



UNION INTERNATIONALE DES TÉLÉCOMMUNICATIONS

UIT-T

SECTEUR DE LA NORMALISATION
DES TÉLÉCOMMUNICATIONS
DE L'UIT

X.209

**INTERCONNEXION DES SYSTÈMES OUVERTS
MODÈLE ET NOTATION**

**SPÉCIFICATION DES RÈGLES DE CODAGE
POUR LA NOTATION DE
SYNTAXE ABSTRAITE NUMÉRO UN (ASN.1)**

Recommandation UIT-T X.209

(Extrait du *Livre Bleu*)

NOTES

1 La Recommandation X.209 de l'UIT-T a été publiée dans le fascicule VIII.4 du Livre Bleu. Ce fichier est un extrait du Livre Bleu. La présentation peut en être légèrement différente, mais le contenu est identique à celui du Livre Bleu et les conditions en matière de droits d'auteur restent inchangées (voir plus loin).

2 Dans la présente Recommandation, le terme «Administration» désigne indifféremment une administration de télécommunication ou une exploitation reconnue.

© UIT 1988, 1993

Droits de reproduction réservés. 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'UIT.

Recommandation X.209

SPÉCIFICATION DES RÈGLES DE CODAGE POUR LA NOTATION DE SYNTAXE ABSTRAITE NUMÉRO UN (ASN.1)¹⁾

(Melbourne, 1988)

Le CCITT,

considérant

- (a) la diversité et la complexité des objets informationnels véhiculés dans la Couche Application;
- (b) la nécessité d'une notation de haut niveau pour spécifier ces objets informationnels;
- (c) l'intérêt de définir et de normaliser les règles de codage de tels objets informationnels,

recommande à l'unanimité

les règles de codage des objets informationnels définies dans la présente Recommandation.

SOMMAIRE

0	<i>Introduction</i>
1	<i>Objet et domaine d'application</i>
2	<i>Références</i>
3	<i>Définitions</i>
4	<i>Abréviations et notation</i>
4.1	Abréviations
4.2	Notation
5	<i>Conformité</i>
6	<i>Règles générales</i>
6.1	Structure d'un codage
6.2	Octets de l'identificateur
6.3	Octets de longueur
6.4	Octets du contenu
6.5	Octets de fin de contenu
7	<i>Codage d'une valeur de booléen</i>

¹⁾ La Recommandation X.209 et la norme ISO 8825, ["Système de traitement de l'information – Interconnexion des systèmes ouverts – Spécification des règles de codage par la notation de syntaxe abstraite numéro un (ASN.1)"], telle que développée par son addendum 1, ont été établies en étroite collaboration et sont techniquement alignées.

- 8 *Codage d'une valeur d'entier*
- 9 *Codage d'une valeur d'énuméré*
- 10 *Codage d'une valeur de réel*
- 11 *Codage d'une valeur de chaîne binaire*
- 12 *Codage d'une valeur de chaîne d'octets*
- 13 *Codage d'une valeur vide*
- 14 *Codage d'une valeur de séquence*
- 15 *Codage d'une valeur de séquence-de*
- 16 *Codage d'une valeur d'ensemble*
- 17 *Codage d'une valeur d'ensemble-de*
- 18 *Codage d'une valeur de choix*
- 19 *Codage d'une valeur de sélection*
- 20 *Codage d'une valeur d'étiqueté*
- 21 *Codage d'une valeur de type Quelconque*
- 22 *Codage d'une valeur d'identificateur d'objet*
- 23 *Codage d'une valeur des types chaîne de caractères*
- 24 *Codage des valeurs des types utiles de la notation ASN.1*
- 25 *Utilisation dans la définition de la syntaxe de transfert*

Appendice I – Exemples de codage

- I.1 Description ASN.1 de la structure d'un enregistrement
- I.2 Description ASN.1 de la valeur d'un enregistrement
- I.3 Représentation de la valeur de cet enregistrement

Appendice II – Affectation des valeurs d'un identificateur d'objet

Appendice III – Illustration du codage d'une valeur de type réel

0 Introduction

La Recommandation X.208 (Spécification de la notation de syntaxe abstraite numéro un) spécifie une notation de définition de syntaxes abstraites, permettant aux spécifications de la Couche Application de définir les types d'informations nécessaires à réaliser un transfert en utilisant le service de présentation. Elle définit également une notation de spécification des valeurs d'un type défini.

La présente Recommandation définit un ensemble de règles de codage qui peuvent s'appliquer aux valeurs des types définis en utilisant la notation spécifiée dans la Recommandation X.208. L'application de ces règles de codage produit une syntaxe de transfert pour ces valeurs. Il est sous-entendu implicitement dans la spécification de ces règles de codage, qu'elles doivent également être utilisées pour le décodage.

Plusieurs ensembles de règles de codage peuvent être appliqués aux valeurs des types qui sont définis en utilisant la notation de la Recommandation X.208. La présente Recommandation définit un ensemble de règles de codage, appelées règles de codage de base.

La présente Recommandation est alignée, sur le plan technique et éditorial, sur la norme internationale ISO 8825, complétée de son additif F.

L'appendice I donne des exemples d'application des règles de codage. Il ne fait pas partie de la présente Recommandation.

L'appendice II résume les affectations des valeurs d'identificateur d'objet stipulées dans la présente Recommandation. Il ne fait pas partie de cette Recommandation.

L'appendice III, qui ne fait pas partie de la présente Recommandation, donne des exemples d'application des règles de codage des réels.

1 Objet et domaine d'application

La présente Recommandation spécifie un ensemble de règles de codage de base qui peuvent être utilisées pour dériver les spécifications d'une syntaxe de transfert des valeurs de type définies en utilisant la notation spécifiée dans la Recommandation X.208. Ces règles de codage de base doivent également être appliquées au décodage d'une telle syntaxe de transfert, pour identifier les valeurs de données transférées.

Ces règles de codage de base sont utilisées au moment de la communication (par le fournisseur du service de présentation, lorsque cela est requis par le contexte de présentation).

2 Références

- [1] Recommandation X. 200, *Modèle de référence pour l'interconnexion des systèmes ouverts pour les applications du CCITT* (voir aussi la norme ISO 7498).
- [2] Recommandation X. 208, *Spécification de la syntaxe abstraite numéro un (ASN.1)* (voir aussi la norme ISO 8824).
- [3] Recommandation X. 226, *Spécification du protocole de présentation (interconnexion des systèmes ouverts) pour les applications du CCITT* (voir aussi la norme ISO 8823).
- [4] ISO 2022, *Traitement de l'information – Jeux ISO de caractères codés à sept et huit éléments – Techniques d'extension de code.*
- [5] ISO 2375, *Traitement de l'information – Procédures pour l'enregistrement des séquences d'échappement.*
- [6] ISO 6093, *Traitement de l'information – Représentation des valeurs numériques dans les chaînes de caractères pour l'échange d'information.*

3 Définitions

Les définitions de la Recommandation X.208 sont utilisées dans la présente Recommandation.

3.1 conformité dynamique

Enoncé du fait qu'une réalisation doit se conformer au comportement prescrit par la présente Recommandation au cours d'une communication.

3.2 conformité statique

Déclaration du fait qu'une réalisation doit présenter un ensemble valide de caractéristiques, parmi celles définies par la présente Recommandation.

3.3 valeur de données

Information spécifiée comme valeur d'un type; le type et la valeur sont définis en ASN.1.

3.4 codage (d'une valeur de données)

Séquence complète des octets utilisés pour représenter la valeur de données.

Remarque – Certaines Recommandations du CCITT utilisent l'expression "élément de données" pour désigner cette séquence d'octets; cette expression n'est pas utilisée dans la présente Recommandation, car d'autres normes internationales ISO l'utilisent pour désigner une "valeur de données".

