



**INTERNATIONAL STANDARD ISO/IEC 8824-1:1998  
TECHNICAL CORRIGENDUM 1**

Published 1999-12-15

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION • МЕЖДУНАРОДНАЯ ОРГАНИЗАЦИЯ ПО СТАНДАРТИЗАЦИИ • ORGANISATION INTERNATIONALE DE NORMALISATION  
INTERNATIONAL ELECTROTECHNICAL COMMISSION • МЕЖДУНАРОДНАЯ ЭЛЕКТРОТЕХНИЧЕСКАЯ КОМИССИЯ • COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

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

## TECHNICAL CORRIGENDUM 1

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

RECTIFICATIF TECHNIQUE 1

Technical Corrigendum 1 to International Standard ISO/IEC 8824-1:1998 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 6, *Telecommunications and information exchange between systems*.

IECNORM.COM: Click to view the full PDF of ISO/IEC 8824-1:1998/COR1:1999

Withdrawn

## INTERNATIONAL STANDARD

## ITU-T RECOMMENDATION

INFORMATION TECHNOLOGY – ABSTRACT SYNTAX NOTATION ONE (ASN.1):  
SPECIFICATION OF BASIC NOTATION

## TECHNICAL CORRIGENDUM 1

1) **Subclause 3.8.53**

Change "Recursive definitions:" to "Recursive definition (of a type):"

2) **Subclause 10.1**

Add to the end of the first sentence in 10.1:

In Table 2, characters are identified by the names they are given in ISO/IEC 10646-1.

Replace the contents of Table 2 with:

**A to Z** (LATIN CAPITAL LETTER A to LATIN CAPITAL LETTER Z)

**a to z** (LATIN SMALL LETTER A to LATIN SMALL LETTER Z)

**0 to 9** (DIGIT ZERO to DIGIT 9)

**:** (COLON)

**=** (EQUALS SIGN)

**,** (COMMA)

**{** (LEFT CURLY BRACKET)

**}** (RIGHT CURLY BRACKET)

**<** (LESS-THAN SIGN)

**.** (FULL STOP)

**@** (COMMERCIAL AT)

**(** (LEFT PARENTHESIS)

**)** (RIGHT PARENTHESIS)

**[** (LEFT SQUARE BRACKET)

**]** (RIGHT SQUARE BRACKET)

**-** (HYPHEN-MINUS)

**'** (APOSTROPHE)

**"** (QUOTATION MARK)

**|** (VERTICAL LINE)

**&** (AMPERSAND)

**^** (CIRCUMFLEX ACCENT)

- \* (ASTERISK)
- ; (SEMICOLON)
- ! (EXCLAMATION MARK)

### 3) New subclause 11.1.6

*Create a new subclause as follows:*

**11.1.6** This Recommendation | International Standard uses the terms "newline", "end of line", "white-space". In representing white-space and newline (end of line) in machine-readable specifications, any of the following characters may be used in any combination (characters are named and identified by a decimal value which is the value in the ISO/IEC 646 encoding of the character):

For white-space:

- HORIZONTAL TABULATION (9)
- SPACE (32)
- LINE FEED (10)
- VERTICAL TAB (11)
- FORM FEED (12)
- CARRIAGE RETURN (13)

For newline:

- LINE FEED (10)
- VERTICAL TAB (11)
- FORM FEED (12)
- CARRIAGE RETURN (13)

### 4) Subclause 11.4

*Replace, at the end of the first sentence "in 11.2" by "in 11.3".*

### 5) Subclause 12.17

*Replace the word "exported" by the word "imported".*

### 6) Subclause 19.4

*Replace:*

The value of each new "AdditionalEnumeration"

*by:*

The value of each new "EnumerationItem"

### 7) Subclause 19.5

*Replace:*

When a "NamedNumber" is used in defining an "AdditionalEnumeration"

*by:*

When a "NamedNumber" is used in defining an "EnumerationItem" in the "AdditionalEnumeration"

**8) Subclause 19.6**

Replace:

The value associated with the first "AdditionalEnumeration"

by:

The value associated with the first "EnumerationItem" in the "AdditionalEnumeration"

**9) Subclause 21.15**

Replace the hexadecimal string of the example:

'A8A'H

by:

'A98A'H

**10) Subclauses 24.4 and 26.2**

Add the following to the end of 24.4 and to the end of 26.2, before the Note:

Any constraint applied to the referenced type is ignored by this transformation.

**11) Subclauses 24.5.2 and 24.5.3**

Replace:

"ComponentTypes"

by:

"ComponentType"s

**12) Subclause 24.6**

Replace 24.6 by:

When the third or the fourth alternative of "ComponentTypeLists" is used, all "ComponentType"s in extension additions shall have tags which are distinct from the tags of the textually following "ComponentType"s up to and including the first such "ComponentType" that is not marked OPTIONAL or DEFAULT in the trailing "RootComponentTypeList", if any.

**13) Subclause 24.9, NOTE 1**

Replace:

"TaggedTypes"

by:

"TaggedType"s

**14) Subclause 26.3**

Replace the reference "(see 6.4)" by "(see 8.4)"

**15) Subclause 26.9**

*Replace:*

"NamedValues"

*by:*

"NamedValue"s

**16) Subclause 28.7**

*Replace:*

"NamedTypes"

*by:*

"NamedType"s

**17) New subclause 29.1 bis**

*Create a new subclause as follows:*

**29.1 bis** When "Type" denotes a constrained type, the selection is performed on the parent type, ignoring the constraint.

**18) New subclause 44.2 bis**

*Create a new subclause as follows:*

**44.2 bis** When the "Constraint" notation follows the selection type notation, it applies to the choice type, and not to the type of the selected alternative.

NOTE – In the following example, the constraint (WITH COMPONENTS {..., a ABSENT}) applies to the CHOICE type T, not to the selected SEQUENCE type (see 29.1 bis).

```
T ::= CHOICE {  
  a SEQUENCE {  
    a INTEGER OPTIONAL,  
    b BOOLEAN  
  },  
  b NULL  
}  
  
V ::= a < T (WITH COMPONENTS {..., a ABSENT})
```

**19) Clause 46**

*Add the following paragraph to the head of clause 46 in front of 46.1:*

When performing set arithmetic involving subtype constraints and value sets, only abstract values that are defined by extension roots are used in the set arithmetic. All instances of value notation (including value references) used in these constructs are required to reference an abstract value of the extension root. Unless there is an extension marker at the outermost level of an "ElementSetSpecs", the result of the set arithmetic is not an extensible type.

When performing set arithmetic involving information object sets, all information objects (not only those in the extension roots) are used in the set arithmetic. If any of the information object sets contributing to the set arithmetic are extensible, or if there is an extension marker at the outermost level of an "ElementSetSpecs", the result of the set arithmetic is extensible.