

JTC1
NORME
INTERNATIONALE

ISO/CEI
8825-2

Première édition
1996-08-01

**Technologies de l'information — Règles de
codage ASN.1: Spécification des règles de
codage compact (PER)**

iTeh STANDARD PREVIEW

*Information technology — ASN.1 encoding rules: Specification of Packed
Encoding Rules (PER)*

ISO/IEC 8825-2:1996

<https://standards.itih.ai/catalog/standards/sist/fd39bd94-9d21-47fc-af50-97c86a800b0b/iso-iec-8825-2-1996>



Numéro de référence
ISO/CEI 8825-2:1996(F)

Sommaire

	<i>Page</i>	
1	Domaine d'application.....	1
2	Références normatives	1
2.1	Recommandations Normes internationales identiques.....	1
2.2	Autres références	2
3	Définitions.....	2
3.1	Définition du service de présentation de base.....	2
3.2	Spécification de la notation de base.....	2
3.3	Extensibilité de la notation ASN.1.....	2
3.4	Spécification des objets informationnels	2
3.5	Spécification des contraintes.....	3
3.6	Spécification du paramétrage en notation ASN.1	3
3.7	Règles de codage de base.....	3
3.8	Autres définitions.....	3
4	Abréviations	6
5	Notation.....	6
6	Conventions.....	6
7	Règles de codage définies dans la présente Recommandation Norme internationale	6
8	Conformité	7
9	Méthode de codage utilisée pour les règles PER 8825-2:1996.....	8
9.1	Utilisation de la notation de types.....	8
9.2	Utilisation d'étiquettes pour établir un ordre canonique	8
9.3	Contraintes visibles par les règles PER.....	8
9.4	Modèle utilisé pour coder les types et les valeurs.....	9
9.5	Structure d'une expression codée	9
9.6	Types à coder	10
10	Procédures de codage.....	11
10.1	Production du codage complet.....	11
10.2	Champs de type ouvert.....	11
10.3	Codage sous forme d'un entier binaire non négatif.....	11
10.4	Codage sous forme d'un entier binaire en complément à deux	12
10.5	Codage d'un nombre entier contraint	12
10.6	Codage d'un nombre entier non négatif normalement petit	13
10.7	Codage d'un nombre entier semi-contraint	13
10.8	Codage d'un nombre entier non contraint	14
10.9	Règles générales pour le codage d'un déterminant de longueur	14

© ISO/CEI 1996

Droits de reproduction réservés. Sauf prescription différente, aucune partie de cette publication ne peut être reproduite ni utilisée sous quelque forme que ce soit et par aucun procédé, électronique ou mécanique, y compris la photocopie et les microfilms, sans l'accord écrit de l'éditeur.

ISO/CEI Copyright Office • Case postale 56 • CH-1211 Genève 20 • Suisse

Version française tirée en 1997

Imprimé en Suisse

11	Codage du type booléen	17
12	Codage du type entier	17
13	Codage du type énuméré	18
14	Codage du type réel	18
15	Codage du type chaîne binaire	19
16	Codage du type chaîne d'octets	20
17	Codage du type néant	20
18	Codage du type séquence	20
19	Codage du type séquence-de	21
20	Codage du type ensemble	22
21	Codage du type ensemble-de	22
22	Codage du type choix	23
23	Codage du type identificateur d'objet	23
24	Codage du type valeur PDV encapsulée	24
25	Codage d'une valeur du type externe	25
26	Codage des types chaîne de caractères restreinte	26
27	Codage du type chaîne de caractères non restreinte	28
28	Identificateurs d'objet pour syntaxes de transfert	29
Annexe A	– Exemples de codages	31
A.1	Enregistrement qui n'utilise pas de contrainte appliquée aux sous-types	31
A.2	Enregistrement utilisant des contraintes appliquées aux sous-types	34
A.3	Enregistrement qui utilise des marqueurs d'extension	36
Annexe B	– Observations sur la combinaison de contraintes visibles par les règles PER	41
Annexe C	– Prise en charge des algorithmes PER	42
Annexe D	– Prise en charge des règles d'extensibilité ASN.1	43
Annexe E	– Complément didactique sur la concaténation de codages conformes aux règles PER	44
Annexe F	– Affectation de valeurs à un identificateur d'objet	45

Avant-propos

L'ISO (Organisation internationale de normalisation) et la CEI (Commission électrotechnique internationale) forment ensemble un système consacré à la normalisation internationale considérée comme un tout. Les organismes nationaux membres de l'ISO ou de la CEI participent au développement de Normes internationales par l'intermédiaire des comités techniques créés par l'organisation concernée afin de s'occuper des différents domaines particuliers de l'activité technique. Les comités techniques de l'ISO et de la CEI collaborent dans des domaines d'intérêt commun. D'autres organisations internationales, gouvernementales ou non gouvernementales, en liaison avec l'ISO et la CEI participent également aux travaux.

Dans le domaine des technologies de l'information, l'ISO et la CEI ont créé un comité technique mixte, l'ISO/CEI JTC 1. Les projets de Normes internationales adoptés par le comité technique mixte sont soumis aux organismes nationaux pour approbation, avant leur acceptation comme Normes internationales. Les Normes internationales sont approuvées conformément aux procédures qui requièrent l'approbation de 75 % au moins des organismes nationaux votants.

La Norme internationale ISO/CEI 8825-2 a été élaborée par le comité technique mixte ISO/CEI JTC 1, *Technologies de l'information*, sous-comité SC 21, *Interconnexion des systèmes ouverts, gestion des données et traitement distribué ouvert*, en collaboration avec l'UIT-T. Le texte identique est publié en tant que Recommandation UIT-T X.691.