3.5 octets de l'identificateur

Partie du codage d'une valeur de données qui sert à identifier le type de la valeur.

3.6 octets de longueur

Partie du codage d'une valeur de données placée à la suite des octets de l'identificateur, et servant à déterminer la longueur du codage.

3.7 octets de fin-de-contenu

Partie du codage d'une valeur de données, apparaissant à la fin de celle-ci et servant à indiquer la fin du codage.

Remarque – Tous les codages ne nécessitent pas d'octets de fin-de-contenu.

3.8 octets de contenu

Partie du codage d'une valeur de données, qui représente une valeur particulière, la distinguant des autres valeurs du même type.

3.9 codage primitif

Codage d'une valeur de données dans lequel les octets de contenu représentent directement la valeur.

3.10 codage constructeur

Codage d'une valeur de données dans lequel les octets de contenu sont le codage complet d'une ou plusieurs valeurs de données.

3.11 émetteur

Réalisation codant une valeur de données afin de la transférer.

3.12 destinataire

Réalisation décodant des octets produits par un émetteur pour déterminer la valeur de données qui a été codée.

4 Abréviations et notation

4.1 Abréviations

ASN.1 Notation de syntaxe abstraite numéro un
(*Abstract Syntax Notation One*)

4.2 Notation

4.2.1 La présente Recommandation se réfère à la notation définie par la Recommandation X.208.

4.2.2 La présente Recommandation spécifie la valeur de chaque octet d'un codage et utilise les expressions "bit de plus fort poids" et "bit de plus faible poids".

Remarque – Les spécifications des couches de niveaux inférieurs utilisent la même notation pour définir l'ordre de transmission des bits sur une ligne série ou les affectations des bits à des canaux parallèles.

4.2.3 Pour les applications de la présente Recommandation, les bits d'un octet sont numérotés de 8 à 1, le bit 8 étant le "bit de plus fort poids" et le bit 1 étant le "bit de plus faible poids".

5 Conformité

5.1 La conformité dynamique est spécifiée par les § 6 à 24 compris.

5.2 La conformité statique est spécifiée par les documents qui précisent l'application de ces règles de codage de base.

5.3 D'autres règles de codage sont autorisées par la présente Recommandation sur option de l'émetteur; pour être conforme, le destinataire doit accepter toutes les règles possibles.

Remarque – Des exemples d'autres règles de codage possibles figurent au § 6.3.2 b) et dans le tableau 2/X.209.

6 Règles générales de codage

6.1 Structure d'un codage

6.1.1 Le codage d'une valeur de données est constitué de quatre composants qui doivent apparaître dans l'ordre suivant:

- a) octets de l'identificateur (voir § 6.2);
- b) octets de longueur (voir § 6.3);
- c) octets de contenu (voir § 6.4);
- d) octets de fin-de-contenu (voir § 6.5).

6.1.2 Les octets de fin-de-contenu ne doivent figurer que lorsque la valeur des octets de longueur exige leur présence (voir § 6.3).

6.1.3 La figure 1/X.209 présente la structure d'un codage (primitif ou constructeur). La figure 2/X.209 présente un autre codage possible de type constructeur.

6.2 Octets de l'identificateur

6.2.1 Les octets de l'identificateur codent l'étiquette ASN.1 (classe et numéro) du type de la valeur de données.

6.2.2 Pour les étiquettes désignées par un numéro de la plage 0 à 30 (inclus), les octets de l'identificateur sont composés d'un seul octet codé comme suit:

- a) les bits 8 et 7 sont codés en sorte de représenter la classe de l'étiquette, comme spécifié dans le tableau 1/X.209;
- b) le bit 6 est à zéro ou un, conformément aux règles du § 6.2.5;
- c) les bits 5 à 1 codent le numéro de l'étiquette, comme entier binaire dont le bit 5 est le plus significatif.

6.2.3 La figure 3/X.209 présente la forme d'un octet d'identificateur d'un type dont l'étiquette a un numéro de la plage 0 à 30 (inclus).

6.2.4 Pour les étiquettes de numéros supérieur ou égal à 31, l'identificateur est composé d'un octet de début, suivi d'un ou de plusieurs autres octets.

TABLEAU 1/X.209

Codage de la classe de l'étiquette

Classe	Bit 8	Bit 7
Universelle	0	0
Particulière à une application	0	1
Spécifique au contexte	1	0
A usage privé	1	1

6.2.4.1 L'octet de début est codé comme suit:

- a) les bits 8 et 7 sont codés pour représenter la classe de l'étiquette, comme indiqué dans le tableau 1/X.209;
- b) le bit 6 est à zéro ou un, conformément aux règles du § 6.2.5;
- c) les bits 5 à 1 sont codés 11111₂.

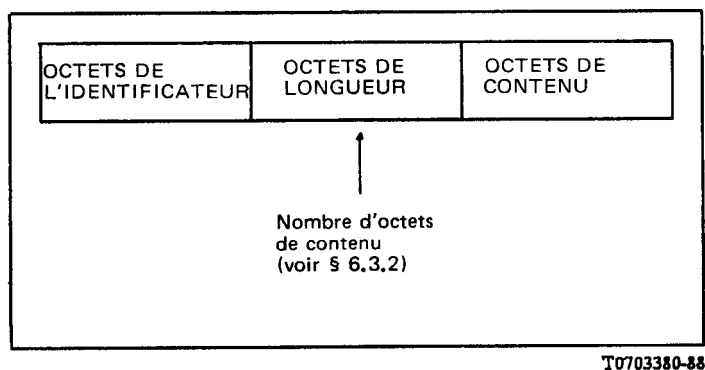


FIGURE 1/X.209
Structure d'un codage

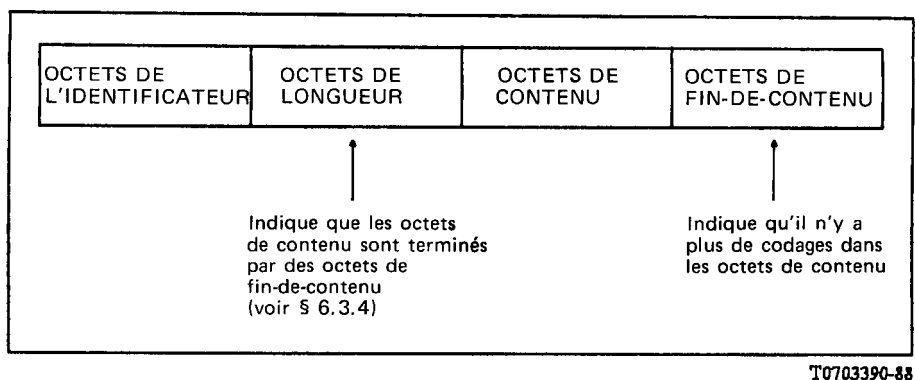


FIGURE 2/X.209
Autre codage possible (constructeur)

6.2.4.2 Les octets suivants doivent coder le numéro d'étiquette comme suit:

- a) le bit 8 de chaque octet doit être mis à un, sauf s'il est le dernier octet de l'identificateur;
- b) les bits 7 à 1 du premier octet suivant, suivis des bits 7 à 1 du deuxième octet suivant, suivis à leur tour des bits 7 à 1 de chacun des octets suivants, jusque et y compris le dernier des octets de l'identificateur, doivent être le codage d'un entier binaire non signé égal au numéro de l'étiquette, le bit 7 du premier octet suivant étant le bit de plus fort poids;
- c) les bits 7 à 1 du premier octet suivant doivent tous être à zéro.

6.2.4.3 La figure 4/X.209 présente la structure des octets d'identificateur pour un type ayant une étiquette dont le numéro est supérieur à 30.

