

# UIT-T

SECTEUR DE LA NORMALISATION  
DES TÉLÉCOMMUNICATIONS  
DE L'UIT

# X.692

**Corrigendum 1**  
(05/2005)

SÉRIE X: RÉSEAUX DE DONNÉES, COMMUNICATION  
ENTRE SYSTÈMES OUVERTS ET SÉCURITÉ

Réseautage OSI et aspects systèmes – Notation de  
syntaxe abstraite numéro un (ASN.1)

---

Technologies de l'information – Règles de codage  
ASN.1: spécification de la notation de contrôle de  
codage (ECN)

## **Corrigendum Technique 1**

Recommandation UIT-T X.692 (2002) – Corrigendum  
Technique 1

RECOMMANDATIONS UIT-T DE LA SÉRIE X  
**RÉSEAUX DE DONNÉES, COMMUNICATION ENTRE SYSTÈMES OUVERTS ET SÉCURITÉ**

<b>RÉSEAUX PUBLICS DE DONNÉES</b>	
Services et fonctionnalités	X.1–X.19
Interfaces	X.20–X.49
Transmission, signalisation et commutation	X.50–X.89
Aspects réseau	X.90–X.149
Maintenance	X.150–X.179
Dispositions administratives	X.180–X.199
<b>INTERCONNEXION DES SYSTÈMES OUVERTS</b>	
Modèle et notation	X.200–X.209
Définitions des services	X.210–X.219
Spécifications des protocoles en mode connexion	X.220–X.229
Spécifications des protocoles en mode sans connexion	X.230–X.239
Formulaires PICS	X.240–X.259
Identification des protocoles	X.260–X.269
Protocoles de sécurité	X.270–X.279
Objets gérés des couches	X.280–X.289
Tests de conformité	X.290–X.299
<b>INTERFONCTIONNEMENT DES RÉSEAUX</b>	
Généralités	X.300–X.349
Systèmes de transmission de données par satellite	X.350–X.369
Réseaux à protocole Internet	X.370–X.379
<b>SYSTÈMES DE MESSAGERIE</b>	<b>X.400–X.499</b>
<b>ANNUAIRE</b>	<b>X.500–X.599</b>
<b>RÉSEAUTAGE OSI ET ASPECTS SYSTÈMES</b>	
Réseautage	X.600–X.629
Efficacité	X.630–X.639
Qualité de service	X.640–X.649
Dénomination, adressage et enregistrement	X.650–X.679
<b>Notation de syntaxe abstraite numéro un (ASN.1)</b>	<b>X.680–X.699</b>
<b>GESTION OSI</b>	
Cadre général et architecture de la gestion-systèmes	X.700–X.709
Service et protocole de communication de gestion	X.710–X.719
Structure de l'information de gestion	X.720–X.729
Fonctions de gestion et fonctions ODMA	X.730–X.799
<b>SÉCURITÉ</b>	<b>X.800–X.849</b>
<b>APPLICATIONS OSI</b>	
Engagement, concomitance et rétablissement	X.850–X.859
Traitement transactionnel	X.860–X.879
Opérations distantes	X.880–X.889
Applications génériques de l'ASN.1	X.890–X.899
<b>TRAITEMENT RÉPARTI OUVERT</b>	<b>X.900–X.999</b>
<b>SÉCURITÉ DES TÉLÉCOMMUNICATIONS</b>	<b>X.1000–</b>

*Pour plus de détails, voir la Liste des Recommandations de l'UIT-T.*

## Technologies de l'information – Règles de codage ASN.1: spécification de la notation de contrôle de codage (ECN)

### Corrigendum technique 1

#### Résumé

Le présent Corrigendum technique corrige les défauts signalés dans la Rec. UIT-T X.692 | ISO/CEI 8825-3 qui empêchaient son utilisation à l'appui de la Rec. UIT-T X.891 | ISO/CEI 24824-1 "Fast Infoset". Le texte suivant récapitule les changements introduits dans ce Corrigendum:

- Suppression de la restriction interdisant la définition récursive des objets de codage. Cette restriction constituait une limitation inopportune (et vraisemblablement involontaire) de l'édition précédente de la Recommandation | Norme internationale, puisque les objets de codage sont des moyens naturels par lesquels de nouvelles "règles de codage" peuvent être définies au moyen de la notation ECN, et la notation de la définition de type ASN.1 est naturellement récursive (comme le sont toutes les règles de codage standard).
- Suppression de l'ambiguïté du concept et de la terminologie liés à la propriété consistant à "offrir un pointeur d'identification". Cette propriété est à présent clairement spécifiée comme étant propre à un objet de codage (et non comme une propriété des codages produits par un objet de codage).
- Clarification du mécanisme par lequel un objet de codage (non fondé sur une syntaxe définie) hérite d'un pointeur d'identification provenant d'un autre objet de codage. Le texte précédent était souvent imprécis ou incomplet à cet égard.
- Suppression de la restriction interdisant les structures binaires multiples (valeurs de pointeur) du pointeur d'identification offert par un objet de codage. Cette restriction ne permettait pas d'utiliser une technique commune et pratique grâce à laquelle, en présence de deux options (par exemple), la première option produit toujours une structure binaire bien connue, tandis que la seconde ne produit jamais la même structure binaire.
- Suppression de la restriction interdisant l'application d'un objet de codage au constructeur dans une structure "EncodeStructure" afin de spécifier le remplacement de la structure. Cette restriction constituait une limitation inopportune (et vraisemblablement involontaire) de l'édition précédente de la Recommandation | Norme internationale, puisqu'il n'y a pas d'autres façons de spécifier un remplacement de structure dans le cas d'une structure de codage.
- Correction du défaut de cohérence introduit par la déclaration d'un pointeur d'identification (**EXHIBITS HANDLE &exhibited-handle**) ayant un nom de type **OPTIONAL**, et l'utilisation d'un pointeur d'identification (**DETERMINED BY handle HANDLE &handle-id**) ayant un nom de type **DEFAULT** (= "**default-handle**") sans lien entre ces deux opérations.
- Correction des erreurs typographiques.

#### Source

Le Corrigendum 1 de la Recommandation UIT-T X.692 (2002) a été approuvé le 14 mai 2005 par la Commission d'études 17 (2005-2008) de l'UIT-T selon la procédure définie dans la Recommandation UIT-T A.8. Un texte identique est publié comme Corrigendum technique 1 de la Norme Internationale ISO/CEI 8825-3.

## AVANT-PROPOS

L'UIT (Union internationale des télécommunications) est une institution spécialisée des Nations Unies dans le domaine des télécommunications. L'UIT-T (Secteur de la normalisation des télécommunications) est un organe permanent de l'UIT. Il est chargé de l'étude des questions techniques, d'exploitation et de tarification, et émet à ce sujet des Recommandations en vue de la normalisation des télécommunications à l'échelle mondiale.

L'Assemblée mondiale de normalisation des télécommunications (AMNT), qui se réunit tous les quatre ans, détermine les thèmes d'étude à traiter par les Commissions d'études de l'UIT-T, lesquelles élaborent en retour des Recommandations sur ces thèmes.

L'approbation des Recommandations par les Membres de l'UIT-T s'effectue selon la procédure définie dans la Résolution 1 de l'AMNT.

Dans certains secteurs des technologies de l'information qui correspondent à la sphère de compétence de l'UIT-T, les normes nécessaires se préparent en collaboration avec l'ISO et la CEI.

## NOTE

Dans la présente Recommandation, l'expression "Administration" est utilisée pour désigner de façon abrégée aussi bien une administration de télécommunications qu'une exploitation reconnue.

Le respect de cette Recommandation se fait à titre volontaire. Cependant, il se peut que la Recommandation contienne certaines dispositions obligatoires (pour assurer, par exemple, l'interopérabilité et l'applicabilité) et considère que la Recommandation est respectée lorsque toutes ces dispositions sont observées. Le futur d'obligation et les autres moyens d'expression de l'obligation comme le verbe "devoir" ainsi que leurs formes négatives servent à énoncer des prescriptions. L'utilisation de ces formes ne signifie pas qu'il est obligatoire de respecter la Recommandation.