L'ISO/CEI 8825 comprend les parties suivantes, présentées sous le titre général *Technologies de l'information - Règles de codage ASN.1*: <https://standards.iso.org/iso/iec/8825-2-1996>

- *Partie 1: Spécification des règles de codage de base (BER), des règles de codage canoniques (CER) et des règles de codage distinctives (DER)*
- *Partie 2: Spécification des règles de codage compact (PER)*

Les annexes A à F de la présente partie de l'ISO/CEI 8825 sont données uniquement à titre d'information.

Introduction

L'ensemble de documents Rec. UIT-T X.680 | ISO/CEI 8824-1, Rec. UIT-T X.681 | ISO/CEI 8824-2, Rec. UIT-T X.682 | ISO/CEI 8824-3, Rec. UIT-T X.683 | ISO/CEI 8824-4, Rec. UIT-T X.680/Amd. 1 | ISO/CEI 8824-1/Amd. 1, Rec. UIT-T X.681/Amd. 1 | ISO/CEI 8824-2/Amd. 1, décrivent la notation de syntaxe abstraite numéro un (ASN.1) qui permet de définir les messages échangés par des applications homologues.

La présente Recommandation | Norme internationale définit les règles de codage qui pourront être appliquées à des valeurs de types définis conformément à la notation spécifiée dans la Rec. UIT-T X.680 | ISO/CEI 8824-1. L'application de ces règles de codage produit une syntaxe de transfert pour de telles valeurs. La spécification de ces règles de codage postule implicitement que ces règles pourront être utilisées telles quelles pour le décodage.

Plusieurs ensembles de règles de codage peuvent être appliqués à des valeurs de types ASN.1. La présente Recommandation | Norme internationale définit un ensemble de règles de codage compact (PER) (*packed encoding rules*), ainsi dénommées parce qu'elles donnent une représentation plus compacte que celle que l'on peut obtenir au moyen des règles de codage de base (BER) et de leurs dérivées, décrites dans la Rec. UIT-T X.690 | ISO/CEI 8825-1, à laquelle font référence certaines parties de la spécification des présentes règles de codage compact.

iTeh STANDARD PREVIEW (standards.iteh.ai)

ISO/IEC 8825-2:1996

<https://standards.iteh.ai/catalog/standards/sist/fd39bd94-9d21-47fc-af50-97c86a800b0b/iso-iec-8825-2-1996>

iTeh STANDARD PREVIEW
(standards.iteh.ai)

ISO/IEC 8825-2:1996

<https://standards.iteh.ai/catalog/standards/sist/fd39bd94-9d21-47fc-af50-97c86a800b0b/iso-iec-8825-2-1996>

NORME INTERNATIONALE

RECOMMANDATION UIT-T

TECHNOLOGIES DE L'INFORMATION – RÈGLES DE CODAGE ASN.1: SPÉCIFICATION DES RÈGLES DE CODAGE COMPACT (PER)

1 Domaine d'application

La présente Recommandation | Norme internationale spécifie un ensemble de règles de codage compact qui peuvent être utilisées pour élaborer une syntaxe de transfert applicable à des valeurs de types définis dans la Rec. UIT-T X.680 | ISO/CEI 8824-1. Ces règles de codage compact sont également applicables au décodage d'une telle syntaxe de transfert afin d'identifier les valeurs de données qui sont transférées.

Les règles de codage spécifiées dans la présente Recommandation | Norme internationale:

- sont utilisées au moment de la communication;
- sont destinées à être utilisées dans des circonstances où la minimisation du volume occupé par la représentation de valeurs est la principale préoccupation lors du choix de règles de codage;
- permettent l'extension d'une syntaxe abstraite par adjonction de valeurs supplémentaires, tout en conservant les codages des valeurs existantes, pour toutes les formes d'extension décrites dans la Rec. UIT-T X.680/Amd. 1 | ISO/CEI 8824-1/Amd. 1.

2 Références normatives

Les Recommandations et les Normes internationales suivantes contiennent des dispositions qui, par suite de la référence qui y est faite, constituent des dispositions valables pour la présente Recommandation | Norme internationale. Au moment de la publication, les éditions indiquées étaient en vigueur. Toutes les Recommandations et Normes sont sujettes à révision et les parties prenantes aux accords fondés sur la présente Recommandation | Norme internationale sont invitées à rechercher la possibilité d'appliquer les éditions les plus récentes des Recommandations et Normes indiquées ci-après. Les membres de la CEI et de l'ISO possèdent le registre des Normes internationales en vigueur. Le Bureau de la normalisation des télécommunications de l'UIT tient à jour une liste des Recommandations de l'UIT-T en vigueur.

2.1 Recommandations | Normes internationales identiques

- Recommandation UIT-T X.200 (1994) | ISO/CEI 7498-1:1994, *Technologies de l'information – Interconnexion des systèmes ouverts – Modèle de référence de base: le modèle de référence de base.*
- Recommandation UIT-T X.216 (1994) | ISO/CEI 8822:1994, *Technologies de l'information – Interconnexion des systèmes ouverts – Définition du service de présentation.*
- Recommandation UIT-T X.226 (1994) | ISO/CEI 8823-1:1994, *Technologies de l'information – Interconnexion des systèmes ouverts – Protocole de présentation en mode connexion: spécification du protocole.*
- Recommandation UIT-T X.680 (1994) | ISO/CEI 8824-1:1995, *Technologies de l'information – Notation de syntaxe abstraite numéro un: spécification de la notation de base.*
- Recommandation UIT-T X.680 (1994)/Amd. 1 (1995) | ISO/CEI 8824-1:1995/Amd. 1:1995, *Technologies de l'information – Notation de syntaxe abstraite numéro un: spécification de la notation de base – Amendement 1: règles d'extensibilité.*
- Recommandation UIT-T X.681 (1994) | ISO/CEI 8824-2:1995, *Technologies de l'information – Notation de syntaxe abstraite numéro un: spécification des objets informationnels.*
- Recommandation UIT-T X.681 (1994)/Amd. 1 (1995) | ISO/CEI 8824-2:1995/Amd. 1:1995, *Technologies de l'information – Notation de syntaxe abstraite numéro un: spécification des objets informationnels – Amendement 1: règles d'extensibilité.*