6.2.5 Le bit 6 doit être mis à zéro si le codage est de forme primitif; il doit être mis à un si le codage est de forme constructeur.

Remarque – Les paragraphes suivants précisent si le codage de chaque type est primitif ou constructeur.

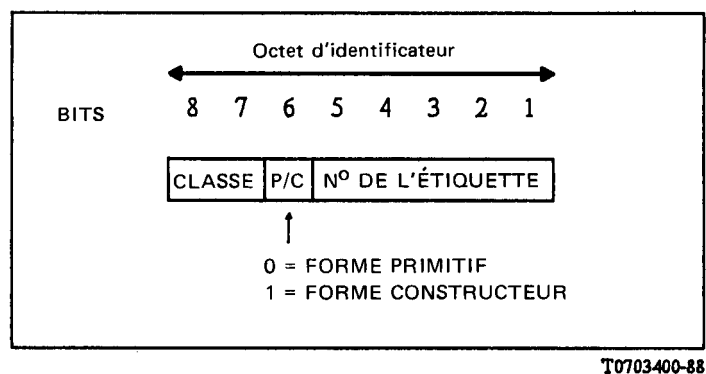


FIGURE 3/X.209

Octet de l'identificateur (étiquette de faible valeur)

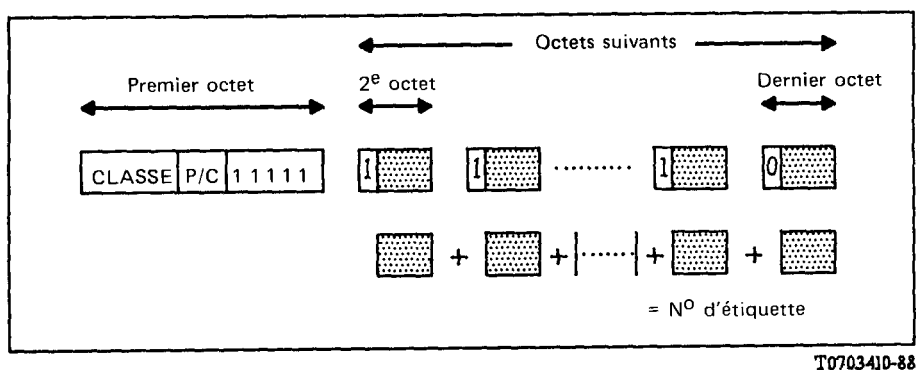


FIGURE 4/X.209

Octets de l'identificateur (étiquette de valeur élevée)

6.2.6 La Recommandation X.208 spécifie que l'étiquette d'un type défini en utilisant le mot clé "CHOICE" prend la valeur de l'étiquette du type auquel appartient la valeur de données choisie.

6.2.7 La Recommandation X.208 spécifie que l'étiquette d'un type défini en utilisant "ANY" est indéterminée. Le type "ANY" est par voie de conséquence défini comme un type ASN.1: le codage complet est alors identique à celui d'une valeur du type affecté (y compris les octets de l'identificateur).

6.3 Octets de longueur

6.3.1 Deux formes d'octets de longueur sont spécifiées:

- a) la forme définie (voir § 6.3.3);
- b) la forme indéfinie (voir § 6.3.4).

6.3.2 Un émetteur doit:

- a) utiliser la forme définie (§ 6.3.3) si le codage est de forme primitif;
- b) utiliser la forme définie (§ 6.3.3) ou la forme indéfinie (§ 6.3.4), à son choix, si le codage est de forme constructeur et immédiatement entièrement disponible;
- c) utiliser la forme indéfinie (§ 6.3.4) si le codage est de forme constructeur mais n'est pas immédiatement entièrement disponible.

6.3.3 Pour la forme définie, les octets de longueur consistent en un ou plusieurs octets, et représentent le nombre d'octets de contenu, en utilisant au choix de l'émetteur la forme courte (§ 6.3.3.1) ou la forme longue (§ 6.3.3.2).

Remarque – La forme courte ne peut être utilisée que si le nombre des octets de contenu est inférieur ou égal à 127.

6.3.3.1 Dans la forme courte, les octets de longueur sont composés d'un seul octet dans lequel le bit 8 a la valeur zéro et les bits 7 à 1 codent le nombre d'octets de contenu (qui peut être zéro), sous forme d'un entier binaire non signé dont le bit 7 est de plus fort poids.

Exemple:

L = 38 peut être codé 00100110_2

6.3.3.2 Dans la forme longue, les octets de longueur comprennent un octet initial suivi d'un ou plusieurs octets. L'octet initial est codé comme suit:

- a) le bit 8 est à un;
- b) les bits 7 à 1 codent le nombre des octets de longueur suivants, sous la forme d'un entier binaire non signé dont le bit 7 est de plus fort poids;
- c) la valeur 1111111_2 ne doit pas être utilisée.

Remarque – Cette restriction est introduite en vue d'une éventuelle extension ultérieure.

Les bits 8 à 1 du premier octet suivant, suivis des bits 8 à 1 du deuxième octet suivant, suivis également des bits 8 à 1 de chacun des octets suivants, jusque et y compris le dernier octet suivant, doivent être le codage d'un entier binaire non signé égal au nombre d'octets de contenu, le bit 8 du premier octet suivant étant celui de plus fort poids.

Exemple:

L = 201 peut être codé 10000001_2
 11001001_2

Remarque – Dans la forme longue, l'émetteur peut choisir d'utiliser plus d'octets de longueur que le minimum nécessaire.

6.3.4 Dans la forme indéfinie, les octets de longueur indiquent que les octets de contenu sont terminés par des octets de fin-de-contenu (voir § 6.5); ils consistent alors en un seul octet.

6.3.4.1 Ce seul octet doit avoir son bit 8 à un et ses bits 7 à 1 à zéro.

6.3.4.2 Si la forme longue est utilisée, les octets de fin-de-contenu (voir § 6.5) doivent figurer dans le codage à la suite des octets de contenu.

6.4 Octets de contenu

Les octets de contenu consistent en zéro, un ou plusieurs octets et codent la valeur de données comme spécifié dans les paragraphes suivants.

Remarque – Les octets de contenu dépendent du type de la valeur de données; les paragraphes ci-après suivent l'ordre des définitions de types dans ASN.1.

6.5 Octets de fin-de-contenu

Les octets de fin-de-contenu doivent figurer si la longueur est codée comme spécifié au § 6.3.4; sinon, ils ne doivent pas figurer.

Les octets de fin-de-contenu consistent en deux octets zéro.

Remarque – Les octets de fin-de-contenu peuvent être considérés comme le codage d'une valeur universelle dont l'étiquette est de classe universelle, la forme primitif, le numéro d'étiquette zéro et le contenu absent:

Fin-de-contenu	Longueur	Contenu
00_{16}	00_{16}	Absent

7 Codage d'une valeur de booléen

7.1 Le codage d'une valeur de booléen doit être de forme primitif. Les octets de contenu sont composés d'un seul octet.

7.2 Si la valeur de booléen est:

FALSE

l'octet doit être zéro.

7.2.1 Si la valeur de booléen est:

TRUE

l'octet doit avoir n'importe quelle valeur différente de zéro, au choix de l'émetteur.

Exemple – Si elle est de type BOOLEAN, la valeur TRUE peut être codée:

Booléen	Longueur	Contenu
01 ₁₆	01 ₁₆	FF ₁₆

8 Codage d'une valeur d'entier

8.1 Le codage d'une valeur d'entier est de forme primitif. Les octets de contenu consistent en un ou plusieurs octets.

8.2 Si les octets de contenu d'une valeur d'entier sont composés de plus d'un octet, les bits du premier octet et le bit 8 du deuxième octet:

- a) ne doivent pas être tous des uns;
- b) ne doivent pas être tous des zéros.