## DROITS DE PROPRIÉTÉ INTELLECTUELLE

L'UIT attire l'attention sur la possibilité que l'application ou la mise en œuvre de la présente Recommandation puisse donner lieu à l'utilisation d'un droit de propriété intellectuelle. L'UIT ne prend pas position en ce qui concerne l'existence, la validité ou l'applicabilité des droits de propriété intellectuelle, qu'ils soient revendiqués par un Membre de l'UIT ou par une tierce partie étrangère à la procédure d'élaboration des Recommandations.

A la date d'approbation de la présente Recommandation, l'UIT n'avait pas été avisée de l'existence d'une propriété intellectuelle protégée par des brevets à acquérir pour mettre en œuvre la présente Recommandation. Toutefois, comme il ne s'agit peut-être pas de renseignements les plus récents, il est vivement recommandé aux responsables de la mise en œuvre de consulter la base de données des brevets du TSB.

© UIT 2006

Tous droits réservés. Aucune partie de cette publication ne peut être reproduite, par quelque procédé que ce soit, sans l'accord écrit préalable de l'UIT.

## TABLE DES MATIÈRES

*Page*

1)	Paragraphe 21.15, 21.15.1, 21.15.2, 22.9.1.1, 22.9.1.3, 22.9.1.4, 22.9.1.7, 22.9.1.8, 22.9.1.9, 23.1.1, 23.2.1, 23.3.1, 23.4.1, 23.5.1, 23.7.1, 23.8.1, 23.9.1, 23.11.1, 23.13.1 et 23.14.1, et table des matières.....	1
2)	Paragraphe 22.9.1.1, 22.9.1.2, 23.1.1, 23.2.1, 23.3.1, 23.4.1, 23.5.1, 23.7.1, 23.8.1, 23.9.1, 23.11.1, 23.13.1 et 23.14.1 .....	1
3)	Nouveau § 3.2.26 <i>bis</i> .....	1
4)	Paragraphe 3.2.27 .....	1
5)	Paragraphe 3.2.38 .....	1
6)	Paragraphe 3.2.39 .....	2
7)	Paragraphe 9.10.3 .....	2
8)	Paragraphe 9.13.3 .....	2
9)	Paragraphe 13.2.12 .....	2
10)	Paragraphe 17.1.4 .....	3
11)	Paragraphe 17.1.6 .....	3
12)	Paragraphe 17.2.6 .....	3
13)	Paragraphe 17.3.6 .....	3
14)	Paragraphe 17.4.8 .....	3
15)	Paragraphe 17.5.4 .....	3
16)	Paragraphe 17.5.16 .....	4
17)	Paragraphe 17.6.5 .....	4
18)	Paragraphe 17.7.3 .....	4
19)	Paragraphe 17.7.4 .....	4
20)	Nouveau § 17.7.4 <i>bis</i> .....	4
21)	Paragraphe 17.8.6 .....	5
22)	Paragraphe 18.1.3 .....	5
23)	Paragraphe 21.5.7 .....	5
24)	Paragraphe 21.6.6 .....	5
25)	Paragraphe 21.7.10 .....	6
26)	Paragraphe 21.15.1 .....	6
27)	Paragraphe 21.15.2 .....	6
28)	Paragraphe 21.15.3 .....	6
29)	Paragraphe 21.15.4 .....	7
30)	Paragraphe 21.15.5 .....	7
31)	Nouveau § 21.15.6 .....	7
32)	Nouveau § 21.15.7 .....	7
33)	Nouveau § 22.1.1.11 .....	7
34)	Paragraphe 22.5.2.5 .....	7
35)	Paragraphe 22.5.4.4 .....	8
36)	Paragraphe 22.6.2.4 .....	8
37)	Paragraphe 22.6.4.4 .....	8
38)	Paragraphe 22.7.2.10 .....	8
39)	Paragraphe 22.7.4.8 .....	8
40)	Paragraphe 22.9.1.1 .....	8
41)	Paragraphe 22.9.1.4 .....	9
42)	Paragraphe 22.9.2.1 .....	9

	<i>Page</i>
43) Paragraphe 22.9.2.2 .....	9
44) Paragraphes 22.9.2.4, 22.9.2.5 et 22.9.2.6 .....	9
45) Paragraphe 22.9.3.1 .....	10
46) Paragraphe 22.10.2.1 .....	10
47) Paragraphe 22.10.4.3 .....	10
48) Paragraphe 22.10.4.4 .....	10
49) Paragraphe 23.1.2.1 .....	10
50) Paragraphe 23.1.2.2 .....	10
51) Paragraphe 23.1.2.3 .....	11
52) Paragraphe 23.1.2.4 .....	11
53) Paragraphe 23.2.3.1 .....	11
54) Paragraphe 23.2.3.4 .....	11
55) Paragraphe 23.2.3.10 .....	11
56) Paragraphe 23.3.2.1 .....	11
57) Paragraphe 23.4.3.1 .....	12
58) Paragraphe 23.4.3.4 .....	12
59) Paragraphe 23.4.3.9 .....	12
60) Paragraphe 23.5.2.1 .....	12
61) Paragraphe 23.5.2.2 .....	12
62) Paragraphe 23.5.2.4 .....	12
63) Paragraphe 23.7.2.10 .....	13
64) Paragraphe 23.8.2.2 .....	13
65) Paragraphe 23.9.3.1 .....	13
66) Paragraphe 23.9.3.4 .....	13
67) Paragraphe 23.9.3.9 .....	13
68) Paragraphe 23.10.2.2 .....	13
69) Paragraphe 23.11.2.3 .....	14
70) Paragraphe 23.13.2.3 .....	14
71) Paragraphe 23.13.2.4 .....	14
72) Paragraphe 23.14.2.2 .....	14

**NORME INTERNATIONALE  
RECOMMANDATION UIT-T**

**Technologies de l'information – Règles de codage ASN.1: spécification  
de la notation de contrôle de codage (ECN)**

**Corrigendum technique 1**

NOTE – Tout texte nouveau ou modifié dans le présent corrigendum est surligné et le texte supprimé est biffé dans les paragraphes à remplacer. Lorsque ces parties de texte seront incorporées dans la Recommandation de base, les soulignements ainsi que le texte biffé seront supprimés.

- 1) Paragraphes 21.15, 21.15.1, 21.15.2, 22.9.1.1, 22.9.1.3, 22.9.1.4, 22.9.1.7, 22.9.1.8, 22.9.1.9, 23.1.1, 23.2.1, 23.3.1, 23.4.1, 23.5.1, 23.7.1, 23.8.1, 23.9.1, 23.11.1, 23.13.1 et 23.14.1, et table des matières**

*Dans les paragraphes ci-dessus et dans la table des matières, remplacer toutes les occurrences de "HandleValue" par "HandleValueSet".*

- 2) Paragraphes 22.9.1.1, 22.9.1.2, 23.1.1, 23.2.1, 23.3.1, 23.4.1, 23.5.1, 23.7.1, 23.8.1, 23.9.1, 23.11.1, 23.13.1 et 23.14.1**

*Dans les paragraphes ci-dessus, remplacer toutes les occurrences de "&handle-value" par "&handle-value-set".*

- 3) Nouveau § 3.2.26 bis**

*Ajouter le nouveau § 3.2.26 bis comme suit:*

**3.2.26 bis handle value set:** ensemble spécifié de toutes les valeurs possibles du pointeur d'identification affiché par un objet de codage.

- 4) Paragraphe 3.2.27**

*Remplacer le § 3.2.27 par le texte suivant:*

**3.2.27 pointeur d'identification:** partie de codage qui sert à distinguer les codages d'une classe de codage produits par objet de codage (d'une classe déterminée) de ceux produits par d'autres objets ~~classes~~ de codage (d'autres classes).

NOTE – Les règles ASN.1 de codage de base utilisent des étiquettes afin d'offrir des pointeurs d'identification dans les codages BER.