- Recommandation UIT-T X.682 (1994) | ISO/CEI 8824-3:1995, *Technologies de l'information – Notation de syntaxe abstraite numéro un: spécification des contraintes.*
- Recommandation UIT-T X.683 (1994) | ISO/CEI 8824-4:1995, *Technologies de l'information – Notation de syntaxe abstraite numéro un: paramétrage des spécifications de la notation de syntaxe abstraite numéro un.*
- Recommandation UIT-T X.690 (1994) | ISO/CEI 8825-1:1995, *Technologies de l'information – Règles de codage de la notation de syntaxe abstraite numéro un: spécification des règles de codage de base, des règles de codage canoniques et des règles de codage distinctives.*

2.2 Autres références

- Recommandation X.208 du CCITT (1988), *Spécification de la syntaxe abstraite numéro un (ASN.1).*
- ISO/CEI 2022:1994, *Technologies de l'information – Structure de code de caractères et techniques d'extension.*
- ISO 2375:1985, *Traitement de l'information – Procédure pour l'enregistrement des séquences d'échappement.*
- ISO 6093:1985, *Traitement de l'information – Représentation des valeurs numériques dans les chaînes de caractères pour l'échange d'information.*
- ISO/CEI 8824:1990, *Technologies de l'information – Interconnexion de systèmes ouverts – Spécification de la notation de syntaxe abstraite numéro un (ASN.1).*
- ISO *Registre international des jeux de caractères codés à utiliser avec une séquence d'échappement.*
- ISO/CEI 10646-1:1993, *Technologies de l'information – Jeu universel de caractères codés à plusieurs octets – Partie 1: Architecture et table multilingue.*

iteh STANDARD PREVIEW
(standards.iteh.ai)

3 Définitions

Pour les besoins de la présente Recommandation | Norme internationale, les définitions suivantes s'appliquent.

ISO/IEC 8825-2:1996
<https://standards.iteh.ai/catalog/standards/sis/1fd39bd94-9d21-47fc-af50-97c86a800b0b/iso-iec-8825-2-1996>

3.1 Définition du service de présentation de base

Termes définis dans la Rec. UIT-T X.216 | ISO/CEI 8822:

- ensemble de contextes défini;
- identificateur de contexte de présentation.

3.2 Spécification de la notation de base

Toutes les définitions contenues dans la Rec. UIT-T X.680 | ISO/CEI 8824-1 s'appliquent.

3.3 Extensibilité de la notation ASN.1

Termes définis dans la Rec. UIT-T X.680/Amd. 1 | ISO/CEI 8824-1/Amd. 1:

- marqueur d'extension;
- arc d'extension;
- racine d'extension;
- prolongement d'extension.

3.4 Spécification des objets informationnels

Toutes les définitions figurant dans la Rec. UIT-T X.681 | ISO/CEI 8824-2 s'appliquent.

3.5 Spécification des contraintes

Termes définis dans la Rec. UIT-T X.682 | ISO/CEI 8824-3:

- a) contrainte relationnelle de composante;
- b) contrainte tabulaire.

3.6 Spécification du paramétrage en notation ASN.1

Terme défini dans la Rec. UIT-T X.683 | ISO/CEI 8824-4:

- contrainte variable.

3.7 Règles de codage de base

Termes définis dans la Rec. UIT-T X.690 | ISO/CEI 8825-1:

- a) conformité dynamique;
- b) conformité statique;
- c) valeur de données;
- d) codage (d'une valeur de données);
- e) expéditeur;
- f) destinataire.

3.8 Autres définitions

De plus, les définitions suivantes s'appliquent.

3.8.1 codage d'entier binaire en complément à 2: codage d'un nombre entier sur un champ binaire aligné à l'octet de longueur spécifiée, ou sur le nombre minimal d'octets permettant de représenter cet entier (égal, supérieur ou inférieur à zéro comme spécifié au 10.4) sous forme d'un entier en complément à deux.

NOTES

1 La représentation d'un nombre binaire en complément à deux est obtenue en numérotant les bits des octets qui le composent, en commençant par le bit 1 du dernier octet qui devient le bit 0 et en terminant par le bit 8 du premier octet. A chaque bit est affectée une valeur numérique de 2^N , N étant la position du bit dans la séquence de numérotation précédente. La valeur du nombre binaire en complément à deux est obtenue en ajoutant les valeurs numériques affectées à chacun des bits qui sont à un, sauf le bit 8 du premier octet, puis en soustrayant de cette valeur la valeur numérique affectée à ce bit 8 du premier octet, s'il est à un.

2 L'expression *nombre entier* est synonyme du terme mathématique *entier*. Elle est utilisée à la place de celui-ci pour éviter une confusion avec le type entier (*integer*) de l'ASN.1.

3.8.2 valeur d'une syntaxe abstraite: valeur d'une syntaxe abstraite (définie comme l'ensemble des valeurs d'un type unique ASN.1), à coder selon les règles PER, ou à générer par un décodage PER.

NOTE – Le type ASN.1 unique associé à une syntaxe abstraite est identifié de façon formelle par un objet de la classe ABSTRACT-SYNTAX.

