



UNIÓN INTERNACIONAL DE TELECOMUNICACIONES

UIT-T

SECTOR DE NORMALIZACIÓN
DE LAS TELECOMUNICACIONES
DE LA UIT

X.694

(01/2004)

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 – Reglas de
codificación de notación de sintaxis abstracta
uno: Correspondencia en notación de sintaxis
abstracta uno de las definiciones W3C de los
esquemas de lenguaje de marcaje extensible**

Recomendación UIT-T X.694

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
SEGURIDAD DE LAS TELECOMUNICACIONES	X.1000–

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

**Tecnología de la información – Reglas de codificación de notación de sintaxis abstracta uno:
Correspondencia en notación de sintaxis abstracta uno de las definiciones W3C
de los esquemas de lenguaje de marcaje extensible**

Resumen

La presente Recomendación | Norma Internacional define reglas para establecer la correspondencia de un esquema XSD (un esquema conforme a la especificación de esquema XML W3C con un esquema ASN.1 con el fin de utilizar reglas de codificación ASN.1 como las reglas de codificación básica (BER, *basic encoding rules*), reglas de codificación distinguida (DER, *distinguished encoding rules*), reglas de codificación compactada (PER, *packed encoding rules*) o reglas de codificación XML (XER, *XML encoding rules*) para la transferencia de información definida por el esquema XSD.

La utilización de esta Recomendación | Norma Internacional con las reglas de codificación XML ampliada (EXTENDED-XER) proporciona la misma representación XML de valores que la definida en el esquema XSD original.

Orígenes

La Recomendación UIT-T X.694 fue aprobada el 13 de enero de 2004 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 8825-5.

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	Alcance	1
2	Referencias normativas	1
2.1	Recomendaciones Normas Internacionales idénticas	1
2.2	Referencias adicionales	2
3	Definiciones	2
3.1	Definiciones importadas	2
3.2	Definiciones adicionales	3
4	Abreviaturas	3
5	Notación	3
6	Objeto y alcance de la normalización	3
7	Correspondencia de esquemas XSD	4
8	Componentes y propiedades de esquema ignorados	6
9	El módulo ASN.1 y los espacios nominales	6
10	Conversión de nombres	7
10.1	Generalidades	7
10.2	Generación de definiciones de tipo ASN.1 que son referencias a asignaciones de tipo ASN.1	7
10.3	Generación de identificadores y nombres de referencia de tipo	8
10.4	Orden en la correspondencia	9
11	Correspondencia de usos de tipos de datos incorporados XSD	11
12	Correspondencia de facetas	12
12.1	Las facetas length, minLength y maxLength	12
12.2	La faceta pattern	12
12.3	La faceta whiteSpace	12
12.4	La faceta enumeration	13
12.5	Otras facetas	14
13	Correspondencia de definiciones de tipo simple	15
14	Correspondencia de declaraciones de elemento	17
15	Correspondencia de declaraciones de atributo	18
16	Correspondencia de valores de definiciones de tipo simple	18
17	Correspondencia de definiciones de grupo de modelo	18
18	Correspondencia de grupos de modelo	18
19	Correspondencia de partículas	19
20	Correspondencia de definiciones de tipo complejas	20
21	Correspondencia de comodines	21
22	Correspondencia de usos de atributo	22
23	Correspondencia de usos de definiciones de tipo simple y de tipo compleja (caso general)	22
24	Correspondencias de usos especiales de definiciones de tipo simple y de tipo compleja (sustituible)	23
25	Correspondencia de usos especiales de definiciones de tipo simple y de tipo compleja (sustituibles, anulables)	24
26	Correspondencia de usos especiales de definiciones de tipo simples (anulables)	25
27	Correspondencia de usos especiales de definiciones de tipo complejas (anulables)	26
28	Correspondencia de usos especiales de declaraciones de elemento (cabecera de grupo de sustitución de elemento)	27
29	Generación de asignaciones de tipo ASN.1 especiales para declaraciones de elemento	27
30	Generación de asignaciones de tipo ASN.1 especiales para definiciones de tipo	28
31	Generación de asignaciones de tipo ASN.1 especial para grupos de sustitución de elemento	29

Anexo A – Definiciones de tipo ASN.1 correspondientes a tipos de datos incorporados XSD	30
Anexo B – Asignación de valores de identificador de objeto	35
Anexo C – Ejemplos de correspondencias.....	36
C.1 Esquema que utiliza definiciones de tipo simples	36
C.2 Definiciones ASN.1 correspondientes	37
C.3 Otros ejemplos	38
C.3.1 Documentos de esquema con los elementos de información import e include	38
C.3.2 Correspondencia de definiciones de tipo simples	39
C.3.3 Correspondencia de facetas.....	40
C.3.4 Correspondencia de declaraciones de elemento	43
C.3.5 Correspondencia de usos de atributo y de declaraciones de atributo	48
C.3.6 Correspondencia de definiciones de grupo de modelo	49
C.3.7 Correspondencia de partículas	49
C.3.8 Correspondencia de definiciones de tipo complejas	51
C.3.9 Correspondencia de comodines.....	56
Anexo D – Uso de la correspondencia para proporcionar codificaciones binarias para el esquema XML W3C	58
D.1 Codificación de esquemas XSD	58
D.2 Transferencia sin utilizar el esquema XSD para los esquemas	58
D.3 Transferencia que utiliza el esquema XSD para los esquemas.....	58

Introducción

La presente Recomendación | Norma Internacional especifica el establecimiento de una correspondencia entre una definición de esquema XML del W3C (un esquema XSD) y un esquema ASN.1. La correspondencia se puede aplicar a cualquier esquema XSD. Especifica la generación de uno o más módulos ASN.1 con definiciones de tipo e instrucciones de codificación XER ASN.1. Estas se describen conjuntamente como un esquema ASN.1 para documentos XML.

Este esquema ASN.1, cuando se utiliza con las reglas de codificación XML ampliada ASN.1 (EXTENDED-XER), se puede utilizar para generar y validar el mismo conjunto de documentos XML 1.0 W3C que el esquema XSD original. Los tipos y codificaciones ASN.1 resultantes sustentan el mismo contenido semántico que el esquema XSD. Por lo tanto, se pueden intercambiar las herramientas ASN.1 con las herramientas XSD para la generación y procesamiento de los documentos XML especificados.

Junto con esta correspondencia normalizada se pueden utilizar otras reglas de codificación ASN.1 normalizadas, tales como las reglas de codificación distinguida (DER) o las reglas de codificación compactada (PER).

La combinación de esta Recomendación | Norma Internacional con las reglas de codificación ASN.1 proporciona codificaciones binarias totalmente normalizadas e independientes del proveedor, compactas y canónicas para datos definidos mediante un esquema XSD.

El esquema ASN.1 proporciona una separación clara entre la especificación del contenido de información de los mensajes (su sintaxis abstracta) y la forma precisa del documento XML (por ejemplo, utilización de atributos en lugar de elementos), lo que da como resultado un esquema más claro y generalmente menos complejo que el esquema XSD original.

El anexo A es parte integrante de la presente Recomendación | Norma Internacional y es un módulo ASN.1 que contiene un conjunto de asignaciones de tipo ASN.1 que se corresponden con cada uno de los tipos de datos imbuidos XSD. La correspondencia entre esquemas XSD y esquemas ASN.1 introducen los nombres de referencia de tipo de dichas asignaciones de tipo o incluyen las definiciones de tipo correspondientes.

El anexo B no es parte integrante de la presente Recomendación | Norma Internacional y resume los valores de los identificadores de objetos asignados a esta Recomendación | Norma Internacional.

El anexo C no es parte integrante de la presente Recomendación | Norma Internacional y ofrece ejemplos de la correspondencia de esquemas XSD con esquemas ASN.1.

El anexo D no es parte integrante de la presente Recomendación | Norma Internacional y describe la utilización de las correspondencias definidas en esta Recomendación | Norma Internacional, junto con reglas de codificación ASN.1 normalizadas, para facilitar codificaciones compactas y canónicas para determinados datos que utilizan un esquema XSD.

**NORMA INTERNACIONAL
RECOMENDACIÓN UIT-T**

**Tecnología de la información – Reglas de codificación de notación de sintaxis abstracta uno:
Correspondencia en notación de sintaxis abstracta uno de las definiciones W3C
de los esquemas de lenguaje de marcaje extensible**

1 Alcance

La presente Recomendación | Norma Internacional especifica una correspondencia entre cualquier esquema XSD y un esquema ASN.1. El esquema ASN.1 soporta la misma semántica y valida el mismo conjunto de documentos XLM.

Esta Recomendación | Norma Internacional especifica las instrucciones finales de codificación XER que deben aplicarse y que forman parte de la correspondencia definida con tipos ASN.1, pero no especifica la forma sintáctica que se debe utilizar para la especificación de dichas instrucciones finales de codificación XER, o el orden y manera de su asignación.

NOTA – Los que utilicen las herramientas que generan estas correspondencias pueden elegir cualquier forma sintáctica u orden de asignación que logre la aplicación de las instrucciones finales de codificación XER especificadas. Los ejemplos en esta Recomendación | Norma Internacional utilizan generalmente la forma de prefijo de tipo, pero puede ser preferible, por cuestiones de estilo, utilizar la sección de control de codificación XER para la correspondencia de un esquema XSD completo.

Existen diferentes formas (sintácticas) para asignar instrucciones de codificación XER para su uso en codificaciones EXTENDED-XER (por ejemplo, utilización de instrucciones de codificación de prefijo de tipo ASN.1 o utilización de una sección de control de codificación XER). La elección de estas formas sintácticas es una cuestión de estilo y se encuentra fuera del ámbito de la presente Recomendación | Norma Internacional.

2 Referencias normativas

Las siguientes Recomendaciones | Normas Internacionales y especificaciones W3C contienen disposiciones que, mediante su referencia en este texto, constituyen disposiciones de la presente Recomendación | Norma Internacional. Al efectuar esta publicación, estaban en vigor las ediciones indicadas. Todas las Recomendaciones, Normas Internacionales y especificaciones W3C son objeto de revisiones, por lo que se preconiza que los participantes en acuerdos basados en la presente Recomendación | Norma Internacional investiguen la posibilidad de aplicar las ediciones más recientes de las Recomendaciones, Normas Internacionales y especificaciones W3C citadas a continuación. Los miembros de la CEI y de la ISO mantienen registros de las Normas Internacionales actualmente vigentes. La Oficina de Normalización de las Telecomunicaciones de la UIT mantiene una lista de las Recomendaciones UIT-T actualmente vigentes. El W3C mantiene una lista de las especificaciones W3C actualmente vigentes. En esta Recomendación | Norma Internacional, la referencia a un documento, en tanto que autónomo, no le otorga el rango de una Recomendación o Norma Internacional.

2.1 Recomendaciones | Normas Internacionales idénticas

NOTA – El conjunto completo de Recomendaciones | Normas Internacionales ASN.1 se numera a continuación, puesto que puede ser aplicable en su totalidad para determinados usos de la presente Recomendación | Norma Internacional. Se añade el símbolo † a la referencia cuando no están directamente referencias en el texto de la presente Recomendación | Norma Internacional.

- Recomendación UIT-T X.680 (2002) | ISO/CEI 8824-1:2002, *Tecnología de la información – Notación de sintaxis abstracta uno: Especificación de la notación básica.*
- Recomendación UIT-T X.681 (2002) | ISO/CEI 8824-2:2002, *Tecnología de la información – Notación de sintaxis abstracta uno: Especificación de objetos de información.* †
- Recomendación UIT-T X.682 (2002) | ISO/CEI 8824-3:2002, *Tecnología de la información – Notación de sintaxis abstracta uno: Especificación de constricciones.*
- Recomendación UIT-T X.683 (2002) | ISO/CEI 8824-4:2002, *Tecnología de la información – Notación de sintaxis abstracta uno: Parametrización de especificaciones de notación de sintaxis abstracta uno.* †

- Recomendación UIT-T X.690 (2002) | ISO/CEI 8825-1:2002, *Tecnología de la información – Reglas de codificación de notación de sintaxis abstracta uno: Especificación de las reglas de codificación básica, de las reglas de codificación canónica y de las reglas de codificación distinguida*. †
- Recomendación UIT-T X.691 (2002) | ISO/CEI 8825-2:2002, *Tecnología de la información – Reglas de codificación de notación de sintaxis abstracta uno: Especificación de las reglas de codificación compactada*. †
- Recomendación UIT-T X.692 (2002) | ISO/CEI 8825-3:2002, *Tecnología de la información – Reglas de codificación de notación de sintaxis abstracta uno: Notación de control de codificación*. †
- Recomendación UIT-T X.693 (2001) | ISO/CEI 8825-4:2002, *Tecnología de la información – Reglas de codificación de notación de sintaxis abstracta uno: Reglas de codificación del lenguaje de marcaje extensible*.
- Recomendación UIT-T X.693 (2001)/enm.1 (2003) | ISO/CEI 8825-4:2002/enm.1:2004, *Tecnología de la información – Reglas de codificación ASN.1: Reglas de codificación del lenguaje de marcaje extensible – Enmienda 1: Instrucciones de codificación XER y EXTENDED-XER*.

2.2 Referencias adicionales

- ISO 8601:2000, *Data elements and interchange formats – Information interchange – Representation of dates and times*.
 - ISO/CEI 10646-1:2000, *Information technology – Universal Multiple-Octet Coded Character Set (UCS) – Part 1: Architecture and Basic Multilingual Plane*.
 - W3C XML 1.0:2000, *Extensible Markup Language (XML) 1.0 (Second Edition)*, W3C Recommendation, Copyright © [6 October 2000] World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University), <http://www.w3.org/TR/2000/REC-xml-20001006>.
 - W3C XML Namespaces:1999, *Namespaces in XML*, W3C Recommendation, Copyright © [14 January 1999] World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University), <http://www.w3.org/TR/1999/REC-xml-names-19990114>.
 - W3C XML Information Set:2001, *XML Information Set*, W3C Recommendation, Copyright © [24 October 2001] World Wide Web Consortium (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University), <http://www.w3.org/TR/2001/REC-xml-info-set-20011024>.
 - W3C XML Schema:2001, *XML Schema Part 1: Structures*, W3C Recommendation, Copyright © [2 May 2001] World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University), <http://www.w3.org/TR/2001/REC-xmlschema-1-20010502>.
 - W3C XML Schema:2001, *XML Schema Part 2: Datatypes*, W3C Recommendation, Copyright © [2 May 2001] World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University), <http://www.w3.org/TR/2001/REC-xmlschema-2-20010502>.
- NOTA – Cuando en esta Recomendación | Norma Internacional se utiliza la referencia "esquema XML W3C", se refiere a la parte 1 del esquema XML del W3C y a la parte 2 del esquema XML del W3C.
- IETF RFC 2396 (1998), *Uniform Resource Identifiers (URI): Generic Syntax*.
 - IETF RFC 1766 (1995), *Tags for the Identification of Languages*.

3 Definiciones

3.1 Definiciones importadas

3.1.1 La presente Recomendación | Norma Internacional utiliza los términos definidos en la Rec. UIT-T X.680 | ISO/CEI 8824-1 y en la Rec. UIT-T X.693 | ISO/CEI 8825-4.