- 5) Paragraphe 3.2.38**

*Remplacer la Note du § 3.2.38 par le texte suivant:*

NOTE – La définition récursive d'une classe de codage (dont une structure de codage) ou d'un objet de codage est permise (se référer toutefois au § 17.1.4). La définition récursive ~~d'un objet de codage~~ ou d'un ensemble d'objets de codage est interdite par les § 17.1.4 et 18.1.3 ~~respectivement~~.

## 6) Paragraphe 3.2.39

Remplacer la Note du § 3.2.39 par le texte suivant:

NOTE – L'instanciation récursive d'une classe de codage (dont une structure de codage) ou d'un objet de codage est permise (se référer toutefois au § 17.1.4). L'instanciation récursive ~~d'un objet de codage~~ ou d'un ensemble d'objets de codage est interdite par les § 17.1.4 et 18.1.3 respectivement.

## 7) Paragraphe 9.10.3

Remplacer le § 9.10.3 par le texte suivant:

**9.10.3** Une troisième caractéristique clé est qu'un objet de codage peut faire apparaître un **pointeur d'identification** dans ses codages. C'est une partie (constituée d'un ensemble fixe de positions binaires) de tous les codages qu'il produit, qui distingue ~~ces~~ ces codages de ceux produits par d'autres objets de codage (d'une classe quelconque) qui font apparaître le même pointeur d'identification. Les pointeurs d'identification ont un nom et sont ~~doivent être~~ visibles par les décodeurs sans la connaissance de la classe de codage ou de la valeur abstraite qui a été codée (mais avec la connaissance du nom du pointeur d'identification qui est utilisé). Ce concept modélise (et généralise) l'utilisation d'étiquettes dans les codages BER: la valeur d'étiquette dans les règles BER peut être déterminée sans la connaissance de la classe de codage, pour tous les codages BER et sert à désigner le codage pour la résolution d'offres d'options, le séquençement d'ensembles, la terminaison de répétitions et le choix d'options.

## 8) Paragraphe 9.13.3

Remplacer le § 9.13.3 par le texte suivant:

**9.13.3** En plus de l'indication de la fin de répétitions, la technique du pointeur d'identification peut également être utilisée afin de déterminer la présence de composants facultatifs ou d'options et le séquençement d'ensembles. Le mécanisme est semblable dans tous ces cas. ~~Le codage de toutes les valeurs d'un codage de "prochaine classe possible" aura la même séquence binaire (leur identification) à certains endroits de ce codage (le pointeur), mais l'identification de différents codages de "prochaine classe possible" sera différente pour chacun d'eux. Pour une classe de codage constituant une "prochaine classe possible" et pour un objet de codage qui lui est appliqué, tout codage ainsi produit contiendra, dans certaines positions binaires (le pointeur d'identification), une séquence binaire coïncidant avec une séquence binaire faisant partie d'un ensemble spécifié de séquences binaires (ensemble de valeurs du pointeur) caractéristique de cette classe, mais non avec toute séquence binaire caractéristique de toute autre "prochaine classe possible".~~ Tous les codages de ce type peuvent être interprétés par un décodeur comme un codage d'une ~~quelconque~~ "prochaine classe possible" et l'identification du pointeur la structure binaire présente dans le codage déterminera quel codage de "prochaine classe possible" est présent. Ce concept est semblable à celui de l'utilisation d'étiquettes à de telles fins dans les règles BER. Des pointeurs d'identification ont des noms qui doivent être uniques à l'intérieur d'une spécification ECN.

## 9) Paragraphe 13.2.12

Remplacer le § 13.2.12 par le texte suivant:

**13.2.12** Dans le processus de codage, des objets de codage appliqués à des constructeurs de codage (et à des classes de la catégorie des offres d'options) peuvent nécessiter que les objets de codage appliqués à leurs-aux composants des constructions définies par ces constructeurs offrent des pointeurs d'identification (d'un nom donné) afin de résoudre les options, ou les offres d'options, une terminaison de répétition ou l'ordre dans une concaténation de type ensemble. ~~Si dans ce cas les codages des composants n'offrent pas les pointeurs d'identification requis, dans ce cas la spécification ECN est en erreur. Ils peuvent également nécessiter que les objets de codage appliqués à d'autres classes de codage (selon ces constructions) offrent les mêmes pointeurs d'identification et que tous les ensembles de valeurs du pointeur de tous les objets de codage impliqués (affichant le même pointeur) soient disjoints. Si ces conditions ne sont pas remplies, alors la spécification ECN est en erreur.~~

NOTE – Ce problème est très susceptible d'apparaître si des objets de codage BER sont appliqués à des constructeurs de codage et non à leurs composants, car les règles BER dépendent beaucoup des pointeurs d'identification. Les objets de codage PER n'utilisent pas de pointeurs d'identification.

**10) Paragraphe 17.1.4**

Remplacer le § 17.1.4 par le texte suivant:

**17.1.4** Il ne doit y avoir aucune définition récursive (voir 3.2.38) d'une référence "encodingobjectreference" et il ne doit y avoir aucune instantiation récursive (voir 3.2.39) d'une référence "encodingobjectreference", si ces opérations se traduisent par une récursion infinie dans la définition du codage.

**11) Paragraphe 17.1.6**

Supprimer la dernière phrase du § 17.1.6. Ce paragraphe s'énonce alors comme suit:

**17.1.6** La production "DefinedEncodingObject" identifie un objet de codage et est spécifiée dans le § 10.9.2. La production "DefinedEncodingObject" doit être de la même classe de codage que le gouverneur, ou d'une classe qui peut être obtenue à partir du gouverneur par déréréfencement.

**12) Paragraphe 17.2.6**

Remplacer le § 17.2.6 par le texte suivant:

**17.2.6** La notation "DefinedSyntax" spécifie si la référence "encodingobjectreference" l'objet encoding\_object définie fait apparaître un pointeur d'identification.

**13) Paragraphe 17.3.6**

Remplacer le § 17.3.6 par le texte suivant:

**17.3.6** ~~Dans l'application des codages spécifiés dans l'article 13, il y a un objet de codage (A par exemple) qui produit le premier champ binaire dans le codage résultant. La référence "encodingobjectreference" qui est définie~~ Soit E l'objet de codage (élément de "CombinedEncodings") appliqué à la classe de gouverneur. Si l'objet de codage E fait apparaître un pointeur d'identification si et seulement si l'objet de codage A fait apparaître ce pointeur d'identification (associé à un ensemble déterminé de valeurs du pointeur), alors l'objet de codage défini (voir § 17.1.5) fait apparaître le même pointeur d'identification que E (associé au même ensemble déterminé de valeurs du pointeur); sinon il ne fait pas apparaître de pointeur.

**14) Paragraphe 17.4.8**

Remplacer le § 17.4.8 par le texte suivant:

**17.4.8** ~~Si la variante "EncodingObject" de la production "ValueMappingEncodingObjects" est utilisée, dans ce cas la référence "encodingobjectreference" définie fait apparaître un pointeur d'identification si et seulement si la variante "EncodingObject" fait apparaître ce pointeur d'identification. Si la variante "DefinedOrBuiltinEncodingObjectSet" de la production "ValueMappingEncodingObjects" est utilisée afin de définir le codage de la classe "DefinedOrBuiltinEncodingClass", dans ce cas la détermination du fait que la référence "encodingobjectreference" fait apparaître un pointeur d'identification est conforme au § 17.3.6. Soit E l'objet de codage appliqué à la classe "DefinedOrBuiltinEncodingClass". Si l'objet de codage E fait apparaître un pointeur d'identification (associé à un ensemble déterminé de valeurs du pointeur), alors l'objet de codage défini (voir 17.1.5) fait apparaître le même pointeur d'identification que E (associé au même ensemble déterminé de valeurs du pointeur); sinon il ne fait pas apparaître de pointeur.~~

NOTE – L'objet de codage E peut être soit la variante "EncodingObject" de la production "ValueMappingEncodingObjects", soit un membre de la classe "DefinedOrBuiltinEncodingObjectSet".