3.8.3 champ binaire: produit d'une partie du processus de codage, qui se compose d'un ensemble ordonné d'éléments binaires. Cet ensemble n'est pas nécessairement un multiple de 8 et ne commence pas nécessairement à une limite d'octet dans le codage complet de la valeur de la syntaxe abstraite.

3.8.4 codage canonique: codage complet d'une valeur dans la syntaxe abstraite, obtenu par application de règles de codage ne comportant aucune option dépendant de la mise en œuvre; de telles règles se traduisent – dans la syntaxe de transfert et dans les valeurs de la syntaxe abstraite – par des correspondances biunivoques entre chaînes binaires non ambiguës et uniques de la syntaxe de transfert et valeurs de la syntaxe abstraite.

3.8.5 type composite: type du genre ensemble, séquence, ensemble-de, séquence-de, choix, valeur PDV encapsulée, externe ou chaîne de caractères.

3.8.6 valeur composite: valeur d'un type composite.

3.8.7 nombre entier contraint: nombre entier soumis par les contraintes visibles des règles PER, de manière à s'inscrire dans un intervalle compris entre une borne inférieure "lb" et une borne supérieure "ub", bornes comprises, avec "lb" inférieure ou égale à "ub".

NOTE – Les nombres entiers contraints apparaissent dans les codages qui identifient l'alternative choisie dans un type choix, ou la longueur d'une chaîne binaire, de caractères ou d'octets lorsque la longueur du type de celle-ci est limitée à un maximum par une contrainte visible des règles PER, ou le nombre de composantes d'une valeur de type séquence-de ou ensemble-de lorsque le nombre des composantes d'un tel type est limité à un maximum par une contrainte visible des règles PER, ou la valeur d'un entier lorsque le type de celui-ci est limité à un intervalle fini par une contrainte visible des règles PER, ou la valeur ordinale d'un élément appartenant à un type énuméré.

3.8.8 contrainte effective de taille (pour un type chaîne contrainte): contrainte unique limitant une taille à une valeur finie, qui peut être appliquée à un type chaîne natif et dont l'effet sera de permettre toutes les longueurs – et seulement celles-ci – qui peuvent être présentes dans le type chaîne contrainte.

NOTE – Par exemple, la définition suivante est soumise à une contrainte effective de taille:

A ::= IA5String (SIZE(1..4) | SIZE(10..15))

car on peut la réécrire sous forme d'une unique contrainte de taille qui s'applique à toutes les valeurs comme suit:

A ::= IA5String (SIZE(1..4 | 10..15))

tandis que l'expression suivante n'est soumise à aucune contrainte effective de taille car la chaîne peut avoir une longueur quelconque si elle ne contient pas d'autres caractères que 'a', 'b' et 'c':

B ::= IA5String (SIZE(1..4) | FROM("abc"))

3.8.9 contrainte effective d'alphabet permis (pour un type chaîne de caractères restreinte et contrainte): contrainte unique d'alphabet permis que l'on peut appliquer à un type natif chaîne de caractères à multiplicateur connu, dont l'effet sera de permettre tous les caractères – et seulement ceux-ci – qui peuvent occuper une position de caractère quelconque dans n'importe quelle valeur contenue dans le type chaîne contrainte.

NOTE – Une contrainte effective d'alphabet permis sera soit l'alphabet entier du type chaîne de caractères non restreinte ou une spécification d'alphabet permis qui se trouvera être un surensemble de toutes les contraintes d'alphabet permis qui sont imposées à ce type. Par exemple, dans la définition suivante:

Ax ::= IA5String (FROM("AB") | FROM("CD"))

Bx ::= IA5String (SIZE(1..4) | FROM("abc"))

où la chaîne "Ax" obéit à une contrainte effective d'alphabet permis qui consiste en l'alphabet IA5String, puisque aucune contrainte d'alphabet permis ne s'applique à toutes les valeurs de "Ax". Il en est de même pour "Bx". Par ailleurs, la définition suivante obéit à la contrainte effective d'alphabet permis pour les caractères "ABCDE" car elle spécifie une contrainte d'alphabet permis applicable à toutes ces valeurs:

A ::= IA5String (FROM("AB") | FROM("CD") | FROM("ABCDE"))

3.8.10 index d'énumération: nombre entier non négatif associé à un item dans un type énuméré (*enumerated*). Les index d'énumération sont déterminés en classant les items par ordre croissant de la valeur énumérée, puis en affectant un index d'énumération égal à 0 pour le premier item, à 1 pour le deuxième, etc., jusqu'au dernier élément de la liste ainsi ordonnée.

NOTE – Les items de base ("RootEnumeration") et les items additionnels ("AdditionalEnumeration") sont triés séparément.

3.8.11 extensibilité au codage compact: propriété d'un type dont la définition contient un marqueur d'extension qui affecte le codage selon les règles PER.

3.8.12 liste de champs: ensemble ordonné de valeurs de champ binaire et/ou de champ binaire calé à l'octet qui résulte de l'application des présentes règles de codage aux composantes d'une valeur.

NOTE – (Didactique) Le modèle employé dans la présente Recommandation | Norme internationale utilise le terme "liste de champs" pour indiquer une liste liée de registres contenant chacun une séquence codée, une longueur en bits ou un indicateur de calage à l'octet du "champ binaire" ou du "champ binaire calé à l'octet". Chaque séquence codée correspond à une valeur d'un type ASN.1. L'indicateur de calage à l'octet précise si cette séquence codée doit être calée sur une limite d'octet lorsqu'elle est utilisée pour former le codage complet de la valeur en syntaxe abstraite; il peut également préciser s'il convient d'insérer cette séquence codée immédiatement après le dernier bit de la précédente séquence codée du codage complet. La notion de "liste de champs" n'a de fonction que descriptive et ne propose aucune méthode de mise en œuvre.