NOTA – En esas Recomendaciones | Normas Internacionales se definen en particular los términos "instrucciones finales de codificación XER", "prefijo de tipo" y "sección de control de codificación XER".

3.1.2 La presente Recomendación | Norma Internacional también utiliza los términos definidos en el esquema XML W3C y en el conjunto de información XML W3C.

NOTA 1 – Se considera que estos términos no entran en conflicto con los términos referenciados en 3.1.1. Si se produjera dicho conflicto, aplica la definición del término en 3.1.1.

NOTA 2 – En particular, se definen en el esquema XML W3C los términos "componente de esquema" y "propiedad (de un componente de esquema)" y en el conjunto de información XML W3C el término "entidad elemento de información".

3.2 Definiciones adicionales

Para los fines de esta Recomendación | Norma Internacional se aplican las definiciones siguientes.

3.2.1 espacio nominal XSD: Un espacio nominal con un URI de "http://www.w3.org/2001/XMLSchema".

3.2.2 espacio nominal XSI: Un espacio nominal con un URI de "http://www.w3.org/2001/XMLSchema-instance".

3.2.3 espacio nominal XML: Un espacio nominal con un URI de "http://www.w3.org/XML/1998/namespace".

4 Abreviaturas

A los fines de esta Recomendación | Norma Internacional, se utilizan las siguientes abreviaturas:

ASN.1	Notación de sintaxis abstracta uno (<i>abstract syntax notation one</i>)
BER	Reglas de codificación básica (ASN.1) [(ASN.1) <i>basic encoding rules</i>]
DER	Reglas de codificación distinguida (ASN.1) [(ASN.1) <i>distinguished encoding rules</i>]
PER	Reglas de codificación compactada (ASN.1) [(ASN.1) <i>packed encoding rules</i>]
URI	Identificador de recursos uniforme (IETF) [(IETF) <i>uniform resource identifier</i>]
XER	Reglas de codificación XML (ASN.1) [(ASN.1) <i>XML encoding rules</i>]
XML	Lenguaje de marcaje extensible (W3C) [(W3C) <i>extensible markup language</i>]
XSD	Esquema XML (W3C) [(W3C) <i>XML schema</i>]

5 Notación

5.1 La presente Recomendación | Norma Internacional hace referencia a la notación definida en la Rec. UIT-T X.680 | ISO/CEI 8824-1, en la Rec. UIT-T X.682 | ISO/CEI 8824-3, en W3C XML 1.0 y en el esquema XML W3C.

5.2 Cuando es necesario especificar en el texto de esta Recomendación | Norma Internacional, formalmente o mediante ejemplos, la asignación de instrucciones de codificación XER, se utiliza generalmente la notación de prefijo de tipo (véanse, no obstante, 6.3 y 6.4). En el anexo A, se utiliza una sección de control de codificación XER.

5.3 En esta Recomendación | Norma Internacional se utiliza **Courier en negrita** para la notación ASN.1 y **Arial en negrita** para la notación XSD y para los términos y conceptos XSD.

5.4 Los esquemas XSD utilizados en los ejemplos en esta Recomendación | Norma Internacional usan el prefijo **xsd:** para identificar el espacio nominal XSD.

6 Objeto y alcance de la normalización

6.1 La correspondencia con ASN.1 que se especifica en esta Recomendación | Norma Internacional asegura que:

- a) cualesquiera módulos ASN.1 resultantes generados mediante herramientas conformes a esta Recomendación | Norma Internacional (a partir del mismo esquema XSD) definen los mismos valores abstractos (estructurados);
- b) todos los BASIC-XER, CXER, EXTENDED-XER y codificaciones binarias de dicha especificación ASN.1 resultante generarán las mismas codificaciones (sujetas a las opciones del codificador); y
- c) todos los documentos XML conformes al esquema XSD de origen son codificaciones EXTENDED-XER válidas de valores abstractos de dicha especificación ASN.1.

6.2 Existen muchos aspectos de una definición ASN.1 (tales como la utilización de espacios en blanco o de secciones de control de codificación o de prefijos de tipo) que no afectan ni a los valores abstractos que se están definiendo ni a las codificaciones XER o binarias de dichos valores. Estos aspectos de la definición ASN.1 generalmente no están normalizados en esta Recomendación | Norma Internacional.

6.3 Existen muchas formas diferentes en ASN.1 para asignar una instrucción de codificación XER a un tipo, incluidas:

- a) la utilización de un prefijo de tipo para toda instrucción de codificación a asignar; o
- b) la utilización de una sección de control de codificación, con una instrucción de codificación diferenciada para cada asignación requerida; o
- c) la utilización de una sección de control de codificación con una única instrucción de codificación que establece una asignación global, complementada normalmente mediante la utilización de una instrucción de codificación negativa para tipos específicos.

6.4 Esta Recomendación | Norma Internacional especifica cuándo debe estar presente una instrucción final de codificación XER y utiliza la sintaxis de 6.3 a) en la mayoría de sus ejemplos. Sin embargo, la utilización de las diferentes opciones de 6.3 no está normalizada y las implementaciones de conformidad de la correspondencia pueden elegir una forma sintáctica (o una mezcla de formas sintácticas) para una asignación de instrucciones finales de codificación XER.

NOTA – La elección entre estas opciones no afecta a las codificaciones finales binaria o XML.

6.5 No se proporciona una especificación formal de la correspondencia requerida.

6.6 Esta Recomendación | Norma Internacional trata exclusivamente de la correspondencia de esquemas XSD conformes con el esquema XML W3C.

NOTA – Esta conformidad se puede lograr mediante la aportación de uno o más documentos de esquema XSD W3C o por otros medios como se especifica en el esquema XML W3C.

7 Correspondencia de esquemas XSD

7.1 Una correspondencia se basa en un esquema XSD de origen, que es un conjunto de componentes de esquema (véase 2.2 de la parte 1 del esquema XML W3C). Para la correspondencia no se requiere ni se presupone ninguna representación particular de los componentes del esquema o de los conjuntos de componentes del esquema, aunque normalmente se supone que se proporciona el esquema XSD de origen como uno o más documentos de esquema XML (véase 3.15.2 de la parte 1 del esquema XML W3C).

NOTA 1 – Puesto que la correspondencia se define en términos de componentes de esquema (y no en términos de su representación XML), no está afectada por detalles de la representación XML, tales como la utilización de múltiples documentos de esquema vinculados por entidades elemento de información **xsd:include** y **xsd:redefine**, la ubicación de los elementos información en uno u otro documentos de esquema, el orden de los elementos de información **xsd:attribute** en una entidad elemento de información **xsd:complexType** y así sucesivamente.

NOTA 2 – Dos conjuntos de documentos de esquema que difieran en muchos aspectos pero representen el mismo conjunto de componentes de esquema generan el mismo conjunto de asignaciones de tipo ASN.1, con las mismas instrucciones finales de codificación asignadas a ellos y a todos sus componentes.

7.2 El esquema XSD de origen cumplirá todas las restricciones impuestas por la especificación XSD. Si el esquema XSD de origen está representado (en parte o en su totalidad) como un conjunto de documentos de esquema XML, cada documento de esquema será válido de conformidad con el esquema XSD para esquemas (véase el apéndice A de la parte 1 del esquema XML W3C).

7.3 Se generará por lo menos un módulo ASN.1 (véase 7.4) para cada **espacio nominal de destino** (ya sea un nombre de espacio nominal o la palabra clave **ausente**) que es el **espacio nominal de destino** de uno o más componentes de esquema en el esquema XSD de origen. Cada módulo ASN.1 contendrá una o más asignaciones de tipo correspondientes a componentes de esquema de nivel superior (véase 7.9) que tienen el mismo **espacio nominal de destino**. Cada ASN.1 también puede contener una o más asignaciones de tipo ASN.1 especiales cuyas asignaciones de tipo ASN.1 asociadas estén en el mismo módulo ASN.1 (véase 7.6).

NOTA – Los componentes de esquema representados en los diversos documentos de esquema llegan a formar parte del mismo esquema XSD mediante la utilización de las entidades elemento de información **xsd:include**, **xsd:redefine** y **xsd:import**.

7.4 El número de módulos ASN.1 generados para cada **espacio nominal de destino** (incluida la palabra clave **ausente**) pueden ser más de uno, pero cada módulo ASN.1 debe incluir asignaciones de tipo correspondientes a componentes de esquema de nivel superior con diferentes **espacios nominales de destino** (incluida la palabra clave **ausente**).

7.5 Cuando se generan múltiples módulos ASN.1 para un determinado **espacio nominal de destino** (incluida la palabra clave **ausente**), todas las asignaciones de tipo presentes en ellos serán generadas como si se añadieran a un único módulo ASN.1 con el fin de generar nombres de referencia de tipo distintos (véase 10.3). Los nombres de referencia de tipo generados a partir de **nombres** de componentes de esquema de nivel superior con un determinado **espacio nominal de destino** serán los mismos nombres de referencia de tipo, independientemente del número de módulos ASN.1

generados para dicho **espacio nominal de destino** y de la forma en que estén divididas las asignaciones de tipo entre los diversos módulos ASN.1.

NOTA – Esto se realiza para proporcionar flexibilidad sin comprometer la interoperabilidad.

7.6 Cada asignación de tipo ASN.1 especial (véanse las cláusulas 29, 30 y 31) se insertará en el mismo módulo ASN.1 que su asignación de tipo ASN.1 asociada (véanse 29.4, 31.4 y 30.4, respectivamente).

7.7 Todos los módulos ASN.1 generados mediante la correspondencia contendrán (en la sección de control de codificación XER) una instrucción de codificación **GLOBAL-DEFAULTS MODIFIED-ENCODINGS** y una instrucción de codificación **GLOBAL-DEFAULTS CONTROL-NAMESPACE** que especifiquen el espacio nominal XSI.

7.8 Un esquema XSD de origen se procesará de la forma siguiente:

- para cada **declaración de elemento** de nivel superior, se generará una asignación de tipo ASN.1 aplicando la cláusula 14 a la **declaración de elemento**;
- para cada **declaración de atributo** de nivel superior, se generará una asignación de tipo ASN.1 aplicando la cláusula 15 a la **declaración de atributo**;
- para cada **definición de tipo simple** de nivel superior definida por el usuario, se generará una asignación de tipo ASN.1 aplicando la cláusula 13 a la **definición de tipo simple**;
- para cada **definición de tipo compleja** de nivel superior, se generará una asignación de tipo ASN.1 aplicando la cláusula 20 a la **definición de tipo compleja**;
- para cada **definición de grupo de modelo** cuyo grupo de modelo tenga un **compositor** de **secuencia** o de **elección**, se generará una asignación de tipo ASN.1 aplicando la cláusula 17 a la **definición de grupo de modelo**.

NOTA 1 – Los componentes de esquema restantes del esquema XSD de origen se procesarán a partir de la correspondencia de estos componentes de esquema.

NOTA 2 – El orden en el que se establece la correspondencia de los componentes de esquema se especifica en 10.4. El orden de los elementos de la lista anterior no tiene significado para la correspondencia.

7.9 La columna 1 del cuadro 1 enumera los componentes de esquema. La columna 2 indica la referencia a la cláusula en el esquema XML W3C que define el componente. La columna 3 enumera la cláusula que define la correspondencia de esos componentes de esquema en ASN.1

Cuadro 1 – Correspondencia de componentes de esquema XSD

Componente de esquema XSD	Referencia al esquema XML W3C	Definido por
attribute declaration (declaración de atributo)	Parte 1, 3.2	Cláusula 15
element declaration (declaración de elemento)	Parte 1, 3.3	Cláusula 14
complex type definition (definición de tipo compleja)	Parte 1, 3.4	Cláusula 20
attribute use (uso de atributo)	Parte 1, 3.5	Cláusula 22
attribute group definition (definición de grupo de atributo)	Parte 1, 3.6	<i>sin correspondencia como tal</i>
model group definition (definición de grupo de modelo)	Parte 1, 3.7	Cláusula 17
model group (grupo de modelo)	Parte 1, 3.8	Cláusula 18
particle (partícula)	Parte 1, 3.9	Cláusula 19
wildcard (comodín)	Parte 1, 3.10	Cláusula 21
identity-constraint definition (definición de construcción de identidad)	Parte 1, 3.11	<i>ignorado por la correspondencia</i>
notation declaration (declaración de notación)	Parte 1, 3.12	<i>ignorado por la correspondencia</i>
annotation (anotación)	Parte 1, 3.13	<i>ignorado por la correspondencia</i>
simple type definition (definición de tipo simple)	Parte 1, 3.14	Cláusulas 11 y 13
schema (esquema)	Parte 1, 3.15	Cláusula 9
ordered (ordenado)	Parte 2, 4.2.2.1	<i>ignorado por la correspondencia</i>
bounded (vinculado)	Parte 2, 4.2.3.1	<i>ignorado por la correspondencia</i>

Cuadro 1 – Correspondencia de componentes de esquema XSD

Componente de esquema XSD	Referencia al esquema XML W3C	Definido por
cardinality (cardinalidad)	Parte 2, 4.2.4.1	<i>ignorado por la correspondencia</i>
numeric (numérico)	Parte 2, 4.2.5.1	<i>ignorado por la correspondencia</i>
length (longitud)	Parte 2, 4.3.1.1	Cláusula 12
minLength (longitud mínima)	Parte 2, 4.3.2.1	Cláusula 12
maxLength (longitud máxima)	Parte 2, 4.3.3.1	Cláusula 12
pattern (plantilla)	Parte 2, 4.3.4.1	Cláusula 12
enumeration (enumeración)	Parte 2, 4.3.5.1	Cláusula 12
whiteSpace (espacio en blanco)	Parte 2, 4.3.6.1	Cláusula 12
maxInclusive (máximo incluido)	Parte 2, 4.3.7.1	Cláusula 12
maxExclusive (máximo excluido)	Parte 2, 4.3.8.1	Cláusula 12
minExclusive (mínimo excluido)	Parte 2, 4.3.9.1	Cláusula 12
minInclusive (mínimo incluido)	Parte 2, 4.3.10.1	Cláusula 12
totalDigits (dígitos totales)	Parte 2, 4.3.11.1	Cláusula 12
fractionDigits (dígitos fraccionarios)	Parte 2, 4.3.12.1	Cláusula 12

8 Componentes y propiedades de esquema ignorados

8.1 La correspondencia ignorará los componentes y propiedades del esquema que se enumeran en esta cláusula.

8.2 Se ignorarán todas las **anotaciones** (véase 3.13 de la parte 1 del esquema XML W3C).

NOTA – Todos los elementos de información de atributo en un documento de esquema con nombres calificados con espacios nominales distintos de los espacios nominales XSD (véase 3.13.1 de la parte 1 del esquema XML W3C) son propiedad de **anotaciones**, y se ignoran.

8.3 Se ignorarán todas las **definiciones identidad-restricción** (véase 3.11 de la parte 1 del esquema XML W3C).

NOTA – La **definición identidad-restricción** proporciona mecanismos para especificar restricciones referenciales que pueden ser necesarias en un ejemplar válido. ASN.1 actualmente no tiene conceptos de dichas restricciones y no se pueden establecer correspondencias en una especificación ASN.1 formal, pero pueden estar incluidas como comentarios normativos vinculados con una aplicación de implementación.