**15) Paragraphe 17.5.4**

Remplacer le § 17.5.4 par le texte suivant:

**17.5.4** Si la liste "ComponentEncodingList" n'est pas vide, l'objet de codage appliqué au constructeur de codage gouvernant (que ce soit à partir de ~~STRUCTURED WITH~~ "StructureEncoding" ou à partir de "CombinedEncodings") ne doit spécifier aucune action de remplacement.

## 16) Paragraphe 17.5.16

Remplacer le § 17.5.16 par le texte suivant:

**17.5.16** La détermination du fait que la référence "encodingobjectreference" l'objet de codage définie (voir § 17.1.5) fait apparaître un pointeur d'identification ~~est conforme au § 17.3.6~~ s'effectue comme suit:

- a) si "TagEncoding" est présent dans "StructureEncoding", on appelle E l'objet de codage appliqué à la classe de codage dans la catégorie des étiquettes; ou
- b) si "TagEncoding" n'est pas présent dans "StructureEncoding", on appelle E l'objet de codage appliqué au constructeur de codage gouvernant (soit l'EncodingObject dans l'ensemble "EncodingOrUseSet" dans "StructureEncoding", soit un membre de la production "CombinedEncodings").

Si l'objet de codage E fait apparaître un pointeur d'identification (associé à un ensemble déterminé de valeurs du pointeur), alors l'objet de codage défini fait apparaître le même pointeur d'identification que E (associé au même ensemble déterminé de valeurs du pointeur); sinon il ne fait pas apparaître de pointeur.

## 17) Paragraphe 17.6.5

Remplacer le § 17.6.5 par le texte suivant:

**17.6.5** ~~La production "encodingobjectreference" qui est définie ne fait apparaître un pointeur d'identification que si et seulement si le même pointeur d'identification est affiché par les productions "SpecForEncoding" et "SpecForDecoders".~~ Si les productions "SpecForEncoding" et "SpecForDecoders" font apparaître le même pointeur d'identification associé au même ensemble de valeurs du pointeur, alors l'objet de codage défini (voir § 17.1.5) fait apparaître ce pointeur d'identification (associé au même ensemble déterminé de valeurs du pointeur); sinon il ne fait pas apparaître de pointeur.

## 18) Paragraphe 17.7.3

Remplacer le § 17.7.3 par le texte suivant:

**17.7.3** La production "AlternativesEncodingObject" doit être un objet de codage d'une classe quelconque dans la catégorie des options. Les codeurs et les décodeurs ne doivent utiliser les codages et procédures spécifiés par cet objet de codage que si les options de codage étaient des codages pour ~~composants-variantes~~ d'une instance de cette classe. L'objet "AlternativesEncodingObject" ne doit pas contenir de spécification **REPLACE** (voir § 23.1.1). Le paramètre **DETERMINED BY** doit être mis à la valeur **handle** et un pointeur d'identification doit être spécifié.

NOTE – Si l'objet "AlternativesEncodingObject" est paramétré avec un champ de référence, dans ce cas la référence "encodingobjectreference" qui est définie doit être paramétrée avec un champ de référence fictif qui est utilisé comme paramètre réel pour l'objet "AlternativesEncodingObject".

## 19) Paragraphe 17.7.4

Remplacer le § 17.7.4 par le texte suivant:

**17.7.4** Tous les objets "EncodingObject" contenus dans la liste "EncodingOptionsList" doivent offrir ce pointeur d'identification et tous les ensembles de valeurs du pointeur doivent être disjoints.

## 20) Nouveau § 17.7.4 bis

Introduire un nouveau § 17.7.4 bis dont le texte s'énonce comme suit:

**17.7.4 bis** Si l'objet "AlternativesEncodingObject" fait apparaître un pointeur d'identification (associé au même ensemble de valeurs du pointeur), alors l'objet de codage défini (voir § 17.1.5) fait apparaître ce pointeur d'identification (associé au même ensemble de valeurs du pointeur); sinon il ne fait pas apparaître de pointeur.

NOTE – Le pointeur d'identification affiché (éventuellement) par la variante "AlternativesEncodingObject" n'est pas lié au pointeur d'identification affiché par les "EncodingObject" de la liste "EncodingOptionsList", même s'ils ont le même nom.

**21) Paragraphe 17.8.6**

Remplacer le § 17.8.6 par le texte suivant:

**17.8.6** Un pointeur d'identification (associé à un ensemble déterminé de valeurs du pointeur) est affiché par la référence "encodingobjectreference" qui n'est définie que l'objet de codage en cours de définition (voir § 17.1.5) si et seulement si la production "anystringexceptnonecnd" spécifie qu'elle agit ainsi. Le moyen d'une telle spécification n'est pas défini dans la présente Recommandation | Norme internationale.

**22) Paragraphe 18.1.3**

Remplacer le § 18.1.3 par le texte suivant:

**18.1.3** Il ne doit y avoir aucune définition récursive (voir 3.2.38) d'une référence "encodingclassobjectsetreference" et il ne doit y avoir aucune instanciation récursive (voir 3.2.39) d'une référence "encodingclassobjectsetreference".

**23) Paragraphe 21.5.7**

Remplacer le § 21.5.7 par le texte suivant (scindé en un § 21.5.7 et un nouveau § 21.5.7 bis):

**21.5.7** La valeur "handle" nécessite qu'un pointeur d'identification soit spécifié. Ce pointeur d'identification doit être conjointement affiché par l'objet de codage pour le composant facultatif et par ~~tout codage en option~~ l'objet de codage appliqué à chaque variante possible de classe de codage pouvant suivre si ce composant facultatif est absent. ~~La valeur du pointeur doit être différente pour le codage du composant facultatif et pour tous les codages suivants qui peuvent être offerts en option. Chaque variante possible de classe de codage peut être un composant de la concaténation contenant ce composant facultatif ou un composant de la classe de codage suivant la concaténation. Les ensembles de valeurs du pointeur spécifiés par tous les objets de codage impliqués (affichant le même pointeur d'identification) doivent être tous disjoints.~~

NOTE – Chaque valeur abstraite d'un composant donné doit avoir une valeur de pointeur correspondant à l'ensemble spécifié de valeurs du pointeur (voir § 22.9.2.2).

**21.5.7 bis** Si l'extrémité d'un quelconque conteneur ouvert (ou l'extrémité de l'unité PDU) est détectée au moment où un décodeur tente de détecter la présence ou l'absence ~~de ce d'un~~ composant facultatif, alors le décodeur doit déterminer que le composant est absent. Sinon, ~~un~~ le décodeur doit déterminer que le composant est présent si et seulement si le décodage des parties restantes du codage produit pour le pointeur d'identification spécifié une valeur qui correspond à ~~elle~~ l'ensemble de valeurs du pointeur du composant facultatif. Il y a erreur de spécification ECN s'il n'en résulte pas une identification correcte de la présence ou de l'absence d'un codage du composant facultatif; mais les codeurs conformes ne doivent pas produire de tels codages.

**24) Paragraphe 21.6.6**

Remplacer le § 21.6.6 par le texte suivant (scindé en un § 21.6.6 et un nouveau § 21.6.7):

**21.6.6** La valeur "handle" nécessite qu'un pointeur d'identification soit spécifié. Ce pointeur d'identification doit être affiché par ~~(les codages de) toutes les options dans la classe et le codage de chaque option doit avoir une valeur différente pour le pointeur d'identification~~ les objets de codage appliqués à chacune des variantes dans la construction définie par la classe dans la catégorie des options. Les ensembles de valeurs du pointeur spécifiés par ces objets de codage doivent être tous disjoints. (La violation de cette règle est une erreur de spécification ECN ~~mais et~~ les codeurs conformes sont tenus de ne pas produire de codages où cette règle est violée.) ~~Cette valeur spécifie qu'un~~

**21.6.7** Un décodeur doit déterminer l'option qui est présente par le décodage des parties restantes du codage afin de produire une valeur pour le pointeur d'identification spécifié. L'option dont ~~le~~ l'ensemble de valeurs du pointeur d'identification a une valeur qui correspond à cette valeur est l'option qui est présente. Si l'extrémité d'un quelconque conteneur ouvert (ou l'extrémité de l'unité PDU) est atteinte avant que le pointeur d'identification puisse être décodé, ou si la valeur du pointeur d'identification ne correspond pas à ~~elle~~ l'ensemble de valeurs du pointeur d'une quelconque option, dans ce cas il y a erreur de codage.

