UNIÓN INTERNACIONAL DE TELECOMUNICACIONES

UIT-T X.680

SECTOR DE NORMALIZACIÓN
DE LAS TELECOMUNICACIONES
DE LA UIT

Enmienda 1
(10/2003)

SERIE X: REDES DE DATOS Y COMUNICACIÓN ENTRE SISTEMAS ABIERTOS

Gestión de redes de interconexión de sistemas abiertos y aspectos de sistemas – Notación de sintaxis abstracta uno

---

Tecnología de la información – Notación de sintaxis abstracta uno: Especificación de la notación básica

Enmienda 1: Soporte de EXTENDED-XER

Recomendación UIT-T X.680 (2002) – Enmienda 1

RECOMENDACIONES UIT-T DE LA SERIE X

REDES DE DATOS Y COMUNICACIÓN ENTRE SISTEMAS ABIERTOS

REDES PÚBLICAS DE DATOS	
Servicios y facilidades	X.1–X.19
Interfaces	X.20–X.49
Transmisión, señalización y conmutación	X.50–X.89
Aspectos de redes	X.90–X.149
Mantenimiento	X.150–X.179
Disposiciones administrativas	X.180–X.199
INTERCONEXIÓN DE SISTEMAS ABIERTOS	
Modelo y notación	X.200–X.209
Definiciones de los servicios	X.210–X.219
Especificaciones de los protocolos en modo conexión	X.220–X.229
Especificaciones de los protocolos en modo sin conexión	X.230–X.239
Formularios para declaraciones de conformidad de implementación de protocolo	X.240–X.259
Identificación de protocolos	X.260–X.269
Protocolos de seguridad	X.270–X.279
Objetos gestionados de capa	X.280–X.289
Pruebas de conformidad	X.290–X.299
INTERFUNCIONAMIENTO ENTRE REDES	
Generalidades	X.300–X.349
Sistemas de transmisión de datos por satélite	X.350–X.369
Redes basadas en el protocolo Internet	X.370–X.399
SISTEMAS DE TRATAMIENTO DE MENSAJES	X.400–X.499
DIRECTORIO	X.500–X.599
GESTIÓN DE REDES DE INTERCONEXIÓN DE SISTEMAS ABIERTOS Y ASPECTOS DE SISTEMAS	
Gestión de redes	X.600–X.629
Eficacia	X.630–X.639
Calidad de servicio	X.640–X.649
Denominación, direccionamiento y registro	X.650–X.679
Notación de sintaxis abstracta uno	**X.680–X.699**
GESTIÓN DE INTERCONEXIÓN DE SISTEMAS ABIERTOS	
Marco y arquitectura de la gestión de sistemas	X.700–X.709
Servicio y protocolo de comunicación de gestión	X.710–X.719
Estructura de la información de gestión	X.720–X.729
Funciones de gestión y funciones de arquitectura de gestión distribuida abierta	X.730–X.799
SEGURIDAD	X.800–X.849
APLICACIONES DE INTERCONEXIÓN DE SISTEMAS ABIERTOS	
Compromiso, concurrencia y recuperación	X.850–X.859
Procesamiento de transacciones	X.860–X.879
Operaciones a distancia	X.880–X.899
PROCESAMIENTO DISTRIBUIDO ABIERTO	X.900–X.999

Para más información, véase la Lista de Recomendaciones del UIT-T.

NORMA INTERNACIONAL ISO/CEI 8824-1
RECOMENDACIÓN UIT-T X.680

Tecnología de la información – Notación de sintaxis abstracta uno: Especificación de la notación básica

Enmienda 1
Soporte de EXTENDED-XER

Resumen

La enmienda 1 se aplica a la Rec. UIT-T X.680 | ISO/CEI 8824-1, Rec. UIT-T X.681 | ISO/CEI 8824-2, Rec. UIT-T X.690 | ISO/CEI 8825-1, Rec. UIT-T X.691 | ISO/CEI 8825-2 y Rec. UIT-T X.693 | ISO/CEI 8825-4. Esta enmienda concierne a los siguientes puntos:

- Corrección de un error de programación de CXER resultante de la admisión de un espacio en blanco entre un signo menos y el siguiente valor INTEGER o REAL (CXER no era canónico), que no se permite en la notación de valor, la notación de valor XML, ni en XER y CXER. **Se trata de una modificación y no una adición**.
- Adición de instrucciones de codificación en un módulo ASN.1 utilizando un prefijo de tipo o dentro de la sección de control de codificación para especificar las variaciones de las codificaciones BASIC-XER. Se supone que estas instrucciones de codificación soportan la correspondencia entre una especificación XSD y una especificación ASN.1. Esta disposición supone un cambio en la terminología, donde un tipo precedido de "`[…]`" es un tipo prefijado y la notación "`[…]`" puede o no ser un rótulo. Esta modificación de la terminología supone un cambio en la forma (pero no en el fondo) de las especificaciones BER y PER.
- Adición de nuevos valores no numéricos (NaN, *not-a-number)* y menos cero para REAL (el soporte de la codificación de estos nuevos valores se establece en la enmienda 1 a la Rec. UIT-T X.690 | ISO/CEI 8825-1 y a la Rec. UIT-T X.691 | ISO/CEI 8825-2, así como en la enmienda 1 a la Rec. X.693 | ISO/CEI 8825-4).
- Adición de nuevas notaciones de valor XML para **`REAL`**, **`BOOLEAN`**, **`ENUMERATED`** e **`INTEGER`** que utilizan rótulos de texto y no de elemento vacío para los valores. Estos valores están disponibles para la notación de valor XML y EXTENDED-XER, pero no para BASIC-XER (por motivos de compatibilidad con versiones anteriores).
- Modificación de la notación de valor XML para secuencia de (y las codificaciones XER) para delimitar los valores que no son elementos XML (esto se da con las notaciones de valor XML adicionales y sólo afecta a la utilización de estas notaciones de valor XML adicionales). Esta modificación sólo afecta la utilización de las notaciones de valor XML que se añaden con esta enmienda, y no está autorizada en BASIC-XER, que no se ve afectada.

Estas modificaciones proporcionan el soporte básico necesario de EXTENDED-XER.

Orígenes

La enmienda 1 a la Recomendación UIT-T X.680 (2002) fue aprobada el 29 de octubre de 2003 por la Comisión de Estudio 17 (2001-2004) del UIT-T por el procedimiento de la Recomendación UIT-T A.8. Se publica también un texto idéntico como Norma Internacional ISO/CEI 8824-1, enmienda 1.

PREFACIO

La UIT (Unión Internacional de Telecomunicaciones) es el organismo especializado de las Naciones Unidas en el campo de las telecomunicaciones. El UIT-T (Sector de Normalización de las Telecomunicaciones de la UIT) es un órgano permanente de la UIT. Este órgano estudia los aspectos técnicos, de explotación y tarifarios y publica Recomendaciones sobre los mismos, con miras a la normalización de las telecomunicaciones en el plano mundial.

La Asamblea Mundial de Normalización de las Telecomunicaciones (AMNT), que se celebra cada cuatro años, establece los temas que han de estudiar las Comisiones de Estudio del UIT-T, que a su vez producen Recomendaciones sobre dichos temas.

La aprobación de Recomendaciones por los Miembros del UIT-T es el objeto del procedimiento establecido en la Resolución 1 de la AMNT.

En ciertos sectores de la tecnología de la información que corresponden a la esfera de competencia del UIT-T, se preparan las normas necesarias en colaboración con la ISO y la CEI.

NOTA

En esta Recomendación, la expresión "Administración" se utiliza para designar, en forma abreviada, tanto una administración de telecomunicaciones como una empresa de explotación reconocida de telecomunicaciones.

La observancia de esta Recomendación es voluntaria. Ahora bien, la Recomendación puede contener ciertas disposiciones obligatorias (para asegurar, por ejemplo, la aplicabilidad o la interoperabilidad), por lo que la observancia se consigue con el cumplimiento exacto y puntual de todas las disposiciones obligatorias. La obligatoriedad de un elemento preceptivo o requisito se expresa mediante las frases "tener que, haber de, hay que + infinitivo" o el verbo principal en tiempo futuro simple de mandato, en modo afirmativo o negativo. El hecho de que se utilice esta formulación no entraña que la observancia se imponga a ninguna de las partes.

PROPIEDAD INTELECTUAL

La UIT señala a la atención la posibilidad de que la utilización o aplicación de la presente Recomendación suponga el empleo de un derecho de propiedad intelectual reivindicado. La UIT no adopta ninguna posición en cuanto a la demostración, validez o aplicabilidad de los derechos de propiedad intelectual reivindicados, ya sea por los miembros de la UIT o por terceros ajenos al proceso de elaboración de Recomendaciones.

