

---

---

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

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

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

[ISO/IEC 8825-2:2002](https://standards.iteh.ai/catalog/standards/sist/69450a17-bed6-4d54-ba3f-a532f3e3e9cd/iso-iec-8825-2-2002)

<https://standards.iteh.ai/catalog/standards/sist/69450a17-bed6-4d54-ba3f-a532f3e3e9cd/iso-iec-8825-2-2002>

**PDF – Exonération de responsabilité**

Le présent fichier PDF peut contenir des polices de caractères intégrées. Conformément aux conditions de licence d'Adobe, ce fichier peut être imprimé ou visualisé, mais ne doit pas être modifié à moins que l'ordinateur employé à cet effet ne bénéficie d'une licence autorisant l'utilisation de ces polices et que celles-ci y soient installées. Lors du téléchargement de ce fichier, les parties concernées acceptent de fait la responsabilité de ne pas enfreindre les conditions de licence d'Adobe. Le Secrétariat central de l'ISO décline toute responsabilité en la matière.

Adobe est une marque déposée d'Adobe Systems Incorporated.

Les détails relatifs aux produits logiciels utilisés pour la création du présent fichier PDF sont disponibles dans la rubrique General Info du fichier; les paramètres de création PDF ont été optimisés pour l'impression. Toutes les mesures ont été prises pour garantir l'exploitation de ce fichier par les comités membres de l'ISO. Dans le cas peu probable où surviendrait un problème d'utilisation, veuillez en informer le Secrétariat central à l'adresse donnée ci-dessous.

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

[ISO/IEC 8825-2:2002](https://standards.iteh.ai/catalog/standards/sist/69450a17-bed6-4d54-ba3f-a532f3e3e9cd/iso-iec-8825-2-2002)

<https://standards.iteh.ai/catalog/standards/sist/69450a17-bed6-4d54-ba3f-a532f3e3e9cd/iso-iec-8825-2-2002>

© ISO/CEI 2002

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'ISO à l'adresse ci-après ou du comité membre de l'ISO dans le pays du demandeur.

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](mailto:copyright@iso.org)  
Web [www.iso.org](http://www.iso.org)

Version française parue en 2006

Publié en Suisse

## TABLE DES MATIÈRES

	<i>Page</i>	
1	Domaine d'application .....	1
2	Références normatives .....	1
2.1	Recommandations   Normes internationales identiques .....	1
2.2	Paires de Recommandations   Normes internationales équivalentes par leur contenu technique .....	1
2.3	Autres références .....	1
3	Définitions .....	2
3.1	Spécification de la notation de base .....	2
3.2	Spécification des objets informationnels .....	2
3.3	Spécification des contraintes .....	2
3.4	Spécification du paramétrage en notation ASN.1 .....	2
3.5	Règles de codage de base .....	2
3.6	Autres définitions .....	2
4	Abréviations .....	5
5	Notation .....	5
6	Conventions .....	5
7	Règles de codage définies dans la présente Recommandation   Norme internationale .....	5
8	Conformité .....	6
9	Méthode de codage utilisée pour les règles PER .....	7
9.1	Utilisation de la notation de types .....	7
9.2	Utilisation d'étiquettes pour établir un ordre canonique .....	7
9.3	Contraintes visibles par les règles PER .....	7
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 .....	10
10.1	Production du codage complet .....	10
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
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 .....	18
16	Codage du type chaîne d'octets .....	19
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 .....	22
23	Codage du type identificateur d'objet .....	23
24	Codage du type identificateur d'objet relatif .....	23