NOTE – Chaque valeur abstraite d'un composant donné doit avoir une valeur de pointeur correspondant à l'ensemble spécifié de valeurs du pointeur (voir § 22.9.2.2).

**25) Paragraphe 21.7.10**

Remplacer le § 21.7.10 par le texte suivant:

**21.7.10** La valeur "handle" nécessite qu'un pointeur d'identification soit spécifié. Ce pointeur d'identification doit être conjointement affiché par ~~l'élément répété et par tous les éléments~~ l'objet de codage appliqué au composant répété et par l'objet de codage appliqué à chacune des classes de codage pouvant suivre (compte tenu de l'offre d'options). ~~La valeur du pointeur d'identification pour l'élément répété doit être différente de celle de tous les éléments pouvant suivre.~~ Les ensembles de valeurs du pointeur spécifiés par ces objets de codage doivent être tous disjoints.

NOTE – Chaque valeur abstraite d'un composant donné doit avoir une valeur de pointeur correspondant à l'ensemble spécifié de valeurs du pointeur (voir § 22.9.2.2).

**26) Paragraphe 21.15.1**

Remplacer le § 21.15.1 par le texte suivant:

**21.15.1** Le type "HandleValueSet" est le suivant:

```
HandleValueSet ::= CHOICE {
  bits          BIT STRING,
  octets        OCTET STRING,
  number        INTEGER (0..MAX),
  tag           ENUMERATED {any},
  range         SEQUENCE {
    low          INTEGER(0..MAX),
    high         INTEGER(0..MAX) },
  ranges        SET (SIZE(1..MAX)) OF SEQUENCE {
    low          INTEGER(0..MAX),
    high         INTEGER(0..MAX) } }
```

**27) Paragraphe 21.15.2**

Remplacer le § 21.15.2 par le texte suivant:

**21.15.2** La production "HandleValueSet" est utilisée afin de spécifier ~~la valeur d'un pointeur d'identification qui est affichée par des objets de codage particuliers~~ l'ensemble de structures binaires (l'ensemble de valeurs du pointeur) caractérisant les codages produits par un objet de codage qui affiche un pointeur d'identification.

**28) Paragraphe 21.15.3**

Remplacer le § 21.15.3 et la note correspondante par le texte suivant:

**21.15.3** ~~Les valeurs d'un quelconque pointeur d'identification qui est affiché par un objet de codage doivent être les mêmes pour toutes les valeurs abstraites codées par cet objet de codage (voir § 22.9.2.2). La valeur d'un pointeur d'identification peut être utilisée afin de désigner la présence ou l'absence de composants facultatifs, le choix d'options, le séquençement d'ensembles ou la fin d'une répétition. Il est également requis dans de telles circonstances que les valeurs de pointeur affichées par le codage de différentes options ou de différents composants soient distinctes (voir § 21.5.7, 21.6.6 et 21.7.10) ensembles de valeurs du pointeur des objets de codage appliqués aux différentes options ou aux différents composants soient tous disjoints (voir § 21.5.7, 21.6.6, 21.7.10 et 22.10.2.1) et que toutes les valeurs possibles du pointeur d'identification présentes dans les codages de toute option ou composant donné correspondent à l'ensemble spécifié de valeurs du pointeur de l'objet de codage appliqué à cette option ou à ce composant (voir § 22.9.2.2).~~

NOTE – Les valeurs de pointeur d'identification affichées par un objet de codage donné peuvent, théoriquement, être déterminées par codage d'une valeur d'essai. Cependant, afin de faciliter la tâche d'implémentation, le Le spécificateur ECN est tenu de spécifier la valeur l'ensemble de valeurs du pointeur dans tous les cas sauf si (pour les codages d'une classe d'étiquettes) la valeur l'ensemble de valeurs du pointeur d'identification est constitué d'une seule valeur et dépend du numéro d'étiquette associé à cette classe d'étiquettes, soit directement par production implicite à partir d'une étiquette ASN.1, ou par mappage à partir d'une structure produite implicitement.

**29) Paragraphe 21.15.4**

*Remplacer le § 21.15.4 par le texte suivant:*

**21.15.4** Les options "**bits**", "**octets**" et "**number**" spécifient respectivement ~~la~~une valeur de pointeur en tant que chaîne de bits, chaîne d'octets ou valeur entière respectivement. Il y a erreur de spécification ECN si cette valeur ne peut pas être codée dans le nombre de bits spécifié pour le pointeur d'identification (voir § 22.9).

**30) Paragraphe 21.15.5**

*Remplacer le § 21.15.5 par le texte suivant:*

**21.15.5** L'option "**tag:any**" spécifie ~~que la~~une valeur du pointeur ~~est~~ déterminée par le numéro spécifié dans une structure de codage ECN pour une classe de la catégorie des étiquettes, ou par le numéro d'étiquette mappé à partir d'une construction d'étiquette ASN.1. Elle ne doit être utilisée que lors de la spécification de l'identification par pointeur pour le codage d'une classe de la catégorie des étiquettes.

**31) Nouveau § 21.15.6**

*Introduire un nouveau § 21.15.6 dont le texte s'énonce comme suit:*

**21.15.6** L'option "**range**" spécifie une étendue de valeurs entières, dont une valeur **high** est supérieure ou égale à une valeur **low**.

**32) Nouveau § 21.15.7**

*Introduire un nouveau § 21.15.7 dont le texte s'énonce comme suit:*

**21.15.7** L'option "**ranges**" spécifie un ensemble d'étendues de valeurs entières, dont chacune comporte une valeur **high** supérieure ou égale à une valeur **low**. Il est possible de spécifier une ou plusieurs étendues de ce type, lesquelles ne doivent pas se chevaucher.

**33) Nouveau § 22.1.1.11**

*Introduire un nouveau § 22.1.1.11 dont le texte s'énonce comme suit:*

**22.1.1.11** Dans une spécification de remplacement complète, si l'objet de codage appliqué à la structure de remplacement affiche un pointeur d'identification (associé à un ensemble donné de valeurs du pointeur), alors l'objet de codage dont la syntaxe définie contient la spécification de remplacement complète affiche le même pointeur d'identification (avec le même ensemble de valeurs du pointeur); sinon il n'affiche pas de pointeur.

**34) Paragraphe 22.5.2.5**

*Remplacer le § 22.5.2.5 par le texte suivant et y supprimer la NOTE.*

**22.5.2.5** Si "**DETERMINED BY**" est "**handle**", alors le § 21.5.7 s'applique.

~~**22.5.2.5** Si "**HANDLE**" est spécifié, dans ce cas le composant dont la présence est déterminée ainsi que tous les codages facultatifs suivants et le prochain codage obligatoire (si présent) doivent tous être produits par des objets de codage dont toutes les spécifications offrent un pointeur d'identification ayant le même nom que "**HANDLE**". Le prochain codage obligatoire peut être un composant de la concaténation contenant le composant facultatif, ou peut être un codage suivant la concaténation. La valeur du pointeur d'identification doit être différente pour tous ces composants.~~

~~NOTE — Il est prescrit que les bits qui forment un pointeur d'identification aient la même valeur pour toutes les valeurs abstraites codées par un objet de codage contenant ce pointeur d'identification (voir § 22.9.2.2).~~

### 35) Paragraphe 22.5.4.4

Remplacer le § 22.5.4.4 par le texte suivant:

**22.5.4.4** Si "DETERMINED BY" est "handle", dans ce cas le décodeur doit déterminer la valeur du pointeur d'identification spécifié. Si la valeur correspond à elle-l'ensemble de valeurs du pointeur d'identification du composant facultatif, dans ce cas le décodeur doit fixer la valeur théorique "element-is-present" à TRUE, sinon le décodeur doit la fixer à FALSE.