En la fecha de aprobación de la presente Recomendación, la UIT no ha recibido notificación de propiedad intelectual, protegida por patente, que puede ser necesaria para aplicar esta Recomendación. Sin embargo, debe señalarse a los usuarios que puede que esta información no se encuentre totalmente actualizada al respecto, por lo que se les insta encarecidamente a consultar la base de datos sobre patentes de la TSB.

© UIT 2004

Reservados todos los derechos. Ninguna parte de esta publicación puede reproducirse por ningún procedimiento sin previa autorización escrita por parte de la UIT.

ÍNDICE

Página

1) Introducción 1
2) Subcláusula 2.2 1
3) Nueva subcláusula 3.6.18 *bis* 1
4) Nuevas subcláusulas 3.6.22 *bis* a 3.6.22 *quat* 1
5) Subcláusula 3.6.54 2
6) Subcláusula 3.6.69 2
7) Subcláusula 3.6.71 2
8) Nueva subcláusula 3.6.74 *bis* 2
9) Subcláusula 8.1 2
10) Subcláusula 8.3 2
11) Nueva subcláusula 8 *bis* 3
12) Nueva subcláusula 11.20 *bis* 3
13) Subcláusula 11.23.2 3
14) Nueva subcláusula 11.23 *bis* 3
15) Subcláusula 11.24.2 4
16) Nuevas subcláusulas 11.24 *bis a 11.24 quat* 4
17) Subcláusula 11.25.5 5
18) Cuadro 4 5
19) Subcláusula 11.27 5
20) Subcláusula 12.1 5
21) Subcláusula 12.2 5
22) Nueva subcláusula 12.4 *bis* 5
23) Nueva subcláusula 12.21 6
24) Subcláusula 13.3 6
25) Subcláusula 16.2 6
26) Subcláusula 16.9 6
27) Subcláusula 16.10 6
28) Subcláusula 17.3 6
29) Nueva subcláusula 17.4 6
30) Subcláusula 18.9 6
31) Nueva subcláusula 18.9 *bis* 7
32) Subcláusula 18.10 7
33) Nueva subcláusula 18.12 7
34) Subcláusula 19.8 7
35) Nueva subcláusula 19.8 *bis* 7
36) Subcláusula 19.9 8
37) Subcláusulas 20.3 a 20.6 8
38) Nueva subcláusula 20.6 *bis* 9
39) Subcláusula 20.7 9
40) Subcláusula 21.9 9
41) Nueva subcláusula 21.9 *bis* 9
42) Subcláusula 21.12 10
43) Nueva subcláusula 24.1 *bis* 10
44) Subcláusula 24.2 10

Página

45) Subcláusula 24.3 ... 10
46) Subcláusula 24.5.1 ... 10
47) Subcláusula 24.8 ... 11
48) Subcláusula 24.9 ... 11
49) Subcláusula 25.3 ... 11
50) Subcláusula 25.4 ... 11
51) Cuadro 5 ... 11
52) Nuevas subcláusulas 25.5 *bis* y 25.5 *ter* ... 12
53) Subcláusulas 25.6, 25.7 y 25.8 ... 12
54) Subcláusula 25.11 ... 12
55) Nuevas subcláusulas 25.11.1 a 25.11 3 ... 12
56) Subcláusula 26.3 ... 12
57) Subcláusula 27.3 ... 12
58) Subcláusula 28.2 ... 13
59) Subcláusula 28.3 ... 13
60) Subcláusula 28.4 ... 13
61) Subcláusula 28.5 ... 13
62) Cláusula 30 ... 13
63) Cláusula 38 ... 15
64) Nuevas subcláusulas 38.1.4 *bis* y 38.1.4 *ter* ... 16
65) Subcláusula 45.4 ... 16
66) Subcláusula 47.4.2 ... 17
67) Subcláusula 48.7.1 ... 17
68) Nueva cláusula 50 ... 17
69) Subcláusula A.2.4 ... 17
70) Subcláusula 8.3.2.2 ... 18
71) Subcláusula B.4.2 ... 18
72) Nuevo anexo C *bis* ... 18
73) Subcláusula E.2.12.2 ... 18
74) Anexo H ... 19

NORMA INTERNACIONAL
RECOMENDACIÓN UIT-T

Tecnología de la información – Notación de sintaxis abstracta uno: Especificación de la notación básica

Enmienda 1
Soporte de EXTENDED-XER

NOTA – Todo texto nuevo o modificado en esta enmienda está subrayado en las cláusulas que han de sustituirse. Al fusionar este texto con la Recomendación de base deberán eliminarse las marcas.

1) Introducción

a) *En la introducción, insértese el siguiente párrafo inmediatamente antes del párrafo que comienza por "Una especificación ASN.1 se producirá inicialmente con un conjunto de tipos ASN.1 totalmente definidos.":*

Es, del mismo modo, posible asignar instrucciones de codificación a un tipo para influir en su codificación. Esto puede hacerse utilizando un prefijo de tipo, situado antes de la definición de tipo, o mediante una referencia de tipo o una sección de control de codificación situada al final del módulo ASN.1. La sintaxis genérica de los prefijos de tipo y las secciones de control de codificación se especifica en la presente Recomendación | Norma Internacional, y se incluye una referencia de codificación para identificar las reglas de codificación modificadas por la instrucción de codificación. La semántica y la sintaxis detallada de las instrucciones de codificación se especifican en las reglas de codificación de las Recomendaciones | Normas Internacionales identificadas por la referencia de codificación.

b) *E la introducción, insértese el siguiente párrafo entre los párrafos que comienzan por "anexo C" y "anexo D":*

El anexo C *bis* forma parte integrante de esta Recomendación | Norma Internacional y especifica las referencias de codificación actualmente definidas y las Recomendaciones | Normas Internacionales que definen la semántica y la sintaxis detallada de las instrucciones de codificación con dichas referencias de codificación.

2) Subcláusula 2.2

En la subcláusula 2.2, sustitúyase la nota 1 por la siguiente:

NOTA 1 – Se incluye la referencia *supra* puesto que proporciona los nombres de los caracteres de control y especifica las categorías de los caracteres.

3) Nueva subcláusula 3.6.18 *bis*

Insértese la nueva subcláusula 3.6.18 bis siguiente:

3.6.18 *bis* referencia de codificación por defecto (para un módulo): Referencia de codificación que se especifica en la cabecera del módulo y que se da por supuesta en todos los prefijos de tipo que no contiene una referencia de codificación.

NOTA – Si en la cabecera del módulo no se especifica la referencia de codificación por defecto, todos los prefijos de tipo que no contengan una referencia de codificación se consideran rótulos de asignación.

4) Nuevas subcláusulas 3.6.22 *bis* a 3.6.22 *quat*

Insértense las tres nuevas subcláusulas 3.6.22 bis – 3.6.22 quat siguientes:

3.6.22 *bis* sección de control de codificación: Parte de un módulo ASN.1 que permite la asignación de instrucciones de codificación a los tipos definidos o utilizados dentro de un módulo ASN.1.

3.6.22 *ter* **instrucción de codificación**: Información que puede asociarse a un tipo utilizando un prefijo de tipo o una sección de control de codificación, y que utiliza para la codificación de dicho tipo una o más reglas de codificación ASN.1.

NOTA – Una instrucción de codificación no afecta a los valores abstractos de un tipo y no se supone que sea visible para una aplicación.

3.6.22 *quat* **referencia de codificación**: Un nombre (véase el anexo C *bis*) que identifica las reglas de codificación que se ven afectadas por una instrucción de codificación en un prefijo de tipo o una sección de control de codificación.

NOTA – La referencia de codificación **`TAG`** puede utilizarse para especificar si un prefijo de tipo asigna un rótulo en vez de una instrucción de codificación (véase 30.2).

5) Subcláusula 3.6.54

Sustitúyase la subcláusula 3.6.54 con el siguiente texto:

3.6.54 **tipo real**: Tipo simple cuyos valores distinguidos (especificados en la cláusula 20) incluyen un conjunto de números reales (números reales numéricos) además de valores especiales como **`NOT-A-NUMBER`**.

6) Subcláusula 3.6.69

Sustitúyase la subcláusula 3.6.69 con el siguiente texto:

3.6.69 **rótulo**: Información adicional, distinta de los valores abstractos del tipo, que se asocia con cada tipo ASN.1 y que puede modificarse o aumentarse con un prefijo de tipo.

NOTA – La información de rótulo se utiliza en algunas reglas de codificación para garantizar que las codificaciones son inequívocas. La información de rótulo difiere de las instrucciones de codificación en que la información de rótulo se asocia con todos los tipos ASN.1, incluso si no tienen un prefijo de tipo.

7) Subcláusula 3.6.71

Sustitúyase la subcláusula 3.6.71 con el siguiente texto:

3.6.71 **rotulación**: Asignación de un nuevo rótulo a un tipo, sustituyendo o añadiendo el rótulo existente (posiblemente por defecto).