	<i>Page</i>
25	Codage d'une valeur type valeur de donnée de présentation enchâssée ..... 24
26	Codage d'une valeur du type externe..... 24
27	Codage des types chaîne de caractères restreinte ..... 25
28	Codage du type chaîne de caractères non restreinte ..... 27
29	Identificateurs d'objet pour syntaxes de transfert ..... 27
Annexe A	– Exemples de codages ..... 29
A.1	Enregistrement qui n'utilise pas de contrainte appliquée aux sous-types ..... 29
A.2	Enregistrement utilisant des contraintes appliquées aux sous-types ..... 32
A.3	Enregistrement qui utilise des marqueurs d'extension ..... 35
A.4	Enregistrement utilisant des groupes d'adjonctions d'extension..... 38
Annexe B	– Combinaison de contraintes visibles par les règles PER et de contraintes non visibles par les règles PER..... 41
B.1	Généralités..... 41
B.2	Extensibilité et visibilité des contraintes liées aux règles PER ..... 42
B.3	Exemples..... 45
Annexe C	– Prise en charge des algorithmes PER..... 47
Annexe D	– Prise en charge des règles d'extensibilité ASN.1 ..... 48
Annexe E	– Complément didactique sur la concaténation de codages conformes aux règles PER ..... 49
Annexe F	– Affectation de valeurs à un identificateur d'objet..... 50

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

[ISO/IEC 8825-2:2002](https://standards.iteh.ai/catalog/standards/sist/69450a17-bed6-4d54-ba3f-a532f3e3e9cd/iso-iec-8825-2-2002)

<https://standards.iteh.ai/catalog/standards/sist/69450a17-bed6-4d54-ba3f-a532f3e3e9cd/iso-iec-8825-2-2002>

## Avant-propos

L'ISO (Organisation internationale de normalisation) et la CEI (Commission électrotechnique internationale) forment le système spécialisé de la normalisation mondiale. 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 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 et 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 Normes internationales sont rédigées conformément aux règles données dans les Directives ISO/CEI, Partie 2.

La tâche principale du comité technique mixte est d'élaborer les Normes internationales. Les projets de Normes internationales adoptés par le comité technique mixte sont soumis aux organismes nationaux pour vote. Leur publication comme Normes internationales requiert l'approbation de 75 % au moins des organismes nationaux votants.

L'attention est appelée sur le fait que certains des éléments du présent document peuvent faire l'objet de droits de propriété intellectuelle ou de droits analogues. L'ISO et la CEI ne sauraient être tenues pour responsables de ne pas avoir identifié de tels droits de propriété et averti de leur existence.

L'ISO/CEI 8825-2 a été élaborée par le comité technique mixte ISO/CEI JTC 1, *Technologies de l'information*, sous-comité SC 6, *Téléinformatique*, en collaboration avec l'UIT-T. Le texte identique est publié en tant que Rec. UIT-T X.691.

Cette troisième édition annule et remplace la deuxième édition (ISO/CEI 8825-2:1998), qui a fait l'objet d'une révision technique. Elle incorpore aussi l'Amendement ISO/CEI 8825-2:1998/Amd.1:2000 ainsi que les Rectificatifs techniques ISO/CEI 8825-2:1998/Cor.1:1999, ISO/CEI 8825-2:1998/Cor.2:2002 et ISO/CEI 8825-2:1998/Cor.3:2002.

L'ISO/CEI 8825 comprend les parties suivantes, présentées sous le titre général *Technologies de l'information — Règles de codage ASN.1*:

- *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)*
- *Partie 3: Spécification de la notation de contrôle de codage (ECN)*
- *Partie 4: Règles de codage XML (XER)*
- *Partie 5: Mappage en ASN.1 des définitions de schéma XML du W3C*

## ISO/CEI 8825-2:2002(F)

### 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 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, *basic encoding rules*) 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 8825-2:2002](https://standards.iteh.ai/catalog/standards/sist/69450a17-bed6-4d54-ba3f-a532f3e3e9cd/iso-iec-8825-2-2002)

<https://standards.iteh.ai/catalog/standards/sist/69450a17-bed6-4d54-ba3f-a532f3e3e9cd/iso-iec-8825-2-2002>

**NORME INTERNATIONALE  
RECOMMANDATION UIT-T**

**Technologies de l'information – Règles de codage ASN.1:  
spécification des règles de codage compact**

## 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 | ISO/CEI 8824-1.

## 2 Références normatives

Les Recommandations et 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 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.

