



**NORME INTERNATIONALE ISO/CEI 8824-1:1998**  
**RECTIFICATIF TECHNIQUE 3**

Publié 2002-05-01

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

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

RECTIFICATIF TECHNIQUE 3

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

TECHNICAL CORRIGENDUM 3

**iTeh STANDARD PREVIEW**  
**(standards.iteh.ai)**

Le Rectificatif technique 3 à la Norme internationale ISO/CEI 8824-1:1998 a été élaboré par le comité technique mixte ISO/CEI JTC 1, *Technologies de l'information*, sous-comité SC 6, *Téléinformatique*.

<https://standards.iteh.ai/catalog/standards/sist/c1551094-bcc7-458c-a518-52bad997f73b/iso-iec-8824-1-1998-cor-3-2002>

**iTeh STANDARD PREVIEW**  
**(standards.iteh.ai)**

[ISO/IEC 8824-1:1998/Cor 3:2002](https://standards.iteh.ai/catalog/standards/sist/c155f094-bee7-438c-a5f8-52bad997f73b/iso-iec-8824-1-1998-cor-3-2002)

<https://standards.iteh.ai/catalog/standards/sist/c155f094-bee7-438c-a5f8-52bad997f73b/iso-iec-8824-1-1998-cor-3-2002>

## NORME INTERNATIONALE

## RECOMMANDATION UIT-T

## Technologies de l'information – Notation de syntaxe abstraite numéro un: spécification de la notation de base

### CORRIGENDUM TECHNIQUE 3

#### 1) Introduction

Dans l'Introduction (page vi), remplacer les 2<sup>e</sup> et 3<sup>e</sup> alinéas (commençant respectivement par les mots "Bien que cette notation normalisée" et "Hors du contexte de l'OSI") par le texte ci-après:

Dans certaines architectures de protocole, chaque message est spécifié comme la valeur binaire d'une séquence d'octets. Les rédacteurs de normes ont cependant besoin de définir des types de données vraiment complexes afin d'exprimer leurs messages, quelle que soit leur représentation binaire. Afin de spécifier ces types de données, ils ont besoin d'une notation qui ne détermine pas nécessairement la représentation de chaque valeur, comme la notation de syntaxe abstraite numéro un (ASN.1). Celle-ci est complétée par la spécification d'un ou de plusieurs algorithmes appelés **règles de codage**, qui déterminent la valeur des octets exprimant la sémantique applicative (appelée **syntaxe de transfert**). La Rec. UIT-T X.690 | ISO/CEI 8825-1 et la Rec. UIT-T X.691 | ISO/CEI 8825-2 spécifient deux familles de règles de codage normalisées, appelées **règles de codage de base (BER, basic encoding rules)** et **règles de codage compact (PER, packed encoding rules)**. Certains utilisateurs souhaitent redéfinir leur protocoles existants au moyen de la notation ASN.1 mais ne peuvent pas utiliser les règles de codage normalisées parce qu'ils ont besoin de conserver leurs représentations binaires existantes. D'autres utilisateurs souhaitent avoir une maîtrise plus complète de la représentation exacte des éléments binaires en ligne directe (la syntaxe de transfert). Ces exigences sont prises en compte par la Rec. UIT-T X.692 | ISO/CEI 8825-3 (en attente d'être approuvée fin 2001), qui spécifie une **notation à maîtrise de codage (ECN, encoding control notation)** pour la notation ASN.1. La notation ECN permet aux concepteurs de spécifier formellement la syntaxe abstraite d'un protocole au moyen de la notation ASN.1 mais de prendre ensuite (s'ils le souhaitent) la maîtrise complète ou partielle des éléments binaires en ligne directe en rédigeant une spécification ECN auxiliaire (qui peut faire référence à des règles de codage normalisées pour certaines parties du codage).

La notion d'une distinction entre une valeur abstraite d'une certaine classe (ou type) et les détails d'un codage particulier fait l'objet d'une reconnaissance croissante. Afin d'interpréter correctement la représentation de la configuration binaire d'une valeur, il faut connaître (habituellement d'après le contexte) la classe (le type) de la valeur à représenter ainsi que le mécanisme de codage employé. L'identification d'un type est donc une partie importante de la présente Recommandation | Norme internationale.

#### 2) Paragraphe 2.1

Ajouter ce qui suit à la fin du § 2.1:

- Recommandation UIT-T X.692 (2001) | ISO/CEI 8825-3:2001, *Technologies de l'information – Règles de codage ASN.1: spécification de la notation à maîtrise de codage*.