8.4 Se ignorarán todas las **declaraciones de notación** (véase 3.12 de la parte 1 del esquema XML W3C).

8.5 Todos los componentes de esquema que sean **facetas fundamentales** (**ordenadas**, **vinculadas**, **cardinalidad**, **numéricas**) de **definiciones de tipo simples** (véase 4.2 de la parte 2 del esquema XML W3C) deben ignorarse.

8.6 Se ignorarán las propiedades **definiciones** de **identidad-restricción**, **exclusiones de grupo de sustitución** y **sustituciones no permitidas de declaraciones de elemento**.

8.7 Se ignorarán las propiedades **final**, **abstracto** y **sustituciones prohibidas** de las **definiciones de tipo complejas**.

8.8 Se ignorará la propiedad **contenidos de procesos** de los **comodines**.

NOTA – No existe ningún soporte en ASN.1 para ninguna acción que no sea **skip**.

8.9 Se ignorarán las propiedades **facetas fundamentales** y **finales** de las **definiciones de tipo simples**.

8.10 Se ignorarán todas las **restricciones de valor** que estén presentes en cualquier **declaración de elemento** o **declaración de atributo** cuya **definición de tipo** sea **xsd:QName** o una **definición de tipo simple** derivada de **xsd:QName** o **xsd:NOTATION**.

8.11 Se ignorarán todas las **definiciones de grupo de atributos**.

NOTA – Los **usos de atributo** en una **definición de grupo de atributos** forman parte de los **usos de atributo** de las **definiciones de tipo complejas** cuya representación XML contenga una referencia a la **definición de grupo de atributos**.

9 El módulo ASN.1 y los espacios nominales

NOTA – En la cláusula 16 de la Rec. UIT-T X.693 | ISO/CEI 8825-4 se incluye una descripción completa de la relación entre el concepto de espacio nominal en XML W3C y la denominación en ASN.1. Se asigna un espacio nominal a los nombres de referencia de tipo y a los identificadores definidos en un módulo ASN.1 mediante una instrucción de codificación **NAMESPACE**, en

otros caso no disponen de espacio nominal. La correspondencia genera instrucciones de codificación **NAMESPACE** cuando es necesario.

9.1 La correspondencia genera uno o más módulos ASN.1 relativos a todos los componentes de esquema en el esquema que tiene el mismo **espacio nominal de destino**.

9.2 El "ModuleIdentifier" ASN.1 (véase la cláusula 12 de la Rec. UIT-T X.680 | ISO/CEI 8824-1) que debe generar la correspondencia no está normalizado. Cuando se utilizan declaraciones **IMPORTS**, los nombres de los módulos ASN.1 y los identificadores de módulos en las declaraciones **IMPORTS** serán los generados para los módulos ASN.1 generados por la correspondencia.

NOTA – La elección de "ModuleIdentifier" no afecta a las codificaciones en ninguna de sus reglas de codificación normalizadas.

9.3 Los módulos ASN.1 tendrán un "TagDefault" de **AUTOMATIC TAGS**.

9.4 En cada módulo ASN.1, existirá una declaración ASN.1 **IMPORTS** que importe los nombres de referencia de tipo ASN.1 en el módulo denominado **XSD**, especificado en el anexo A, que se encuentra referenciado en el módulo ASN.1.

9.5 La declaración **IMPORTS** también importará los nombres de referencia de tipo ASN.1 de asignaciones de tipo que se estén situados (a consecuencia de la correspondencia) en otros módulos ASN.1 pero que estén referenciados en ese módulo ASN.1.

9.6 No habrá declaraciones **EXPORTS**.

NOTA – Esto significa que todos los nombres de referencia de tipo ASN.1 en el módulo ASN.1 se pueden importar en otros módulos.

10 Conversión de nombres

10.1 Generalidades

10.1.1 La presente Recomendación | Norma Internacional especifica la generación de:

- nombres de referencia de tipo ASN.1 correspondientes a los **nombres** y de **definiciones de grupo de modelo**, **declaraciones de elemento**, **declaraciones de atributo** de nivel superior, **definiciones de tipo complejas** de nivel superior y **definiciones de tipo simples** de nivel superior definidas por el usuario;
- identificadores ASN.1 correspondientes a los **nombres** de **declaraciones de elemento** de nivel superior, **declaraciones de atributo** de nivel superior, **declaraciones de elemento** locales y **declaraciones de atributo** locales;
- identificadores ASN.1 para la correspondencia de ciertas **definiciones de tipo simples** con una faceta de **enumeración** (véanse 12.4.1 y 12.4.2);
- nombres de referencia de tipo ASN.1 de asignaciones de tipo especiales (véanse las cláusulas 29, 30 y 31); y
- identificadores ASN.1 de ciertos componentes de secuencia introducidos por la correspondencia (véase la cláusula 20).

10.1.2 Todos estos nombres ASN.1 son generados mediante la aplicación de 10.3 ya sea al **nombre** del componente de esquema correspondiente o a un miembro del **valor** de una faceta de **enumeración**, o a una cadena de caracteres especificada, como se indica en las cláusulas pertinentes en la presente Recomendación | Norma Internacional.

10.2 Generación de definiciones de tipo ASN.1 que son referencias a asignaciones de tipo ASN.1

10.2.1 Esta subcláusula se aplica cuando la invocan explícitamente otras cláusulas de la presente Recomendación | Norma Internacional para generar una definición de tipo ASN.1 que es una referencia ("DefinedType") a una asignación de tipo ASN.1.

10.2.2 Si una definición de tipo ASN.1 (por ejemplo, R) que es un "DefinedType" debe insertarse en un módulo ASN.1 (por ejemplo, M) distinto del módulo ASN.1 al que se inserta la asignación de tipo ASN.1 referenciada (por ejemplo, TA) y el nombre de referencia de tipo de TA es idéntico al nombre de referencia de tipo de otra asignación de tipo ASN.1 que se está insertando en el módulo M o de otro nombre de referencia de tipo que se está importando al módulo M, entonces, R será una "ExternalTypeReference" (construida convenientemente para el módulo M) para TA, en otro caso será una "typereference" para TA.

10.3 Generación de identificadores y nombres de referencia de tipo

10.3.1 Esta subcláusula se aplica cuando la invocan explícitamente otras cláusulas de la presente Recomendación | Norma Internacional para generar un nombre o un identificador de referencia de tipo ASN.1.

10.3.2 Los **nombres de declaraciones de atributo**, las **declaraciones de elemento**, las **definiciones de grupo de modelo**, las **definiciones de tipo simples** de nivel superior definidas por el usuario y las **definiciones de tipo complejas** de nivel superior pueden ser idénticas a las palabras reservadas ASN.1 o pueden contener caracteres no permitidos en identificadores ASN.1 o en nombres de referencia de tipo ASN.1. Además, existen casos en los que los nombres ASN.1 tienen que ser distintos cuando está permitido que los **nombres** de los componentes de esquema XSD correspondientes (de los que se obtienen los nombres ASN.1 mediante correspondencia) sean idénticos.

10.3.3 Se aplicarán las transformaciones siguientes, en su orden, para cada cadena de caracteres a la que se está estableciendo una correspondencia a un nombre ASN.1, donde cada transformación (salvo la primera) se aplica al resultado de la transformación anterior:

- los caracteres " " (ESPACIO), "." (PUNTO), y "_" (GUIÓN BAJO) deben sustituirse todos por "-" (SIGNO MENOS); y
- cualquier carácter salvo "A" a "Z" (MAYÚSCULA LATINA A a MAYÚSCULA LATINA Z), "a" a "z" (MINÚSCULA LATINA A a MINÚSCULA LATINA Z), "0" a "9" (DÍGITO CERO a DÍGITO NUEVE), y "-" (SIGNO MENOS) deben suprimirse; y
- una secuencia de dos o más caracteres SIGNO MENOS serán sustituidas por un único SIGNO MENOS; y
- se suprimirán los caracteres SIGNO MENOS que se encuentren al principio o al final del nombre; y
- si una cadena de caracteres que se tiene que utilizar como nombre de referencia de tipo empieza con una letra minúscula, la primera letra estará en mayúsculas (convertida en una mayúscula); si empieza con un dígito (DÍGITO CERO a DÍGITO NUEVE) deberá tener un prefijo con un carácter "x" (MAYÚSCULA LATINA X); y
- si una cadena de caracteres que se tiene que utilizar como identificador comienza con una letra mayúscula, la primera letra estará en minúsculas (convertida en una minúscula); si empieza con un dígito (DÍGITO CERO a DÍGITO NUEVE), deberá tener como prefijo un carácter "x" (MINÚSCULA LATINA X); y
- si la cadena de caracteres que se tiene que utilizar como nombre de referencia de tipo está vacía, se sustituirá por "x" (MAYÚSCULA LATINA X); y
- si la cadena de caracteres que se tiene que utilizar como identificador está vacía, se sustituirá por "x" (MINÚSCULA LATINA X).

10.3.4 Dependiendo del tipo de nombre que se está generando, se aplicará una o las tres subcláusulas siguientes.

10.3.4.1 Si el nombre que se está generando es el nombre de referencia de tipo de una asignación de tipo ASN.1 y la cadena de caracteres generada por 10.3.3 es idéntica al nombre de referencia de tipo de otra asignación de tipo ASN.1 generada previamente en el mismo módulo ASN.1 o en otro módulo ASN.1 con el mismo espacio nominal (incluida la ausencia de un espacio nominal), o es una de las palabras reservadas especificadas en la Rec. UIT-T X.680 | ISO/CEI 8824-1, 11.27, entonces, se añadirá un sufijo a la cadena de caracteres generada mediante 10.3.3. El sufijo consistirá en un SIGNO MENOS seguido de una representación léxica canónica (véase 2.3.1 de la parte 2 del esquema XML W3C) de un número entero. Este número entero será el último entero positivo para que el nuevo nombre sea diferente del nombre de referencia de tipo de cualquier otra asignación de tipo ASN.1 generada con anterioridad en cualquiera de esos módulos ASN.1.

10.3.4.2 Si el nombre que se está generando es el identificador de un componente de una secuencia, conjunto o tipo de elección, y la cadena de caracteres generada mediante 10.3.3 es idéntica al identificador de un componente generado anteriormente de la misma secuencia, conjunto o tipo de elección, entonces, se añadirá un sufijo a la cadena de caracteres generada mediante 10.3.3. El sufijo consistirá en un SIGNO MENOS seguido de una representación léxica canónica (véase 2.3.1 de la Parte 2 del esquema XML W3C) de un número entero. El número entero será el último entero positivo para que el nuevo identificador sea diferente del identificador de cualquier componente generado con anterioridad de dicha secuencia, conjunto o tipo de elección.

10.3.4.3 Si el nombre que se está generado es el "identificador" en un "EnumerationItem" de un tipo enumerado, y la cadena de caracteres generada mediante 10.3.3 es idéntica al "identificador" en otro "EnumerationItem" generado con anterioridad en el mismo tipo enumerado, entonces, se añadirá un sufijo a la cadena de caracteres generada mediante 10.3.3. El sufijo consistirá en un SIGNO MENOS seguido de una representación léxica canónica (véase 2.3.1 de la parte 2 del esquema XML W3C) de un número entero. Este número entero será el último entero positivo para que

el nuevo identificador sea diferente del "identificador" de cualquier otro "EnumerationItem" ya presente en ese tipo enumerado ASN.1.

10.3.5 Para un nombre de referencia tipo ASN.1 (o un identificador) que se genere aplicando esta subcláusula 10.3 al **nombre** de una **declaración de elemento**, **declaración de atributo**, **definición de tipo compleja** de nivel superior o de una **definición de tipo simple** de nivel superior definida por el usuario, si el nombre de referencia de tipo (o el identificador) generado es diferente del **nombre**, se asignará una instrucción de codificación final **NAME** a la asignación de tipo ASN.1 con ese nombre de referencia de tipo (o al componente con ese identificador) como se especifica en las tres subcláusulas siguientes.

10.3.5.1 Si la única diferencia está en la primera letra (que es una mayúscula en el nombre de referencia de tipo y una minúscula en el **nombre**), entonces, la "palabra clave" en la instrucción de codificación **NAME** deberá estar **SIN MAYÚSCULAS**.

10.3.5.2 Si la única diferencia está en la primera letra (que es una minúscula en el identificador y una mayúscula en el **nombre**) entonces la "palabra clave" en la instrucción de codificación **NAME** deberá estar en **MAYÚSCULAS**.

10.3.5.3 En otros casos, el "NewName" en la instrucción de codificación **NAME** será el **nombre**.

EJEMPLO – La **definición de tipo compleja** de nivel superior:

```
<xsd:complexType name="COMPONENTS">
  <xsd:sequence>
    <xsd:element name="Elem" type="xsd:boolean"/>
    <xsd:element name="elem" type="xsd:integer"/>
    <xsd:element name="Elem-1" type="xsd:boolean"/>
    <xsd:element name="elem-1" type="xsd:integer"/>
  </xsd:sequence>
</xsd:complexType>
```

se corresponde con la asignación de tipo ASN.1:

```
COMPONENTS-1 ::= [NAME AS "COMPONENTS"] SEQUENCE {
  elem      [NAME AS CAPITALIZED] BOOLEAN,
  elem-1    [NAME AS "elem"] INTEGER,
  elem-1-1  [NAME AS "Elem-1"] BOOLEAN,
  elem-1-2  [NAME AS "elem-1"] INTEGER }
```

10.3.6 Para un nombre de referencia de tipo ASN.1 (o un identificador) que se genere aplicando esta subcláusula 10.3 al **nombre** de una **declaración de elemento**, **declaración de atributo**, **definición de tipo compleja** de nivel superior o **definición de tipo simple** de nivel superior definida por el usuario, si el **espacio nominal de destino** del componente de esquema no es **ausente**, entonces, se asignará una instrucción de codificación final **NAMESPACE** a la asignación de tipo ASN.1 con dicho nombre de referencia de tipo (o al tipo denominado con dicho identificador) y especificará el **espacio nominal de destino** del componente de esquema.

10.3.7 Para un identificador ASN.1 generado mediante esta subcláusula 10.3 para la correspondencia de una **definición de tipo simple** con una faceta **enumeración** en la que el identificador generado sea diferente del miembro correspondiente del **valor** de la faceta **enumeración**, se asignará una instrucción de codificación final **TEXT** al tipo enumerado ASN.1, con información de calificación que especifique el "identificador" en el "EnumerationItem" del tipo enumerado. Aplica una de las dos subcláusulas siguientes.

10.3.7.1 Si la única diferencia está en la primera letra (que es una minúscula en el identificador y una mayúscula en el miembro del **valor** de la faceta **enumeración**), entonces, la "palabra clave" en la instrucción de codificación **TEXT** estará en **MAYÚSCULAS**.

10.3.7.2 En otro caso, "NewName" en la instrucción de codificación **TEXT** será el miembro del **valor** de la faceta **enumeración**.