[https://standards.iteh.ai/catalog/standards/sist/69450a17-bed6-4d54-ba3f-](https://standards.iteh.ai/catalog/standards/sist/69450a17-bed6-4d54-ba3f-532b3c0d5e1e/iso-8825-2-2002)

### 2.1 Recommandations | Normes internationales identiques

- Recommandation UIT-T X.680 (2002) | ISO/CEI 8824-1:2002, *Technologies de l'information – Notation de syntaxe abstraite numéro un: spécification de la notation de base.*
- Recommandation UIT-T X.681 (2002) | ISO/CEI 8824-2:2002, *Technologies de l'information – Notation de syntaxe abstraite numéro un: spécification des objets informationnels.*
- Recommandation UIT-T X.682 (2002) | ISO/CEI 8824-3:2002, *Technologies de l'information – Notation de syntaxe abstraite numéro un: spécification des contraintes.*
- Recommandation UIT-T X.683 (2002) | ISO/CEI 8824-4:2002, *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 (2002) | ISO/CEI 8825-1:2002, *Technologies de l'information – Règles de codage ASN.1: spécification des règles de codage de base, des règles de codage canoniques et des règles de codage distinctives.*

### 2.2 Paires de Recommandations | Normes internationales équivalentes par leur contenu technique

### 2.3 Autres références

- ISO/CEI 646:1991, *Technologies de l'information – Jeu ISO de caractères codés à 7 éléments pour l'échange d'information.*
- ISO/CEI 2022:1994, *Technologies de l'information – Structure de code de caractères et techniques d'extension.*
- ISO 2375:2003, *Traitement de l'information – Procédure pour l'enregistrement des séquences d'échappement et des jeux de caractères codés.*

## ISO/CEI 8825-2:2002 (F)

- ISO 6093:1985, *Traitement de l'information – Représentation des valeurs numériques dans les chaînes de caractères pour l'échange d'information.*
- *ISO Registre international des jeux de caractères codés à utiliser avec une séquence d'échappement.*
- ISO/CEI 10646-1:2003, *Technologies de l'information – Jeu universel de caractères codés sur plusieurs octets (JUC).*

### 3 Définitions

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

#### 3.1 Spécification de la notation de base

Pour les besoins de la présente Recommandation | Norme internationale, toutes les définitions contenues dans la Rec. UIT-T X.680 | ISO/CEI 8824-1 s'appliquent.

#### 3.2 Spécification des objets informationnels

Pour les besoins de la présente Recommandation | Norme internationale, toutes les définitions figurant dans la Rec. UIT-T X.681 | ISO/CEI 8824-2 s'appliquent.

#### 3.3 Spécification des contraintes

La présente Recommandation | Norme internationale utilise le terme suivant défini dans la Rec. UIT-T X.682 | ISO/CEI 8824-3:

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

#### 3.4 Spécification du paramétrage en notation ASN.1

La présente Recommandation | Norme internationale utilise le terme suivant défini dans la Rec. UIT-T X.683 | ISO/CEI 8824-4:

- contrainte variable.

#### 3.5 Règles de codage de base

La présente Recommandation | Norme internationale utilise les termes suivants 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.

#### 3.6 Autres définitions

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

**3.6.1 codage d'entier binaire en complément à 2:** codage d'un nombre entier sur un champ binaire (calé à l'octet suivant la variante ALIGNED) 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.

NOTE 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.



NOTE 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.

**3.6.2 valeur de 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**.

**3.6.3 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.

NOTE – Si l'utilisation de ce terme est suivie de la mention "calé à l'octet suivant la variante ALIGNED", cela signifie que ce champ binaire doit commencer au niveau d'une limite d'octet dans le cadre du codage complet pour la variante ALIGNED des règles PER.

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

**3.6.5 type composite:** type du genre ensemble, séquence, ensemble-de, séquence-de, choix, valeur pdv incorporée, externe ou chaîne de caractères.