#### 3) Paragraphe 3.8

Ajouter un nouveau paragraphe 3.8.15 bis comme suit:

**3.8.15 bis contrainte de contenu:** contrainte associée à un type chaîne binaire ou à un type chaîne d'octets, indiquant que le contenu doit correspondre à un codage d'un type ASN.1 spécifié ou que des procédures spécifiées doivent être utilisées pour produire le contenu.

#### 4) Article 4

Dans l'article 4, ajouter la ligne suivante après celle qui définit le code DNIC:

ECN Notation à maîtrise de codage de l'ASN.1 (*encoding control notation of ASN.1*)

#### 5) Nouveau paragraphe 11.8 bis

Ajouter un nouveau paragraphe 11.8 bis comme suit:

##### 11.8 bis Unité lexicale: nombre réel

Nom de l'unité lexicale – realnumber

Un nombre réel (*realnumber*) doit se composer d'un élément entier constitué d'un ou de plusieurs chiffres avec, facultativement, une virgule décimale (.). La virgule décimale peut éventuellement être suivie d'une fraction d'élément constituée d'un ou de plusieurs chiffres. L'élément entier, la virgule décimale ou la fraction d'élément (c'est-à-dire celui de ces trois éléments qui est présent en dernier) peut éventuellement être suivi d'un e ou d'un E ainsi que d'un exposant, le cas échéant signé, constitué d'un ou de plusieurs chiffres. Le premier chiffre de l'élément entier doit être différent de zéro sauf s'il est immédiatement suivi de la virgule décimale. Le premier chiffre de l'exposant doit être différent de zéro sauf s'il est composé d'un seul chiffre.

NOTE – On fait toujours correspondre l'unité lexicale "realnumber" à la valeur du nombre réel en l'interprétant comme étant une notation décimale.

#### 6) Paragraphe 11.18

**ITeh STANDARD PREVIEW**

(standards.iteh.ai)

Ajouter les mots réservés "CONTAINING" et "ENCODED" aussi bien à la liste des mots réservés qu'à la Note 2. Ajouter "PATTERN" à la liste des mots réservés.

[ISO/IEC 8824-1:1998/Cor 3:2002](https://standards.iteh.ai/catalog/standards/sist/c155f094-bee7-438c-a5f8-621ad997f73b/iso-iec-8824-1-1998-cor-3-2002)

[https://standards.iteh.ai/catalog/standards/sist/c155f094-bee7-438c-a5f8-](https://standards.iteh.ai/catalog/standards/sist/c155f094-bee7-438c-a5f8-621ad997f73b/iso-iec-8824-1-1998-cor-3-2002)

[621ad997f73b/iso-iec-8824-1-1998-cor-3-2002](https://standards.iteh.ai/catalog/standards/sist/c155f094-bee7-438c-a5f8-621ad997f73b/iso-iec-8824-1-1998-cor-3-2002)

#### 7) Paragraphe 20.5 – Note 1

Modifier comme suit la Note 1:

NOTE 1 – Les valeurs différentes de zéro qui sont représentées en "base" 2 ou en "base" 10 sont considérées comme des valeurs abstraites distinctes même si elles correspondent à la même valeur de nombre réel. Elles peuvent exprimer des sémantiques d'application différentes.

#### 8) Paragraphe 20.6

Modifier ce paragraphe comme suit:

**20.6** La valeur d'un type réel doit être définie par la notation "RealValue":

**RealValue ::= NumericRealValue | SpecialRealValue**

**NumericRealValue ::=**

**realnumber** |

**"-" realnumber** |

**SequenceValue** -- Valeur du type séquence associé

**SpecialRealValue ::=**

**PLUS-INFINITY | MINUS-INFINITY**

La forme "SequenceValue" de la notation "NumericRealValue" ne doit pas être utilisée pour les valeurs zéro.

**9) Nouveau paragraphe 20.7**

Ajouter un nouveau paragraphe 20.7 comme suit:

**20.7** Lorsque la notation "realnumber" est utilisée, elle désigne la valeur abstraite correspondante en "base" 10. Si le type réel est contraint à la "base" 2, la notation "realnumber" désigne la valeur abstraite en "base" 2 qui correspond soit à la valeur décimale spécifiée par "realnumber" soit à une précision définie localement si une représentation exacte n'est pas possible.

**10) Paragraphe 38.1**

Ajouter "UTF8String" après "UniversalString".