10.4 Orden en la correspondencia

10.4.1 Se impone un orden en los componentes de esquema de nivel superior del esquema XSD de origen en el que se realiza la correspondencia. Esto aplica a las **definiciones de grupo de modelo**, a las **definiciones de tipo complejas** de nivel superior, a las **definiciones de tipo simples** de nivel superior definidas por el usuario, a las **declaraciones de atributo** de nivel superior y a las **declaraciones de elemento** de nivel superior.

NOTA – No se establecen correspondencias con ASN.1 de otros componentes de esquema de nivel superior, y los tipos de datos XSD incorporados se corresponden de una forma especial.

10.4.2 El orden se especifica en las tres subcláusulas siguientes.

10.4.2.1 Los componentes de esquema de nivel superior se ordenarán en primer lugar por su **espacio nominal de destino** con el espacio nominal **ausente**, antes que todos los nombres de espacio nominal en orden lexicográfico ascendente.

10.4.2.2 En cada espacio nominal de destino, se dividirán los componentes de esquema de nivel superior en cuatro conjuntos ordenados como sigue:

- a) **declaraciones de elemento**;
- b) **declaraciones de atributo**;
- c) **definiciones de tipo complejas y definiciones de tipo sencillas**;
- d) **definiciones de grupo de modelo**.

10.4.2.3 En cada conjunto (véase 10.4.2.2), se ordenarán los componentes de esquema por su **nombre** en orden lexicográfico ascendente.

10.4.3 La correspondencia genera algunas asignaciones de tipo ASN.1 que no se corresponden directamente con ningún componente de esquema XSD. Éstas son:

- a) tipos elección (con una instrucción de codificación final **USE-TYPE**) correspondientes a una jerarquía de derivación de tipo; los nombres de referencia de tipo de estos tipos tienen un sufijo **"-derivations"** (véase la cláusula 29).
- b) tipos elección (con una instrucción de codificación final **USE-TYPE** en el tipo y una instrucción de codificación final **USE-NIL** en cada alternativa) correspondientes a una jerarquía de derivación de tipo en la que la **definición de tipo simple** o la **definición de tipo compleja** de nivel superior definida por el usuario, raíz de la jerarquía de derivación, se utiliza como la **definición de tipo** de una o más **declaraciones de elemento** que son **anulables**; los nombres de referencia de tipo de estos tipos tienen un sufijo **"-deriv-nillable"** (véase la cláusula 29);
- c) tipos elección (con una instrucción de codificación final **USE-TYPE** en el tipo e instrucciones de codificación finales **DEFAULT-FOR-EMPTY** en cada alternativa) correspondientes a una jerarquía de derivación de tipo en la que la **definición de tipo simple** o la **definición de tipo compleja** de nivel superior definida por el usuario, raíz de la jerarquía de derivación, se utiliza como la **definición de tipo** de una o más **declaraciones de elemento** que no son **anulables** y que tienen una **restricción de valor** que es un valor **por defecto**; los nombres de referencia de tipo de estos tipos tienen un sufijo **"-deriv-default-"** (véase la cláusula 29);
- d) tipos elección (con una instrucción de codificación final **USE-TYPE** en el tipo e instrucciones de codificación finales **DEFAULT-FOR-EMPTY** en cada alternativa) correspondientes a una jerarquía de derivación de tipo en la que la **definición de tipo simple** o la **definición de tipo compleja** de nivel superior definida por el usuario, raíz de la jerarquía de derivación, se utiliza como la **definición de tipo** de una o más **declaraciones de elemento** que no son **anulables** y que tienen una **restricción de valor** que es un valor **fijo**; los nombres de referencia de tipo de estos tipos tienen un sufijo **"-deriv-fixed-"** (véase la cláusula 29);
- e) tipos elección (con una instrucción de codificación final **USE-TYPE** en el tipo e instrucciones de codificación finales **USE-NIL** y **DEFAULT-FOR-EMPTY** en cada alternativa) correspondientes a una jerarquía de derivación de tipo en la que la **definición de tipo simple** o la **definición de tipo compleja** de nivel superior definida por el usuario, raíz de la jerarquía de derivación, se utiliza como la **definición de tipo** de una o más **declaraciones de elemento** que son **anulables** y que tienen una **restricción de valor** que es un valor **por defecto**; los nombres de referencia de tipo de estos tipos tienen un sufijo **"-deriv-nillable-default-"** (véase la cláusula 29);
- f) tipos elección (con una instrucción de codificación final **USE-TYPE** en el tipo e instrucciones de codificación finales **USE-NIL** y **DEFAULT-FOR-EMPTY** en cada alternativa) correspondientes a una jerarquía de derivación de tipo en la que la **definición de tipo simple** o la **definición de tipo compleja** de nivel superior definida por el usuario, raíz de la jerarquía de derivación, se utiliza como la **definición de tipo** de una o más **declaraciones de elemento** que son **anulables** y que tienen una **restricción de valor** que es un valor **fijo**; los nombres de referencia de tipo de estos tipos tienen un sufijo **"-deriv-nillable-fixed-"** (véase la cláusula 29);
- g) tipos elección (con una instrucción de codificación final **UNTAGGED**) correspondientes a un grupo de sustitución de elemento; los nombres de referencia de tipo de estos tipos tienen un sufijo **"-group-"** (véase la cláusula 31);
- h) tipos secuencia (con una instrucción de codificación final **USE-NIL**) correspondientes a la utilización de una **definición de tipo simple** o una **definición de tipo compleja** de nivel superior definida por el usuario

como la **definición de tipo** de una o más **declaraciones de elemento** que son **anulables**; los nombres de referencia de tipo de estos tipos tienen un sufijo **"-nillable-"** (véase la cláusula 30).

10.4.4 Todas las asignaciones de tipo ASN.1 que corresponden directamente a los componentes de esquema XSD en el esquema XSD de origen se generarán antes que todas las asignaciones de tipo ASN.1 enumeradas en 10.4.3 (si existen).

10.4.5 Las asignaciones de tipo ASN.1 que corresponden directamente a los componentes de esquema XSD se generarán en el orden de los componentes de esquema XSD correspondientes (véase 10.4.1). Las asignaciones de tipo ASN.1 enumeradas en 10.4.3 (si existen) se generarán en el orden de los componentes de esquema XSD (véase 10.4.1) correspondiente a la "asignación de tipo asociada" (véanse las cláusulas 29, 30 y 31).

10.4.6 Para 10.4.3 c) a f), si se utiliza la **definición de tipo simple** o la **definición de tipo compleja**, raíz de la jerarquía de derivación, como la **definición de tipo** de múltiples **declaraciones de elemento** que tienen diferentes valores en la **restricción de valor**, las asignaciones de tipo ASN.1 se generarán en orden lexicográfico ascendente de la representación léxica canónica (véase 2.3.1 de la parte 2 del esquema XML W3C) del valor en la **restricción de valor**.

11 Correspondencia de usos de tipos de datos incorporados XSD

11.1 Esta cláusula se aplica cuando la invoquen explícitamente otras cláusulas de la presente Recomendación | Norma Internacional para generar una definición de tipo ASN.1 correspondiente al uso de tipos de datos incorporados XSD

11.2 Debe establecerse una correspondencia entre el uso de tipos de datos incorporados XSD y una definición de tipo ASN.1 de conformidad con el cuadro 2. El cuadro indica la definición de tipo ASN.1 a utilizar. La notación "XSD.Name" indica que la definición de tipo ASN.1 debe ser la definición de tipo ASN.1 (un "DefinedType") generada aplicando 10.2 a la asignación de tipo ASN.1 correspondiente presente el módulo **xsd**.

**Cuadro 2 – Definiciones de tipo ASN.1 correspondientes
a los usos de tipos de datos incorporados XSD**

Tipo de datos incorporados XSD	Definición de tipo ASN.1	Tipo de datos incorporados XSD	Definición de tipo ASN.1
anyURI	XSD.AnyURI	int	XSD.Int
anySimpleType	XSD.AnySimpleType	integer	INTEGER
anyType	XSD.AnyType	language	XSD.Language
base64Binary	[BASE64] OCTET STRING	long	XSD.Long
boolean	BOOLEAN	Name	XSD.Name
byte	INTEGER (-128..127)	NCName	XSD.NCName
date	XSD.Date	negativeInteger	INTEGER (MIN..-1)
dateTime	XSD.DateTime	NMTOKEN	XSD.NMTOKEN
decimal	XSD.Decimal	NMTOKENS	XSD.NMTOKENS
double	XSD.Double	nonNegativeInteger	INTEGER (0..MAX)
duration	XSD.Duration	nonPositiveInteger	INTEGER (MIN..0)
ENTITIES	XSD.ENTITIES	normalizedString	XSD.NormalizedString
ENTITY	XSD.ENTITY	NOTATION	XSD.NOTATION
float	XSD.Float	positiveInteger	INTEGER (1..MAX)
gDay	XSD.GDay	QName	XSD.QName
gMonth	XSD.GMonth	short	XSD.Short
gMonthDay	XSD.GMonthDay	string	XSD.String
gYear	XSD.GYear	time	XSD.Time
gYearMonth	XSD.GYearMonth	token	XSD.Token
hexBinary	OCTET STRING	unsignedByte	INTEGER (0..255)
ID	XSD.ID	unsignedInt	XSD.UnsignedInt
IDREF	XSD.IDREF	unsignedLong	XSD.UnsignedLong
IDREFS	XSD.IDREFS	unsignedShort	XSD.UnsignedShort

12 Correspondencia de facetas

Esta cláusula se aplica cuando la invoquen explícitamente otras cláusulas de la presente Recomendación | Norma Internacional para establecer una correspondencia con una **faceta** de una **definición de tipo simple**. Una **faceta** de una **definición de tipo simple** STD se hace corresponder con una restricción ASN.1 aplicada a una definición de tipo ASN.1 correspondiente a STD, a menos que STD tenga una faceta **enumeration** que se está haciendo corresponder con una "Enumeration" ASN.1 (véanse 12.4.1 y 12.4.2). En este caso, no se genera ninguna restricción ASN.1 a partir de la faceta (véanse 12.1.2, 12.2.1, 12.3.1 y 12.5.1).

12.1 Las facetas length, minLength y maxLength

12.1.1 Las facetas **length**, **minLength** y **maxLength** se ignorarán para los tipos de datos incorporados **xsd:QName** y **xsd:NOTATION** y para cualquier **definición de tipo simple** derivada de ellos por restricciones.

12.1.2 Si una faceta **length**, **minLength** o **maxLength** pertenece a una **definición de tipo simple** que también tiene una faceta **enumeración** que está siendo correspondida con una "Enumeration" ASN.1 (véanse 12.4.1 y 12.4.2), entonces, no se incluirá ningún "EnumerationItem" en "Enumeration" para los miembros (si existen) del **valor** de la faceta **enumeration** que no satisfacen las facetas **length**, **minLength** o **maxLength**.

12.1.3 En otro caso, las facetas **length**, **minLength** y **maxLength** de la **definición de tipo simple** se harán corresponder con una restricción de tamaño ASN.1 según el cuadro 3.

Cuadro 3 – Restricciones de tamaño ASN.1 correspondientes a las facetas length, minLength y maxLength

XSD facet	ASN.1 size constraint
length=value	(SIZE (<i>value</i>))
minLength=min	(SIZE (<i>min</i> .. MAX))
maxLength=max	(SIZE (0 .. <i>max</i>))
minLength=min maxLength=max	(SIZE (<i>min</i> .. <i>max</i>))

12.2 La faceta pattern

12.2.1 Si una faceta **pattern** pertenece a una **definición de tipo simple** que también tiene una faceta **enumeration** que se está correspondiendo con una "Enumeration" ASN.1 (véanse 12.4.1 y 12.4.2), entonces, no se incluirá ningún "EnumerationItem" en la "Enumeration" para los miembros (si existen) del **valor** de la faceta **enumeration** que no satisfagan la faceta **pattern**.

12.2.2 En otro caso, la faceta **pattern** se hará corresponder con una restricción definida por el usuario. Se aplicará una de las dos subcláusulas siguientes.

12.2.2.1 Si el **valor** de la faceta **pattern** es una expresión normal simple, la restricción definida por el usuario será:

(**CONSTRAINED BY** {*/* XML representation of the XSD pattern "xyz" */*})

donde "xyz" es la representación XML del **valor** de la faceta **pattern**, pero si aparece la subcadena "*" en el **valor** de la faceta **pattern** será sustituida por la cadena de caracteres "/".

12.2.2.2 Si el **valor** de la faceta **pattern** es una conjunción de uniones de expresiones normales (caso general), la restricción definida por el usuario no está especificada (no obstante, véase 12.5.4).

12.3 La faceta whiteSpace

12.3.1 Si una faceta **whiteSpace** con un **valor sustitución** o **colapso** pertenece a una **definición de tipo simple** que también tiene una faceta **enumeration** que se está correspondiendo con una "Enumeration" ASN.1 (véanse 12.4.1 y 12.4.2), entonces, no se incluirá ningún "EnumerationItem" en "Enumeration" para los miembros (si existen) del **valor** de la faceta **enumeration** que contenga cualquiera de los caracteres TABULACIÓN HORIZONTAL, NUEVO RENGLÓN o RETORNO DEL CARRO, o (en el caso de **colapso**) que contenga múltiples caracteres ESPACIO consecutivos.

12.3.2 En otro caso, aplica por lo menos una de las tres cláusulas siguientes.

12.3.2.1 Si el **valor** de la faceta **whiteSpace** es **preserve**, entonces la faceta **whiteSpace** se ignorará.

12.3.2.2 Si el **valor** de la faceta **whiteSpace** es **replace** y la definición de tipo ASN.1 correspondiente a la **definición de tipo simple** es un tipo de cadena de caracteres restringida ASN.1, entonces, se añadirá una restricción alfabética

permitida a la definición de tipo ASN.1 para suprimir los caracteres TABULACIÓN HORIZONTAL, NUEVO RENGLÓN y RETORNO DEL CARRO. Se asignará una instrucción de codificación final **WHITESPACE REPLACE** a la definición de tipo ASN.1. Se utilizará la siguiente restricción alfabética permitida o una equivalente:

```
(FROM ({0, 0, 0, 32} .. {0, 16, 255, 255}))
```

12.3.2.3 Si el **valor** de la faceta **whiteSpace** es **collapse** y la definición de tipo ASN.1 correspondiente a la **definición de tipo simple** es un tipo de cadena de caracteres restringida ASN.1, entonces, se añadirá a la definición de tipo ASN.1 una restricción alfabética permitida de las especificadas en 12.3.2.2 y una restricción de plantilla que prohíba los caracteres de múltiples SPACE consecutivos. Se asignará a la definición de tipo ASN.1 una instrucción de codificación final **WHITESPACE COLLAPSE**. Se utilizará la siguiente restricción de plantilla o una equivalente:

```
(PATTERN "([^\ ]([^\ ]| [^\ ])*)?")
```

12.4 La faceta enumeration

12.4.1 Una faceta **enumeration** que pertenezca a una **definición de tipo simple** con una **variedad de atomic** que esté derivada mediante una restricción (directa o indirectamente) de **xsd:string** no se corresponderá con una restricción ASN.1. En su lugar, se hará corresponder la faceta con la "Enumeración" del tipo ASN.1 enumerado, correspondiente a la **definición de tipo simple** (véase 13.5) como se especifica en las tres subcláusulas siguientes.

