3.6.15 constraint: A notation which can be used in association with a type, to define a subtype of that type.

3.6.16 contents constraint: A constraint on a bit string or octet string type that specifies either that the contents are to be an encoding of a specified ASN.1 type, or that specified procedures are to be used to produce and process the contents.

3.6.17 control characters: Characters appearing in some character repertoires that have been given a name (and perhaps a defined function in relation to certain environments) but which have not been assigned a graphic symbol, and which are not spacing characters.

NOTE – HORIZONTAL TABULATION (9) and LINE FEED (10) are examples of control characters that have been assigned a formatting function in a printing environment. DATA LINK ESCAPE (16) is an example of a control character that has been assigned a function in a communication environment.

3.6.18 Coordinated Universal Time (UTC): The time scale maintained by the Bureau International de l'Heure (International Time Bureau) that forms the basis of a coordinated dissemination of standard frequencies and time signals.

NOTE 1 – The source of this definition is ITU-R Rec. TF.460-5. ITU-R has also defined the acronym for Coordinated Universal Time as UTC.

NOTE 2 – UTC and Greenwich Mean Time (GMT) are two alternative time standards which for most practical purposes determine the same time.

3.6.19 element: A value of a governing type or an information object of a governing information object class, distinguable from all other values of the same type or information objects of the same class, respectively.

3.6.20 element set: A set of elements, all of which are values of a governing type, or information objects of a governing class.

NOTE – Governing class is defined in ITU-T Rec. X.681 | ISO/IEC 8824-2, 3.4.7.

3.6.21 embedded-pdv type: A type whose set of values is formally the union of the sets of values in all possible abstract syntaxes. This type can be used in an ASN.1 specification that wishes to carry in its protocol an abstract value whose type may be defined externally to that ASN.1 specification. It carries an identification of the abstract syntax (the type) of the abstract value being carried, as well as an identification of the encoding rules used to encode that abstract value.

3.6.22 encoding: The bit-pattern resulting from the application of a set of encoding rules to an abstract value.

3.6.23 (ASN.1) encoding rules: Rules which specify the representation during transfer of the values of ASN.1 types. Encoding rules also enable the values to be recovered from the representation, given knowledge of the type.

NOTE – For the purpose of specifying encoding rules, the various referenced type (and value) notations, which can provide alternative notations for built-in types (and values), are not relevant.

3.6.24 enumerated types: Simple types whose values are given distinct identifiers as part of the type notation.

3.6.25 extension addition: One of the added notations in an extension series. For set, sequence and choice types, each extension addition is the addition of either a single extension addition group or a single component type. For enumerated types it is the addition of a single further enumeration. For a constraint it is the addition of (only) one subtype element.

NOTE – Extension additions are both textually ordered (following the extension marker) and logically ordered (having increasing enumeration values, and, in the case of **CHOICE** alternatives, increasing tags).

3.6.26 extension addition group: One or more components of a set, sequence or choice type grouped within version brackets. An extension addition group is used to clearly identify the components of a set, sequence or choice type that were added in a particular version of an ASN.1 module, and can identify that version with a simple integer.

3.6.27 extension addition type: A type contained within an extension addition group or a single component type that is itself an extension addition (in such a case it is not contained within an extension addition group).

3.6.28 extensible constraint: A subtype constraint with an extension marker at the outer level, or that is extensible through the use of set arithmetic with extensible sets of values.

3.6.29 extension insertion point (or insertion point): The location within a type definition where extension additions are inserted. This location is the end of the type notation of the immediately preceding type in the extension series if there is

a single ellipsis in the type definition, or immediately before the second ellipsis if there is an extension marker pair in the definition of the type.

NOTE – There can be at most one insertion point within the components of any choice, sequence, or set type.

3.6.30 extension marker: A syntactic flag (an ellipsis) that is included in all types that form part of an extension series.

3.6.31 extension marker pair: A pair of extension markers between which extension additions are inserted.

3.6.32 extension-related: Two types that have the same extension root, where one was created by adding zero or more extension additions to the other.

3.6.33 extension root: An extensible type that is the first type in an extension series. It carries either the extension marker with no additional notation other than comments and white-space between the extension marker and the matching "}" or ")", or an extension marker pair with no additional notation other than a single comma, comments and white-space between the extension markers.

NOTE – Only an extension root can be the first type in an extension series.

3.6.34 extension series: A series of ASN.1 types which can be ordered in such a way that each successive type in the series is formed by the addition of text at the extension insertion point.

3.6.35 extensible type: A type with an extension marker, or to which an extensible constraint has been applied.

3.6.36 external reference: A type reference, value reference, information object class reference, information object reference, or information object set reference (which may be parameterized), that is defined in some other module than the one in which it is being referenced, and which is being referred to by prefixing the module name to the referenced item.

EXAMPLE – **`ModuleName.TypeReference`**

3.6.37 external type: A type which is a part of an ASN.1 specification that carries a value whose type may be defined externally to that ASN.1 specification. It also carries an identification of the type of the value being carried.

3.6.38 false: One of the distinguished values of the boolean type (see also "true").

3.6.39 governing (type); governor: A type definition or reference which affects the interpretation of a part of the ASN.1 syntax, requiring that part of the ASN.1 syntax to reference values in the governing type.

3.6.40 identical type definitions: Two instances of the ASN.1 "Type" production (see clause 16) are defined as identical type definitions if, after performing the transformations specified in Annex B, they are identical ordered lists of identical lexical items (see clause 11).

3.6.41 integer type: A simple type with distinguished values which are the positive and negative whole numbers, including zero (as a single value).

NOTE – When particular encoding rules limit the range of an integer, such limitations are chosen so as not to affect any user of ASN.1.

3.6.42 lexical item: A named sequence of characters from the ASN.1 character set, specified in clause 11, which is used in forming the ASN.1 notation.

3.6.43 module: One or more instances of the use of the ASN.1 notation for type, value, value set, information object class, information object, and information object set (as well as the parameterized variant of those), encapsulated using the ASN.1 module notation (see clause 12).

NOTE – The terms information object class (etc.) are specified in ITU-T Rec. X.681 | ISO/IEC 8824-2, and parameterization is specified in ITU-T Rec. X.683 | ISO/IEC 8824-4.

3.6.44 null type: A simple type consisting of a single value, also called null.

3.6.45 object: A well-defined piece of information, definition, or specification which requires a name in order to identify its use in an instance of communication.

NOTE – Such an object may be an information object as defined in ITU-T Rec. X.681 | ISO/IEC 8824-2.

3.6.46 object descriptor type: A type whose distinguished values are human-readable text providing a brief description of an object (see 3.6.45).

NOTE – An object descriptor value is usually associated with a single object. Only an object identifier value unambiguously identifies an object.

3.6.47 object identifier: A globally unique value associated with an object to unambiguously identify it.

3.6.48 object identifier type: A simple type whose values are the set of all object identifiers allocated in accordance with the rules of ITU-T Rec. X.660 | ISO/IEC 9834 series.