8) Nueva subcláusula 3.6.74 *bis*

Insértese la nueva subcláusula 3.6.74 bis siguiente:

3.6.74 *bis* prefijo de tipo: Parte de la notación ASN.1 que puede utilizarse para asignar una instrucción de codificación o un rótulo a un tipo.

9) Subcláusula 8.1

Sustitúyase la subcláusula 8.1 con el siguiente texto:

8.1 Un rótulo se especifica (bien en el texto de esta Recomendación | Norma Internacional o utilizando un prefijo de tipo) dando su clase y el número dentro de la clase. La clase es una de las siguientes:

- universal;
- aplicación;
- privada;
- específica al contexto.

10) Subcláusula 8.3

En la subcláusula 8.3 sustitúyase "cláusula 30" *por* "30.2" *y en la nota, sustitúyase* "Cláusula 30" *por* "Subcláusula 30.2".

11) Nueva subcláusula 8 *bis*

Insértese la nueva cláusula 8 bis siguiente:

8 *bis* Instrucciones de codificación

8 *bis*.1 Se asigna una instrucción de codificación a un tipo utilizando un prefijo de tipo (véase 30.3) o una sección de control de codificación (véase la cláusula 50).

8 *bis*.2 Un prefijo de tipo puede contener una referencia de codificación. En caso contrario, la referencia de codificación viene determinada por la referencia de codificación por defecto del módulo (véase 12.4 *bis*).

8 *bis*.3 Una sección de control de codificación siempre contiene una referencia de codificación. Puede haber múltiples secciones de control de codificación, pero cada una de ellas tendrá una referencia de codificación distinta.

8 *bis*.4 Una instrucción de codificación consiste de una secuencia de ítems léxicos especificados en la Recomendación | Norma Internacional determinada por la referencia de codificación (véase anexo C *bis*).

8 *bis*.5 Pueden asignarse múltiples instrucciones de codificación con la misma referencia de codificación, o con distintas referencias de codificación, a un tipo (utilizando prefijos de tipos o una sección de control de codificación, o ambos). Las instrucciones de codificación asignadas con una referencia de codificación determinada son independientes de las asignadas con referencias de codificación distintas, e independientes también de la utilización de un prefijo de tipo para la rotulación.

8 *bis*.6 El efecto de asignar diversas instrucciones de codificación con la misma referencia de codificación (usando prefijos de tipo o una sección de control de codificación, o ambos) se especifica en la Recomendación | Norma Internacional determinada por la referencia de codificación (véase el anexo C *bis*), y no se especifica en la presente Recomendación | Norma Internacional.

8 *bis*.7 Si se asigna una instrucción de codificación a un "Type" en "TypeAssignment", éste se asocia con el tipo y se aplica siempre que se utiliza el "typereference" del "TypeAssignment". También puede utilizarse en otros módulos mediante declaraciones de exportación e importación.

12) Nueva subcláusula 11.20 *bis*

Insértese la nueva subcláusula 11.20 bis siguiente:

11.20 *bis* Referencias de codificación

Nombre del ítem – encodingreference

Un "encodingreference" deberá estar formado por una secuencia de caracteres, como se especifica para el "typereference" en 11.2, excepto que no deberán incluirse letras en minúscula.

NOTA – Las referencias de codificación actualmente definidas se enumeran en el anexo C *bis*, además de la Recomendación | Norma Internacional que especifica la sintaxis y la semántica de las instrucciones de codificación correspondientes. El "en codingreference" deberá estar formado únicamente por las secuencias enumeradas en el anexo C *bis* de esta Recomendación | Norma Internacional o sus futuras versiones.

13) Subcláusula 11.23.2

Sustitúyase la subcláusula 11.23.2 con el siguiente texto:

11.23.2 Al analizar un ejemplar de uso de esta notación, se distingue un "true" de un "valuereference" o un "identifier", o un ejemplar booleano de XML "extended-true", según el contexto en que aparece.

NOTA – Esta secuencia no contiene ningún carácter de espacio en blanco (véase 11.1.2).

14) Nueva subcláusula 11.23 *bis*

Insértese la nueva subcláusula 11.23 bis siguiente:

11.23 *bis* Ítem booleano de XML "extended-true"

Nombre del ítem – extended-true

11.23 *bis*.1 Este ítem estará formado por una secuencia de caracteres:

true

o por un único carácter

1 (DÍGITO UNO)

11.23 *bis*.2 Al analizar un ejemplar de uso de esta notación, se distingue un "extended-true" de un "valuereference" o un "identifier", o un ejemplar booleano de XML "true", según el contexto en que aparece.

NOTA – Esta secuencia no contiene ningún carácter de espacio en blanco (véase 11.1.2).

15) Subcláusula 11.24.2

Sustitúyase la subcláusula 11.24.2 con el siguiente texto:

11.24.2 Al analizar un ejemplar de uso de esta notación, se distingue un "false" de un "valuereference" o un "identifier", o un ejemplar booleano de XML "extended-false", según el contexto en que aparece.

NOTA – Esta secuencia no contiene ningún carácter de espacio en blanco (véase 11.1.2).

16) Nuevas subcláusulas 11.24 *bis a 11.24 quat*

Insértense las nuevas subcláusulas 11.24 bis, ter y quat siguientes:

11.24 *bis* Ítem booleano de XML "extended-false"

Nombre del ítem – extended-false

11.24 *bis*.1 Este ítem estará formado por una secuencia de caracteres:

false

o por un único carácter

0 (DÍGITO CERO)

11.24 *bis*.2 Al analizar un ejemplar de uso de esta notación, se distingue un "false" de un "valuereference" o un "identifier", o un ejemplar booleano de XML "false", según el contexto en que aparece.

NOTA – Esta secuencia no contiene ningún carácter de espacio en blanco (véase 11.1.2).

11.24 *ter* Ítem real de XML "not-a-number"

Nombre del ítem – "NaN"

11.24 *ter*.1 Este ítem estará formado por una secuencia de caracteres:

NaN

11.24 *ter*.2 Al analizar un ejemplar de uso de esta notación, se distingue un "NaN" de cualquier otro ítem léxico que comience por mayúscula según el contexto en que aparece.

NOTA – Esta secuencia no contiene ningún carácter de espacio en blanco (véase 11.1.2).

11.24 *quat* Ítem real de XML "infinity"

Nombre del ítem – "INF"

11.24 *quat*.1 Este ítem estará formado por una secuencia de caracteres:

INF

11.24 *quat*.2 Al analizar un ejemplar de uso de esta notación, se distingue un "INF" de cualquier otro ítem léxico que comience por mayúscula según el contexto en que aparece.

NOTA – Esta secuencia no contiene ningún carácter espacio en blanco (véase 11.1.2)

17) Subcláusula 11.25.5

Sustitúyase la cláusula 11.25.5 con el siguiente texto:

11.25.5 Si el tipo incorporado ASN.1 es un "PrefixedType", el tipo que determina el "xmlasn1typename" será "Type" en el "PrefixedType" (véase 30.1.5). Si éste es a su vez un "PrefixedType", se aplicará sucesivamente la subcláusula 11.25.5.

NOTA – La subcláusula 25.11 especifica los "Type" que han de utilizarse en un "SelectionType" y un "ConstrainedType".

18) Cuadro 4

En el cuadro 4, sustitúyase "TaggedType" *por* "PrefixedType".

19) Subcláusula 11.27

En la subcláusula 11.27, insértense las siguientes tres nuevas palabras reservadas en nuevas celdas del cuadro, según el orden alfabético correspondiente:

ENCODING-CONTROL

INSTRUCTIONS

NOT-A-NUMBER

20) Subcláusula 12.1

En la subcláusula 12.1, sustitúyase la producción de "ModuleDefinition" con el siguiente texto:

ModuleDefinition ::=
ModuleIdentifier
DEFINITIONS
EncodingReferenceDefault
TagDefault
ExtensionDefault
"::="
BEGIN
ModuleBody
EncodingControlSections
END

e insértese la siguiente nueva producción inmediatamente antes de la producción de "TagDefault":

EncodingReferenceDefault ::=
encodingreference INSTRUCTIONS
| **empty**

21) Subcláusula 12.2

En la subcláusula 12.2, en la nota, sustitúyase "Cláusula 30" *por* "Subcláusula 30.2".

22) Nueva subcláusula 12.4 *bis*

Insértese la nueva subcláusula 12.4bis siguiente:

12.4 *bis* El "EncodingReferenceDefault" especifica que el "encodingreference" es la referencia de codificación por defecto del módulo. Si el "EncodingReferenceDefault" es "empty", la referencia de codificación por defecto del módulo es **TAG**.