12.4.1.1 Para cada miembro del **valor** de la faceta **enumeration**, se añadirá a "Enumeration" (sujeto a 12.1.2, 12.2.1, 12.3.1 y 12.5.1) un "EnumerationItem" que es un "identificador".

12.4.1.2 Cada "identificador" se generará aplicando 10.3 al miembro correspondiente del **valor** de la faceta **enumeration**.

12.4.1.3 Los miembros del **valor** de la faceta **enumeration** se harán corresponder en orden lexicográfico ascendente y se descartará cualquier miembro duplicado.

12.4.2 Una faceta **enumeration** que pertenezca a una **definición de tipo simple** con una **variedad de atomic** que esté derivada mediante una restricción (directa o indirectamente) de **xsd:integer** no se corresponderá con una restricción ASN.1. En su lugar, se hará corresponder la faceta con la "Enumeration" del tipo ASN.1 enumerado, correspondiente a la **definición de tipo simple** (véase 13.6) como se especifica en las tres subcláusulas siguientes.

12.4.2.1 Para cada miembro del **valor** de la faceta **enumeration**, se añadirá a "Enumeration" (sujeto a 12.1.2, 12.2.1, 12.3.1 y 12.5.1) un "EnumerationItem" que es un "identificador".

12.4.2.2 Cada "identificador" en cada "NamedNumber" se generará concatenando la cadena de caracteres "int" con la representación léxica canónica (véase 2.3.1 de la parte 2 del esquema XML W3C) del miembro correspondiente del **valor** de la faceta **enumeration**. "SignedNumber" en "NamedNumber" debe ser la notación del valor ASN.1 para el miembro (un número entero).

12.4.2.3 Los miembros del **valor** de la faceta **enumeration** se harán corresponder en orden numérico ascendente y se descartará cualquier miembro duplicado.

12.4.3 Cualquier otra faceta **enumeration** se hará corresponder con una restricción ASN.1 que sea un valor único o una unión de valores únicos correspondiente a los miembros del **valor de enumeration**.

NOTA – La faceta **enumeration** se aplica al valor espacio de la **definición de tipo básica**. Por lo tanto, para una **enumeración** de tipos de datos incorporados **xsd:QName** o **xsd:NOTATION**, el valor del componente **uri** de la **[USE-QNAME] SEQUENCE** producida por una restricción ASN.1 de valor único viene determinada, en la representación XML de un esquema XSD, por las declaraciones namespace cuyo ámbito incluya **xsd:QName** o **xsd:NOTATION** y por el prefijo (en su caso) de **xsd:QName** o **xsd:NOTATION**.

EJEMPLO 1 – Lo siguiente representa una **definición de tipo simple** de nivel superior definida por el usuario que es una restricción de **xsd:string** con una faceta **enumeration**:

```
<xsd:simpleType name="state">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="off"/>
    <xsd:enumeration value="on"/>
  </xsd:restriction>
</xsd:simpleType>
```

se corresponde con la asignación de tipo ASN.1:

```
State ::= [NAME AS UNCAPITALIZED] ENUMERATED {off, on}
```

EJEMPLO 2 – Lo siguiente representa una **definición de tipo simple** de nivel superior definida por el usuario que es una restricción de **xsd:integer** con una faceta **enumeration**:

```
<xsd:simpleType name="integer-0-5-10">
  <xsd:restriction base="xsd:integer">
    <xsd:enumeration value="0"/>
    <xsd:enumeration value="5"/>
    <xsd:enumeration value="10"/>
  </xsd:restriction>
</xsd:simpleType>
```

se corresponde con la asignación de tipo ASN.1:

```
Integer-0-5-10 ::= [NAME AS UNCAPITALIZED] ENUMERATED {int0(0), int5(5), int10(10)}
```

EJEMPLO 3 – Lo siguiente representa una **definición de tipo simple** de nivel superior definida por el usuario que es una restricción de **xsd:integer** con una faceta **minInclusive** y otra **maxInclusive**:

```
<xsd:simpleType name="integer-1-10">
  <xsd:restriction base="xsd:integer">
    <xsd:minInclusive value="1"/>
    <xsd:maxInclusive value="10"/>
  </xsd:restriction>
</xsd:simpleType>
```

se corresponde con la asignación de tipo ASN.1:

```
Integer-1-10 ::= [NAME AS UNCAPITALIZED] INTEGER(1..10)
```

EJEMPLO 4 – Lo siguiente representa una **definición de tipo simple** de nivel superior definida por el usuario que es una restricción (con una faceta **minExclusive**) de otra **definición de tipo simple**, derivada mediante restricción a partir de **xsd:integer** a la que se ha añadido una faceta **minInclusive** y otra **maxInclusive**:

```
<xsd:simpleType name="multiple-of-4">
  <xsd:restriction>
    <xsd:simpleType>
      <xsd:restriction base="xsd:integer">
        <xsd:minInclusive value="1"/>
        <xsd:maxInclusive value="10"/>
      </xsd:restriction>
    </xsd:simpleType>
    <xsd:minExclusive value="5"/>
  </xsd:restriction>
</xsd:simpleType>
```

se corresponde con la asignación de tipo ASN.1:

```
Multiple-of-4 ::= [NAME AS UNCAPITALIZED] INTEGER(5<..10)
```

EJEMPLO 5 – Lo siguiente representa una **definición de tipo simple** de nivel superior definida por el usuario que es una restricción (con una faceta **minLength** y una **maxLength**) de otra **definición de tipo simple**, derivada mediante restricción a partir de **xsd:string** a la que se ha añadido una faceta **enumeration**:

```
<xsd:simpleType name="color">
  <xsd:restriction>
    <xsd:simpleType>
      <xsd:restriction base="xsd:string">
        <xsd:enumeration value="white"/>
        <xsd:enumeration value="black"/>
        <xsd:enumeration value="red"/>
      </xsd:restriction>
    </xsd:simpleType>
    <xsd:minLength value="2"/>
    <xsd:maxLength value="4"/>
  </xsd:restriction>
</xsd:simpleType>
```

se corresponde con la asignación de tipo ASN.1:

```
Color ::= [NAME AS UNCAPITALIZED] ENUMERATED {red}
```

12.5 Otras facetas

12.5.1 Si una faceta **totalDigits**, **fractionDigits**, **maxInclusive**, **maxExclusive**, **minExclusive** o **minInclusive** pertenece a una **definición de tipo simple** que también tiene una faceta **enumeration** que se corresponde con una "Enumeration" ASN.1 (véanse 12.4.1 y 12.4.2), entonces no se incluirá ningún "EnumerationItem" en "Enumeration" para los miembros (si

existen) del **valor** de la faceta **enumeration** que no satisfagan la faceta **totalDigits**, **fractionDigits**, **maxInclusive**, **maxExclusive**, **minExclusive** o **minInclusive**.

12.5.2 Si una faceta **maxInclusive**, **maxExclusive** o **minExclusive** pertenece a una **definición de tipo simple** sin una faceta **enumeration** o con una faceta **enumeration** que no está siendo correspondida con una "Enumeration" ASN.1 (véanse 12.4.1 y 12.4.2), entonces aplica una de las dos subcláusulas siguientes:

12.5.2.1 Si la **definición de tipo simple** se deriva mediante restricción (directa o indirectamente) a partir de un tipo de datos de fecha u hora incorporado XSD (**xsd:date**, **xsd:dateTime**, **xsd:duration**, **xsd:gDay**, **xsd:gMonth**, **xsd:gYear**, **xsd:gYearMonth**, **xsd:gMonthDay** o **xsd:time**), entonces, las facetas **maxInclusive**, **maxExclusive**, **minExclusive** y **minInclusive** de la **definición de tipo simple** se harán corresponder con una restricción definida por el usuario ASN.1 (véase 12.5.4).

12.5.2.2 En otro caso, las facetas **maxInclusive**, **maxExclusive**, **minExclusive** y **minInclusive** de una **definición de tipo simple** se harán corresponder con una gama de valores o una restricción de valores única ASN.1 de conformidad con el cuadro 4.

**Cuadro 4 – Restricciones ASN.1 correspondientes a las facetas
maxInclusive, maxExclusive, minExclusive y minInclusive**

Faceta XSD	Restricción ASN.1
maxInclusive=ub	(MIN .. ub)
maxExclusive=ub	(MIN .. < ub)
minExclusive=lb	(lb < .. MAX)
minInclusive=lb	(lb .. MAX)
minInclusive=ub maxInclusive=lb	(lb .. ub)
minInclusive=v maxInclusive=v	(v)
minInclusive=ub maxExclusive=lb	(lb .. < ub)
minExclusive=ub maxInclusive=lb	(lb < .. ub)
minExclusive=ub maxExclusive=lb	(lb < .. < ub)

12.5.3 Si una faceta **totalDigits** o **fractionDigits** pertenece a una **definición de tipo simple** sin una faceta **enumeration** o con una faceta **enumeration** que no se corresponde con una "Enumeration" ASN.1 (véanse 12.4.1 y 12.4.2), entonces las facetas **totalDigits** y **fractionDigits** de la **definición de tipo simple** se harán corresponder con una restricción definida por el usuario (véase 12.5.4).

12.5.4 Cuando una faceta se corresponde con una restricción definida por el usuario ASN.1, se recomienda que la faceta y su **valor** aparezcan en un comentario ASN.1 en la restricción definida por el usuario. La forma precisa de la restricción definida por el usuario no se especifica.

13 Correspondencia de definiciones de tipo simple

13.1 Esta cláusula se aplica cuando la invoquen explícitamente otras cláusulas de la presente Recomendación | Norma Internacional para generar una asignación de tipo ASN.1 o una definición de tipo ASN.1 correspondiente a una **definición de tipo simple**.

13.2 Esta cláusula especifica la correspondencia de **definiciones de tipo simple** que no son tipos de datos incorporados XSD. El conjunto de tipos de datos incorporados XSD se corresponde con el módulo ASN.1 predefinido, especificado en el anexo A (el módulo **XSD**), que se incluirá en las especificaciones ASN.1 generadas por la correspondencia.

13.3 Una **definición de tipo simple** de nivel superior definida por el usuario se hará corresponder con una asignación de tipo ASN.1. La "typerefence" en "TypeAssignment" se generará aplicando 10.3 al **nombre** de la **definición de tipo simple** y "Type" en "TypeAssignment" será una definición de tipo ASN.1 como se especifica en 13.5 a 13.10.

13.4 Una **definición de tipo simple** anónima se corresponderá con una definición de tipo ASN.1 como se especifica en 13.5 a 13.10.

13.5 Para una **definición de tipo simple** con una **variedad** de **atomic** con una faceta **enumeration**, obtenida mediante restricción (directa o indirectamente) a partir de **xsd:string**, la definición de tipo ASN.1 será un tipo enumerado ASN.1 cuya "Enumeration" se generará como se especifica en 12.4.1.

13.6 Para una **definición de tipo simple** con una **variedad** de **atomic** con una faceta **enumeration** obtenida mediante restricción (directa o indirectamente) a partir de **xsd:integer**, la definición de tipo ASN.1 será un tipo enumerado ASN.1

cuya "Enumeration" se generará como se especifica en 12.4.2. Se asignará al tipo enumerado ASN.1 una instrucción de codificación final **USE-NUMBER**.

13.7 Para cualquier otra **definición de tipo simple** (por ejemplo, D) con una **variedad** obtenida mediante restricción (directa o indirectamente) a partir de una **definición de tipo simple** de nivel superior definida por el usuario, la definición de tipo ASN.1 se generará aplicando la cláusula 23 a la **definición de tipo simple** de nivel superior definida por el usuario (por ejemplo, B) de forma que:

- a) D se obtiene mediante restricción (directa o indirectamente) a partir de B; y
- b) B es la **definición de tipo básica** de D o todos los pasos de derivación intermedios entre B y D son **definiciones de tipo simple** anónimas.

Entonces, para cada una de las **facetas** de D (si existen), se añadirá a la definición ASN.1 una restricción ASN.1 generada aplicando la cláusula 12 a la faceta.

13.8 Para cualquier otra **definición de tipo simple** (por ejemplo, D) con una **variedad de atomic**, la definición de tipo ASN.1 se generará aplicando la cláusula 23 al tipo de datos incorporado XSD (por ejemplo, B) de forma que:

- a) D se obtiene mediante restricción (directa o indirectamente) a partir de B; y
- b) B es la **definición de tipo básica** de D o todos los pasos de derivación intermedios entre B y D son **definiciones de tipo simple** anónimas.

Entonces, para cada una de las **facetas** de D, se añadirá a la definición ASN.1 una restricción ASN.1 generada aplicando la cláusula 12 a la faceta.

13.9 Para cualquier otra **definición de tipo simple** (por ejemplo, D) con una **variedad de list**, aplican las tres subcláusulas siguientes:

13.9.1 La definición de tipo ASN.1 será un tipo ASN.1 secuencia de cuyo componente deberá ser un "Type" generado aplicando la cláusula 23 a la **definición de tipo elemento**.

13.9.2 Para cada una de las **facetas** de D, se añadirá al tipo ASN.1 secuencia de una restricción ASN.1 generada aplicando la cláusula 12 a la faceta.

13.9.3 Se asignará al tipo secuencia de ASN.1 una instrucción final de codificación **LIST**.

EJEMPLO – Lo siguiente representa una **definición de tipo simple** de nivel superior definida por el usuario que es una **lista** de **xsd:float**:

```
<xsd:simpleType name="list-of-float">
  <xsd:list itemType="xsd:float"/>
</xsd:simpleType>
```

Se corresponde con una asignación de tipo ASN.1:

```
List-of-float ::= [LIST] [NAME AS UNCAPITALIZED] SEQUENCE OF XSD.Float
```

13.10 Para cualquier otra **definición de tipo simple** (por ejemplo, D) con una **variedad de unión**, aplican las cinco subcláusulas siguientes:

13.10.1 La definición de tipo ASN.1 será un tipo elección ASN.1 con una alternativa para cada **miembro** de las **definiciones de tipo miembro**.

13.10.2 Para cada miembro de las **definiciones de tipo miembro**, el "identificador" en "NamedType" de la alternativa correspondiente se generará aplicando 10.3 al **nombre** del miembro (si el miembro es un tipo de datos incorporados XSD o una **definición de tipo simple** de nivel superior definida por el usuario) o a la cadena de caracteres **"alt"** (si el miembro es una **definición de tipo simple** anónima) y "Type" en "NamedType" será la definición de tipo ASN.1 generada aplicando la cláusula 23 al miembro de las **definiciones de tipo miembro**.

13.10.3 Para cada miembro de las **definiciones de tipo miembro** que sea una **definición de tipo simple** anónima, el "NamedType" correspondiente tendrá una instrucción de codificación final **NAME AS ""**.

13.10.4 Para cada una de las **facetas** de D, se añadirá al tipo elección ASN.1 una restricción ASN.1 generada aplicando la cláusula 12 a la faceta.