**3.6.6 valeur composite:** valeur d'un type composite.

**3.6.7 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.

**3.6.8 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 1 – 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"))
```

NOTE 2 – La contrainte effective de taille sert uniquement à déterminer le codage des longueurs.

**3.6.9 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 au moins une position de caractère dans n'importe quelle valeur contenue dans le type chaîne contrainte.

NOTE 1 – Par exemple dans:

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

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

la chaîne "Ax" obéit à une contrainte effective d'alphabet permis "ABCD", et "Bx" obéit à une contrainte effective d'alphabet permis qui consiste en l'alphabet IA5String complet, puisque aucune contrainte d'alphabet permis plus petite ne s'applique à toutes les valeurs de "Bx".

NOTE 2 – La contrainte effective d'alphabet permis sert uniquement à déterminer le codage des caractères.

**3.6.10 indice 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.

**3.6.11 extensibilité pour un codage compact:** propriété d'un type qui requiert que les règles PER identifient le codage d'une valeur comme étant celui d'une valeur racine ou celui d'une addition d'extension.

NOTE – Les valeurs racine sont généralement codées d'une manière plus efficace que les additions d'extension.

**3.6.12 liste de champs:** ensemble ordonné de champs binaires qui résulte de l'application des présentes règles de codage aux composantes d'une valeur abstraite.

**3.6.13 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.

**3.6.14 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.

**3.6.15 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.

**3.6.16 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**.

**3.6.17 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.

**3.6.18 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.

**3.6.19 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.

**3.6.20 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 (aligné à l'octet suivant la variante ALIGNED) 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.

**3.6.21 type le plus externe:** type ASN.1 dont le codage est compris dans une porteuse non ASN.1 ou en tant que valeur d'autres structures ASN.1 (voir le § 10.1.1).

NOTE – Le codage PER d'un type le plus externe se fait toujours sur un multiple entier de huit bits.

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

**3.6.23 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'environnement d'exécution du codage.

**3.6.24 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.

**3.6.25 type simple:** type qui n'est pas composite.

**3.6.26 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 contextuellement de la variable "Bar", bien que celle-ci n'ait aucun effet sur l'ensemble des valeurs de "Foo" (selon le § 9.3.5, la contrainte sur la variable "Foo" n'est pas visible, puisque "Bar" est soumise à une contrainte tabulaire et que "Foo" dépend contextuellement 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)
```

**3.6.27 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

Pour les besoins de la présente Recommandation | Norme internationale, les abréviations suivantes s'appliquent:

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> )
CER	Règles de codage canonique de l'ASN.1 ( <i>canonical encoding rules of ASN.1</i> )
DER	Règles de codage distinctives de l'ASN.1 ( <i>distinguished encoding rules of ASN.1</i> )
PER	Règles de codage compact de l'ASN.1 ( <i>packed 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 iTech STANDARD PREVIEW

**6.1** La présente Recommandation | Norme internationale définit la valeur de chaque octet codé en utilisant les expressions "bit de plus fort poids" et "bit de plus faible poids".

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<sup>e</sup> bit étant le " bit de plus fort poids " et le 1<sup>er</sup> bit le " bit de plus faible poids ".

**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 OSI.

**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 l'implémentation dans le codage de type BASIC-PER.

NOTE 1 – L'algorithme CANONICAL-PER produit des codages canoniques qui ont des applications lorsqu'il faut appliquer des authenticateurs à des valeurs abstraites.

NOTE 2 – Toute implémentation codable selon les règles de type CANONICAL-PER est conforme aux règles de codage BASIC-PER. Toute implémentation 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 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 implémenter afin de maximiser les chances d'interfonctionnement.

STANDARD PREVIEW  
(standards.itech.ai)

## 8 Conformité

[ISO/IEC 8825-2:2002](https://standards.itech.ai/catalog/standards/sist/69450a17-bed6-4d54-ba3f-900000000000/iso-iec-8825-2-2002)

[https://standards.itech.ai/catalog/standards/sist/69450a17-bed6-4d54-ba3f-](https://standards.itech.ai/catalog/standards/sist/69450a17-bed6-4d54-ba3f-900000000000/iso-iec-8825-2-2002)

8.1 La conformité dynamique est spécifiée à partir du § 9.8825-2-2002

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

NOTE – L'Annexe C 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 implémentations 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.

NOTE 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, *distinguished encoding rules*) qu'aux règles de codage canonique (CER, *canonical encoding rules*) (voir la Rec. UIT-T X.690 | ISO/CEI 8825-1), lorsqu'il est nécessaire de disposer d'un codage canonique à compatibilité assurée.

NOTE 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).