NOTA – En el anexo C *bis* se enumeran las referencias de codificación permitidas, además de la Recomendación | Norma Internacional que especifica la forma y significado de las instrucciones de codificación correspondientes.

23) Nueva subcláusula 12.21

Insértese la nueva subcláusula 12.21 siguiente:

12.21 En la cláusula 50 se especifica "EncodingControlSections".

24) Subcláusula 13.3

En la nota de la subcláusula 13.3, sustitúyase "rótulos XML" *por "*nombres de rótulos XML*"*.

25) Subcláusula 16.2

En la subcláusula 16.2, sustitúyase, las dos veces que aparece, "TaggedType" *por "*PrefixedType*"*.

26) Subcláusula 16.9

En la subcláusula 16.9, sustitúyase "TaggedValue" *por "*PrefixedValue*"*.

27) Subcláusula 16.10

En la subcláusula 16.10, sustitúyase "XMLTaggedValue" *por* "XMLPrefixedValue".

28) Subcláusula 17.3

Sustitúyase la subcláusula 17.3 por el siguiente texto:

17.3 El valor de un tipo booleano (véase 3.6.73 y 3.6.38) quedará definido por la notación "BooleanValue", o, cuando se utilice como un "XMLValue", por la notación "XMLBooleanValue". Las producciones son:

BooleanValue ::= TRUE | FALSE

XMLBooleanValue ::=
EmptyElementBoolean
| TextBoolean

EmptyElementBoolean ::=
"<" & "true" "/>"
| "<" & "false" "/>"

TextBoolean ::=
extended-true
| extended-false

29) Nueva subcláusula 17.4

Insértese la nueva subcláusula 17.4 siguiente:

17.4 Si en un "XMLValueAssignment" aparece un "EmptyElementBoolean", no habrá ninguna ocurrencia de "TextBoolean" en dicho "XMLValueAssignment".

30) Subcláusula 18.9

Sustitúyase la subcláusula 18.9 por el siguiente texto:

18.9 El valor de un tipo entero quedará definido por la notación "IntegerValue", o, cuando se utilice como un "XMLValue", por la notación "XMLIntegerValue". Las producciones son:

IntegerValue ::=
SignedNumber |
identifier

XMLIntegerValue ::=
XMLSignedNumber
| EmptyElementInteger
| TextInteger

XMLSignedNumber ::=
number
| "-" & number

EmptyElementInteger ::=
"<" & identifier "/>"

TextInteger ::=
identifier

31) Nueva subcláusula 18.9 *bis*

Añádase la nueva subcláusula 18.9 bis siguiente:

18.9 *bis* Si en un "XMLValueAssignment" aparece un "EmptyElementInteger", no habrá ninguna ocurrencia de "TextInteger" en dicho "XMLValueAssignment".

32) Subcláusula 18.10

Sustitúyase la subcláusula 18.10 y su nota con el siguiente texto:

18.10 El "identifier" del "IntegerValue", y de dos últimas alternativas de "XMLIntegerValue", será uno de los "identifier" del "IntegerType" con el que está asociado el valor y representará el número correspondiente.

NOTA – Cuando se referencia un valor entero para el cual se ha definido un "identifier", deberá preferirse la utilización de la forma "identifier" del "IntegerValue" y una de las formas "identifier" del "XMLIntegerValue".

33) Nueva subcláusula 18.12

Añádase la nueva subcláusula 18.12 siguiente:

18.12 No deberá utilizarse la segunda alternativa de "XMLSignedNumber" si el "number" es cero.

34) Subcláusula 19.8

Sustitúyase la subcláusula 19.8 por el siguiente texto:

19.8 El valor de un tipo enumerado quedará definido por la notación "EnumeratedValue", o, cuando se utilice como un "XMLValue", por la notación "XMLEnumeratedValue". Las producciones son:

EnumeratedValue ::= identifier

XMLEnumeratedValue ::=
EmptyElementEnumerated
| TextEnumerated

EmptyElementEnumerated ::= "<" & identifier "/>"

TextEnumerated ::= identifier

35) Nueva subcláusula 19.8 *bis*

Añádase la nueva subcláusula 19.8 bis siguiente:

19.8 *bis* Si en un "XMLValueAssignment" aparece un "EmptyElementEnumerated", no habrá ninguna ocurrencia de "TextEnumerated" en dicho "XMLValueAssignment".

36) Subcláusula 19.9

Sustitúyase la subcláusula 19.9 por el siguiente texto:

19.9 El "identifier" del "EnumeratedValue" y de las dos alternativas de "XMLEnumeratedValue", será igual al "identifier" de la secuencia "EnumeratedType" con la que está asociado el valor.

37) Subcláusulas 20.3 a 20.6

Sustitúyanse las subcláusulas, 20.3, 20.4, 20.5 y 20.6 con el siguiente texto:

20.3 Los valores abstractos del tipo real son los valores especiales **PLUS-INFINITY**, **MINUS-INFINITY** y **NOT-A-NUMBER** además de los números reales numéricos más cero o menos cero, o que pueden ser especificados por la siguiente fórmula que consta de tres enteros, M, B y E:

$$M \times B^E$$

donde M (no cero) se denomina la mantisa, B (2 ó 10) la base y E el exponente. Los valores con B = 2 (valores abstractos "**base**" 2) y B = 10 (valores abstractos "**base**" 10) se definen como valores abstractos distintos. En cualquier otro caso, los valores de $M \times B^E$ que se refieren al mismo valor numérico son un valor abstracto único.

NOTA – Menos cero y más cero son dos valores abstractos distintos para un cero matemático, y los valores abstractos "**base**" 2 y "**base**" 10 son valores abstractos distintos para cualquier otro número real numérico.

20.4 El tipo real tiene un tipo asociado que se utiliza para soportar las notaciones de valor y subtipo de valores numéricos del tipo real (además de la notación para los valores especiales del tipo real y para más cero y menos cero).

NOTA – Las reglas de codificación pueden definir un tipo diferente que se utiliza para especificar codificaciones, o puede especificar codificaciones sin referencia a los tipos asociados. En particular, la codificación en BER y PER proporciona una codificación decimal codificado en binario (BCD, *binary-coded decimal*) si la "base" es 10, y una codificación que permite la transformación hacia y desde las representaciones de coma flotante de los soportes físicos si la "base" es 2.

20.5 El tipo asociado para la definición de valores (y a efectos de subtipificación) de los valores numéricos es (con comentarios normativos):

```
SEQUENCE {
            mantissa    INTEGER (ALL EXCEPT 0),
            base  INTEGER (2|10),
            exponent    INTEGER

            -- The associated mathematical real number is "mantissa"

            -- multiplied by "base" raised to the power "exponent"
        }
```

NOTA 1 – Los valores representados por "**base**" 2 y "**base**" 10 se consideran valores abstractos distintos incluso si equivalen al mismo valor en números reales, y pueden llevar semánticas de aplicación diferentes.

NOTA 2 – La notación **REAL (WITH COMPONENTS { ... , base (10)})** puede utilizarse para restringir el conjunto de valores a valores numéricos en "**base**" 10 (y de manera similar para valores numéricos en "**base**" 2). Esta notación no incluye valores (valores reales especiales y más cero y menos cero) que no pueden representarse con un tipo asociado. De ser necesario, estos valores pueden añadirse utilizando un conjunto aritmético.

NOTA 3 – Este tipo es capaz de llevar una representación finita exacta de cualquier número que pueda ser almacenado en soportes físicos de coma flotante y de cualquier número que tenga una representación finita decimal en caracteres.

20.6 El valor de un tipo real se definirá por la notación "RealValue", o, cuando se utilice en un "XMLValue", por la notación "XMLRealValue":

```
RealValue ::=
        NumericRealValue
    |   SpecialRealValue

NumericRealValue ::=
        realnumber
    |   "-" realnumber
    |   SequenceValue       -- Value of the associated sequence type
```

SpecialRealValue ::=
    PLUS-INFINITY
  | **MINUS-INFINITY**
  | **NOT-A-NUMBER**

NOTA – No puede utilizarse la tercera alternativa de "NumericRealValue" para valores más cero o menos cero. Estos valores abstractos se especifican utilizando la primera o la segunda alternativa, respectivamente, con un único carácter "**0**" para "realnumber".

XMLRealValue ::=
    XMLNumericRealValue | XMLSpecialRealValue

XMLNumericRealValue ::=
    realnumber
  | **"-" & realnumber**

XMLSpecialRealValue ::=
    EmptyElementReal
  | **TextReal**

EmptyElementReal ::=
    "<" & PLUS-INFINITY "/>"
  | **"<" & MINUS-INFINITY "/>"**
  | **"<" & NOT-A-NUMBER "/>"**

TextReal ::=
    "INF"
  | **"-" & "INF"**
  | **"NaN"**

38) Nueva subcláusula 20.6 *bis*

Añádase la nueva subcláusula 20.6 bis siguiente:

20.6 *bis* Si en un "XMLValueAssignment" aparece un "EmptyElementReal", no habrá ninguna ocurrencia de "TextReal" en dicho "XMLValueAssignment".