**11) Paragraphe 38.4**

Ajouter "et UTF8String" après "UniversalString".

**12) Paragraphe 48.1**

Ajouter ce qui suit à la fin du module "SubtypeElements":

**PatternConstraint**

A la fin de la ligne "InnerTypeConstraints", ajouter "|".

Ajouter une nouvelle colonne au Tableau 6, intitulée "PatternConstraint", avec une entrée "Oui <sup>a)</sup>" pour la rangée "Types de chaînes de caractères à alphabet restreint" et "Non" pour toutes les autres rangées.

Ajouter "UTF8String" après "VisibleString" dans la note de bas de page <sup>w)</sup> du Tableau 6.

ISO/IEC 8824-1:1998/Cor 3:2002

**13) Paragraphe 48.9**

<https://standards.iteh.ai/catalog/standards/sist/c155f094-bee7-438c-a5f8-52bad997f73b/iso-iec-8824-1-1998-cor-3-2002>

Ajouter un nouveau paragraphe 48.9 comme suit:

**48.9 Contrainte de structure**

**48.9.1** La notation "PatternConstraint" doit être la suivante:

**PatternConstraint ::= PATTERN Value**

**48.9.2** L'élément "Value" doit être une chaîne "cstring" de type "UniversalString" (ou une référence à une telle chaîne de caractères) contenant une expression régulière comme défini dans l'Annexe H. La notation "PatternConstraint" sélectionne les valeurs du type parent qui satisfont à l'expression régulière. La valeur entière doit satisfaire à l'expression régulière entière, c'est-à-dire que "PatternConstraint" ne sélectionne pas les valeurs dont les caractères de début répondent à l'expression régulière (entière) mais qui contient d'autres caractères en fin de séquence.

NOTE – L'élément "Value" est défini formellement comme étant une valeur de type UniversalString; mais les ensembles de valeurs de type UniversalString et UTF8String sont les mêmes (voir § 36.13). Une définition totalement équivalente aurait donc pu consister à énoncer que "Value" est une valeur de type UTF8String.

**14) Annexe C, paragraphe C.2.8**

Modifier les deux identificateurs d'objet dans l'exemple du C.2.8 afin d'utiliser des valeurs assignées sous l'arc "asn1(1) examples(123)":

```
packedBCDStringAbstractSyntaxId OBJECT IDENTIFIER ::=
    { joint-iso-itu-t asn1(1) examples(123) packedBCD(2) charSet(0) }
```

```
packedBCDStringTransferSyntaxId OBJECT IDENTIFIER ::=
    { joint-iso-itu-t asn1(1) examples(123) packedBCD(2) characterTransferSyntax(1) }
```

**15) Annexe F, paragraphe F.3.2.2 de l'Amendement 2 (Modèle sémantique)**

*Ajouter ce qui suit après l'alinéa c):*

- c bis)* Chaque occurrence de la notation "realnumber" doit être transformée en une valeur de séquence (SequenceValue) associée à la "base" 10. Chaque occurrence de la notation "RealValue" associée à la notation "SequenceValue" doit être transformée en la "SequenceValue" de la même "base", de façon que le dernier chiffre de la mantisse soit différent de zéro.

**16) Annexe G**

*Ajouter ce qui suit à la fin du module "SubtypeElements":*

**PatternConstraint**

A la fin de la ligne "InnerTypeConstraints", ajouter "|".

*De même, ajouter ce qui suit à l'Annexe G, après la ligne commençant par "PresenceConstraint ::=":*

**PatternConstraint ::= PATTERN Value**

**17) Nouvelle Annexe H**

*Ajouter ce qui suit en tant que nouvelle Annexe H.*

**iTeh STANDARD PREVIEW  
(standards.iteh.ai)**

[ISO/IEC 8824-1:1998/Cor 3:2002](https://standards.iteh.ai/catalog/standards/sist/c155f094-bee7-438c-a5f8-52bad997f73b/iso-iec-8824-1-1998-cor-3-2002)

<https://standards.iteh.ai/catalog/standards/sist/c155f094-bee7-438c-a5f8-52bad997f73b/iso-iec-8824-1-1998-cor-3-2002>

## Annexe H

## Expressions régulières en notation ASN.1

(Cette annexe fait partie intégrante de la présente Recommandation | Norme internationale)

NOTE – Cette annexe recevra une nouvelle lettre, de façon qu'elle précède toutes les annexes informatives lorsque la présente Recommandation | Norme internationale sera rééditée.