## 9 Méthode de codage utilisée pour les règles PER

### 9.1 Utilisation de la notation de types

**9.1.1** Les présentes règles de codage font spécifiquement appel à la notation des types ASN.1 qui est spécifiée dans la Rec. UIT-T X.680 | ISO/CEI 8824-1. Elles ne peuvent être appliquées que pour coder les valeurs d'un seul type ASN.1, spécifié au moyen de cette notation.

**9.1.2** Ces règles dépendent en particulier, mais non exclusivement, de la conservation des informations suivantes dans le modèle de type et de valeur ASN.1 sur lequel est fondée l'utilisation de cette notation:

- a) l'imbrication de types choix à l'intérieur de types choix;
- b) les étiquettes attribuées aux composantes d'un type ensemble et aux variantes d'un type choix ainsi que les valeurs attribuées à une énumération;
- c) le fait qu'une composante d'un type ensemble ou séquence soit ou non facultative;
- d) le fait qu'une composante d'un type ensemble ou séquence possède ou non une valeur par **défaut**;
- e) la restriction de l'étendue des valeurs d'un type en raison de l'application de contraintes visibles (seulement) par les règles PER;
- f) le fait qu'une composante soit de type ouvert;
- g) le fait qu'un type soit extensible pour le codage PER.

### 9.2 Utilisation d'étiquettes pour établir un ordre canonique

La présente Recommandation | Norme internationale prescrit que les composantes d'un type ensemble ou choix soient en relation d'ordre canonique, indépendamment de leur ordre contextuel. L'ordre canonique est déterminé par tri de l'étiquette la plus externe attribuée à chaque composante, comme spécifié au § 8.6 de la Rec. UIT-T X.680 | ISO/CEI 8824-1.

iTeh STANDARD PREVIEW  
(standards.iteh.ai)

### 9.3 Contraintes visibles par les règles PER

NOTE – Le fait que certaines contraintes de notation ASN.1 puissent ne pas être visibles par les règles PER dans le cadre d'un codage et d'un décodage n'a aucune incidence que ce soit sur l'utilisation de telles contraintes pour traiter des erreurs détectées au cours du décodage; il n'en découle pas non plus que les valeurs violant de telles contraintes puissent être émises par un expéditeur conforme. Toutefois, la présente Recommandation | Norme internationale n'utilise en aucune façon de telles contraintes pour la spécification des codages.

**9.3.1** Les contraintes exprimées sous forme de texte en clair ou de commentaire ASN.1 ne sont pas visibles par les règles PER.

**9.3.2** Les contraintes variables ne sont pas visibles par les règles PER (voir les § 10.3 et 10.4 de la Rec. UIT-T X.683 | ISO/CEI 8824-4).

**9.3.3** Les contraintes tabulaires ne sont pas visibles par les règles PER (voir la Rec. UIT-T X.682 | ISO/CEI 8824-3).

**9.3.4** Les contraintes relationnelles sur composantes (voir la Rec. UIT-T X.682 | ISO/CEI 8824-3, 10.7) ne sont pas visibles par les règles PER.

**9.3.5** Les contraintes dont l'évaluation est contextuellement dépendante d'une contrainte tabulaire ou d'une contrainte relationnelle sur composantes ne sont pas visibles par les règles PER (voir la Rec. UIT-T X.682 | ISO/CEI 8824-3).

**9.3.6** Les contraintes sur des types chaîne de caractères restreinte autres que des types chaîne de caractères à multiplicateur connu (voir le § 37 de la Rec. UIT-T X.680 | ISO/CEI 8824-1) ne sont pas visibles par les règles PER (voir le § 3.6.16).

**9.3.7** Les contraintes sur la configuration ne sont pas visibles par les règles PER.

**9.3.8** Sous réserve de ce qui précède, toutes les contraintes de taille sont visibles par les règles PER.

**9.3.9** La contrainte effective de taille pour un type contraint est une contrainte de taille unique telle qu'une taille ne soit permise que si ce type contraint possède une valeur qui a cette taille (permise).

**9.3.10** Les contraintes d'alphabet permis sur des types chaîne de caractères à multiplicateur connu qui ne sont pas extensibles après l'application des § 48.3 à 48.5 de la Rec. UIT-T X.680 | ISO/CEI 8824-1, sont visibles par les règles PER. Les contraintes d'alphabet permis qui sont extensibles ne sont visibles par les règles PER.