39) Subcláusula 20.7

Sustitúyase la subcláusula 20.7 con el siguiente texto:

20.7 Cuando se utiliza la notación "realnumber", se identifica el valor abstracto "**base**" 10 correspondiente, o más cero. Cuando un valor "realnumber" va precedido de "-", identifica el valor abstracto "**base**" 10 correspondiente que sea número negativo, o menos cero. Si el "RealType" se reduce a "**base**" 2, el "realnumber" o "-" "realnumber" identifica el valor abstracto "**base**" 2 correspondiente a un valor decimal especificado por el "realnumber" o a una precisión local si no es posible una representación exacta.

40) Subcláusula 21.9

Sustitúyase la subcláusula 21.9 con el siguiente texto:

21.9 El valor de un tipo cadena de bits se definirá por la notación "BitStringValue", o, cuando se utilice como un "XMLValue", por la notación "XMLBitStringValue". Las producciones son:

BitStringValue ::=
    bstring
  | **hstring**
  | **"{" IdentifierList "}"**
  | **"{" "}"**
  | **CONTAINING Value**

IdentifierList ::=
    identifier
  | **IdentifierList "," identifier**

XMLBitStringValue ::=
XMLTypedValue
| xmlbstring
| XMLIdentifierList
| empty

XMLIdentifierList ::=
EmptyElementList
| TextList

EmptyElementList ::=
"<" & identifier "/>"
| EmptyElementList "<" & identifier "/>"

TextList ::
identifier
| TextList identifier

41) Nueva subcláusula 21.9 *bis*

Añádase la nueva subcláusula 21.9 bis siguiente:

21.9 *bis* Si en un "XMLValueAssignment" aparece un "EmptyElementList", no habrá ninguna ocurrencia de "TextList" en dicho "XMLValueAssignment".

42) Subcláusula 21.12

Sustitúyase la subcláusula 21.12 con el siguiente texto:

21.12 El "identifier" de "BitStringValue" o de las alternativas de "XMLBitStringValue", será el mismo que el "identifier" de la frecuencia "BitStringType" con la que está asociado el valor.

43) Nueva subcláusula 24.1 *bis*

Añádase la nueva subcláusula 24.1 bis siguiente:

24.1 *bis* A los efectos de las siguientes cláusulas, se supone que un "PrefixedType" es un tipo rotulado textualmente si:

a) el "PrefixedType" es un "TaggedType"; o

b) el "Type" del "PrefixedType" es un tipo rotulado textualmente.

44) Subcláusula 24.2

Sustitúyase el texto de la subcláusula 24.2 con el siguiente texto (manténganse las dos notas):

24.2 Cuando aparece una producción de "ComponentTypeLists" dentro de la definición de un módulo para el que se ha seleccionado rotulación automática (véase 12.3), y ninguna de las ocurrencias de "NamedType" en cualquiera de las tres primeras alternativas para "ComponentType" es un tipo rotulado textualmente (véase 24.1 *bis*), se selecciona la transformación de rotulación automática para la totalidad de la "ComponentTypeLists"; de no ser así, no se selecciona.

45) Subcláusula 24.3

En la nota de la subcláusula 24.3, sustitúyase "explícita" *por* "textual".

46) Subcláusula 24.5.1

En la subcláusula 24.5.1, sustitúyase "cláusula 30" *con* "30.2".

47) Subcláusula 24.8

Sustitúyase la subcláusula 24.8 con el siguiente texto:

24.8 Si está en vigor la rotulación automática y los "ComponentType" en la raíz de extensión no tienen rótulos, entonces ningún "ComponentType" dentro de la "ExtensionAdditionList" deberá ser un tipo rotulado textualmente.

48) Subcláusula 24.9

En la NOTA 1 de la subcláusula 24.9, sustitúyase "30.6" *con* "30.2.7".

49) Subcláusula 25.3

En la subcláusula 25.3, elimínese la alternativa "XMLSpaceSeparatedList" de la producción "XMLSequenceOfValue"; elimínese la producción "XMLSpaceSeparatedList"; elimínese "1" en la NOTA 1, y elimínese la NOTA 2.

50) Subcláusula 25.4

Después de la subcláusula 25.4, añádase la siguiente nota:

NOTA – Esto ocurre sólo para **`SEQUENCE OF NULL.`**

51) Cuadro 5

Sustitúyase el cuadro 5 por el siguiente cuadro:

Cuadro 5 – Notación "XMLSequenceOfValue" y "XMLSetOfValue" para los tipos ASN.1

Tipo ASN.1	Notación de valor XML
BitStringType	XMLDelimitedItemList
BooleanType	*Véase 25.5 bis*
CharacterStringType	XMLDelimitedItemList
ChoiceType	XMLValueList
EmbeddedPDVType	XMLDelimitedItemList
EnumeratedType	*Véase 25.5 ter*
ExternalType	XMLDelimitedItemList
InstanceOfType	*Véase la Rec. UIT-T X.681 | ISO/CEI 8824-2, C.9*
IntegerType	XMLDelimitedItemList
NullType	XMLValueList
ObjectClassFieldType	*Véase la Rec. UIT-T X.681 | ISO/CEI 8824-2, 14.10 y 14.11*
ObjectIdentifierType	XMLDelimitedItemList
OctetStringType	XMLDelimitedItemList
RealType	XMLDelimitedItemList
RelativeOIDType	XMLDelimitedItemList
SequenceType	XMLDelimitedItemList
SequenceOfType	XMLDelimitedItemList
SetType	XMLDelimitedItemList
SetOfType	XMLDelimitedItemList
PrefixedType	*Véase 25.11.1*
UsefulType (GeneralizedTime)	XMLDelimitedItemList
UsefulType (UTCTime)	XMLDelimitedItemList
UsefulType (ObjectDescriptor)	XMLDelimitedItemList
TypeFromObject	*Véase la Rec. UIT-T X.681 | ISO/CEI 8824-2, 15.6*
ValueSetFromObjects	*Véase la Rec. UIT-T X.681 | ISO/CEI 8824-2, 15.6*

52) Nuevas subcláusulas 25.5 *bis* y 25.5 *ter*

Insértense las nuevas subcláusulas 25.5 bis y 25.5 ter siguientes:

25.5 *bis* Si se utiliza "EmptyElementBoolean" para un valor de un tipo booleano, se utilizará "XMLValueList"; en cualquier otro caso se utilizará "XMLDelimitedItemList".

25.5 *ter* Si se utiliza un "EmptyElementEnumerated" para el valor de un tipo enumerado, se utilizará "XMLValueList", en cualquier otro caso se utilizará "XMLDelimitedItemList".

53) Subcláusulas 25.6, 25.7 y 25.8

Elimínense las subcláusulas 25.6, 25.7 y 25.8. Nótese que estas cláusulas han pasado a ser, con modificaciones menores, como 25.11.1, 25.11.2 y 25.11.3, respectivamente.

54) Subcláusula 25.11

Sustitúyase la subcláusula 25.11 con el siguiente texto:

25.11 Si se utiliza la primera alternativa de "XMLDelimitedItem", entonces, si el componente de un tipo secuencia de (ignorando cualquier ocurrencia de "TypePrefix") es un "typereference" o un "ExternalTypeReference", el "NonParameterizedTypeName" será el "typereference" o el "typereference" del "ExternalTypeReference", respectivamente. En cualquier otro caso será el "xmlasn1typename" especificado en el cuadro 4 correspondiente al tipo incorporado del componente.

55) Nuevas subcláusulas 25.11.1 a 25.11 3

Insértense las tres nuevas subcláusulas 25.11.1, 25.11.2 y 25.11.3 siguientes:

25.11.1 Si el "Type" de un componente es un "PrefixedType", entonces el tipo que determina la alternativa "XMLSequenceOfValue" y el "xmlasn1typename" (si es necesario) será el "Type" del "PrefixedType" (véase 30.1.5). Si éste es un "PrefixedType", un "ConstrainedType" o un "SelectionType", se aplicarán sucesivamente estas subcláusulas de 25.11.

25.11.2 Si el "Type" de un componente es un "ConstrainedType", entonces el tipo que determina la alternativa "XMLSequenceOfValue" y el "xmlasn1typename" (si es necesario) será el "Type" del "ConstrainedType" (véase 45.1). Si éste es un "PrefixedType", un "ConstrainedType" o un "SelectionType", se aplicarán sucesivamente a estas subcláusulas de 25.11.