## H.1 Définition

**H.1.1** Une expression régulière est une structure qui décrit un ensemble de chaînes dont le format est conforme à cette structure. Une expression régulière est elle-même une chaîne; elle est construite de façon analogue aux expressions arithmétiques, au moyen de divers opérateurs combinant des expressions plus petites. Les plus petites expressions, constituées (habituellement) de 1 ou 2 caractères, sont des structures génériques qui représentent un jeu de caractères. Les expressions régulières présentées ici sont très semblables à celles des langages d'information en code machine comme PERL et à celles des schémas XML, où l'on peut trouver quelques autres exemples d'utilisation.

**H.1.2** La plupart des caractères, y compris toutes les lettres et tous les chiffres, sont des expressions régulières qui correspondent à elles-mêmes.

## EXEMPLE

L'expression régulière "fred" ne correspond qu'à la chaîne "fred".

**H.1.3** Deux expressions régulières peuvent être concaténées: l'expression régulière résultante correspond à toute chaîne formée par concaténation de deux sous-chaînes qui correspondent respectivement aux sous-expressions concaténées.

**iTeh STANDARD PREVIEW**  
(standards.iteh.ai)

## H.2 Métacaractères

H.2.1 Une séquence de métacaractères (ou un métacaractère) est un ensemble d'un ou de plusieurs caractères continus qui ont une signification spéciale dans le contexte d'une expression régulière. La liste suivante contient toutes les séquences de métacaractères. Leur signification est expliquée dans les paragraphes suivants.

[ ]	Correspond à tout caractère du jeu où les séries sont indiquées par "-". Un "^" après le premier crochet complète le jeu qui le suit.
{g,p,r,c}	Quadruplet qui identifie un caractère de l'ISO/CEI 10646-1 (voir § 36.7)
\N{nom}	Correspond au caractère nommé (ou à tout caractère du jeu de caractères nommé) comme défini au § 37.1
.	Correspond à tout caractère (à moins qu'il ne soit un des caractères d'interligne définis au § 11.1.6)
\d	Correspond à un chiffre quelconque (équivalent à "[0-9]")
\w	Correspond à n'importe quel caractère alphanumérique (équivalent à "[a-zA-Z0-9]")
\t	Correspond au caractère TABULATION HORIZONTALE (9) (voir § 11.1.6)
\n	Correspond à l'un quelconque des caractères d'interligne définis au § 11.1.6
\r	Correspond au caractère RETOUR DE CHARIOT (13) (voir § 11.1.6)
\s	Correspond à l'un quelconque des caractères d'espacement définis au § 11.1.6
\b	Correspond à une limite de mot
\ (préfixe)	Cite le métacaractère suivant et lui attribue une interprétation littérale
\\	Correspond au caractère BARRE OBLIQUE INVERSE (92)
""	Correspond au caractère de double apostrophe (")
(infixe)	Choix entre deux expressions
()	Groupement de l'expression enclose
*	(suffixe) Correspond à l'expression précédente zéro, une ou plusieurs fois
+	(suffixe) Correspond à l'expression précédente une ou plusieurs fois
?	(suffixe) Correspond à l'expression précédente une seule ou aucune fois
#(n)	(suffixe) Correspond à l'expression précédente exactement n fois
#(n,)	(suffixe) Correspond à l'expression précédente au moins n fois
#(n,m)	(suffixe) Correspond à l'expression précédente au moins n mais au plus m fois
#(,m)	(suffixe) Correspond à l'expression précédente au plus m fois

NOTE 1 – Les caractères ACCENT CIRCONFLEXE (94) "^" et TRAIT D'UNION - SIGNE MOINS (45) "-" sont des métacaractères additionnels dans certaines positions de la chaîne définie au § H.2.2.

NOTE 2 – La valeur entre parenthèses après un nom de caractère dans la présente annexe est la valeur décimale du caractère contenu dans l'ISO/CEI 10646.

NOTE 3 – Cette notation ne fournit pas les métacaractères "^" et "\$" correspondant au caractère situé au début ou à la fin, respectivement, d'une chaîne. Une chaîne doit donc correspondre à une expression régulière dans sa totalité sauf si cette expression contient les caractères "."\* à son début, à sa fin ou de part et d'autre.