**9.3.11** La contrainte effective d'alphabet permis sur un type contraint est une contrainte sur alphabet permis isolée, telle qu'un caractère ne soit permis que si ce type contraint possède une valeur qui contient ce caractère. Si tous les caractères du type soumis à la contrainte peuvent être présents dans une valeur du type contraint, la contrainte effective d'alphabet permis est l'ensemble des caractères définis pour le type non contraint.

**9.3.12** Des contraintes appliquées à un type REAL ne sont pas visibles par les règles PER.

**9.3.13** Une contrainte de type interne, appliquée à un type chaîne de caractères non restreinte ou à un type valeur pdv incorporée, n'est visible par les règles PER que lorsqu'elle sert à restreindre à une seule variante la valeur de la composante "syntaxes", ou lorsqu'elle sert à restreindre la composante "identification" à la seule valeur "fixed" (voir les § 25 et 28).

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

**9.3.15** Les contraintes de sous-typage à valeur unique appliquées à un type chaîne de caractères ne sont pas visibles par les règles PER.

**9.3.16** Sous réserve de ce qui précède, toutes les autres contraintes ne sont visibles par les règles PER que si elles sont appliquées à un type entier ou à un type chaîne de caractères à multiplicateur connu.

**9.3.17** En général, la contrainte sur un type associera plusieurs contraintes en utilisant quelques-unes ou l'ensemble des opérations arithmétiques, des contraintes de sous-typage contenu et une série d'applications de contraintes. Les paragraphes suivants indiquent ce qui se passe lorsque certaines des composantes de la contrainte totale sont visibles par les règles PER et que d'autres ne le sont pas.

NOTE – Voir l'Annexe B, pour une étude plus approfondie de la combinaison de contraintes visibles par les règles PER et de contraintes non visibles par les règles PER.

**9.3.18** Si une contrainte consiste en une série d'applications de contraintes, les éventuelles contraintes qui ne sont pas visibles par les règles PER n'affectent pas les codages PER, mais entraînent la suppression de l'extensibilité (et des additions d'extension) figurant dans des contraintes antérieures, comme il est spécifié au § 46.8 de la Rec. UIT-T X.680 | ISO/CEI 8824-1

NOTE 1 – Si la contrainte finale d'une série d'applications n'est pas visible par les règles PER, le type n'est alors pas extensible pour un codage PER et on le code sans bit d'extension.

NOTE 2 – Par exemple:

**A ::= IA5String(SIZE(1..4))(FROM("ABCD",...))**

a une contrainte effective d'alphabet permis qui consiste en la totalité de l'alphabet IA5String, puisque la contrainte extensible d'alphabet permis n'est pas visible par les règles PER. Elle a néanmoins une contrainte effective de taille, à savoir SIZE(1..4).

De même,

**B ::= IA5String(A)**

a la même contrainte effective de taille et la même contrainte effective d'alphabet permis.

**9.3.19** Si une contrainte visible par les règles PER fait partie d'une structure INTERSECTION, la contrainte résultante est visible par les règles PER et correspond à l'INTERSECTION de toutes les composantes visibles par les règles PER (les composantes qui ne sont pas visibles par les règles PER étant alors ignorées). Si une contrainte qui n'est pas visible par les règles PER appartient à une structure UNION, la contrainte résultante n'est pas visible par les règles PER. Si une contrainte contient une déclaration EXCEPT, EXCEPT et l'ensemble de valeurs qui suivent sont complètement ignorés, que cet ensemble de valeurs soit ou non visible par les règles PER.

NOTE – Par exemple:

**A ::= IA5String(SIZE(1..4) INTERSECTION FROM("ABCD",...))**

a pour contrainte effective de taille (1..4) mais la contrainte d'alphabet n'est pas visible par les règles PER parce qu'elle est extensible.