13.10.5 Se asignará al tipo elección ASN.1 una instrucción de codificación final **USE-UNION**.

EJEMPLO – Lo siguiente representa una **definición de tipo simple** de nivel superior definida por el usuario que es una **unión** de dos **definiciones de tipo simple** anónimas:

```
<xsd:simpleType name="decimalOrBinary">
  <xsd:union>
    <xsd:simpleType>
```



```

        <xsd:restriction base="xsd:decimal"/>
    </xsd:simpleType>
    <xsd:simpleType>
        <xsd:restriction base="xsd:float"/>
    </xsd:simpleType>
</xsd:union>
</xsd:simpleType>

```

Se corresponde con una asignación de tipo ASN.1:

```

DecimalOrBinary ::= [NAME AS UNCAPITALIZED] [USE-UNION] CHOICE {
    alt          [NAME AS ""] XSD.Decimal,
    alt-1        [NAME AS ""] XSD.Float }

```

14 Correspondencia de declaraciones de elemento

14.1 Esta cláusula se aplica cuando la invoquen explícitamente otras cláusulas de la presente Recomendación | Norma Internacional para generar una asignación de tipo ASN.1 o una definición de tipo ASN.1 correspondiente a una **declaración de elemento**.

NOTA – La presencia de una **restricción de valor** en una **declaración de elemento** normalmente afecta a la correspondencia. Sin embargo, 8.10 implica que una **declaración de elemento** que tenga una **restricción de valor** y cuya **definición de tipo** sea **xsd:QName** o **xsd:NOTATION** o una restricción de estos tipos de datos incorporados XSD se hace corresponder como si no tuviera **restricción de valor**.

14.2 Se ignorará una **declaración de elemento** de nivel superior que sea **abstracta**.

14.3 Una **declaración de elemento** de nivel superior que no sea **abstracta** se corresponderá con una asignación de tipo ASN.1. La "typereference" en "TypeAssignment" se generará aplicando 10.3 al **nombre** de la **declaración de elemento** y "Type" en "TypeAssignment" será una definición de tipo ASN.1 como se especifica en 14.5.

14.4 Una **declaración de elemento** local se corresponderá con una definición de tipo ASN.1 como se especifica en 14.5.

14.5 Aplica una de las dos subcláusulas siguientes (14.5.1 y 14.5.2).

14.5.1 Si la **definición de tipo** de la **declaración de elemento** es una **definición de tipo simple** o una **definición de tipo compleja** anónima o un tipo de datos incorporados XSD (por ejemplo, A), entonces aplica una de las dos subcláusulas siguientes.

14.5.1.1 Si una **declaración de elemento** no es **anulable**, entonces la definición de tipo ASN.1 se generará aplicando a A la cláusula 23.

14.5.1.2 Si la **declaración de elemento** es **anulable**, entonces la definición de tipo ASN.1 se generará aplicando a A la cláusula 26 (si A es una **definición de tipo simple**) o la cláusula 27 (si A es una **definición de tipo compleja**).

14.5.2 Si la **definición de tipo** de la **declaración de elemento** es una **definición de tipo simple** o una **definición de tipo compleja** de nivel superior definida por el usuario (por ejemplo, T) entonces aplica una de las cuatro subcláusulas siguientes.

14.5.2.1 Si la **declaración de elemento** no es **anulable** y no tiene una **definición de tipo** sustituible (véase 14.6), entonces la definición de tipo ASN.1 se generará aplicando a T la cláusula 23.

14.5.2.2 Si la **declaración de elemento** es **anulable** y no tiene una **definición de tipo** sustituible (véase 14.6), entonces la definición de tipo ASN.1 se generará aplicando a T la cláusula 26 (si T es una **definición de tipo simple**) o la cláusula 27 (si T es una **definición de tipo compleja**).

14.5.2.3 Si la **declaración de elemento** no es **anulable** y tiene una **definición de tipo** sustituible (véase 14.6), entonces la definición de tipo ASN.1 se generará aplicando a T la cláusula 24.

14.5.2.4 Si la **declaración de elemento** es **anulable** y tiene una **definición de tipo** sustituible (véase 14.6), entonces la definición de tipo ASN.1 se generará aplicando a T la cláusula 25.

14.6 La frase "tiene una **definición de tipo** sustituible" aplicada a una **declaración de elemento**, significa que la **definición de tipo** de la **declaración de elemento** es una **definición de tipo simple** o una **definición de tipo compleja** de nivel superior definida por el usuario que se produce como la **definición de tipo básica** de otra **definición de tipo simple** o **definición de tipo compleja** de nivel superior.

15 Correspondencia de declaraciones de atributo

15.1 Esta cláusula se aplica cuando la invoquen explícitamente otras cláusulas de la presente Recomendación | Norma Internacional para generar una asignación de tipo ASN.1 o una definición de tipo ASN.1 correspondiente a una **declaración de atributo**.

15.2 Una **declaración de atributo** de nivel superior se corresponderá con una asignación de tipo ASN.1. La "typereference" en "TypeAssignment" se generará aplicando 10.3 al **nombre** de la **declaración de atributo** y "Type" en "TypeAssignment" será una definición de tipo ASN.1 como se especifica en 15.4. Se asignará a la asignación de tipo ASN.1 una instrucción de codificación final **ATTRIBUTE**.

15.3 Una **declaración de atributo** local se corresponderá con una definición de tipo ASN.1 como se especifica en 15.4.

15.4 La definición de tipo ASN.1 se generará aplicando la cláusula 23 a la **definición de tipo** de la **declaración de atributo**.

16 Correspondencia de valores de definiciones de tipo simple

16.1 Esta cláusula se aplica cuando la invoquen explícitamente otras cláusulas de la presente Recomendación | Norma Internacional para generar un "Valor" ASN.1 correspondiente a un valor en el espacio de valor correspondiente a una **definición de tipo simple**.

16.2 Dado un valor V en el espacio de valor de una **definición de tipo simple**, y:

- a) la definición de tipo ASN.1 obtenida a partir de esta **definición de tipo simple**; y
- b) la representación léxica canónica de V (véase 2.3.1 de la parte 2 del esquema XML W3C),

V se corresponderá con una notación de valor básica ASN.1 para el valor abstracto de la definición de tipo ASN.1 para la cuál la representación léxica canónica sea una codificación "ExtendedXMLValue" válida en EXTENDED-XER.

17 Correspondencia de definiciones de grupo de modelo

17.1 Esta cláusula se aplica cuando la invoquen explícitamente otras cláusulas de la presente Recomendación | Norma Internacional para generar una asignación de tipo ASN.1 correspondiente a una **definición de grupo de modelo**.

17.2 Una **definición de grupo de modelo** cuyo **grupo de modelo** tenga un **compositor** de **secuencia** o **elección** se corresponderá con una asignación de tipo ASN.1. "typereference" en "TypeAssignment" se generará aplicando 10.3 al **nombre** de la **definición de grupo de modelo** y "Type" en "TypeAssignment" se generará aplicando la cláusula 18 al **grupo de modelo** de la **definición de grupo de modelo**.

NOTA – Las **definiciones de grupo de modelo** cuyo **grupo de modelo** tenga un **compositor** de **todo** no tienen correspondencia con ASN.1.

18 Correspondencia de grupos de modelo

18.1 Esta cláusula se aplica cuando la invoquen explícitamente otras cláusulas de la presente Recomendación | Norma Internacional para generar una definición de tipo ASN.1 correspondiente a un **grupo de modelo**.

NOTA – Esta cláusula no se invoca para todos los **grupos de modelo**. Por ejemplo, un **grupo de modelo** con un **compositor** de **todo** no se corresponde con ASN.1, pero sus **partículas** tienen correspondencia como se especifica en 20.9.

18.2 Un **grupo de modelo** con un **compositor** de **secuencia** se corresponderá con un tipo de secuencia ASN.1. Para cada **partícula** en el **grupo de modelo** en su orden, se generará un "NamedType" aplicando la cláusula 19 a la **partícula** y se añadirá dicho "NamedType" al tipo de secuencia como uno de sus componentes. Se asignará al tipo de secuencia una instrucción de codificación final **UNTAGGED**.

18.3 Un **grupo de modelo** con un **compositor** de **elección** se corresponderá con un tipo elección ASN.1. Para cada **partícula** en el **grupo de modelo** en su orden, se generará un "NamedType" aplicando la cláusula 19 a la **partícula** y se añadirá dicho "NamedType" al tipo elección como una de sus alternativas. Se asignará al tipo elección una instrucción de codificación final **UNTAGGED**.

19 Correspondencia de partículas

19.1 Esta cláusula se aplica cuando la invoquen explícitamente otras cláusulas de la presente Recomendación | Norma Internacional para generar un "NamedType" ASN.1 correspondiente a una **partícula**.

NOTA – Esta cláusula no se invoca para todas las **partículas**. Por ejemplo, la **partícula** (superior) del **tipo contenido** de una **definición de tipo compleja** se corresponde de forma particular si su **término** es un **grupo de modelo** con un **compositor** de **secuencia** o **todo** (véase 20.8).

19.2 Las tres subcláusulas siguientes definen términos que se utilizan en el resto de esta cláusula 19.

19.2.1 Si **min occurs** y **max occurs** de una **partícula** son uno, la **partícula** se denomina "partícula de presencia obligatoria".

19.2.2 Si **min occurs** es cero y **max occurs** en uno, entonces:

- a) si la correspondencia de la **partícula** es para generar un componente de un tipo secuencia ASN.1, la **partícula** se denomina "partícula de presencia opcional";
- b) en otro caso, la **partícula** se denomina "partícula de episodio único".

19.2.3 Si **max occurs** es dos o más, la **partícula** se denomina "partícula de episodio múltiple".

19.3 Una "partícula de presencia obligatoria" o una "partícula de presencia opcional" se corresponderá con un "NamedType" como se especifica en las dos subcláusulas siguientes.

19.3.1 El "identificador" en "NamedType" se generará aplicando 10.3 a la cadena de caracteres especificada en 19.5 y "Type" en "NamedType" se generará aplicando 19.6 al **término** de la **partícula**.

19.3.2 Si la **partícula** es una "partícula de presencia opcional", "NamedType" estará seguido de una palabra clave **OPTIONAL**.

19.4 Una "partícula opcional de episodio único" o una "partícula de episodio múltiple" se corresponderá con "NamedType" como se especifica en las seis subcláusulas siguientes.

19.4.1 El "identificador" en "NamedType" se generará aplicando 10.3 a la cadena de caracteres obtenida anexando el sufijo **-list** a la cadena de caracteres especificada en 19.5. "Type" en "NamedType" será un tipo secuencia de.

19.4.2 Si la **partícula** es una "partícula opcional de episodio único" o una "partícula de episodio múltiple", se añadirá una restricción de tamaño al tipo secuencia de, de conformidad con el cuadro 5.

**Cuadro 5 – Restricción de tamaño ASN.1 correspondiente
a min occurs y max occurs**

min occurs y max occurs	Restricción de tamaño ASN.1
min occurs = n max occurs = n $n \geq 2$	SIZE (n)
min occurs = min max occurs = max $max > min$ y $max \geq 2$	SIZE (min .. max)
min occurs = 0 max occurs = 1	SIZE (0 .. 1)
min occurs = min max occurs = unbounded $min \geq 1$	SIZE (min .. MAX)
min occurs = 0 max occurs = unbounded	sin restricción de tamaño

19.4.3 Si el **término** de la **partícula** es una **declaración de elemento**, entonces, el componente del tipo secuencia de será un "NamedType". El "identificador" en "NamedType" se generará aplicando 10.3 al **nombre** de la **declaración de elemento** y "Type" en ese "NamedType" se generará aplicando 19.6 al **término** de la **partícula**.

19.4.4 Si el **término** de la **partícula** es un **comodín**, entonces el componente del tipo secuencia de será un "NamedType". El "identificador" en "NamedType" será **elem** y "Type" en ese "NamedType" se generará aplicando 19.6 al **término** de la **partícula**.

19.4.5 Si el **término** de la **partícula** es un **grupo de modelo**, entonces el componente del tipo secuencia de será un "Type" y se generará aplicando 19.6 al **término** de la **partícula**.

19.4.6 Se asignará al tipo secuencia de una instrucción de codificación final **UNTAGGED**.

19.5 La cadena de caracteres utilizada en la generación del "identificador" en "NamedType" correspondiente a la **partícula** será:

- a) si la **partícula** es el **tipo contenido** de una **definición de tipo compleja**, la cadena de caracteres "content";
- b) si el **término** de la **partícula** es una **declaración de elemento**, el nombre de la **declaración de elemento**;
- c) si el **término** de la **partícula** es el **grupo de modelo** de una **definición de grupo de modelo**, el nombre de la **definición de grupo de modelo**;
- d) si el **término** de la **partícula** es un **grupo de modelo** con un **compositor de secuencia** no relacionado a una **definición de grupo de modelo**, la cadena de caracteres "sequence";
- e) si el **término** de la **partícula** es un **grupo de modelo** con un **compositor de elección** no relacionado a una **definición de grupo de modelo**, la cadena de caracteres "choice";
- f) si el **término** de la **partícula** es un **comodín**, la cadena de caracteres "elem".

19.6 El "Type" en "NamedType" correspondiente a la **partícula** (véase 19.3) o el "Type" en "NamedType" en el "SequenceOfType" correspondiente a la **partícula** (véase 19.4) será:

- a) si el **término** de la **partícula** es una **declaración de elemento** de nivel superior que no sea el encabezamiento de un grupo de sustitución de elemento, la definición de tipo ASN.1 (un "DefinedType") generada aplicando 10.2 a la asignación de tipo ASN.1 generada aplicando la cláusula 14 a la **declaración de elemento**;
- b) si el **término** de la **partícula** es una **declaración de elemento** de nivel superior que sea el encabezamiento de un grupo de sustitución de elemento, la definición de tipo ASN.1 (un "DefinedType") generada aplicando 10.2 a la asignación de tipo ASN.1 generada aplicando la cláusula 31 a la **declaración de elemento**;
- c) si el **término** de la **partícula** es una **declaración de elemento** local la definición de tipo ASN.1 generada aplicando la cláusula 14 a la **declaración de elemento**;
- d) si el **término** de la **partícula** es el **grupo de modelo** de una **definición de grupo de modelo**, la definición de tipo ASN.1 (un "DefinedType") generada aplicando 10.2 a la asignación de tipo ASN.1 generada aplicando la cláusula 17 a la **definición de grupo de modelo**;
- e) si el **término** de la **partícula** es un **grupo de modelo** no relacionado con una **definición de grupo de modelo**, la definición de tipo ASN.1 generada aplicando la cláusula 18 al **grupo de modelo**;
- f) si el **término** de la **partícula** es un **comodín**, la definición de tipo ASN.1 generada aplicando la cláusula 21 al **comodín**.

20 Correspondencia de definiciones de tipo complejas

20.1 Esta cláusula se aplica cuando la invoquen explícitamente otras cláusulas de la presente Recomendación | Norma Internacional para generar una asignación de tipo ASN.1 o una definición de tipo ASN.1 correspondiente a una **definición de tipo compleja**.