25.11.3 Si el "Type" de un componente es un "SelectionType", entonces el tipo que determina la alternativa "XMLSequenceOfValue" y el "xmlasn1typename" (si es necesario) será el tipo referenciado por "SelectionType" (véase la cláusula 29). Si éste es un "PrefixedType", un "ConstrainedType" o un "SelectionType", se aplicarán sucesivamente estas subcláusulas de 25.11.

56) Subcláusula 26.3

En la subcláusula 26.3, sustitúyase "cláusula 30" *por* "30.2".

57) Subcláusula 27.3

Sustitúyase la subcláusula 27.3 con el siguiente texto:

27.3 La notación para definir un valor de un tipo conjunto será "SetOfValue", o, cuando se utilice como un "XMLValue", será "XMLSetOfValue". Las producciones son:

```
SetOfValue ::=
        "{" ValueList "}"
    |   "{" NamedValueList "}"
    |   "{" "}"
```

XMLSetOfValue ::=
XMLValueList
| **XMLDelimitedItemList**
| **empty**

"ValueList", "NamedValueList" y las alternativas de "XMLSetOfValue" se especifican en 25.3 y la elección de alternativa es la misma que si se utiliza "XMLSequenceOfValue". La notación "**{**" "**}**" o "empty" se utiliza cuando "SetOfValue" o "XMLSetOfValue" es una lista vacía.

NOTA 1 – No debe darse significado semántico al orden de estos valores.

NOTA 2 – No es necesario que las reglas de codificación conserven el orden de estos valores.

NOTA 3 – El tipo conjunto no es un conjunto matemático de valores, por lo que, por ejemplo, para **SET OF INTEGER** los valores **{ 1 }** y **{ 1 1 }** son distintos.

58) Subcláusula 28.2

Sustitúyase la subcláusula 28.2 con el siguiente texto:

28.2 Cuando la producción "AlternativeTypeLists" ocurre dentro de la definición de un módulo para el que se ha seleccionado rotulación automática (véase 12.3), y ninguna de las ocurrencias de "NamedType" en ella es un tipo rotulado textualmente (véase 24.1 b*is*), se selecciona la transformación de rotulación automática para la totalidad de la "AlternativeTypeLists", de no ser así, no se selecciona.

59) Subcláusula 28.3

En la subcláusula 28.3 sustitúyase "cláusula 30" *por* "30.2".

60) Subcláusula 28.4

En la subcláusula 28.4, sustitúyase "tipo rotulado" *con* "tipo rotulado textualmente".

61) Subcláusula 28.5

*En la subcláusula 28.5, en la nota 1, sustitúyase "*30.6*" con "*30.2.7*".*

62) Cláusula 30

Sustitúyase la cláusula 30, y sus subcláusulas correspondientes, con el siguiente texto:

30 Notación para tipos prefijados

30.1 Aspectos generales

30.1.1 Un tipo prefijado es un tipo nuevo que es isomórfico con un tipo antiguo, pero que tiene un rótulo diferente o adicional y puede tener instrucciones de codificación asociadas diferentes o adicionales.

30.1.2 Un tipo prefijado es un "TaggedType" o un "EncodingPrefixedTyp-e".

30.1.3 Un tipo prefijado que sea un tipo rotulado se utiliza principalmente donde esta Recomendación | Norma Internacional requiera el empleo de tipos con rótulos distintos (véanse 24.5 a 24.6, 26.3 y 28.3). Mediante el empleo de un "TagDefault" de **AUTOMATIC TAGS** en un módulo puede conseguirse esto sin que aparezca explícitamente en este módulo un "TaggedType".

NOTA – Cuando un protocolo determina que pueden transmitirse valores de varios tipos de datos en cualquier momento, pueden necesitarse rótulos distintos para hacer posible que el receptor decodifique correctamente el valor.

30.1.4 La asignación de una instrucción de codificación utilizando un "EncodingPrefixedType" sólo es pertinente a las codificaciones identificadas por la referencia de codificación asociada y no afecta a los valores abstractos del tipo.

30.1.5 La notación para un tipo prefijado será "PrefixedType":

PrefixedType ::=
TaggedType
| EncodingPrefixedType

NOTA – La especificación de la sintaxis de "PrefixedType" será más simple y clara si el rotulado se describe como la asignación de una instrucción de codificación. No obstante, tradicionalmente, el rotulado se introdujo en las primeras versiones de las especificaciones ASN.1 y puede afectar a la legalidad de la definición de tipo. Se realizaron cambios mínimos en los conceptos de rotulado (y a las descripciones sintácticas conexas) al introducir la codificación de tipos prefijados. El rotulado difiere asimismo sintácticamente de la asignación de instrucciones de codificación: la especificación de que el rotulado es **EXPLICIT** o **IMPLICIT** ocurre tras el cierre "]" del rótulo, y no dentro de éste, "[" y "]", como ocurre con las instrucciones de codificación normales.

30.1.6 La notación de un valor de "PrefixedType" será "PrefixedValue", o, cuando se utilice como un "XMLValue", será "XMLPrefixedValue". Las producciones son:

PrefixedValue ::= Value

XMLPrefixedValue ::= XMLValue

donde "Value" o "XMLValue" es una notación para un valor de "Type" en el "TaggedType" o el "EncodingPrefixedType" del "PrefixedType".

NOTA 1 – Ni el "Tag" ni ninguna parte del "EncodingPrefix" aparecen en esta notación.

NOTA 2 – También pueden asignarse instrucciones de codificación a un tipo en una sección de control de codificaciones (véase la cláusula 50). Esta asignación no afecta a la notación de valor del tipo.

30.2 Tipo rotulado

30.2.1 La notación para un tipo rotulado será "TaggedType":

TaggedType ::=
Tag Type
| Tag IMPLICIT Type
| Tag EXPLICIT Type

Tag ::= "[" EncodingReference Class ClassNumber "]"

EncodingReference ::=
encodingreference ":"
| empty

ClassNumber ::=
number
| DefinedValue

Class ::=
UNIVERSAL
| APPLICATION
| PRIVATE
| empty

30.2.2 Dentro de un "Tag", el "encodingreference" será **TAG**. El "EncodingReference" del "Tag" no podrá ser "empty", a menos que la referencia de codificación por defecto para el módulo sea **TAG** (véase 12.4 *bis*).

30.2.3 La "valuereference" del "DefinedValue" deberá ser de tipo entero y se le asignará un valor no negativo.

30.2.4 El nuevo tipo es isomórfico con el tipo antiguo, pero tiene un rótulo con la clase "Class" y el número "ClassNumber", a menos que la "Class" sea "empty", en cuyo caso el rótulo es de clase específica del contexto y el número es "ClassNumber".

30.2.5 La "Class" no será **UNIVERSAL** salvo para los tipos definidos en esta Recomendación | Norma Internacional.

NOTA 1 – El uso de rótulos de clase universal es aprobado cada cierto tiempo por el UIT-T y la ISO.

NOTA 2 – La subcláusula E.2.12 contiene directrices y sugerencias sobre la utilización estilística de las clases de rótulos.

30.2.6 Toda aplicación de rótulos es rotulación implícita o rotulación explícita. La rotulación implícita indica, para aquellas reglas de codificación que proporcionan la opción, que la identificación explícita del rótulo del "Type" del "TaggedType" no se necesita durante la transferencia.

NOTA – Puede ser útil retener el rótulo antiguo cuando sea de clase universal y, por lo tanto, identifique sin ambigüedad el tipo antiguo sin conocer la definición ASN.1 del tipo nuevo. La transferencia mínima de octetos, sin embargo, se consigue normalmente con el uso de **IMPLICIT**. En la Rec. UIT-T X.690 | ISO/CEI 8825-1 se da un ejemplo de una codificación utilizando **IMPLICIT**.

30.2.7 La construcción de la rotulación especifica rotulación explícita si se cumple cualquiera de los siguientes requisitos:

a) se utiliza la alternativa "Tag **EXPLICIT** Type";

b) se utiliza la alternativa "Tag Type" y el valor de "TagDefault" para el módulo es **EXPLICIT TAGS** o está vacío;

c) se utiliza la alternativa "Tag Type" y el valor de "TagDefault" para el módulo es **IMPLICIT TAGS** o **AUTOMATIC TAGS**, pero el tipo definido por "Type" es un tipo choice (selección), un tipo open (abierto), o una "DummyReference" sin rótulo (véase la Rec. UIT-T X.683 | ISO/CEI 8824-4, 8.3).

En todos los demás casos, la construcción de rotulación especifica rotulación implícita.

30.2.8 Si la "Class" es "empty", no hay otras restricciones a la utilización del "Tag" que las implicadas por los requisitos de rótulos distintos de 24.5 a 24.6, 26.3 y 28.3.

30.2.9 La alternativa **IMPLICIT** no deberá utilizarse si el tipo definido por "Type" es un tipo choice o un tipo open o una "DummyReference" sin rótulo (véase la Rec. UIT-T X.683 | ISO/CEI 8824-4, 8.3).