Remarque – Ces règles garantissent qu'une valeur d'entier est toujours codée dans le plus petit nombre d'octets possible.

8.3 Les octets du contenu doivent être un nombre binaire en complément à deux, égal à la valeur d'un entier et composé des bits 8 à 1 du premier octet, suivis des bits 8 à 1 du deuxième octet, suivis des bits 8 à 1 de chaque octet suivant, jusque et y compris le dernier des octets de contenu.

Remarque – La valeur d'un nombre binaire en complément à deux, est obtenue en numérotant les bits des octets de contenu, à partir du bit 1 du dernier octet (bit 0) jusqu'au bit 8 du premier octet. A chaque bit est affectée une valeur numérique de 2^N , N étant sa position dans la séquence numérotée précédente. La valeur du nombre binaire en complément à 2 est obtenue en faisant la somme des valeurs numériques affectées à chacun des bits mis à un (sauf le bit 8 du premier octet) de laquelle on soustrait la valeur affectée au bit 8 du premier octet, s'il est à un.

9 Codage d'une valeur d'énuméré

9.1 Le codage d'une valeur d'énuméré est celui de la valeur d'entier à laquelle il est associé.

10 Codage d'une valeur de réel

10.1 Le codage d'une valeur de réel est primitif.

10.2 Si la valeur de réel est la valeur zéro, aucun octet de contenu ne doit figurer dans le codage.

10.3 Si la valeur de réel est différente de zéro, il faut utiliser pour le codage une base B' choisie par l'émetteur. Si B' est 2, 8 ou 16, il faut utiliser un codage binaire, spécifié au § 10.5. Si B' est 10, il faut utiliser un codage par caractère, spécifié au § 10.6.

Remarque – La forme utilisée pour l'enregistrement, la génération ou le traitement par les émetteurs ou les destinataires et la forme utilisée dans la notation de valeur ASN.1, sont toutes indépendantes de la base utilisée pour le transfert.

10.4 Le bit 8 du premier octet de contenu doit être positionné comme suit:

- a) si le bit 8 = 1, le codage binaire spécifié au § 10.5 s'applique;
- b) si le bit 8 = 0 et le bit 7 = 0, le codage décimal spécifié au § 10.6 s'applique;
- c) si le bit 8 = 0 et le bit 7 = 1, alors une “valeur-réel-spécial” (voir Recommandation X.208) est codée comme au § 10.7.

10.5 Quand un codage binaire est utilisé (bit 8 = 1), si la mantisse, M, est différente de zéro, il doit être représenté par un signe S, une valeur entière non négative N et un facteur d'échelle binaire F:

$$M = S \times N \times 2^F, 0 \leq F < 4, S = +1 \text{ ou } -1$$

Remarque – Cette liberté de choix de F est destinée à faciliter la génération du format de transfert, en supprimant la nécessité d'aligner la virgule décimale implicite de la mantisse sur une limite d'octet (voir appendice III). L'existence de F ne complique pas de façon sensible la tâche des destinataires.

10.5.1 Le bit 7 des premiers octets de contenu doit être 1 si S est –1, et 0 autrement.

10.5.2 Les bits 6 et 5 du premier octet de contenu codent la valeur de la base B' comme suit:

Bits 6 et 5	Base
00	base 2
01	base 8
10	base 16
11	Réservé pour des versions ultérieures de la présente Recommandation

10.5.3 Les bits 3 et 4 du premier octet de contenu codent la valeur du facteur d'échelle binaire, sous la forme d'un entier binaire non signé.

10.5.4 Les bits 2 et 1 du premier octet de contenu codent le format de l'exposant comme suit:

- a) si les bits 2 et 1 sont 00, le deuxième octet de contenu code la valeur de l'exposant sous la forme d'un nombre binaire en complément à deux;
- b) si les bits 2 et 1 sont 01, le deuxième et le troisième octet de contenu codent la valeur de l'exposant sous la forme d'un nombre binaire en complément à deux;
- c) si les bits 2 et 1 sont 10, le deuxième, le troisième et le quatrième octet de contenu codent la valeur de l'exposant sous la forme d'un nombre binaire en complément à deux;
- d) si les bits 2 et 1 sont 11, le deuxième octet de contenu code le nombre d'octets, disons X, (sous la forme d'un nombre binaire non signé) utilisé pour coder la valeur de l'exposant, et les troisième à (X plus 3)^e (compris) octets de contenu codent la valeur de l'exposant sous la forme d'un nombre binaire en complément à deux; la valeur de X doit être au moins égale à un, les neuf premiers bits de l'exposant transmis ne doivent pas être tous des zéros ou tous des uns.

10.5.5 Les octets de contenu restants codent la valeur de l'entier N (voir le § 10.5) sous la forme d'un nombre binaire non signé.

Remarque 1 – Ce codage ne spécifie pas une représentation “normalisée” car plusieurs représentations sont possibles pour chaque valeur (sauf zéro). Ces variantes sont au choix de l'émetteur et peuvent être utilisées comme indication grossière de la précision.

Remarque 2 – Cette représentation des nombres réels est très différente des formats normalement utilisés dans les matériels de virgule flottante, mais elle est conçue pour une conversion facile dans ou à partir de tels formats (voir appendice III).

10.6 Quand le codage décimal est utilisé (bits 8 et 7 = 00), tous les octets de contenu qui suivent le premier octet constituent un champ, selon le terme utilisé dans la norme ISO 6093, de longueur choisie par l'émetteur et codé conformément à cette norme. La forme de représentation des nombres, telle que définie dans la norme ISO 6093, est spécifiée par les bits 6 à 1 du premier octet de contenu:

Bits 6 à 1	Représentation des nombres
00 0001	Forme NR1 ISO 6093
00 0010	Forme NR2 ISO 6093
00 0011	Forme NR3 ISO 6093

Les valeurs restantes des bits 6 à 1 sont réservées pour des versions ultérieures de la présente Recommandation.

Remarque 1 – Les recommandations de la norme ISO 6093 concernant l'utilisation d'au moins un chiffre à gauche du point décimal sont également formulées dans la présente Recommandation, mais ne sont pas obligatoires.

Remarque 2 – La documentation associée ne doit pas spécifier de facteur d'échelle (voir ISO 6093).

Remarque 3 – L'utilisation de la forme normalisée (voir ISO 6093) est au choix de l'émetteur et est sans importance.

10.7 Dans le cas où il faut coder des "Valeur-Réel-Spécial" (bit 8 à 7 = 01), l'octet de contenu est unique et peut prendre les valeurs suivantes:

01000000 la valeur est PLUS-INFINITY
 01000001 la valeur est MINUS-INFINITY

Toutes les autres valeurs dont les bits 8 et 7 sont égaux à 01, sont réservées pour des versions ultérieures de la présente Recommandation.

11 Codage d'une valeur de chaîne binaire

11.1 Le codage d'une valeur de chaîne binaire est de forme primitif ou constructeur, au choix de l'émetteur.

Remarque – Pour transférer une partie d'une chaîne binaire, lorsque la chaîne complète n'est pas encore disponible, il faut utiliser le codage constructeur.

11.2 Les octets de contenu du codage primitif contiennent un octet initial suivi de zéro, un ou plusieurs octets.

11.2.1 Les bits de la chaîne, à partir du premier bit et en allant jusqu'au dernier, doivent être placés dans les bits 8 à 1 du premier octet de contenu, suivis des bits 8 à 1 du deuxième octet, suivis des bits 8 à 1 de chacun des octets suivants, suivis d'autant de bits que nécessaire dans le dernier octet, en commençant par le bit 8.