**9.3.20** Un type est également extensible au codage compact (qu'il fasse ou non l'objet de contraintes ultérieures) si une des conditions suivantes est vérifiée:

- a) ce type est dérivé d'un type ENUMERATED (par sous-typage, par référencement de type ou par étiquetage) et la production "Enumerations" contient un marqueur d'extension;
- b) ou bien ce type est dérivé d'un type SEQUENCE (par sous-typage, par référencement de type ou par étiquetage) et la production "ComponentTypeLists" ou "SequenceType" contient un marqueur d'extension;
- c) ou bien ce type est dérivé d'un type SET (par sous-typage, par référencement de type ou par étiquetage) et la production "ComponentTypeLists" ou "SetType" contient un marqueur d'extension;
- d) ou bien ce type est dérivé d'un type CHOICE (par sous-typage, par référencement de type ou par étiquetage) et la production "AlternativeTypeLists" contient un marqueur d'extension.

**9.4 Modèle utilisé pour coder les types et les valeurs**

**9.4.1** En notation ASN.1, un type est simple ou construit au moyen d'autres types. Cette notation permet de faire appel à la référénciation et à l'étiquetage des types. Dans le cadre des présentes règles de codage, la référénciation et l'étiquetage des types n'ont aucune incidence sur le codage et sont invisibles dans le modèle, sauf ce qui est indiqué au § 9.2. La notation permet également d'appliquer des contraintes et de spécifier des erreurs. Les contraintes visibles par les règles PER sont présentes dans le modèle en tant que restrictions sur les valeurs d'un type. D'autres contraintes et spécifications d'erreur n'ont pas d'incidence sur le codage et sont invisibles dans le modèle type et valeur PER.

**9.4.2** Une valeur à coder peut être considérée comme étant simple ou composite, c'est-à-dire construite au moyen des mécanismes de structuration à partir de composantes qui sont des valeurs simples ou des valeurs composites, ce qui correspond à la structure des définitions de type en notation ASN.1.

**9.4.3** Lorsqu'une contrainte comprend une valeur présente dans la racine en tant qu'extension d'addition, cette valeur est toujours codée en tant que valeur dans la racine et pas en tant que valeur correspondant à une addition d'extension.

**EXEMPLE**

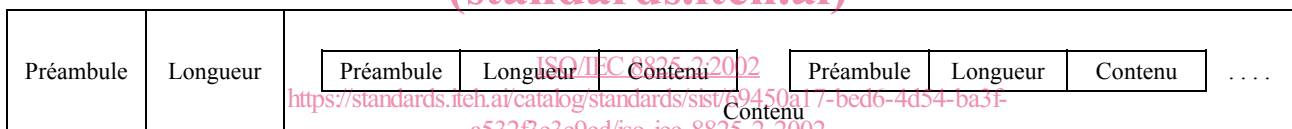
```

INTEGER (0..10, ..., 5)
-- La valeur 5 est codée en tant que valeur racine et pas en tant qu'addition
-- d'extension.
    
```

**9.5 Structure d'une expression codée**

**9.5.1** Les présentes règles de codage spécifient:

- a) le codage d'une valeur simple pour la transformer en liste de champs;
- b) le codage d'une valeur composite pour la transformer en liste de champs, au moyen des listes de champs produites par l'application des présentes règles de codage aux composantes de la valeur composite;
- c) la transformation de la liste des champs de la valeur la plus externe en codage complet de la valeur de la syntaxe abstraite (voir § 10.1).

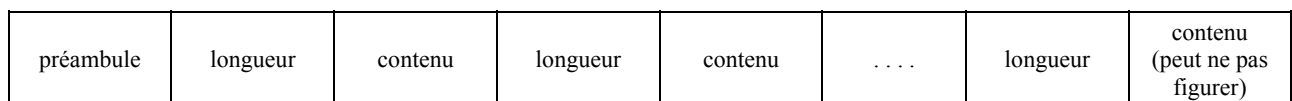


NOTE – Le préambule, la longueur et le contenu sont des "champs" dont la concaténation forme une "liste de champs". La liste de champs d'un type composite différent du type choix (*Choice*) peut être formée par la concaténation de champs de valeur différente. Le préambule, la longueur ou le contenu peuvent 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 les § 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 § 18, 20 et 22), les parties suivantes étant des paires de champs binaires (calés à l'octet suivant la variante ALIGNED), 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.



**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);