### 36) Paragraphe 22.6.2.4

Remplacer le § 22.6.2.4 par le texte suivant et y supprimer la NOTE:

**22.6.2.4** Si "DETERMINED BY" est "handle", alors le § 21.6.6 s'applique.

~~22.6.2.4~~ Si "HANDLE" est spécifié, dans ce cas toutes les options de la classe de codage de la catégorie des options doivent être codées par des objets de codage dont la spécification fait apparaître et définit un pointeur d'identification ayant le même nom que "HANDLE" et avec la même valeur du pointeur d'identification. La valeur du pointeur d'identification doit être différente pour toutes ces options.

~~NOTE~~ Il est prescrit qu'un pointeur d'identification doit avoir la même valeur pour toutes les valeurs abstraites codées par un objet de codage contenant ce pointeur d'identification (voir 22.9.2.2).

### 37) Paragraphe 22.6.4.4

Remplacer le § 22.6.4.4 par le texte suivant:

**22.6.4.4** Si "DETERMINED BY" est "handle", dans ce cas le décodeur doit déterminer la valeur du pointeur d'identification. Cette valeur doit être comparée à la valeur-l'ensemble de valeurs du pointeur d'identification de chaque option. Si aucune ne correspond, dans ce cas le décodeur doit diagnostiquer une erreur de codeur. Sinon la valeur théorique "alternative-index" doit être mise à l'option concordante.

### 38) Paragraphe 22.7.2.10

Remplacer le § 22.7.2.10 par le texte suivant et y supprimer la NOTE:

**22.7.2.10** Si "DETERMINED BY" est "handle", alors le § 21.7.10 s'applique.

~~22.7.2.10~~ Si "HANDLE" est spécifié, dans ce cas l'élément répété, ainsi que tout élément qui (par l'utilisation d'offres d'options) peut suivre l'élément répété, doivent tous être codés par des objets de codage dont la spécification fait apparaître un pointeur d'identification ayant le même nom que "HANDLE". La valeur du pointeur d'identification dans l'élément de répétition doit être différente de celle d'un éventuel élément suivant.

~~NOTE~~ Il est prescrit qu'un pointeur d'identification ait la même valeur pour toutes les valeurs abstraites codées par un objet de codage contenant ce pointeur d'identification (voir 22.9.2.2).

### 39) Paragraphe 22.7.4.8

Remplacer le § 22.7.4.8 par le texte suivant:

**22.7.4.8** Si "DETERMINED BY" est "handle", dans ce cas le décodeur doit déterminer la valeur du pointeur d'identification et tenter de décoder l'élément-le codage suivant (en parallèle), soit en tant que nouvelle occurrence de la répétition ou en tant que élément-que classe de codage suivante, au moyen de la valeur du pointeur d'identification afin de distinguer ces options. Si le décodage réussit pour plusieurs de ces éléments ou pour aucun d'entre eux, il y a erreur de codage ou de spécification.

### 40) Paragraphe 22.9.1.1

Remplacer le § 22.9.1.1 par le texte suivant:

**22.9.1.1** La spécification de pointeur d'identification utilise les propriétés de codage suivantes:

```

&exhibited-handle      PrintableString OPTIONALDEFAULT "default-handle",
&handle-positions      INTEGER (0..MAX) OPTIONAL,
&handle-value-set      HandleValueSet DEFAULT tag:any

```

#### 41) Paragraphe 22.9.1.4

Remplacer le § 22.9.1.4 par le texte suivant:

~~22.9.1.4 Cette spécification est utilisée afin d'indiquer qu'un objet de codage fait apparaître un pointeur d'identification à l'intérieur de tous ses codages (c'est-à-dire pour toutes les valeurs abstraites possibles qu'il code). Le nom du pointeur d'identification est spécifié, ainsi que les bits qui sont associés à ce pointeur d'identification. La valeur du pointeur d'identification est spécifiée par "HandleValue".~~

22.9.1.4 L'objet de la présente spécification consiste à indiquer qu'un objet de codage fait apparaître un pointeur d'identification et à préciser ses propriétés, c'est-à-dire:

- a) le nom du pointeur;
- b) les bits qui le constituent; et
- c) les structures binaires possibles (pour les positions binaires qui constituent le pointeur) présentes dans les codages produits par cet objet de codage (l'ensemble de valeurs du pointeur).

#### 42) Paragraphe 22.9.2.1

Remplacer le § 22.9.2.1 et la note correspondante par le texte suivant:

~~22.9.2.1 Dans une application-spécification ECN quelconque de la notation ECN, tous les pointeurs d'identification ayant le même nom doivent spécifier le même ensemble de bits pour l'emplacement du pointeur d'identification positions binaires.~~

~~NOTE— Il n'y a pas d'exigence générale que la valeur du pointeur d'identification (révélée par différents objets de codage) soit distincte, mais des valeurs distinctes sont requises lorsque le pointeur d'identification est utilisé afin de résoudre des offres d'options, un choix d'options, ou une terminaison de répétition (voir 21.5.7, 21.6.6 et 21.7.10).~~

~~NOTE – Il n'y a pas d'exigence générale que les ensembles de valeurs du pointeur de différents objets de codage définis dans une spécification ECN soient disjoints; par contre, les ensembles de valeurs du pointeur doivent être disjoints lorsque le pointeur d'identification est utilisé afin de résoudre des offres d'options, un choix d'options, une terminaison de répétition ou un séquençement d'ensembles (voir § 21.5.7, 21.6.6, 21.7.10 et 22.10.2.1).~~

#### 43) Paragraphe 22.9.2.2

Remplacer le § 22.9.2.2 par le texte suivant:

~~22.9.2.2 — Le spécificateur ECN doit veiller à ce qu'un objet de codage contenant un pointeur d'identification produise la même valeur du pointeur d'identification pour chaque valeur abstraite qui est codée.~~

22.9.2.2 Dans le cas d'un objet de codage qui fait apparaître un pointeur d'identification (associé à un ensemble donné de valeurs du pointeur), la valeur du pointeur obtenue dans chacun des codages possibles produits par l'objet de codage (pour toutes les valeurs abstraites possibles) doit figurer dans l'ensemble spécifié de valeurs du pointeur.

#### 44) Paragraphes 22.9.2.4, 22.9.2.5 et 22.9.2.6

Supprimer les § 22.9.2.4, 22.9.2.5, et 22.9.2.6.

~~22.9.2.4 — Si un objet de codage pour une classe de la catégorie des répétitions fait apparaître un pointeur d'identification, dans ce cas ce pointeur d'identification doit également être affiché (avec la même valeur) par le codage de l'élément répété.~~

~~22.9.2.5 — Si un objet de codage pour une classe de la catégorie des options fait apparaître un pointeur d'identification, dans ce cas ce pointeur d'identification doit également être affiché par (le codage de) toutes les options, et la valeur du pointeur d'identification doit être la même pour toutes les options.~~

~~NOTE— Dans ce cas, ce pointeur d'identification ne peut pas être utilisé pour la détermination des options dans cette option. La détermination des options doit donc être effectuée au moyen d'un autre pointeur d'identification ou par d'autres moyens.~~

~~22.9.2.6~~ Si un objet de codage pour une classe de la catégorie des concaténations fait apparaître un pointeur d'identification, dans ce cas le premier composant codé (si présent) (ou, s'il est étiqueté, l'étiquette), compte tenu des offres d'options, doit offrir ce pointeur d'identification avec la même valeur.

#### 45) Paragraphe 22.9.3.1

Remplacer le § 22.9.3.1 par le texte suivant:

**22.9.3.1** Si un objet de codage fait apparaître un pointeur d'identification, le codeur doit vérifier que le codage a la valeur du pointeur d'identification présente dans le codage produit figure dans l'ensemble spécifié de valeurs du pointeur et doit diagnostiquer une erreur de spécification ou d'application si ce n'est pas le cas.