3.8.13 longueur non définie: codage dont la longueur est supérieure à 64K – 1 ou dont la longueur maximale ne peut pas être déterminée d'après la notation ASN.1.

3.8.14 type de longueur fixe: type tel que l'on puisse déterminer – à partir de la notation de type (après application des seules contraintes visibles par les règles PER) – la valeur du déterminant de longueur le plus extérieur dans une séquence codée de ce type (au moyen des mécanismes spécifiés dans la présente Recommandation | Norme internationale) et tel que cette valeur soit la même pour toutes les valeurs possibles de ce type.

3.8.15 valeur fixe: valeur telle qu'elle puisse être déterminée (au moyen des mécanismes spécifiés dans la présente Recommandation | Norme internationale) comme étant la seule valeur permise (après application des seules contraintes visibles par les règles PER) du type dont elle dépend.

3.8.16 type chaîne de caractères à multiplicateur connu: type de chaîne de caractères restreinte dont le nombre d'octets codés est un multiple fixe et connu du nombre de caractères contenus dans la chaîne de caractères pour toutes les valeurs permises de la chaîne de caractères. Les types chaîne de caractères à multiplicateur connu sont les suivants: *IA5String*, *PrintableString*, *VisibleString*, *NumericString*, *UniversalString* et *BMPString*.

3.8.17 déterminant de longueur: compte (de bits, d'octets, de caractères ou de composantes) qui détermine la longueur de tout ou partie d'une séquence à codage PER.

3.8.18 nombre entier non négatif normalement petit: partie d'une séquence codée qui représente un entier non négatif et non délimité, mais dont les petites valeurs sont normalement plus fréquentes que les grandes.

3.8.19 longueur normalement petite: codage d'une longueur qui représente les valeurs d'une longueur non délimitée, mais telle que les petites valeurs de cette longueur soient normalement plus fréquentes que les grandes.

3.8.20 champ binaire calé à l'octet: produit issu d'une partie du mécanisme de codage, composé d'un ensemble ordonné d'éléments binaires qui n'est pas nécessairement un multiple de 8 mais qui doit commencer à une limite d'octet, dans le codage complet de la valeur de syntaxe abstraite.

3.8.21 codage d'entier binaire non négatif: codage d'un nombre entier contraint ou semi-contraint pour obtenir soit un champ binaire de longueur spécifiée, soit un champ binaire calé à l'octet et de longueur spécifiée, soit encore le nombre minimal d'octets permettant de représenter ce nombre entier sous la forme d'un entier binaire non négatif, dont le codage permet de représenter des nombres entiers supérieurs ou égaux à zéro, comme spécifié au 10.3.

NOTE – La valeur d'un nombre binaire en complément à deux est obtenue en numérotant les bits des octets qui le composent, en commençant par le bit 1 du dernier octet qui devient le bit 0 et en terminant par le bit 8 du premier octet. A chaque bit est affectée une valeur numérique de 2^N , N est la position du bit dans la séquence de numérotation précédente. La valeur du nombre binaire en complément à deux est obtenue en ajoutant les valeurs numériques affectées à chacun des bits qui sont mis à 1.

3.8.22 contrainte visible par les règles PER: instance d'utilisation de la notation de contraintes ASN.1 qui affecte le codage PER d'une valeur.

3.8.23 codage à compatibilité assurée: codage complet d'une valeur de syntaxe abstraite qui peut être décodée (y compris tous modules encastrés) sans connaissance de l'ensemble contextuel qui a été défini pour la couche présentation afin de former l'environnement d'exécution du codage.

3.8.24 nombre entier semi-contraint: nombre entier obéissant à des contraintes visibles par les règles PER de façon à être égal ou supérieur à une certaine valeur "lb", celle-ci étant une valeur permise et non pas un nombre entier contraint.

NOTE – Des nombres entiers semi-contraints apparaissent dans le codage de la longueur des types chaîne de caractères, chaîne d'octets et chaîne binaire non contraints (et parfois contraints), dans le compte du nombre de composantes contenues dans des types séquence-de et ensemble-de non contraints (et parfois contraints) et dans la valeur d'un type entier qui a été contraint à dépasser une certaine valeur minimale.

3.8.25 type simple: type qui n'est pas composite.

3.8.26 contextuellement dépendante: terme utilisé pour qualifier le cas où, si un certain nom de référence est utilisé pour évaluer un ensemble d'éléments, la valeur de celui-ci est considérée comme dépendant de ce nom de référence, que l'opération arithmétique en cours d'exécution réelle soit ou non telle que la valeur de l'ensemble d'éléments soit indépendante de la valeur d'ensemble d'éléments réellement affectée au nom de référence.

NOTE – Par exemple, la définition suivante de la variable métasyntaxique "Foo" dépend textuellement de la variable "Bar", bien que celle-ci n'ait aucun effet sur l'ensemble des valeurs de "Foo" (selon le 9.3.4, la contrainte sur la variable "Foo" n'est pas visible, puisque "Bar" est soumise à une contrainte tabulaire et que "Foo" dépend textuellement de "Bar").

```
MY-CLASS ::= CLASS { &name PrintableString, &age INTEGER } WITH SYNTAX{&name, &age}
```

```
MyObjectSet MY-CLASS ::= { {"Jack", 7} | {"Jill", 5} }
```

```
Bar ::= MY-CLASS.&age ({MyObjectSet})
```

```
Foo ::= INTEGER (Bar | 1..100)
```

3.8.27 nombre entier non contraint: nombre entier qui n'est pas soumis à des contraintes visibles par les règles PER.

NOTE – Des nombres entiers non contraints n'apparaissent que dans le codage d'une valeur de type entier.