Remarque – Les notations "premier bit" et "bit de fin" sont spécifiées dans la Recommandation X. 208.

11.2.2 Le premier octet code, sous la forme d'un entier binaire non signé dont le bit 1 est de plus faible poids, le nombre de bits non utilisés dans l'octet final. Ce nombre est de la plage 0 à 7.

11.2.3 Si la chaîne binaire est vide, l'octet initial doit être zéro et doit être suivi d'aucun octet.

11.3 Les octets de contenu du codage constructeur sont composés du codage complet de zéro, une ou plusieurs valeurs de données.

Remarque – Dans la forme constructeur, chacun de ces codages comprend des octets d'identificateur, de longueur, de contenu, et peut comprendre des octets de fin-de-contenu.

11.3.1 Dans les octets de contenu, le codage de chaque valeur de données est le codage d'une valeur du type chaîne binaire.

Remarque – En particulier, les étiquettes des octets de contenu sont toujours de classe universelle, numéro 3.

11.3.2 Les bits de la valeur de la chaîne binaire qui est codée, sont placés (en commençant par le premier bit et en allant jusqu'au bit de fin) comme suit: du premier bit au bit de fin de la première valeur de données codée des octets de contenu, suivis du premier bit au bit de fin de la seconde valeur codée des octets de contenu, suivis du premier bit au bit de fin de chacune des valeurs de données successives, suivis du premier bit au bit de fin de la dernière valeur de données codée des octets de contenu.

11.3.3 Chaque valeur de données codée des octets de contenu, sauf la dernière, est composée d'un nombre de bits qui est un multiple de huit.

Remarque – Une valeur de données codée dans les octets de contenu peut être une chaîne binaire de longueur zéro.

11.3.4 Avec un codage de forme constructeur, aucune signification ne doit être accordée à la limite entre les valeurs de données codées des octets de contenu.

11.3.5 Le codage de chaque valeur de données codée des octets de contenu peut être de forme primitif ou constructeur.

Remarque – Il est en général de forme primitif.

Exemple – Dans le cas d'un type chaîne binaire, la valeur '0A3B5F291CD'H peut être codée comme suit (codage de forme primitif):

Chaîne binaire	Longueur	Contenu
03 ₁₆	07 ₁₆	040A3B5F291CD0 ₁₆

La valeur ci-dessus peut également être codée comme suit (codage de forme constructeur):

Chaîne binaire	Longueur	Contenu
23 ₁₆	80 ₁₆	
Chaîne binaire	Longueur	Contenu
03 ₁₆	03 ₁₆	000A3B ₁₆
Chaîne binaire	Longueur	Contenu
03 ₁₆	05 ₁₆	045F291CD0 ₁₆
EOC	Longueur	
00 ₁₆	00 ₁₆	

12 Codage d'une valeur de chaîne d'octets

12.1 Le codage d'une valeur de chaîne d'octets est de forme primitif ou constructeur, au choix de l'émetteur.

Remarque – Pour transférer une partie d'une chaîne d'octets, lorsque la chaîne complète n'est pas encore disponible, il faut utiliser le codage constructeur.

12.2 Le codage de forme primitif contient zéro, un ou plusieurs octets, de valeurs égales à celles des octets de la valeur de données, dans l'ordre dans lequel ils apparaissent dans cette valeur de données, le bit de plus fort poids d'un octet de la valeur de données correspondant au bit de plus fort poids d'un octet de contenu.

12.3 Dans le codage de forme constructeur, les octets de contenu sont composés du codage complet de zéro, une ou plusieurs valeurs de données.

Remarque – Dans la forme constructeur, chacun de ces codages comprend des octets d'identificateur, de longueur, de contenu et peut comprendre des octets de fin-de-contenu.

12.3.1 Chaque codage de valeur de données des octets de contenu est le codage d'une valeur du type chaîne d'octets.

Remarque – En particulier, les étiquettes des octets de contenu sont toujours de classe universelle, numéro 4.

12.3.2 Les octets de valeur de la chaîne d'octets qui est codée, sont placés (en commençant par le premier octet et en allant jusqu'au dernier octet) comme suit: du premier au dernier octet de la première valeur de données codée des octets de contenu, suivis du premier au dernier octet de la deuxième valeur de données codée des octets de contenu, suivis du premier au dernier octet de chaque valeur de données successive, suivis du premier au dernier octet de la dernière valeur de données codée des octets de contenu.

Remarque – Une valeur de données codée des octets de contenu peut être une chaîne d'octets de longueur zéro.

12.3.3 Avec un codage de forme constructeur, les limites séparant les valeurs de données des octets de contenu n'ont pas de signification particulière.

12.3.4 Le codage de chaque valeur de données codée dans les octets de contenu peut être primitif ou constructeur.

Remarque – Il est en général primitif.

13 Codage d'une valeur vide

13.1 Le codage d'une valeur vide est de forme primitif.

13.2 Les octets de contenu ne contiennent aucun octet.

Remarque – L'octet de longueur est zéro.

Exemple: Dans le cas d'un type vide, la valeur peut être codée comme suit:

Vide	Longueur
05 ₁₆	00 ₁₆

14 Codage d'une valeur de séquence

14.1 Le codage d'une valeur de séquence est de forme constructeur.

14.2 Les octets de contenu sont composés du codage complet d'une valeur de données de chacun des types énumérés dans la définition ASN.1 du type séquence, dans l'ordre de leur apparition dans la définition, sauf si le type a été dénoté par le mot clé "OPTIONAL" ou par le mot clé "DEFAULT".

14.3 Le codage d'une valeur de données peut figurer, mais pas nécessairement, pour un type qui a été dénoté par le mot clé "OPTIONAL" ou par le mot clé "DEFAULT". S'il figure, il doit apparaître dans le codage au point correspondant à l'apparition de ce type dans la définition ASN.1.

Exemple: Dans le cas d'un type:

SEQUENCE {nom Chaîne-IA5, ok BOOLEAN}

la valeur:

{nom "Smith", ok TRUE}

peut être codée comme suit:

Séquence	Longueur	Contenu
30 ₁₆	80 ₁₆	
Chaîne IA5	Longueur	Contenu
16 ₁₆	05 ₁₆	"Smith"
Booléen	Longueur	Contenu
01 ₁₆	01 ₁₆	FF ₁₆

15 Codage d'une valeur de séquence-de

15.1 Le codage d'une valeur de séquence-de est de forme constructeur.

15.2 Les octets de contenu sont composés de zéro, un ou plusieurs codages complets des valeurs de données des types indiqués dans la définition ASN.1.

15.3 L'ordre des codages des valeurs de données doit être identique à celui des valeurs de données de la valeur de séquence-de à coder.

16 Codage d'une valeur d'ensemble

16.1 Le codage d'une valeur d'ensemble est de forme constructeur.

16.2 Les octets de contenu sont composés du codage complet d'une valeur de données de chacun des types indiqués dans la définition ASN.1 du type ensemble, dans un ordre choisi par l'émetteur, sauf si le type a été dénoté par le mot clé "OPTIONAL" ou par le mot "DEFAULT".

16.3 Le codage d'une valeur de données peut, mais ne doit pas, être présent pour un type qui a été dénoté par le mot clé "OPTIONAL" ou par le mot clé "DEFAULT".

Remarque – L'ordre des valeurs de données d'une valeur d'ensemble n'est pas significatif et n'impose aucune contrainte sur leur ordre de transfert.

17 Codage d'une valeur d'ensemble-de

17.1 Le codage d'une valeur d'ensemble-de est de forme constructeur.

17.2 Le texte du § 15.2 s'applique.

17.3 L'ordre des valeurs de données doit être conservé par le codage et le décodage subséquents.

18 Codage d'une valeur de choix

Le codage d'une valeur de choix est le même que le codage d'une valeur du type choisi.