#### 46) Paragraphe 22.10.2.1

Remplacer le § 22.10.2.1 par le texte suivant:

**22.10.2.1** Si "ORDER" est "random", dans ce cas "HANDLE" prend la valeur par défaut "default-handle" s'il n'est pas activé et ~~le codage de les objets de codage appliqués à~~ tous les composants ~~doit offrir "HANDLE" avec des valeurs distinctes pour le~~ doivent afficher ce pointeur d'identification. Les ensembles de valeurs du pointeur de ces objets de codage doivent être tous disjoints.

#### 47) Paragraphe 22.10.4.3

Remplacer le § 22.10.4.3 par le texte suivant:

**22.10.4.3** Si "ORDER" est "random", le décodeur doit déterminer l'ordre des composants par examen de la valeur des bits associés à "HANDLE" du pointeur d'identification.

#### 48) Paragraphe 22.10.4.4

Supprimer la première phrase du § 22.10.4.4. Le paragraphe s'énonce alors comme suit:

**22.10.4.4** ~~Chaque composant a une valeur distincte pour les bits associés à "HANDLE", qui permet que le composant soit identifié.~~ Le décodage doit continuer jusqu'à ce qu'une valeur abstraite ait été obtenue pour chaque composant et un décodeur doit diagnostiquer une erreur de codeur si plusieurs codages sont identifiés pour un composant, ou si des valeurs inattendues apparaissent pour des pointeurs d'identification pendant le décodage.

NOTE – Des valeurs inattendues peuvent apparaître dans le cadre d'une disposition d'extensibilité, mais cela n'est pas pris en charge dans cette version de la présente Recommandation | Norme internationale. De telles instances doivent être traitées comme des erreurs de codeur.

#### 49) Paragraphe 23.1.2.1

Remplacer le § 23.1.2.1 par le texte suivant:

**23.1.2.1** Cette syntaxe est utilisée afin de définir le début de l'espace de codage pour une classe de codage de la catégorie des options, la détermination de l'option qui a été codée, et une déclaration facultative que tous les codages l'objet de codage offrent un pointeur d'identification spécifié avec des valeurs de pointeur d'identification distinctes (avec un ensemble donné de valeurs du pointeur).

#### 50) Paragraphe 23.1.2.2

Remplacer le § 23.1.2.2 par le texte suivant:

**23.1.2.2** Si "REPLACE STRUCTURE" est activé, dans ce cas aucun autre groupe de propriétés de codage ne doit être activé. Si l'objet de codage de la structure de remplacement fait apparaître un pointeur (avec un ensemble donné de valeurs du pointeur), l'objet de codage défini fait alors apparaître le même pointeur d'identification (avec le même ensemble de valeurs du pointeur – voir § 22.1.1.11).

**51) Paragraphe 23.1.2.3**

Remplacer le § 23.1.2.3 par le texte suivant:

**23.1.2.3** ~~Les codages~~ Un objet de codage de cette classe n'~~offre~~ pas de pointeur d'identification à moins que "EXHIBITS HANDLE" ne soit activé (même si ~~tous les composants de la construction définie~~ offrent un pointeur d'identification, ~~qui peut être ou ne pas être le même ou à moins que~~ "REPLACE STRUCTURE" ne soit activé et que l'objet de codage de la structure de remplacement ne produise un pointeur d'identification (voir § 22.1.1.11).

**52) Paragraphe 23.1.2.4**

Remplacer le § 23.1.2.4 par le texte suivant et y supprimer la NOTE:

**23.1.2.4** Si "EXHIBITS HANDLE" est activé, dans ce cas ~~les codages de toutes les options de cette classe doivent offrir le pointeur d'identification défini, et offrir des valeurs distinctes pour ce pointeur d'identification.~~ l'objet de codage offre le pointeur d'identification spécifié.

NOTE – Cela nécessitera normalement que chaque composant ait un champ "EXHIBITS HANDLE" mis à la même valeur, à moins qu'une insertion en tête de séquence n'ait affiché le pointeur d'identification (voir 9.10.3).

**53) Paragraphe 23.2.3.1**

Remplacer le § 23.2.3.1 par le texte suivant:

**23.2.3.1** Cette syntaxe est utilisée afin de définir le début de l'espace de codage pour une classe de la catégorie des chaînes de bits, le codage des valeurs abstraites de cette classe, une déclaration facultative que ~~tous les codages de bits offrent~~ l'objet de codage offre un pointeur d'identification spécifié, (avec un ensemble déterminé de valeurs du pointeur) et une spécification de la façon de coder un type contenu.

**54) Paragraphe 23.2.3.4**

Remplacer le § 23.2.3.4 par le texte suivant:

**23.2.3.4** S'il y a une clause "REPLACE STRUCTURE" dans les objets de codage de la classe #CONDITIONAL-REPETITION, dans ce cas aucun autre paramètre ne doit être activé. Si l'objet de codage de la structure de remplacement fait apparaître un pointeur (avec un ensemble donné de valeurs du pointeur), l'objet de codage défini fait alors apparaître le même pointeur d'identification (avec le même ensemble de valeurs du pointeur – voir § 22.1.1.11).

**55) Paragraphe 23.2.3.10**

Remplacer le § 23.2.3.10 par le texte suivant:

**23.2.3.10** Si "EXHIBITS HANDLE" est activé, dans ce cas ~~tous les codages de valeurs associés à cette classe doivent offrir~~ l'objet de codage offre le pointeur d'identification spécifié.

NOTE – Cela nécessitera en général des restrictions applicables aux valeurs abstraites du type associé ou l'adjonction de bits redondants dans la transformation en bits, ou les deux.

**56) Paragraphe 23.3.2.1**

Remplacer le § 23.3.2.1 par le texte suivant:

**23.3.2.1** Cette syntaxe est utilisée afin de définir le début de l'espace de codage pour une classe de la catégorie des booléens, le codage des valeurs abstraites de cette classe, leur positionnement à l'intérieur de l'espace de codage, une déclaration facultative que ~~tous les codages de bits offrent~~ l'objet de codage offre un pointeur d'identification spécifié (avec un ensemble donné de valeurs du pointeur), et une éventuelle inversion de l'ordre des bits de l'espace de codage pour les booléens.

### 57) Paragraphe 23.4.3.1

Remplacer le § 23.4.3.1 par le texte suivant:

**23.4.3.1** Cette syntaxe est utilisée afin de définir le début de l'espace de codage pour une classe de la catégorie des chaînes de caractères, le codage des valeurs abstraites associées à cette classe, une déclaration facultative que ~~tous les codages de caractères offrent~~ l'objet de codage offre un pointeur d'identification spécifié (avec un ensemble donné de valeurs du pointeur).

### 58) Paragraphe 23.4.3.4

Remplacer le § 23.4.3.4 par le texte suivant:

**23.4.3.4** S'il n'y a pas de clause "**REPLACE STRUCTURE**" dans les objets de codage de la classe **#CONDITIONAL-REPETITION**, dans ce cas la spécification "**TRANSFORMS**" doit être activée. S'il y a une clause "**REPLACE STRUCTURE**" dans les objets de codage de la classe **#CONDITIONAL-REPETITION**, dans ce cas aucun autre paramètre ne doit être activé. Si l'objet de codage de la structure de remplacement fait apparaître un pointeur (avec un ensemble donné de valeurs du pointeur), l'objet de codage défini fait alors apparaître le même pointeur d'identification (avec le même ensemble de valeurs du pointeur – voir § 22.1.1.11).

### 59) Paragraphe 23.4.3.9

Remplacer le § 23.4.3.9 par le texte suivant:

**23.4.3.9** Si "**EXHIBITS HANDLE**" est activé, dans ce cas ~~tous les codages de valeurs associés à cette classe doivent offrir~~ l'objet de codage offre le pointeur d'identification spécifié.

NOTE – Cela nécessitera en général des restrictions applicables aux valeurs abstraites du type associé, ou l'inclusion de bits redondants dans le codage de chaque caractère, ou les deux.