30.3 Tipo prefijado de codificación

30.3.1 La notación para un tipo prefijado de codificación será "EncodingPrefixedType":

EncodingPrefixedType ::=
EncodingPrefix Type

EncodingPrefix ::=
"[" EncodingReference EncodingInstruction "]"

"EncodingReference" se define en 30.2.1.

30.3.2 La producción de "EncodingInstruction" se especifica en la Recomendación | Norma Internacional identificada por el "EncodingReference" (véase el anexo C *bis*) y puede estar formado por cualquier secuencia de ítems léxicos ASN.1 (con inclusión de comment, cstring y white-space).

NOTA 1 – Los ítems léxicos "`[`" y "`]`" no aparecen nunca en "EncodingInstruction".

NOTA 2 – En futuras versiones de esta Recomendación | Norma Internacional podrán añadirse referencias de codificación al anexo C *bis*. Se recomienda que las aplicaciones ASN.1 (únicamente) hagan una advertencia si un "encodingreference" no es ninguno de los especificados en el anexo C *bis* y, entonces, ignore todo el "EncodingPrefix" que tenga un "`]`" al final (véase la NOTA 1 *supra*).

30.3.3 Si el "EncodingReference" está vacío, entonces la referencia de codificación para el prefijo de codificación es la referencia de codificación por defecto para el módulo.

NOTA – Si la referencia de codificación por defecto para el módulo es **TAG** (véase 30.2.2) y el "EncodingReference" es "empty", entonces el "PrefixedType" es un "TaggedType" y no un "EncodingPrefixedType".

30.3.4 En general, se imponen restricciones a las instrucciones de codificación (con la misma referencia de codificación) que pueden utilizarse en combinación, y restricciones a los tipos a los que pueden aplicarse instrucciones o combinaciones de instrucciones particulares. Estas restricciones se especifican en la Recomendación | Norma Internacional asociada con la referencia de codificación (véase el anexo C *bis*) y no se especifican en la presente Recomendación | Norma Internacional.

63) Cláusula 38

Sustitúyase la cláusula 38 con el siguiente texto:

38 Denominación de caracteres, colecciones y conjuntos de categorías de propiedades

Esta cláusula especifica un módulo incorporado ASN.1 que contiene la definición de un nombre de referencia de valor para cada carácter de ISO/CEI 10646-1, donde cada carácter referencia un valor **UniversalString** de tamaño 1. Este módulo contiene también la definición de un nombre de referencia de tipo para cada colección de caracteres de ISO/CEI 10646-1, donde cada nombre referencia un subconjunto de **UniversalString**. Por último, contiene la

definición de un nombre "typereference" para cada conjunto de caracteres de cada categoría general de propiedades de caracteres enumerados en 4.5 de la norma Unicode, donde cada nombre referencia un subconjunto de **`UniversalString`**.

NOTA – Estos valores están disponibles para uso en la notación de valor del tipo **`UniversalString`** y de los tipos derivados de éste. Todas las referencias de valores y tipos definidas en el módulo especificado en 38.1 son exportadas y tienen que ser importadas por cualquier módulo que las utilice.

64) Nuevas subcláusulas 38.1.4 *bis* y 38.1.4 *ter*

Insértense las nuevas subcláusulas 38.1.4 bis y 38.1.4 ter siguientes:

38.1.4 ***bis*** Para cada abreviatura y cada descripción enumeradas en la norma Unicode, cuadro 4-5, se incluyen dos enunciados en el módulo de la forma:

<categoryabbreviation> ::= UniversalString (FROM (<alternativelist>))

```
-- represents the set of characters with the property
-- category <categoryabbreviation>.
```

<categorydescription> UniversalString ::= <categoryabbreviation>

donde:

a) <categoryabbreviation> es la abreviatura de la categoría general de propiedades de caracteres enumerada en la norma Unicode, cuadro 4-5 (por ejemplo, **`Lu`** o **`Nd`** o **`Pi`**);

b) <categorydescription> es la descripción de la misma categoría general de caracteres donde la inicial de todas las palabras va en mayúscula, se eliminan las comas y los espacios, y los paréntesis en las descripciones (por ejemplo, **`LetterUppercase`** o **`NumericDigit`** o **`PunctuationInitialQuote`**);

c) La <alternativelist> de cada <categoryabbreviation> es una lista de nombres <namedcharacter> producidos por 38.2 para cada uno de los caracteres enumerados en la base de datos Unicode Character (versión 3.2.0) de la norma Unicode, que tiene una <categoryabbreviation> correspondiente.

NOTA – El nombre Unicode para un carácter es el mismo que el <iso10646name> para dicho carácter.

38.1.4 ***ter*** Para la inicial de cada abreviatura enumerada en la norma Unicode, cuadro 4-5, se incluyen dos enunciados en el módulo de la forma:

<categoryabbreviationletter> ::= UniversalString (FROM (<alternativelist>))

```
-- represents the set of characters with any category property
-- with the initial letter <categoryabbreviationletter>.
```

<maincategorydescription> UniversalString ::= <categoryabbreviationletter>

donde:

a) <categoryabbreviationletter> es la primera letra de la abreviatura de una categoría general de propiedades de caracteres enumerada en la norma Unicode, cuadro 4-5 (por ejemplo, **`L`** o **`N`** o **`P`**);

b) <categorydescription> es la primera palabra de la descripción de la misma categoría general de caracteres (por ejemplo, **`Letter`** o **`Numeric`** o **`Punctuation`**);

c) La <alternativelist> de cada <categoryabbreviationletter> es una lista de nombres <namedcharacter> producidos por 38.2 para cada carácter enumerado en la base de datos Unicode Character (versión 3.2.0) de la norma Unicode, que tiene una <categoryabbreviationletter> correspondiente.

NOTA – El nombre Unicode para un carácter es el mismo que él <iso10646name> para dicho carácter.

65) Subcláusula 45.4

*En la subcláusula 45.4 sustitúyase, las dos veces que aparece, "*TaggedType*" por "*PrefixedType*".*

66) Subcláusula 47.4.2

Sustitúyase la nota de la subcláusula 47.4.2 con el siguiente texto:

NOTA – A los fines de la subtipificación, **NOT-A-NUMBER** es mayor que todos los valores reales, **PLUS–INFINITY** es mayor que todos los valores reales excepto **NOT-A-NUMBER**, menos cero es mayor que todos los valores reales negativos y es menor que más cero, y **MINUS-INFINITY** es menor que todos los valores reales. En cualquier otro caso, se aplica el orden matemático normal.

67) Subcláusula 48.7.1

*En la subcláusula 48.7.1a), sustitúyase "*encabezamiento*" por "*cabecera*".*

68) Nueva cláusula 50

Insértese la nueva cláusula 50 siguiente:

50 Secciones de control de codificación

50.1 Las "EncodingControlSections" se especifican en las siguientes producciones:

EncodingControlSections ::=
EncodingControlSection EncodingControlSections
|empty

EncodingControlSection ::=
ENCODING-CONTROL
encodingreference
EncodingInstructionAssignmentList

50.2 Cada "EncodingControlSection" dentro de un módulo ASN.1 tendrá distintas "encodingreference", y asignará instrucciones de codificación para dicha referencia de codificación a uno o más tipos en el módulo.

50.3 La "encodingreference" no será **TAG**.

50.4 La producción de "EncodingInstructionAssignmentList" y la semántica asociada se especifica en la Recomendación | Norma Internacional identificada por la "encodingreference" (véase el anexo C *bis*) y podrá estar formada por cualquier secuencia de ítems léxicos ASN.1 (con inclusión de comment, cstring y white-space) excepto los ítems léxicos **END** y **ENCODING-CONTROL**, que no aparecerán en una "EncodingInstructionAssignmentList".

NOTA 1 – Futuras versiones de esta Recomendación | Norma Internacional podrán añadir otras referencias de codificación al anexo C *bis*. Se recomienda que las aplicaciones ASN.1 (únicamente) hagan una advertencia si la "EncodingControlSection" de una "EncodingControlSection" no es ninguna de las especificadas en el anexo C *bis* y, entonces, ignore la "EncodingControlSection", hasta la siguiente ocurrencia de **END** o **ENCODING-CONTROL**, lo que suceda primero.

NOTA 2 – No puede omitirse la "encodingreference" en una "EncodingControlSection". La referencia de codificación por defecto del módulo no afecta a la "EncodingControlSection".