Remarque 1 – Le codage peut être de forme primitif ou constructeur, selon le type choisi.

Remarque 2 – L'étiquette utilisée dans les identificateurs d'octet est celle du type choisi, telle que spécifiée dans la définition ASN.1 du type choix.

19 Codage d'une valeur de sélection

Le codage d'une valeur de sélection est le même que celui d'une valeur du type sélectionné.

Remarque – Le codage peut être de forme primitif ou constructeur, selon le type sélectionné.

20 Codage d'une valeur d'étiqueté

20.1 Le codage d'une valeur d'étiqueté dérive du codage complet de la valeur de données correspondante du type apparaissant dans la notation "Type-Etiqueté" (appelé codage de base) tel que spécifié aux § 20.2 et 20.3.

20.2 Si le mot clé "IMPLICIT" n'a pas été utilisé dans la définition du type, le codage doit être de forme constructeur et les octets de contenu doivent constituer le codage de base complet.

20.3 Si le mot clé "IMPLICIT" a été choisi dans la définition du type:

- a) le codage doit être de forme constructeur si le codage de base est de forme constructeur, sinon il doit être primitif;
- b) les octets de contenu doivent être les mêmes que les octets de contenu du codage de base.

Exemple: Avec les définitions de types ASN.1 suivantes:

Type1 ::= Chaîne-Visible
Type2 ::= [APPLICATION 3] IMPLICIT Type1
Type3 ::= [2] Type2
Type4 ::= [APPLICATION 7] IMPLICIT Type3
Type5 ::= [2] IMPLICIT Type2

une valeur:

“Jones”

sera codée comme suit:

Pour le Type-1:

Chaîne-Visible	Longueur	Contenu
1A ₁₆	05 ₁₆	4A6F6E6573 ₁₆

Pour le Type-2:

[Application 3]	Longueur	Contenu
43 ₁₆	05 ₁₆	4A6F6E6573 ₁₆

Pour le Type-3:

[2]	Longueur	Contenu
A2 ₁₆	07 ₁₆	
[Application 3]	Longueur	Contenu
43 ₁₆	05 ₁₆	4A6F6E6573 ₁₆

Pour le Type-4

[Application 7]	Longueur	Contenu
67 ₁₆	07 ₁₆	
[Application 3]	Longueur	Contenu
43 ₁₆	05 ₁₆	4A6F6E6573 ₁₆

Pour le Type-5:

[2]	Longueur	Contenu
82 ₁₆	05 ₁₆	4A6F6E6573 ₁₆

21 Codage d'une valeur de type Quelconque

Le codage d'un type Quelconque doit être le codage complet spécifié dans la présente Recommandation pour le type de la valeur du type Quelconque.

22 Codage d'une valeur d'identificateur d'objet

22.1 Le codage d'une valeur d'identificateur d'objet est de forme primitif.

22.2 Les octets de contenu sont une liste (ordonnée) des codages des sous-identificateurs (voir § 22.3 et 22.4) concaténés.

Chaque sous-identificateur est représenté par une suite d'un ou plusieurs octets. Le bit 8 de chaque octet indique s'il est le dernier de la suite. Le bit 8 du dernier octet est zéro, le bit 8 de chaque octet précédent est un. Les bits 7 à 1 des octets de la suite codent collectivement le sous-identificateur. Conceptuellement, ces groupes de bits sont concaténés sous la forme d'un nombre binaire non signé dont le bit de plus fort poids est le bit 7 du premier octet et dont le bit de plus faible poids est le bit 1 du dernier octet. Le sous-identificateur doit être codé avec le moins d'octets possible, c'est-à-dire que l'octet de début ne doit pas avoir la valeur 80 (en hexadécimal).

22.3 Le nombre de sous-identificateurs (N) doit être inférieur d'une unité au nombre de composants d'identificateur d'objet de la valeur d'identificateur d'objet à coder.

22.4 La valeur numérique du premier sous-identificateur est obtenue à partir des valeurs des deux composants d'identificateur d'objet de la valeur d'identificateur d'objet à coder, en appliquant la formule:

$$(X * 40) + Y$$

où X est la valeur du premier composant de l'identificateur d'objet et Y la valeur du deuxième composant.

Remarque – Ce regroupement des deux premiers composants de l'identificateur d'objet suppose que trois valeurs seulement sont affectées à partir d'une racine, et au moins 39 valeurs aux niveaux suivants, à partir des noeuds atteints par $X = 0$ et $X = 1$.

22.5 La valeur numérique du i ème sous-identificateur ($2 \leq i \leq N$) est celle du $(i + 1)$ ème composant de l'identificateur d'objet.

Exemple: Une valeur d'identificateur d'objet de:

{joint-iso-ccitt 100 3}

qui est la même que:

{2 100 3}

a un premier sous-identificateur de 180 et un second sous-identificateur de 3. Le codage résultant est:

Identificateur d'objet	Longueur	Contenu
06 ₁₆	03 ₁₆	813403 ₁₆

23 Codage des valeurs des types chaîne de caractères

23.1 La valeur de données est composée d'une chaîne de caractères du jeu de caractères spécifié dans la définition de type ASN.1.

23.2 Chaque valeur de données est codée indépendamment des autres valeurs de données de même type.

23.3 Chaque type chaîne de caractères est codé comme s'il a été déclaré:

[UNIVERSAL x] IMPLICIT OCTET STRING

où x est le numéro de l'étiquette de classe universelle affecté au type de la chaîne de caractères dans la Recommandation X.208. La valeur de la chaîne d'octets est spécifiée au § 23.4 et 23.5.

23.4 Lorsqu'un type chaîne de caractères est spécifié dans la Recommandation X.208, par référence directe à un tableau récapitulatif (Chaîne-Numérique et Chaîne-Imprimable), la valeur de la chaîne d'octets est celle spécifiée au § 23.5 pour un type Chaîne-Visible de même valeur de chaîne de caractères.

23.5 La chaîne d'octets contient les octets spécifiés dans la norme ISO 2022 pour les codages à huit éléments, utilisant les codages de séquences d'échappement et de caractères enregistrés conformément à la norme ISO 2375.

23.5.1 Seules peuvent être utilisées les séquences d'échappement spécifiées par un des numéros d'enregistrement utilisés pour définir le type chaîne de caractères dans la Recommandation X.208.

TABLEAU 2/X.209

Utilisation des séquences d'échappement

Type	G0 implicite (numéro d'enregistrement)	C0 et C1 implicites (numéro d'enregistrement)	Séquence(s) d'échappement implicite(s) et remplacement bloquant zéro (LS0) quand applicable	Séquences d'échappement explicites autorisées?
Chaîne-Numérique	2	Aucun	ESC 2/8 4/0 LS0	NON
Chaîne-Imprimable	2	Aucun	ESC 2/8 4/0 LS0	NON
Chaîne-Télétext (Chaîne-T61)	102	106(C0) 107(C1)	ESC 2/8 7/5 LS0 ESC 2/1 4/5 ESC 2/2 4/8	OUI
Chaîne-Vidéotex	102	1(C0) 73(C1)	ESC 2/8 7/5 LS0 ESC 2/1 4/0 ESC 2/2 4/1	OUI
Chaîne-Visible (Chaîne-ISO646)	2	Aucun	ESC 2/8 4/0 LS0	NON
Chaîne-IA5	2	1(C0)	ESC 2/8 4/0 LS0 ESC 2/1 4/0	NON
Chaîne-Graphique	2	Aucun	ESC 2/8 4/0 LS0	OUI
Chaîne-Générale	2	1(C0)	ESC 2/8 4/0 LS0 ESC 2/1	OUI

Remarque – Beaucoup de caractères d'usage courant (par exemple, A à Z) apparaissent dans un certain nombre de répertoire de caractères avec des numéros d'enregistrement et des séquences d'échappement individuelles. Quand les types ASN.1 autorisent les séquences d'échappement, plusieurs codages sont possibles pour une chaîne de caractères donnée (voir également le § 5.3).