20.2 Una **definición de tipo compleja** de nivel superior tendrá una correspondencia con una asignación de tipo ASN.1. La "typereference" en "TypeAssignment" se generará aplicando 10.3 al **nombre** de la **definición de tipo compleja** y "Type" en "TypeAssignment" será una definición de tipo ASN.1 como se especifica en 20.4 a 20.11.

20.3 Una **definición de tipo compleja** anónima tendrá una correspondencia con una definición de tipo ASN.1 como se especifica en 20.4 a 20.11.

20.4 La definición de tipo ASN.1 será un tipo secuencia ASN.1. Se añadirán cero o más componentes al tipo secuencia ASN.1 como indican las subcláusulas siguientes, en el orden especificado.

20.5 Si el **tipo contenido** de la **definición de tipo compleja** es un modelo de contenido **mixto**, entonces, se añadirá un componente al tipo secuencia ASN.1. El "identificador" en "NamedType" de ese componente será **embed-values** y "Type" en "NamedType" será un tipo secuencia de, cuyo componente será un "Type" generado aplicando la cláusula 23 al tipo de datos incorporados XSD **xsd:string**. Se asignará al tipo secuencia ASN.1 una instrucción de codificación final **EMBED-VALUES**.

20.6 Si el **tipo contenido** de la **definición de tipo compleja** es una **partícula** cuyo **término** es un **grupo de modelo** con un **compositor de todo**, entonces se añadirá un componente al tipo secuencia ASN.1. El "identificador" en "NamedType" de ese componente será **order** y el "Type" en "NamedType" será un tipo secuencia de, cuyo componente será un "EnumeratedType". Para cada **partícula** del **grupo de modelo** (cuyo **término** sea siempre una **declaración de elemento**), se añadirá a "Enumeration" en su orden un "EnumerationItem" que sea un "identificador" idéntico al "identificador" en

"NamedType" correspondiente a cada **partícula**. Se asignará al tipo secuencia ASN.1 una instrucción de codificación final **USE-ORDER**

NOTA – Los "identificadores" en los "NamedType" que están estableciendo una correspondencia a partir de **partículas** se generan (aplicando 10.3) como si cada componente se añadiera al tipo secuencia. Por lo tanto, aunque el componente **order** está situado en una posición que precede textualmente a las posiciones de esos componentes en el tipo secuencia ASN.1, la generación del componente **order** solo puede completarse después de que todas las **partículas** hayan establecido correspondencias con componentes secuencia.

20.7 Si la **definición de tipo complejo** tiene **usos de atributo**, entonces, componentes generados aplicando la cláusula 22 a los **usos de atributo** serán añadidos al tipo secuencia ASN.1 en un orden basado en el **espacio nominal de destino** y en el nombre de la **declaración de atributo** de cada **uso de atributo**. Los **usos de atributo** se ordenarán en primer lugar por el **espacio nominal de destino** de la **declaración de atributo** (precediendo la palabra clave **ausente** todos los nombres de espacios nominales establecidos en orden lexicográfico ascendente) y posteriormente por el **nombre** de la **declaración de atributo** en cada **espacio nominal de destino** (también en orden lexicográfico ascendente).

20.8 Si la **definición de tipo compleja** tiene un **comodín de atributo**, entonces se añadirá al tipo secuencia ASN.1 un componente generado a partir del **comodín de atributo** (véase 21.3).

20.9 Si el **tipo contenido** de la **definición de tipo compleja** es una **partícula**, entonces, aplica una de las cuatro subcláusulas siguientes.

20.9.1 Si el **término** de la **partícula** es un **grupo de modelo** con un **compositor** de secuencia cuyos **min occurs** y **max occurs** son ambos uno, entonces, para cada **partícula** del **grupo de modelo** en su orden, se añadirá al tipo de secuencia ASN.1 un componente generado aplicando la cláusula 19 a la **partícula de modelo**.

20.9.2 Si el **término** de la **partícula** es un **grupo de modelo** con un **compositor de secuencia** cuyos **min occurs** y **max occurs** no son ambos uno, entonces, se añadirá al tipo de secuencia ASN.1 un componente generado aplicando la cláusula 19 a la **partícula** en el **tipo contenido**.

20.9.3 Si el **término** de la **partícula** es un **grupo de modelo** con un **compositor de todo**, entonces, para cada **partícula** del **grupo de modelo** en su orden, se añadirá al tipo de secuencia ASN.1 un componente generado aplicando la cláusula 19 a la **partícula de grupo de modelo**. Si la **partícula** en el **tipo contenido** de la **definición de tipo compleja** tiene **min occurs** cero, cada una de las **partículas** del **grupo de modelo** con **min occurs** uno se harán corresponder como si tuvieran **min occurs** cero.

20.9.4 Si el **término** de la **partícula** es un **grupo de modelo** con un **compositor de elección**, entonces se añadirá al tipo de secuencia ASN.1 un componente generado aplicando la cláusula 19 a la **partícula** en el **tipo contenido**.

20.10 Si el **tipo contenido** de la **definición de tipo compleja** es una **definición de tipo simple**, entonces se añadirá al tipo de secuencia ASN.1 un componente. El "identificador" en "NamedType" del componente se generará aplicando 10.3 a la cadena de caracteres "**base**" y "Type" en "NamedType" será la definición de tipo ASN.1 generada aplicando la cláusula 23 al **tipo contenido**. Se asignará al componente una instrucción de codificación final **UNTAGGED**.

20.11 Si el **tipo contenido** de la **definición de tipo compleja** está **vacío**, entonces no se añadirá ningún componente más al tipo secuencia ASN.1.

21 Correspondencia de comodines

21.1 Esta cláusula se aplica cuando la invoquen explícitamente otras cláusulas de la presente Recomendación | Norma Internacional para generar una asignación de tipo ASN.1 o una definición de tipo ASN.1 correspondiente a una **definición de tipo simple**.

21.2 Un **comodín** que sea el **término** de una **partícula** tendrá una correspondencia con una definición de tipo ASN.1 generada aplicando la cláusula 23 al tipo de datos incorporados XSD **xsd:string**. Se asignará a la definición de tipo ASN.1 una instrucción de codificación final **ANY-ELEMENT**.

21.3 Un **comodín** que sea un **comodín de atributo** de un **tipo complejo** tendrá una correspondencia con un "NamedType". El "identificador" en "NamedType" se generará aplicando 10.3 a la cadena de caracteres "**attr**" y "Type" en "NamedType" será un tipo secuencia de. El componente del tipo secuencia de será un "Type" generado aplicando la cláusula 23 al tipo de datos incorporados XSD **xsd:string**. Se aplicará al tipo secuencia de la restricción definida por el usuario siguiente:

```
(CONSTRAINED BY
  { /* Each item shall conform to the "AnyAttributeFormat" specified in
    ITU-T Rec. X.693 | ISO/IEC 8825-4, clause 18 */ })
```

Se asignará al tipo secuencia de una instrucción de codificación final **ANY-ATTRIBUTES**.

21.4 Si el **comodín** tiene una **restricción de espacio nominal**, se hará corresponder con una "NamespaceRestriction" en la instrucción de codificación **ANY-ELEMENTS** o **ANY-ATTRIBUTES**.

22 Correspondencia de usos de atributo

22.1 Esta cláusula se aplica cuando la invoquen explícitamente otras cláusulas de la presente Recomendación | Norma Internacional para generar un "NamedType" ASN.1 correspondiente a un **uso de atributo**.

22.2 Un **uso de atributo** tendrá una correspondencia con un "NamedType".

22.3 El "identificador" en "NamedType" se generará aplicando 10.3 al **nombre** de la **declaración de atributo** del **uso del atributo** y "Type" en "NamedType" será:

- a) si el **uso de atributo** tiene una **declaración de atributo** de nivel superior, la definición de tipo ASN.1 (un "DefinedType") generada aplicando 10.2 a la asignación de tipo ASN.1 generada aplicando la cláusula 15 a la **declaración de atributo**;
- b) si el **uso de atributo** tiene una **declaración de atributo** local, la definición de tipo ASN.1 generada aplicando la cláusula 15 a la **declaración de atributo**.

22.4 Si el **uso de atributo** o su **declaración de atributo** tiene una **restricción de valor**, "NamedType" estará seguido de la palabra clave **DEFAULT** y de un "Valor" generado aplicando la cláusula 16 al valor en la **restricción de valor** del **uso de atributo** (si el **uso de atributo** tiene una **restricción de valor**) o al valor en la **restricción de valor** de su **declaración de atributo** (en los demás casos).

22.5 Si el **uso de atributo** o su **declaración de atributo** tiene una **restricción de valor** que es un valor **fijo**, entonces se añadirá a "NamedType" una restricción de valor única ASN.1 con un "Valor" idéntico al "Valor" siguiente a la palabra clave **DEFAULT**.

22.6 Si el **uso de atributo** no es **necesario** y ni el **uso de atributo** ni su **declaración de atributo** tiene una **restricción de valor**, "NamedType" estará seguido de la palabra clave **OPTIONAL**.

22.7 Se asignará al "Type" en "NamedType" una instrucción de codificación final **ATTRIBUTE**.

23 Correspondencia de usos de definiciones de tipo simple y de tipo compleja (caso general)

23.1 Esta cláusula se aplica cuando la invoquen explícitamente otras cláusulas de la presente Recomendación | Norma Internacional para generar una definición de tipo ASN.1 correspondiente a un uso de una **definición de tipo simple** o de una **definición de tipo compleja**. Esto incluye su utilización como **definición de tipo de declaraciones de elemento** que no tengan una **definición de tipo** sustituible (véase 14.6), no sean **anulables** y puedan o no tener una **restricción de valor**.

23.2 Un uso de una **definición de tipo simple** de nivel superior que es un tipo de datos incorporados XSD tendrá una correspondencia como se especifica en la cláusula 11.

23.3 Un uso de una **definición de tipo simple** de nivel superior definida por el usuario tendrá una correspondencia con la definición de tipo ASN.1 (un "DefinedType") generada aplicando 10.2 a la asignación de tipo ASN.1 generada aplicando la cláusula 13 a la **definición de tipo simple**.

23.4 Un uso de una **definición de tipo compleja** de nivel superior definida por el usuario tendrá una correspondencia con la definición de tipo ASN.1 (un "DefinedType") generada aplicando 10.2 a la asignación de tipo ASN.1 generada aplicando la cláusula 20 a la **definición de tipo compleja**.

23.5 Un uso de una **definición de tipo simple** anónima no se distingue de la propia **definición de tipo simple** y tendrá una correspondencia como se especifica en la cláusula 13 de la **definición de tipo simple**.

23.6 Un uso de una **definición de tipo compleja** anónima no se distingue de la propia **definición de tipo compleja** y tendrá una correspondencia como se especifica en la cláusula 20 de la **definición de tipo compleja**.

23.7 Si una **definición de tipo simple** o una **definición de tipo compleja** se utiliza como la **definición de tipo** de una **declaración de elemento** con una **restricción de valor**, entonces, se asignará a la definición de tipo ASN.1 una instrucción de codificación final **DEFAULT-FOR-EMPTY** y aplica una de las tres subcláusulas siguientes:

23.7.1 Para una **definición de tipo simple**, el "Valor" en la instrucción de codificación final **DEFAULT-FOR-EMPTY** se generará aplicando la cláusula 16 al valor en la **restricción de valor** considerado como un valor en el espacio de valor de la **definición de tipo simple**.

23.7.2 Para una **definición de tipo compleja** cuyo **tipo contenido** es una **definición de tipo simple**, el "Valor" en la instrucción de codificación final **DEFAULT-FOR-EMPTY** se generará aplicando la cláusula 16 al valor en la **restricción de valor** considerado como un valor en el espacio de valor de la **definición de tipo simple**.

23.7.3 Para una **definición de tipo compleja** con un tipo contenido **mixto**, el "Valor" en la instrucción de codificación final **DEFAULT-FOR-EMPTY** se generará aplicando la cláusula 16 al valor en la **restricción de valor** considerado como un valor en el espacio de valor de **xsd:string** con **whiteSpace preserve**.

23.8 Si se utiliza una **definición de tipo simple** o una **definición de tipo compleja** como la **definición de tipo** de una **declaración de elemento** con una **restricción de valor** que tiene un valor **fijo**, entonces aplica una de las tres subcláusulas siguientes.

23.8.1 Para una **definición de tipo simple**, se añadirá a la definición ASN.1 una restricción de valor única ASN.1 con un "Valor" idéntico al "Valor" de la instrucción de codificación final **DEFAULT-FOR-EMPTY**.

23.8.2 Para una **definición de tipo compleja** cuyo **tipo contenido** es una **definición de tipo simple**, se añadirá a la definición ASN.1 una restricción de subtipo interna ASN.1 y aplicará al componente **base** una restricción de valor única con un "Valor" idéntico al "Valor" de la instrucción de codificación final **DEFAULT-FOR-EMPTY**.

23.8.3 Para una **definición de tipo compleja** con un tipo contenido **mixto**, se añadirá a la definición ASN.1 una restricción de subtipo interno ASN.1 y se aplicará:

- a) al componente **embed-values**, una restricción de valor única ASN.1 con un "Valor" constituido por un único episodio de un "Valor" idéntico al "Valor" de la instrucción de codificación final **DEFAULT-FOR-EMPTY**;
- b) a cada componente que sea **OPTIONAL** y que no tenga una instrucción de codificación final **ATTRIBUTE**, la palabra clave **ABSENT**; y
- c) a cada componente cuyo tipo sea un tipo secuencia de, una restricción **SIZE (0)**.

24 Correspondencias de usos especiales de definiciones de tipo simple y de tipo compleja (sustituible)

24.1 Esta cláusula se aplica cuando la invoquen explícitamente otras cláusulas de la presente Recomendación | Norma Internacional para generar una definición de tipo ASN.1 correspondiente a una **definición de tipo simple** o de una **definición de tipo compleja** utilizada como la **definición de tipo** de **declaraciones de elemento** que tengan una **definición de tipo** sustituible (véase 14.6), no sean **anulables** y puedan o no tener una **restricción de valor**.

24.2 Un uso de una **definición de tipo simple** (por ejemplo, STD) o de una **definición de tipo compleja** (por ejemplo, CTD) tendrá una correspondencia con un tipo elección ASN.1.

24.3 Se añadirá una alternativa al tipo elección ASN.1 para el propio STD o CTD y se añadirá una alternativa para cada **definición de tipo simple** o **definición de tipo compleja** de nivel superior definida por el usuario en el esquema XSD de origen que se derive mediante restricción o extensión (directa o indirectamente) a partir de STD o CTD.

24.4 Para cada alternativa, el "identificador" en "NamedType" se generará aplicando 10.3 al **nombre** de la **definición de tipo simple** o de la **definición de tipo compleja** correspondiente a la alternativa y "Type" en "NamedType" será la definición de tipo ASN.1 generada aplicando la cláusula 23 a la **definición de tipo simple** o a la **definición de tipo compleja**.

24.5 La primera alternativa añadida al tipo elección será la correspondiente al propio STD o CTD. Las alternativas subsiguientes se añadirán al tipo elección en un orden basado en el **espacio nominal de origen** y en el **nombre** de las **definiciones de tipo simples** y de las **definiciones de tipo complejas**. Las definiciones de tipo se ordenarán en primer lugar por **espacios nominales de destino** (con el espacio nominal **ausente** precediendo a todos los nombres espacios nominales clasificados en orden lexicográfico ascendente) y posteriormente por el **nombre** (también en orden lexicográfico ascendente) dentro de cada **espacio nominal de destino**.