### 60) Paragraphe 23.5.2.1

Remplacer le § 23.5.2.1 par le texte suivant:

**23.5.2.1** Cette syntaxe est utilisée afin de définir le début de l'espace de codage pour une classe de la catégorie des concaténations, la façon dont les codages des composants doivent être combinés, leur positionnement à l'intérieur de l'espace de codage, une déclaration facultative que ~~tous les codages offrent~~ l'objet de codage offre un pointeur d'identification spécifié (avec un ensemble donné de valeurs du pointeur), et une éventuelle inversion de l'ordre des bits de l'espace de codage.

### 61) Paragraphe 23.5.2.2

Remplacer le § 23.5.2.2 par le texte suivant:

**23.5.2.2** Si "**REPLACE STRUCTURE**" est activé, dans ce cas aucun autre paramètre de codage des groupes ne doit être activé. Si l'objet de codage de la structure de remplacement fait apparaître un pointeur (avec un ensemble donné de valeurs du pointeur), l'objet de codage défini fait alors apparaître le même pointeur d'identification (avec le même ensemble de valeurs du pointeur – voir § 22.1.1.11).

### 62) Paragraphe 23.5.2.4

Remplacer le § 23.5.2.4 par le texte suivant et y supprimer la NOTE:

**23.5.2.4** Si "**EXHIBITS HANDLE**" est activé, alors ~~le codage de toutes les valeurs abstraites possibles associées à cette classe doivent offrir~~ le pointeur d'identification défini l'objet de codage offre le pointeur d'identification spécifié.

NOTE – Cela sera souvent réalisé par vérification du fait que le premier composant de la concaténation, ou une insertion en tête de séquence, a affiché le pointeur d'identification.

**63) Paragraphe 23.7.2.10**

Remplacer le § 23.7.2.10 par le texte suivant:

**23.7.2.10** Si "EXHIBITS HANDLE" est activé, dans ce cas ~~le spécificateur déclare par assertion que le codage de toutes les valeurs~~ l'objet de codage fait apparaître le pointeur d'identification spécifié.

NOTE – Cela nécessitera normalement l'utilisation de "VALUE-PADDING" avec justification à partir de la gauche afin de permettre au bourrage d'offrir le pointeur d'identification.

**64) Paragraphe 23.8.2.2**

Remplacer le § 23.8.2.2 par le texte suivant:

**23.8.2.2** Si "REPLACE STRUCTURE" est activé, dans ce cas aucun autre groupe de propriétés de codage ne doit être activé. Si l'objet de codage de la structure de remplacement fait apparaître un pointeur (avec un ensemble donné de valeurs du pointeur), l'objet de codage défini fait alors apparaître le même pointeur d'identification (avec le même ensemble de valeurs du pointeur – voir § 22.1.1.11).

**65) Paragraphe 23.9.3.1**

Remplacer le § 23.9.3.1 par le texte suivant:

**23.9.3.1** Cette syntaxe est utilisée afin de définir le début de l'espace de codage pour une classe de la catégorie des chaînes d'octets, le codage des valeurs abstraites associées à cette classe, une déclaration facultative que ~~tous les codages de chaîne d'octets offrent~~ l'objet de codage offre un pointeur d'identification spécifié (avec un ensemble donné de valeurs de pointeur), une spécification de la façon de coder un type contenu.

**66) Paragraphe 23.9.3.4**

Remplacer le § 23.9.3.4 par le texte suivant:

**23.9.3.4** S'il y a une clause "REPLACE STRUCTURE" dans les objets de codage de la classe #CONDITIONAL-REPETITION, dans ce cas aucun autre paramètre ne doit être activé. Si l'objet de codage de la structure de remplacement fait apparaître un pointeur (avec un ensemble donné de valeurs du pointeur), l'objet de codage défini fait alors apparaître le même pointeur d'identification (avec le même ensemble de valeurs du pointeur – voir § 22.1.1.11).

**67) Paragraphe 23.9.3.9**

Remplacer le § 23.9.3.9 par le texte suivant:

**23.9.3.9** Si "EXHIBITS HANDLE" est activé, dans ce cas ~~tous les codages de valeurs de cette classe doivent offrir~~ l'objet de codage offre le pointeur d'identification spécifié.

NOTE – Cela nécessitera en général des restrictions applicables aux valeurs abstraites du type associé.

**68) Paragraphe 23.10.2.2**

Remplacer le § 23.10.2.2 par le texte suivant:

**23.10.2.2** Si "REPLACE STRUCTURE" est activé, dans ce cas aucun autre groupe de propriétés de codage ne doit être activé. Si l'objet de codage de la structure de remplacement fait apparaître un pointeur (avec un ensemble donné de valeurs du pointeur), l'objet de codage défini fait alors apparaître le même pointeur d'identification (avec le même ensemble de valeurs du pointeur – voir § 22.1.1.11).

**69) Paragraphe 23.11.2.3**

*Remplacer le § 23.11.2.3 par le texte suivant:*

**23.11.2.3** Si "REPLACE STRUCTURE" est activé, dans ce cas aucun autre groupe de propriétés de codage ne doit être activé. Si l'objet de codage de la structure de remplacement fait apparaître un pointeur (avec un ensemble donné de valeurs du pointeur), l'objet de codage défini fait alors apparaître le même pointeur d'identification (avec le même ensemble de valeurs du pointeur – voir § 22.1.1.11).

**70) Paragraphe 23.13.2.3**

*Remplacer le § 23.13.2.3 par le texte suivant:*

**23.13.2.3** Si "REPLACE STRUCTURE" est activé, dans ce cas aucun autre groupe de propriétés de codage ne doit être activé. Si l'objet de codage de la structure de remplacement fait apparaître un pointeur (avec un ensemble donné de valeurs du pointeur), l'objet de codage défini fait alors apparaître le même pointeur d'identification (avec le même ensemble de valeurs du pointeur – voir § 22.1.1.11).

**71) Paragraphe 23.13.2.4**

*Remplacer le § 23.13.2.4 par le texte suivant:*

**23.13.2.4** Si "EXHIBITS HANDLE" est activé, ~~eela indique que tous les codages de cette classe offrent dans ce cas~~ l'objet de codage offre le pointeur d'identification spécifié (voir également § 22.9.2.4).

**72) Paragraphe 23.14.2.2**

*Remplacer le § 23.14.2.2 par le texte suivant:*

**23.14.2.2** Si "REPLACE STRUCTURE" est activé, dans ce cas aucune autre spécification ne doit être activée. Si l'objet de codage de la structure de remplacement fait apparaître un pointeur (avec un ensemble donné de valeurs du pointeur), l'objet de codage défini fait alors apparaître le même pointeur d'identification (avec le même ensemble de valeurs du pointeur – voir § 22.1.1.11).



## SÉRIES DES RECOMMANDATIONS UIT-T

Série A	Organisation du travail de l'UIT-T
Série D	Principes généraux de tarification
Série E	Exploitation générale du réseau, service téléphonique, exploitation des services et facteurs humains
Série F	Services de télécommunication non téléphoniques
Série G	Systèmes et supports de transmission, systèmes et réseaux numériques
Série H	Systèmes audiovisuels et multimédias
Série I	Réseau numérique à intégration de services
Série J	Réseaux câblés et transmission des signaux radiophoniques, télévisuels et autres signaux multimédias
Série K	Protection contre les perturbations
Série L	Construction, installation et protection des câbles et autres éléments des installations extérieures
Série M	Gestion des télécommunications y compris le RGT et maintenance des réseaux
Série N	Maintenance: circuits internationaux de transmission radiophonique et télévisuelle
Série O	Spécifications des appareils de mesure
Série P	Qualité de transmission téléphonique, installations téléphoniques et réseaux locaux
Série Q	Commutation et signalisation
Série R	Transmission télégraphique
Série S	Equipements terminaux de télégraphie
Série T	Terminaux des services télématiques
Série U	Commutation télégraphique
Série V	Communications de données sur le réseau téléphonique
<b>Série X</b>	<b>Réseaux de données, communication entre systèmes ouverts et sécurité</b>
Série Y	Infrastructure mondiale de l'information, protocole Internet et réseaux de prochaine génération
Série Z	Langages et aspects généraux logiciels des systèmes de télécommunication