23.5.2 Au début de chaque chaîne, certains numéros d'enregistrement doivent être implicitement désignés par G0 et/ou C0 et/ou C1 et appelés (selon la terminologie de la norme ISO 2022). Le tableau 2/X.209 spécifie ces numéros d'enregistrement pour chaque type, avec les séquences d'échappement implicites qu'ils impliquent.

23.5.3 Certains types de chaînes de caractères ne doivent pas contenir de séquences d'échappement explicites dans leurs codages; dans tous les autres cas, toute séquence d'échappement autorisée par le § 23.5.1 peut apparaître à tout moment, y compris au début du codage. Le tableau 2/X.209 donne la liste des types pour lesquels sont autorisées les séquences d'échappement explicites.

23.5.4 L'utilisation des avertisseurs est interdite, sauf autorisation explicite de l'utilisateur d'ASN.1.

Remarque – Le choix d'un type ASN.1 fournit une forme limitée de la fonction d'avertisseur. Des protocoles d'application spécifiques peuvent véhiculer des avertisseurs dans d'autres éléments de protocole, ou spécifier en détail la façon d'utiliser les avertisseurs.

Exemple: Avec définition de type ASN.1:

Nom ::= Chaîne-Visible

une valeur:

“Jones”

peut être codée comme suit (forme primitif):

Chaîne-Visible	Longueur	Contenu
1A ₁₆	05 ₁₆	4A6F6E6573 ₁₆

ou (forme constructeur, longueur définie), comme:

Chaîne-Visible	Longueur	Contenu
3A ₁₆	09 ₁₆	
Chaîne-Octets	Longueur	Contenu
04 ₁₆	03 ₁₆	4A6F6E ₁₆
Chaîne-Octets	Longueur	Contenu
04 ₁₆	02 ₁₆	6573 ₁₆

ou (forme constructeur, longueur indéfinie), comme:

Chaîne-Visible	Longueur	Contenu
3A ₁₆	80 ₁₆	
Chaîne-Octets	Longueur	Contenu
04 ₁₆	03 ₁₆	4A6F6E ₁₆
Chaîne-Octets	Longueur	Contenu
04 ₁₆	02 ₁₆	6573 ₁₆
EOC	Longueur	
00 ₁₆	00 ₁₆	

L'exemple ci-dessus illustre trois des (nombreuses) formes possibles, au choix de l'émetteur. Les destinataires doivent traiter toutes les formes autorisées (voir le § 5.3).

24 Codage des valeurs des types utiles ASN.1

La Recommandation X.208 donne une définition de ces types en ASN.1. Le codage est obtenu en appliquant à cette définition de types les règles spécifiées dans la présente Recommandation.

25 Utilisation dans une définition de syntaxe de transfert

25.1 Il est possible de se référer aux règles de codage spécifiées dans la présente Recommandation, et de les appliquer chaque fois qu'il est nécessaire de spécifier une représentation en chaîne d'octets non ambiguë, non divisée et auto-délimitante de toutes les valeurs d'un type unique ASN.1.

Remarque - Toutes ces chaînes d'octets sont non ambiguës dans le cas d'un type unique ASN.1. Elles ne le seraient pas nécessairement si elles étaient combinées avec des codages d'un type ASN.1 différent.

25.2 Les valeurs d'identificateur d'objet et de descripteur d'objet:

{joint-iso-ccitt asn1 (1) codage-base (1)}

et

“Codage de base d'un seul type ASN.1”

sont affectées à l'identification et à la description des règles de codage spécifiées dans la présente norme internationale.

25.3 Lorsqu'une spécification d'application définit une syntaxe abstraite comme un ensemble de valeurs de données de présentation, dont chacun est une valeur d'un type ASN.1 nommé – en général (mais pas nécessairement) un type choix – la valeur d'identificateur d'objet spécifiée au § 25.2 peut être utilisée avec le nom de la syntaxe abstraite pour identifier la syntaxe de transfert résultant de l'application des règles de codage spécifiées dans la présente Recommandation, au type ASN.1 nommé utilisé dans la définition de la syntaxe abstraite.

Remarque – En particulier, cette identification des règles de codage peut apparaître dans le champ des paramètres “nom de la syntaxe de transfert” du protocole de présentation (Recommandation X.226).

25.4 Le nom spécifié au § 25.2 ne doit pas être utilisé avec un nom de syntaxe abstraite pour identifier une syntaxe de transfert, si les conditions définies au § 25.3 pour la définition de la syntaxe abstraite ne sont pas remplies.

APPENDICE I

(à la Recommandation X.209)

Exemples de codages

Cet appendice illustre les règles de codage de base spécifiées dans la présente Recommandation, en montrant la représentation en octets d'un enregistrement “salarié” (fictif), qui est défini en ASN.1.

I.1 Description ASN.1 de la structure de l'enregistrement

La structure de l'enregistrement fictif (salarié) est décrite formellement ci-après, en utilisant la notation spécifiée dans la Recommandation X.208 pour définir des types.

```
Enregistrement-Salarié ::= [APPLICATION 0] IMPLICIT SET
{
  fonction          [0] Chaîne-Visible
  matricule         Matricule-Salarié
  Date-engagement  [1] Date
  Nom-conjoint     [2] Nom
  Enfants          [3] IMPLICIT
                  SEQUENCE OF Renseignements-Enfant
                  DEFAULT {} }

Renseignements-Enfant ::= SET
{
  date-Naissance   [0] Date }

Nom ::= [APPLICATION 1] IMPLICIT SEQUENCE
{ Prénom          Chaîne-Visible
  Initiale        Chaîne-Visible
  nom-Famille     Chaîne-Visible }

Matricule-Employé ::= [APPLICATION 2] IMPLICIT INTEGER

Date ::= [APPLICATION 3] IMPLICIT Chaîne-Visible
--YYYYMMDD
```

I.2 Description ASN.1 d'une valeur de l'enregistrement

La valeur de l'enregistrement “salarié” de John Smith est décrite formellement ci-dessous en utilisant ASN.10.

```
{
  fonction          {prénom “John”,initiale “P”,nom-Famille “Smith”}
  matricule         “Directeur”
  date-Engagement  51
  nom-Conjoint     “19710917”
  enfants          {prénom “Mary”,initiale “T”,nom-Famille“Smith”}
                  {{prénom “Ralph”,initiale “T”,nom-Famille “Smith”}
                   date-Naissance “19571111”}
                   {{prénom “Susan”,initiale “B”,nom-Famille “Jones”}
                    date-Naissance “19590717” }}}}
```

I.3 Représentation de la valeur de cet enregistrement

La représentation en octets de la valeur d'enregistrement donnée ci-dessus (après application des règles de codage de base définies dans la présente Recommandation) est présentée ci-après. Les valeurs des identificateurs, les longueurs, et les contenus des entiers sont indiqués en hexadécimal, avec deux chiffres hexadécimaux par octet. Les valeurs du contenu des chaînes de caractères sont représentées comme un texte, avec un caractère par octet.