4 Abréviations

ASN.1	Notation de syntaxe abstraite numéro un (<i>abstract syntax notation one</i>)
BER	Règles de codage de base de l'ASN.1 (<i>basic encoding rules of ASN.1</i>)
PER	Règles de codage compact de l'ASN.1 (<i>packed encoding rules of ASN.1</i>)
CER	Règles de codage canonique de l'ASN.1 (<i>canonical encoding rules of ASN.1</i>)
16K	16384
32K	32768
48K	49152
64K	65536

5 Notation

La présente Recommandation | Norme internationale fait référence à la notation définie par la Rec. UIT-T X.680 | ISO/CEI 8824-1.

6 Conventions

6.1 La présente Recommandation | Norme internationale définit la valeur de chaque octet codé en utilisant les expressions "bit le plus significatif" et "bit le moins significatif".

NOTE – Les spécifications relatives aux couches inférieures utilisent la même notation pour définir l'ordre de transmission des bits sur un circuit série, ou d'affectation des bits à des canaux parallèles.

6.2 Pour les besoins de la présente Recommandation | Norme internationale, les bits d'un octet sont numérotés de 8 à 1, le 8^e bit étant le "bit le plus significatif" et le 1^{er} bit le "bit le moins significatif".

6.3 Dans la présente Recommandation | Norme internationale, le terme "octet" a souvent le sens de "huit éléments binaires". L'emploi du terme "octet" au lieu de "8 éléments binaires" n'implique aucune prescription d'alignement. Si celui-ci est recherché, cela est explicitement déclaré dans la présente Recommandation | Norme internationale.

7 Règles de codage définies dans la présente Recommandation | Norme internationale

7.1 La présente Recommandation | Norme internationale spécifie quatre règles de codage (ainsi que leurs identificateurs d'objet associés). Ces quatre règles pourront être utilisées pour coder et décoder les valeurs d'une syntaxe abstraite définie comme contenant les valeurs d'un seul type ASN.1 (connu). Le présent article décrit l'applicabilité et les propriétés de ces règles.

7.2 Si l'on ne connaît pas le type de la valeur à coder, il n'est pas possible de déterminer la structure du codage (selon l'un des algorithmes des règles de codage compact). En particulier, la fin d'une séquence codée ne peut pas être déterminée d'après cette séquence si l'on ne connaît pas le type qui est codé.

7.3 Les codages PER sont toujours à compatibilité assurée, à condition que les valeurs abstraites des types EXTERNAL, EMBEDDED PDV et CHARACTER STRING soient contraintes de manière à empêcher l'acheminement d'identificateurs de contexte de présentation.

7.4 L'algorithme de règle de codage le plus général, spécifié dans la présente Recommandation | Norme internationale, est de type BASIC-PER, qui ne produit généralement pas de codage canonique.

7.5 Un deuxième algorithme de règle de codage, spécifié dans la présente Recommandation | Norme internationale, est de type CANONICAL-PER, qui produit généralement des codages canoniques. Il est défini sous la forme d'une restriction des choix dépendant de la mise en œuvre dans le codage de type BASIC-PER. L'algorithme CANONICAL-PER produit des codages canoniques qui ont des applications lorsqu'il faut appliquer des authentificateurs à des valeurs abstraites, comme décrit dans l'Annexe D de la Rec. UIT-T X.690 | ISO/CEI 8825-1.

NOTE – Toute mise en œuvre codable selon les règles de type CANONICAL-PER est conforme aux règles de codage BASIC-PER. Toute mise en œuvre décodable selon les règles de type BASIC-PER est conforme aux règles de décodage CANONICAL-PER. Les codages effectués selon les règles CANONICAL-PER sont donc autorisés par les règles BASIC-PER.

7.6 Si un type codé selon les règles BASIC-PER ou CANONICAL-PER contient des types comme EMBEDDED PDV, CHARACTER STRING ou EXTERNAL, le codage extérieur perd son assurance de compatibilité, à moins que la syntaxe de transfert utilisée pour tous ces types (EMBEDDED PDV, CHARACTER STRING et EXTERNAL) soit elle-même à compatibilité assurée. Si un type codé selon les règles BASIC-PER ou CANONICAL-PER contient des types comme EMBEDDED PDV, EXTERNAL ou CHARACTER STRING, le codage extérieur perd son caractère canonique, à moins que la syntaxe de transfert utilisée pour tous ces types (EMBEDDED PDV, EXTERNAL et CHARACTER STRING) ne soit elle-même canonique.

NOTE – Les syntaxes de transfert de caractères, prenant en charge toutes les syntaxes abstraites en mode caractère de la forme {iso standard 10646 level-1 (1)} sont canoniques. Celles qui prennent en charge des syntaxes de la forme {iso standard 10646 level-2 (2)} et {iso standard 10646 level-3 (3)} ne sont pas toujours canoniques. Toutes les syntaxes de transfert de caractères susmentionnées sont à compatibilité assurée.

7.7 Les règles BASIC-PER et CANONICAL-PER ont chacune deux variantes: ALIGNED et UNALIGNED. Dans la variante ALIGNED, des bits de bourrage sont insérés de temps en temps afin de restaurer l'alignement en octets. Dans la variante UNALIGNED, aucun bit de bourrage n'est jamais inséré.