50.5 Existen interacciones y restricciones en la asignación de instrucciones de codificación (con la misma referencia de codificación) para un tipo que utilice un prefijo de tipo y una "EncodingControlSection". Siempre es posible (por cuestiones de estilo) utilizar únicamente "EncodingControlSection", pero, en general, hay algunas instrucciones de codificación (en concreto las que se aplican a todos los tipos de un módulo) que tan sólo pueden asignarse en una "EncodingControlSection". Del mismo modo hay restricciones para los tipos a los que pueden aplicarse instrucciones o combinaciones de instrucciones particulares. Estas interacciones y restricciones se especifican en la Recomendación | Norma Internacional asociada con la referencia de codificación (véase el anexo C *bis*), y no se especifican en la presente Recomendación | Norma Internacional.

69) Subcláusula A.2.4

En la subcláusula A.2.4, añádase el siguiente texto al final de la cláusula:

Las expresiones normales "**\N{LetterUppercase}" y "\N{Lu}**" se adaptan a cualquier carácter (único) de la categoría general "Letter, uppercase" (cuya abreviatura es "Lu") como se define en la norma Unicode.

70) Subcláusula 8.3.2.2

En la subcláusula B.3.2.2, sustitúyase, las dos veces que aparece, "cláusula 30" *por* "30.2".

71) Subcláusula B.4.2

En la subcláusula B.4.2, sustitúyase "cláusula 30" *por* "30.2".

72) Nuevo anexo C *bis*

Insértese el nuevo anexo C bis siguiente:

Anexo C *bis*

Referencias de codificación

(Este anexo forma parte integrante de la presente Recomendación | Norma Internacional)

C *bis*.1 En este anexo se especifican las referencias de codificación actualmente definidas y la Recomendación | Norma Internacional que especifica la forma sintáctica (y la semántica) de las instrucciones de codificación con dicha referencia de codificación (excepto para la referencia de codificación **TAG**, que no tiene ninguna instrucción de codificación asociada).

NOTA – Se recomienda que, si una referencia de codificación no especificada aquí aparece en una especificación ASN.1, se ignoren las instrucciones de codificaciones asociadas y se haga (únicamente) un diagnóstico de advertencia.

C *bis*.2 Las referencias de codificación de la columna 1 del cuadro C *bis*.1 ya están definidas. La sintaxis y la semántica de las instrucciones de codificación asociadas (si procede) se definen en la Recomendación | Norma Internacional que figura en la columna 2 del cuadro C *bis*.1.

Cuadro C *bis*.1 – Normas que definen la semántica asociada con una determinada referencia de codificación

Referencia de codificación	Referencia a la norma
TAG	La presente Recomendación | Norma Internacional
XER	Rec. UIT-T X.693 (2001), enmienda 1 (10/2003) | ISO/CEI 8825-4 (2002)/enmienda 1: 2004 EXTENDED-XER

73) Subcláusula E.2.12.2

Sustitúyase la subcláusula E.2.12.2 con el siguiente texto:

E.2.12.2 Un estilo de utilización de los rótulos, que se da frecuentemente, consiste en asignar un rótulo de clase de aplicación exactamente una vez en la totalidad de la especificación, utilizándolo para especificar un tipo que se emplea de una manera amplia y dispersa dentro de la especificación. También se utiliza frecuentemente (solamente una vez) un rótulo de clase de aplicación para rotular los tipos de la **CHOICE** más externa de una aplicación, proporcionando la identificación de mensajes individuales mediante el rótulo de clase de aplicación. Lo que sigue es un ejemplo de utilización en el primer caso:

EJEMPLO

```
FileName ::= [APPLICATION 8] SEQUENCE {
                    directoryName                 VisibleString,
                    directoryRelativeFileName     VisibleString}
```

En este ejemplo se supone que la referencia de codificación por defecto es "empty" o **TAG**. En cualquier otro caso, el ejemplo sería el siguiente:

```
FileName ::= [TAG: APPLICATION 8] SEQUENCE {
                    directoryName                 VisibleString,
                    directoryRelativeFileName     VisibleString}
```

Sería necesaria una modificación semejante en los ejemplos posteriores.

74) Anexo H

Háganse los siguientes cambios en el anexo H:

a) *Insértense los siguientes ítems en la lista de ítems léxicos:*

encodingreference

extended-true

extended-false

"NaN"

"INF"

ENCODING-CONTROL

INSTRUCTIONS

NOT-A-NUMBER

b) *Sustitúyase la producción de "XMLBooleanValue" con las siguientes tres producciones:*

```
XMLBooleanValue ::=
        EmptyElementBoolean
    |   TextBoolean

EmptyElementBoolean ::=
        "<" & "true" "/>"
    |   "<" & "false" "/>"

TextBoolean  ::=
        extended-true
    |   extended-false
```

c) *Sustitúyanse la producción de "XMLIntegerValue" con las siguientes cuatro producciones:*

```
XMLIntegerValue ::=
            XMLSignedNumber
    |       EmptyElementInteger
    |       TextInteger

XMLSignedNumber ::=
            number
    |       "-" & number

EmptyElementInteger ::=
            "<" & identifier "/>"

TextInteger ::=
            identifier
```

d) *Sustitúyase la producción de "XMLEnumeratedValue" con las siguientes tres producciones:*

```
XMLEnumeratedValue ::=
        EmptyElementEnumerated
    |   TextEnumerated

EmptyElementEnumerated ::= "<" & identifier "/>"

TextEnumerated ::= identifier
```

e) *Sustitúyanse las producciones de "XMLNumericRealValue" y "XMLSpecialRealValue" con las siguientes cuatro producciones:*

XMLNumericRealValue ::=
    realnumber
| **"-" & realnumber**

XMLSpecialRealValue ::=
    EmptyElementReal
| **TextReal**

EmptyElementReal ::=
    "<" & PLUS-INFINITY "/>"
| **"<" & MINUS-INFINITY "/>"**
| **"<" & NOT-A-NUMBER "/>"**

TextReal ::=
    "INF"
| **"-" & "INF"**
| **"NaN"**

f) *Sustitúyase la producción de "XMLIdentifierList" con las siguientes tres producciones:*

XMLIdentifierList ::=
    EmptyElementList
| **TextList**

EmptyElementList ::=
    "<" & identifier "/>"
| **EmptyElementList "<" & identifier "/>"**

TextList ::=
    identifier
| **TextList identifier**

g) *En la producción de "XMLSequenceOfValue", elimínese la alternativa "XMLSpaceSeparatedList".*

h) *Elimínese la producción de "XMLSpaceSeparatedList".*

i) *En la producción de "XMLSetOfValue", elimínese la alternativa "XMLSpaceSeparatedList".*

j) *Insértense las siguientes producciones después de la producción de "SelectionType":*

PrefixedType ::=
    TaggedType
| **EncodingPrefixedType**

EncodingPrefixedType ::=
    EncodingPrefix Type

EncodingPrefix ::=
    "[" EncodingReference EncodingInstruction "]"

PrefixedValue ::= Value

XMLPrefixedValue ::= XMLValue

k) *Sustitúyase la producción de "Tag" con las siguientes dos producciones:*

Tag ::= "[" EncodingReference Class ClassNumber "]"

EncodingReference ::=
    encodingreference ":"
| **empty**

l) *Elimínense las producciones de "TaggedValue" y "XMLTaggedValue".*

SERIES DE RECOMENDACIONES DEL UIT-T

Serie A	Organización del trabajo del UIT-T
Serie B	Medios de expresión: definiciones, símbolos, clasificación
Serie C	Estadísticas generales de telecomunicaciones
Serie D	Principios generales de tarificación
Serie E	Explotación general de la red, servicio telefónico, explotación del servicio y factores humanos
Serie F	Servicios de telecomunicación no telefónicos
Serie G	Sistemas y medios de transmisión, sistemas y redes digitales
Serie H	Sistemas audiovisuales y multimedios
Serie I	Red digital de servicios integrados
Serie J	Redes de cable y transmisión de programas radiofónicos y televisivos, y de otras señales multimedios
Serie K	Protección contra las interferencias
Serie L	Construcción, instalación y protección de los cables y otros elementos de planta exterior
Serie M	RGT y mantenimiento de redes: sistemas de transmisión, circuitos telefónicos, telegrafía, facsímil y circuitos arrendados internacionales
Serie N	Mantenimiento: circuitos internacionales para transmisiones radiofónicas y de televisión
Serie O	Especificaciones de los aparatos de medida
Serie P	Calidad de transmisión telefónica, instalaciones telefónicas y redes locales
Serie Q	Conmutación y señalización
Serie R	Transmisión telegráfica
Serie S	Equipos terminales para servicios de telegrafía
Serie T	Terminales para servicios de telemática
Serie U	Conmutación telegráfica
Serie V	Comunicación de datos por la red telefónica
Serie X	**Redes de datos y comunicación entre sistemas abiertos**
Serie Y	Infraestructura mundial de la información, aspectos del protocolo Internet y Redes de la próxima generación
Serie Z	Lenguajes y aspectos generales de soporte lógico para sistemas de telecomunicación

Impreso en Suiza
Ginebra, 2004