Enregistrement «salarié» 60	Longueur 8185	Contenu							
		Nom	Longueur	Contenu					
		61	10						
				Chaîne-Visible	Longueur	Contenu			
				1A	04	"John"			
				Chaîne-Visible	Longueur	Contenu			
				1A	01	"P"			
				Chaîne-Visible	Longueur	Contenu			
				1A	05	"Smith"			
		Fonction	Longueur	Contenu					
		A0	0A						
				Chaîne-Visible	Longueur	Contenu			
				1A	08	"Directeur"			
		Matricule	Longueur	Contenu					
		42	01	33					
		Date engagement	Longueur	Contenu					
		A1	0A						
				Date	Longueur	Contenu			
				43	08	"19710917"			
		Nom conjoint	Longueur	Contenu					
		A2	12						
				Nom	Longueur	Contenu			
				61	10				
				Chaîne-Visible	Longueur	Contenu			
				1A	04	"Mary"			
				Chaîne-Visible	Longueur	Contenu			
				1A	01	"T"			
				Chaîne-Visible	Longueur	Contenu			
				1A	05	"Smith"			
		Fonction	Longueur	Contenu					
		A3	42						
				Ensemble	Longueur	Contenu			
				31	1F				
				Nom	Longueur	Contenu			
				61	11				
				Chaîne-Visible	Longueur	Contenu			
				1A	05	"Ralph"			
				Chaîne-Visible	Longueur	Contenu			
				1A	01	"T"			
				Chaîne-Visible	Longueur	Contenu			
				1A	05	"Smith"			
				Date naissance	Longueur	Contenu			
				A0	0A				
				Date	Longueur	Contenu			
				43	08	"19571111"			
				Ensemble	Longueur	Contenu			
				31	1F				
				Nom	Longueur	Contenu			
				61	11				
				Chaîne-Visible	Longueur	Contenu			
				1A	05	"Susan"			
				Chaîne-Visible	Longueur	Contenu			
				1A	01	"B"			
				Chaîne-Visible	Longueur	Contenu			
				1A	05	"Smith"			
				Date Naissance	Longueur	Contenu			
				A0	0A				
				Date	Longueur	Contenu			
				43	08	"19590717"			

APPENDICE II

(à la Recommandation X.209)

Affectation des valeurs d'identificateur d'objet

Les valeurs suivantes sont affectées par la présente Recommandation.

Paragraphe	Valeur d'identificateur d'objet Valeur de descripteur d'objet
25.2	{joint-iso-ccitt asn1 (1) codage-base (1)} "Codage de base d'un type ASN.1"

APPENDICE III

(à la Recommandation X.209)

Illustration du codage d'une valeur de réel

III.1 Normalement, un émetteur détermine d'après sa propre représentation en virgule flottante câblée, les algorithmes (indépendamment de la valeur) à utiliser pour transférer des valeurs entre cette représentation en virgule flottante et les octets de longueur et de contenu du codage d'une valeur de réel ASN.1. Cet appendice illustre la démarche à suivre, en prenant la représentation en virgule flottante (artificielle) présentée sur la figure III-1/X.209.

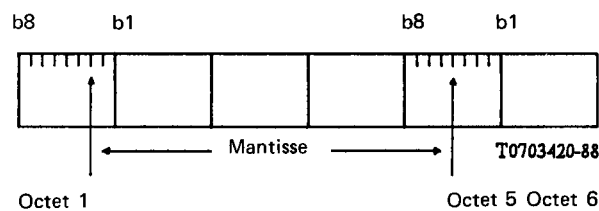


FIGURE III-1/X.209

Représentation pour une virgule flottante câblée hypothétique

Il est supposé que l'exposant peut facilement être obtenu à partir de la virgule flottante câblée, sous la forme d'une valeur entière E.

III.2 Les octets de contenu qui doivent être générés pour émettre une valeur différente de zéro (comme spécifié dans le corps de la présente Recommandation) sont:

1 S bb ff ee Octets de E Octets de N

où S (le signe de la mantisse) dépend de la valeur à convertir, bb est une valeur fixe (disons 10) pour représenter la base (dans ce cas supposons une base 16), ff est la valeur fixe F calculée comme décrit au § III.3 et ee est une valeur fixe de longueur d'exposant calculée comme décrit au § III.4. (Cet appendice ne traite pas le cas où E doit dépasser trois octets.)

III.3 L'algorithme transmettra les octets 1 à 5 de la représentation matérielle comme valeur de N, après avoir forcé les bits 8 à 3 de l'octet 1 et les bits 4 à 1 de l'octet 5 à zéro. Le point décimal implicite est supposé être positionné entre les bits 2 et 1 de l'octet 1 dans la représentation matérielle qui fournit la valeur de E. Sa position implicite peut être décalée au premier point après la fin de l'octet 5 en réduisant la valeur de E avant la transmission. Dans le système de notre exemple, nous pouvons décaler de quatre bits pour chaque décrémentation de l'exposant (parce que nous avons supposé

une base 16) en sorte qu'une décrémentation de 9 positionnera le point implicite entre les bits 6 et 5 de l'octet 6. Pour positionner correctement le point à M, la valeur de N multiplié par 2^3 (la position implicite dans N, après transfert des octets, se trouve après le bit 1 de l'octet 5). Nous avons ainsi les paramètres cruciaux:

$$F = 3 \text{ (ainsi ff est 11)}$$

$$\text{décrémentation exposant} = 9$$

III.4 La longueur nécessaire pour l'exposant est alors calculée en prenant le nombre maximal d'octets nécessaires à représenter les valeurs:

$$E_{\min} - \text{excès} - \text{décrémentation de l'exposant}$$

$$E_{\max} - \text{excès} - \text{décrémentation de l'exposant}$$

où E_{\min} et E_{\max} sont les valeurs entières minimale et maximale de la représentation de l'exposant, excès est n'importe quelle valeur qui doit être soustraite pour donner la vraie valeur de l'exposant, et la décrémentation de l'exposant est telle que calculée au § III.3. Supposons que ceci donne une longueur de trois octets, alors ee est 10. Supposons également que l'excès est zéro.

III.5 L'algorithme de transmission est alors:

- examiner si la valeur est zéro, si oui transmettre une longueur ASN.1 zéro (aucun octet de contenu) et terminer l'algorithme;
- examiner et enregistrer le signe de la mantisse, et prendre l'opposé de la mantisse si elle est négative;
- transmettre une longueur ASN.1 de 9, alors:

11101110 si négative

ou

10101110 si positive;

- produire et transmettre l'exposant de trois octets, avec la valeur:

$$E - 9$$

- mettre à zéro les bits 8 à 3 de l'octet 1 et les bits 4 à 1 de l'octet 5, puis transmettre la mantisse de 5 octets.

III.6 L'algorithme du destinataire doit être préparé à traiter n'importe quel format ASN.1, mais ici l'unité de virgule flottante peut être directement utilisée. Procéder comme suit:

- examiner l'octet 1 du contenu; s'il est 1x101110, la transmission est compatible avec la représentation du destinataire: il suffit d'inverser l'algorithme de l'émetteur;
- autrement, pour le codage des caractères, appeler le logiciel de conversion du codage décimal par caractère standard en virgule flottante, et traiter un "Valeur-Réel-Spécial" conformément à la sémantique de l'application (peut-être en définissant les plus grand et plus petit nombres que la virgule flottante câblée peut traiter);
- pour une transmission binaire, entrer N dans l'unité de virgule flottante, en perdant des octets si nécessaire du côté des octets de plus faible poids, multiplier par 2^F et par B^E , puis prendre l'opposé si nécessaire. Les réalisateurs trouveront que dans certains cas, l'optimisation est possible, mais ils constateront qu'ils perdraient alors en tests plus qu'elle ne leur rapporterait (en dehors de l'optimisation dans le cas des transmissions à partir d'une machine compatible).

III.7 Les algorithmes ci-dessus sont fournis à titre indicatif. Il appartient, bien entendu, aux réalisateurs de déterminer les stratégies qui leur sont les plus avantageuses.