7.8 Il n'existe aucune possibilité d'interfonctionnement entre la variante ALIGNED et la variante UNALIGNED.

7.9 Les codages compacts (selon les règles PER) ne sont autodélimitants que si l'on connaît le type de la valeur codée. Les codages sont toujours un multiple de 8 éléments binaires. Lorsqu'ils sont acheminés dans un type EXTERNAL, ils doivent figurer dans l'option OCTET STRING, à moins que le type EXTERNAL soit lui-même en codage compact, auquel cas la valeur peut être codée sous la forme d'un unique type ASN.1 (c'est-à-dire comme un type ouvert). Lorsque les codages compacts sont acheminés dans un protocole de couche présentation de l'OSI, le "codage complet" (tel que défini dans la Rec. UIT-T X.226 | ISO/CEI 8823-1) doit être utilisé avec l'option OCTET STRING.

7.10 Les règles de la présente Recommandation | Norme internationale s'appliquent aux deux algorithmes et aux deux variantes, sauf indication contraire.

7.11 L'Annexe C est informative et donne des recommandations sur les combinaisons de règles PER à mettre en œuvre afin de maximiser les chances d'interfonctionnement.

8 Conformité

8.1 La conformité dynamique est spécifiée à partir de l'article 9.

8.2 La conformité statique est spécifiée par les règles d'application des présentes règles de codage compact.

NOTE – L'Annexe C de la présente Recommandation | Norme internationale donne des directives sur la conformité statique afin d'assurer le support des deux variantes des deux algorithmes de codage. Ces directives sont conçues de façon à assurer l'interfonctionnement, tout en admettant que, pour certaines applications, il peut être préférable de suivre des règles de codage qui ne sont ni à compatibilité assurée ni canoniques.

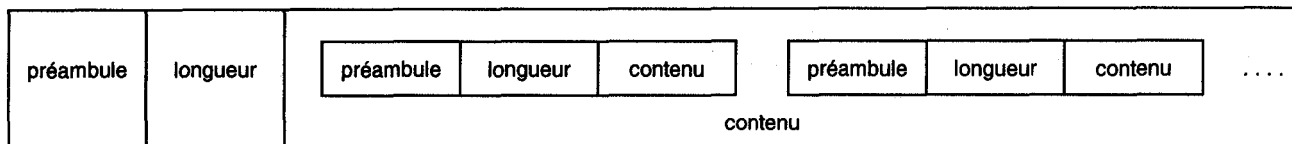
8.3 Les règles contenues dans la présente Recommandation | Norme internationale sont spécifiées en termes de procédure de codage. Les mises en œuvre ne sont pas tenues de refléter intégralement la procédure spécifiée, à condition que la chaîne binaire produite comme codage complet d'une valeur de syntaxe abstraite, soit identique à l'une des chaînes binaires spécifiées dans la présente Recommandation | Norme internationale pour la syntaxe de transfert applicable.

8.4 Les mises en œuvre effectuant le décodage sont tenues de produire la valeur de syntaxe abstraite correspondant à toute chaîne binaire reçue en provenance d'un expéditeur se conformant aux règles de codage indiquées dans la syntaxe de transfert associée aux données à décoder.

NOTES

1 En général, on ne définit pas de variantes de codage pour les règles BASIC-PER qui sont explicitement déclarées dans la présente Recommandation | Norme internationale. Le codage BASIC-PER devient canonique lorsque l'on spécifie un fonctionnement à compatibilité assurée et que l'on restreint certaines des options de codage indiquées par d'autres Normes ISO/CEI citées en référence. L'algorithme CANONICAL-PER offre une variante, aussi bien aux règles de codage distinctif (DER) qu'aux règles de codage canonique (CER) (voir la Rec. UIT-T X.690 | ISO/CEI 8825-1), lorsqu'il est nécessaire de disposer d'un codage canonique à compatibilité assurée.

2 Lorsque l'algorithme CANONICAL-PER est utilisé pour produire un codage canonique, il est recommandé que toute valeur chiffrée à codage dispersé qui en est dérivée dispose d'un identificateur d'algorithme associé qui indique que l'algorithme CANONICAL-PER a été utilisé pour transformer la valeur abstraite en une chaîne binaire initiale (dispersée par la suite).



NOTE – Le préambule, la longueur et le contenu sont des "champs" dont la concaténation forme une "liste de champs". La liste de champs d'un type composite différent du type choix (*Choice*) peut être formée par la concaténation de champs de valeur différente. Le préambule, la longueur et/ou le contenu peuvent ne pas figurer, quelle que soit leur valeur.

Figure 1 – Codage d'une valeur composite dans une liste de champs

9.5.2 Le codage d'une composante de valeur de données est constitué d'une des deux manières suivantes:

- a) comprend trois parties, comme représenté à la Figure 1, qui apparaissent dans l'ordre suivant:
 - 1) un préambule (voir articles 18, 20 et 22);
 - 2) un déterminant de longueur (voir 10.9);
 - 3) un contenu;
- b) ou bien (si le contenu est important), un nombre quelconque de parties (comme représenté à la Figure 2), dont la première est un préambule (voir articles 18, 20 et 22), les parties suivantes étant des paires de champs binaires calés à l'octet, le premier champ étant un déterminant de longueur pour un fragment du contenu et le deuxième étant ce fragment de contenu; la dernière paire de champs est identifiée par la partie contenant le déterminant de longueur, comme spécifié au 10.9.

(standards.iteh.ai)

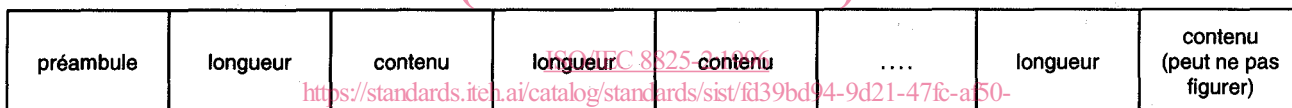


Figure 2 – Codage d'une valeur de données longue

9.5.3 Chacune des parties mentionnées au 9.5.2 produit un des résultats suivants:

- a) un champ nul (néant);
- b) un champ binaire;
- c) un champ binaire calé à l'octet;
- d) une liste de champs pouvant contenir des champs binaires, ou des champs binaires calés à l'octet, ou les deux sortes.