NOTE 4 – Les séquences de métacaractères suivantes ne peuvent contenir un espace blanc (voir le § 3.8.75) sauf si celui-ci est situé immédiatement avant ou après une fin de ligne:

```
{g,p,r,c}
\N{name}
#n
#(n)
#(n,)
#(n,m)
#(,m)
```

Dans le cas d'une expression régulière occupant plus d'une ligne de texte, un blanc situé immédiatement avant ou après une fin de ligne n'a pas de signification et ne correspond à rien (voir le § 11.11.1).

**H.2.2** Une liste de caractères enclos par "["et"]" correspond à tout caractère isolé de cette liste. Si le premier caractère de la liste est le signe circonflexe "^", il correspond à tout caractère qui n'est pas dans la liste. Une série de caractères peut être spécifiée par l'indication des premier et dernier caractères, séparés par un trait d'union (selon la relation d'ordre définie au § 38.4). Toutes les séquences de métacaractères, sauf "]" et "\", perdent leur signification spéciale dans une liste. Pour inclure un littéral ACCENT CIRCONFLEXE (94) "^", le placer n'importe où sauf dans la première position ou le faire précéder d'une barre oblique inverse. Pour inclure un littéral TRAIT D'UNION - SIGNE MOINS (45) "-", le placer en premier ou en dernier dans la liste, ou le faire précéder d'une barre oblique inverse. Pour inclure un littéral DEUXIÈME CROCHET (93) "]", le placer en premier. Si le premier caractère dans une liste est le signe circonflexe "^" alors les caractères "-" et "]" correspondent à eux-mêmes lorsqu'ils suivent immédiatement le signe circonflexe. Les séquences de métacaractères définies aux § H.2.3, H.2.4, H.2.6 et H.2.7 peuvent être utilisées entre les crochets où elles conservent leur signification.

#### EXEMPLES

L'expression régulière "[0123456789]", ou son équivalent "[0-9]", correspond à tout chiffre isolé.

L'expression régulière "[^0]" correspond à tout caractère isolé sauf 0.

L'expression régulière "[d^.-]" correspond à tout chiffre isolé, un signe circonflexe, un trait d'union ou un point.

**H.2.3** Afin d'éviter toute ambiguïté entre deux caractères ISO/CEI 10646-1 ayant le même glyphe, deux notations sont prévues. Une notation de la forme "{group,plane,row,cell}" renvoie à un caractère (isolé) selon la production "par quadruplet" définie dans les paragraphes 36.9 et 36.10.

**H.2.4** Une notation de la forme "\N{valuereference}" correspond au caractère référencé si "valuereference" est une référence à une chaîne de caractères restreinte dont la longueur a la valeur 1 (voir 36) qui est définie ou importée dans le module actuel. Une notation de la forme "\N{typereference}" correspond à tout caractère du jeu de caractères référencé si "typereference" est une référence à un sous-type d'un type "RestrictedCharacterStringType" qui est défini dans le module actuel, ou d'un type "RestrictedCharacterStringType" défini au § 36.

NOTE – En particulier, "valuereference" ou "typereference" peut être une des références définies dans le module ASN1-CHARACTER-MODULE (voir § 37.1) et importée dans le module actuel (voir § 36.7).

#### EXEMPLES

L'expression régulière "\N{greekCapitalLetterSigma}" correspond à la lettre majuscule grecque SIGMA.

L'expression régulière "\N{BasicLatin}" correspond à tout caractère (isolé) du jeu de caractères LATIN DE BASE.

"[\N{BasicLatin}\N{Cyrillic}\N{BasicGreek}]" et son équivalent "(\N{BasicLatin}\N{Cyrillic}\N{BasicGreek})+" sont des expressions régulières qui correspondent à une chaîne constituée de tout nombre (non nul) de caractères provenant des trois jeux de caractères spécifiés.

**H.2.5** Le point "." correspond à tout caractère isolé, à moins qu'il ne soit un des caractères d'interligne définis au § 11.1.6.

**H.2.6** Le symbole "\d" est un synonyme de "[0-9]", c'est-à-dire qu'il correspond à tout chiffre isolé. Le symbole "\t" correspond au caractère TABULATION HORIZONTALE (9). Le symbole "\w" est un synonyme de "[a-zA-Z0-9]", c'est-à-dire qu'il correspond à tout caractère isolé (minuscule ou majuscule) ou à tout chiffre isolé.

#### EXEMPLE

L'expression régulière "\w+(\s\w+)\*\." correspond à une phrase constituée d'au moins un mot (alphanumérique). Les mots sont séparés par un caractère d'espacement. Il n'y a pas de caractère d'espacement devant le point final.