9.6 Types à coder

9.6.1 Les articles suivants spécifient le codage des types suivants pour créer une liste de champs: booléen, entier, énuméré, réel, chaîne binaire, chaîne d'octets, néant, séquence, séquence-de, ensemble, ensemble-de, choix, ouvert, identificateur d'objet, valeurs PDV encapsulées, externe, chaîne de caractères restreinte et chaîne de caractères non restreinte.

9.6.2 Le type ANY, défini dans la Rec. X.208 du CCITT (1988) | ISO/CEI 8824:1990 doit être codé comme un type ouvert.

9.6.3 Le type de sélection doit être codé sous forme de séquence codée du type sélectionné.

9.6.4 Le codage d'un type étiqueté n'est pas traité dans la présente Recommandation | Norme internationale car, sauf dans le cas stipulé au 9.2, l'étiquetage n'est pas visible dans le modèle type et valeur utilisé pour les présentes règles de codage. Un type étiqueté est donc codé comme le type correspondant qui a été étiqueté.

9.6.5 Les "types utiles" suivants, définis à l'article 38 de la Rec. UIT-T X.680 | ISO/CEI 8824-1, doivent être codés comme s'ils avaient été remplacés par leurs définitions selon la Rec. UIT-T X.680 | ISO/CEI 8824-1:

- temps généralisé;
- temps universel;
- descripteur d'objet.

Les contraintes sur les types utiles ne sont pas visibles par les règles PER.

10 Procédures de codage

10.1 Production du codage complet

10.1.1 La liste de champs produite par l'application des règles de la présente Recommandation | Norme internationale à la valeur la plus externe doit être utilisée pour produire le codage complet de la valeur en syntaxe abstraite, comme suit: on prend tour à tour chaque champ de la liste et on le concatène à la fin de la chaîne binaire qui forme le codage complet de la valeur en syntaxe abstraite, tout en ajoutant des bits 0 de bourrage, comme spécifié ci-après.

10.1.2 Dans la variante UNALIGNED des présentes règles de codage, tous les champs doivent être concaténés sans bourrage. Si le résultat du codage de la valeur la plus externe est une chaîne binaire vide, celle-ci doit être remplacée par un octet unique dont tous les bits sont forcés à 0. Si le résultat est une chaîne binaire non vide et que cette chaîne ne soit pas un multiple de huit éléments binaires (zéro à sept), les bits 0 doivent être ajoutés à cette chaîne afin d'obtenir un multiple de huit éléments binaires.

10.1.3 Dans la variante ALIGNED des présentes règles de codage, tous les champs binaires de la liste de champs doivent être concaténés sans bourrage et tous les champs binaires alignés en octets doivent être concaténés après adjonction d'une série de (zéro à sept) bits 0 afin que la longueur du codage produit jusque-là soit un multiple de huit éléments binaires. Si le résultat du codage de la valeur la plus externe est une chaîne binaire vide, celle-ci doit être remplacée par un octet unique dont tous les bits sont forcés à 0. Si le résultat est une chaîne binaire non vide et que cette chaîne ne soit pas un multiple de huit éléments binaires (zéro à sept), les bits 0 doivent être ajoutés à cette chaîne afin d'obtenir un multiple de huit éléments binaires.

NOTE – Le codage de la valeur la plus extérieure est la chaîne binaire vide si, par exemple, la valeur de la syntaxe abstraite est du type néant ou un type entier contraint à être une valeur unique.

10.1.4 La chaîne binaire résultante est le codage complet de la valeur de syntaxe abstraite.

10.2 Champs de type ouvert

10.2.1 Pour coder un champ de type ouvert, la valeur du type occupant déjà ce champ doit être codée de manière à créer une liste de champs, laquelle est ensuite convertie dans la séquence de codage complète d'une valeur de syntaxe abstraite, comme spécifié au 10.1, afin de produire une chaîne d'octets de longueur "n" (par exemple).

10.2.2 La liste des champs de la valeur dans laquelle le type ouvert doit être imbriqué doit alors être complétée (comme spécifié au 10.9) d'une longueur non contrainte de "n" (en octets) et d'un champ binaire calé à l'octet, contenant les éléments binaires produits au 10.2.1.

NOTE – Lorsque le nombre d'octets contenus dans le type ouvert est grand, on applique les procédures de fragmentation indiquées au 10.9 et le codage du type ouvert sera haché sans tenir compte de la position des limites de fragment dans le codage du type occupant le champ de type ouvert.

10.3 Codage sous forme d'un entier binaire non négatif

NOTE – (Didactique) Ce paragraphe précise le sens du terme "codage d'entier binaire non négatif", ce nombre étant placé dans un champ dont le nombre d'éléments binaires est fixe ou dans un champ dont le nombre d'octets est le minimum requis pour contenir ce nombre.

10.3.1 Les paragraphes suivants se rapportent à la production d'un codage entier binaire non négatif représentant un nombre entier non négatif afin d'obtenir un champ binaire de longueur spécifiée, un seul octet, deux octets ou le nombre minimal d'octets permettant de représenter la valeur. Ce paragraphe spécifie le codage précis à appliquer lorsque de telles références sont faites.

10.3.2 Le bit initial du champ est défini comme étant le bit le plus significatif du premier octet et le bit final du champ est défini comme étant le bit le moins significatif du dernier octet.

10.3.3 Pour la définition suivante seulement, les bits doivent être numérotés comme suit: zéro pour le bit final du champ, un pour le bit suivant et ainsi de suite jusqu'au bit initial du champ.