24.6 Se asignará al tipo elección ASN.1 una instrucción de codificación final **USE-TYPE**.

24.7 Si existe una **restricción de valor**, entonces, se asignará a cada alternativa del tipo elección ASN.1 una instrucción de codificación final **DEFAULT-FOR-EMPTY**. Aplica una de las tres subcláusulas siguientes.

24.7.1 Si la alternativa corresponde a una **definición de tipo simple**, el "Valor" en la instrucción de codificación final **DEFAULT-FOR-EMPTY** se generará aplicando la cláusula 16 al valor en la **restricción de valor** considerado como un valor en el espacio de valor de la **definición de tipo simple**.

24.7.2 Si la alternativa corresponde a una **definición de tipo compleja** cuyo **tipo contenido** es una **definición de tipo simple**, el "Valor" en la instrucción de codificación final **DEFAULT-FOR-EMPTY** se generará aplicando la cláusula 16 al valor en la **restricción de valor** considerado como un valor en el espacio de valor de la **definición de tipo simple**.

24.7.3 Si la alternativa corresponde a una **definición de tipo compleja** con un tipo de contenido **mixto**, el "Valor" en la instrucción de codificación final **DEFAULT-FOR-EMPTY** se generará aplicando la cláusula 16 al valor en la **restricción de valor** considerado como un valor en el espacio de valor de **xsd:string** con **whiteSpace preserve**.

24.8 Si existe una **restricción de valor** que tenga un valor **fijo**, entonces se añadirá al tipo elección ASN.1 una restricción de subtipo interna ASN.1. Aplica una de las tres subcláusulas siguientes.

24.8.1 Si la alternativa corresponde a una **definición de tipo simple**, la restricción de subtipo interna aplicará a la alternativa una restricción de valor única ASN.1 con un "Valor" idéntico al "Valor" en la instrucción de codificación final **DEFAULT-FOR-EMPTY**.

24.8.2 Si la alternativa corresponde a una **definición de tipo compleja** cuyo **tipo contenido** es una **definición de tipo simple**, la restricción de subtipo interna aplicará a la alternativa otra restricción de subtipo interna ASN.1 que aplique al componente **base** una restricción de valor única con un "Valor" idéntico al "Valor" en la instrucción de codificación final **DEFAULT-FOR-EMPTY**.

24.8.3 Si la alternativa corresponde a una **definición de tipo compleja** con un tipo de contenido **mixto**, la restricción de subtipo interna aplicará a la alternativa otra restricción de tipo interna ASN.1 que aplica:

- a) al componente **embed-values**, una restricción de valor única ASN.1 con un "Valor" constituido por un único episodio de un "Valor" idéntico al "Valor" de la instrucción de codificación final **DEFAULT-FOR-EMPTY**.
- b) para cada componente que sea **OPTIONAL** y que no tenga una instrucción de codificación final **ATTRIBUTE**, la palabra clave **ABSENT**; y
- c) para cada componente cuyo tipo sea un tipo secuencia de, una restricción **SIZE (0)**.

25 Correspondencia de usos especiales de definiciones de tipo simple y de tipo compleja (sustituibles, anulables)

25.1 Esta cláusula se aplica cuando la invoquen explícitamente otras cláusulas de la presente Recomendación | Norma Internacional para generar una definición de tipo ASN.1 correspondiente a una **definición de tipo simple** o de una **definición de tipo compleja** utilizada como la **definición de tipo de declaraciones de elemento** que tengan una **definición de tipo** sustituible (véase 14.6), no sean **anulables** y puedan o no tener una **restricción de valor**.

25.2 Un uso de una **definición de tipo simple** (por ejemplo, STD) o de una **definición de tipo compleja** (por ejemplo, CTD) tendrá una correspondencia con un tipo elección ASN.1.

25.3 Se añadirá una alternativa al tipo elección ASN.1 para el propio STD o CTD y se añadirá una alternativa para cada **definición de tipo simple** o **definición de tipo compleja** de nivel superior definida por el usuario en el esquema XSD de origen que se derive mediante restricción o extensión (directa o indirectamente) a partir de STD o CTD.

25.4 Para cada alternativa, el "identificador" en "NamedType" se generará aplicando 10.3 al **nombre** de la **definición de tipo simple** o de la **definición de tipo compleja** correspondiente a la alternativa y "Type" en "NamedType" será la definición de tipo ASN.1 (un "DefinedType") generada aplicando 10.2 a la asignación de tipo ASN.1, generada aplicando la cláusula 30 a la **definición de tipo simple** o a la **definición de tipo compleja**.

25.5 La primera alternativa añadida al tipo elección será la correspondiente al propio STD o CTD. Las alternativas subsiguientes se añadirán al tipo elección en un orden basado en el **espacio nominal de origen** y en el **nombre** de las **definiciones de tipo simples** y de las **definiciones de tipo complejas**. Las definiciones de tipo se ordenarán en primer lugar por **espacios nominales de destino** (con el espacio nominal **ausente** precediendo a todos los nombres de espacios nominales clasificados en orden lexicográfico ascendente) y posteriormente por el **nombre** (también en orden lexicográfico ascendente) dentro de cada **espacio nominal de destino**.

25.6 Se asignará al tipo elección ASN.1 una instrucción de codificación final **USE-TYPE**.

25.7 Si existe una **restricción de valor**, entonces se asignará a cada alternativa del tipo elección ASN.1 una instrucción de codificación final **DEFAULT-FOR-EMPTY**. Aplica una de las tres subcláusulas siguientes.

25.7.1 Si la alternativa corresponde a una **definición de tipo simple**, el "Valor" en la instrucción de codificación final **DEFAULT-FOR-EMPTY** se generará aplicando la cláusula 16 al valor en la **restricción de valor** considerado como un valor en el espacio de valor de la **definición de tipo simple**.

25.7.2 Si la alternativa corresponde a una **definición de tipo compleja**, cuyo **tipo contenido** es una **definición de tipo simple**, el "Valor" en la instrucción de codificación final **DEFAULT-FOR-EMPTY** se generará aplicando la cláusula 16 al valor en la **restricción de valor** considerado como un valor en el espacio de valor de la **definición de tipo simple**.

25.7.3 Si la alternativa corresponde a una **definición de tipo compleja** con un tipo de contenido **mixto**, el "Valor" en la instrucción de codificación final **DEFAULT-FOR-EMPTY** se generará aplicando la cláusula 16 al valor en la **restricción de valor** considerado como un valor en el espacio de valor de **xsd:string** con **whiteSpace preserve**.

25.8 Si existe una **restricción de valor** que tenga un valor **fijo**, entonces, se añadirá al tipo elección ASN.1 una restricción de subtipo interna ASN.1. Aplica una de las tres subcláusulas siguientes.

25.8.1 Si la alternativa corresponde a una **definición de tipo simple**, la restricción de subtipo interna aplicará a la alternativa (que es un tipo secuencia ASN.1 con una instrucción de codificación final **USE-NIL**) otra restricción de subtipo interna ASN.1 que, a su vez, aplicará al componente **content** la palabra clave **PRESENT** y una restricción de valor único ASN.1 con un "Valor" idéntico al "Valor" en la instrucción de codificación final **DEFAULT-FOR-EMPTY**.

25.8.2 Si la alternativa corresponde a una **definición de tipo compleja** cuyo **tipo contenido** es una **definición de tipo simple**, la restricción de subtipo interna aplicará a la alternativa (que es un tipo secuencia ASN.1 con una instrucción de codificación final **USE-NIL**) otra restricción de subtipo interna ASN.1 que aplique al componente **content** la palabra clave **PRESENT** y una restricción de valor único ASN.1 con un "Valor" idéntico al "Valor" en la instrucción de codificación final **DEFAULT-FOR-EMPTY**.

25.8.3 Si la alternativa corresponde a una **definición de tipo compleja** con un tipo de contenido **mixto**, la restricción de subtipo interna aplicará a la alternativa (que es un tipo secuencia ASN.1 con una instrucción de codificación final **USE-NIL**) otra restricción de tipo interna ASN.1 que aplica:

- a) al componente **embed-values**, una restricción de valor única ASN.1 con un "Valor" constituido por un único episodio de un "Valor" idéntico al "Valor" de la instrucción de codificación final **DEFAULT-FOR-EMPTY**;
- b) al componente **content** (que es un tipo secuencia ASN.1), la palabra clave **PRESENT** y otra restricción de subtipo interna que aplica la palabra clave **ABSENT** a cada uno de sus componentes que sea **OPTIONAL** y una restricción **SIZE(0)** a cada uno de sus componentes cuyo tipo sea un tipo secuencia de;
- c) para cada componente que sea **OPTIONAL** y que no tenga una instrucción de codificación final **ATTRIBUTE**, la palabra clave **ABSENT**; y
- d) para cada componente cuyo tipo sea un tipo secuencia de, una restricción **SIZE(0)**.

26 Correspondencia de usos especiales de definiciones de tipo simples (anulables)

26.1 Esta cláusula se aplica cuando la invoquen explícitamente otras cláusulas de la presente Recomendación | Norma Internacional para generar una definición de tipo ASN.1 correspondiente a una **definición de tipo simple** utilizada como la **definición de tipo** de las **declaraciones de elemento** que no tengan una **definición de tipo** sustituible (véase 14.6), sean **anulables** y puedan o no tener una **restricción de valor**.

26.2 Un uso de una **definición de tipo simple** tendrá una correspondencia con un tipo de secuencia ASN.1 con un componente **OPTIONAL**.

26.3 El "identificador" en "NamedType" del componente será **content** y "Type" en "NamedType" será la definición de tipo ASN.1 generada aplicando la cláusula 23 a la **definición de tipo simple**.

26.4 Se asignará al tipo de secuencia ASN.1 una instrucción de codificación final **USE-NIL**.

26.5 Si existe una **restricción de valor**, entonces, se asignará al tipo de secuencia ASN.1 una instrucción de codificación final **DEFAULT-FOR-EMPTY**. El "Valor" en la instrucción de codificación final **DEFAULT-FOR-EMPTY** se generará aplicando la cláusula 16 al valor en la **restricción de valor**.

26.6 Si existe una **restricción de valor** que tenga un valor **fijo**, entonces se añadirá al tipo secuencia ASN.1 una restricción de subtipo interna ASN.1. La restricción de subtipo interna aplicará al componente **content** una restricción de valor única ASN.1 con un "Valor" idéntico al "Valor" en la instrucción de codificación final **DEFAULT-FOR-EMPTY**. La restricción de subtipo interna también aplicará la palabra clave **PRESENT** al componente **content**.

27 Correspondencia de usos especiales de definiciones de tipo complejas (anulables)

27.1 Esta cláusula se aplica cuando la invoquen explícitamente otras cláusulas de la presente Recomendación | Norma Internacional para generar una definición de tipo ASN.1 correspondiente a una **definición de tipo compleja** utilizada como la **definición de tipo** de las **declaraciones de elemento** que no tengan una **definición de tipo** sustituible (véase 14.6), sean **anulables** y puedan o no tener una **restricción de valor**.

27.2 El uso de una **definición de tipo compleja** tendrá una correspondencia con un tipo secuencia ASN.1. Se añadirán al tipo secuencia ASN.1 uno o más componentes como se especifica en las subcláusulas siguientes, en el orden especificado.

27.3 Si el **tipo contenido** de la **definición de tipo compleja** es un modelo de contenido **mixto**, entonces, se añadirá un componente **embed-values** al tipo secuencia ASN.1 como se especifica en 20.5.

27.4 Si el **tipo contenido** de la **definición de tipo compleja** es una **partícula** cuyo **término** es un **grupo de modelo** con un **compositor** de **todo**, entonces se añadirá un componente **order** al tipo secuencia ASN.1 como se especifica en 20.6.

27.5 Si la **definición de tipo compleja** tiene **usos de atributo**, se añadirán al tipo secuencia ASN.1 componentes con correspondencias a partir de **usos de atributo** como se especifica en 20.7.

27.6 Si la **definición de tipo compleja** tiene un **comodín de atributo**, entonces se añadirá al tipo secuencia ASN.1 un componente generado a partir del **comodín de atributo** como se especifica en 20.8.

27.7 Si el **tipo de contenido** de la **definición de tipo compleja** es una **partícula**, entonces aplica una de las dos subcláusulas siguientes.

27.7.1 Si el **término** de la **partícula** es un **grupo de modelo** con un **compositor** de **secuencia** o de **elección**, entonces, se añadirá al tipo secuencia ASN.1 un componente **OPTIONAL**. El "identificador" en "NamedType" del componente se generará aplicando 10.3 a la cadena de caracteres "**content**" y "Type" en "NamedType" será un tipo secuencia ASN.1 con un único componente, que se generará aplicando la cláusula 19 a la **partícula** en el **tipo contenido**.

27.7.2 Si el **término** de la **partícula** es un **grupo de modelo** con un **compositor** de **todo**, entonces se añadirá al tipo secuencia ASN.1 un componente **OPTIONAL**. El "identificador" en "NamedType" del componente se generará aplicando 10.3 a la cadena de caracteres "**content**" y "Type" en "NamedType" será un tipo secuencia ASN.1. Para cada **partícula** del **grupo de modelo** en su orden, se añadirá al tipo secuencia ASN.1 interno un componente generado aplicando la cláusula 19 a la **partícula** del **grupo de modelo**. Si la **partícula** en el **tipo contenido** de la **definición de tipo compleja** tiene **min occurs** cero, cada una de las **partículas** del **grupo de modelo** con **min occurs** uno se hará corresponder como si tuviera **min occurs** cero.

27.8 Si el **tipo contenido** de la **definición de tipo compleja** es una **definición de tipo simple**, entonces se añadirá al tipo secuencia ASN.1 un componente **OPTIONAL**. El "identificador" en "NamedType" del componente se generará aplicando 10.3 a la cadena de caracteres "**content**" y "Type" en "NamedType" será la definición de tipo ASN.1 generada aplicando la cláusula 23 al **tipo contenido**.

27.9 Si el **tipo contenido** de la **definición de tipo compleja** está **vacío**, entonces, no se añadirá ningún componente más al tipo secuencia ASN.1.

27.10 Se asignará al tipo secuencia ASN.1 una instrucción de codificación final **USE-NIL**.

27.11 Si existe una **restricción de valor**, entonces, se asignará al tipo secuencia ASN.1 una instrucción de codificación final **DEFAULT-FOR-EMPTY**. Aplica una de las dos subcláusulas siguientes:

27.11.1 Si el **tipo contenido** de la **definición de tipo compleja** es una **definición de tipo simple**, el "Valor" en la instrucción de codificación final **DEFAULT-FOR-EMPTY** se generará aplicando la cláusula 16 al valor en la **restricción de valor** considerado como un valor en el espacio de valor de la **definición de tipo simple**.

27.11.2 Si el **tipo contenido** de la **definición de tipo compleja** es un tipo de contenido **mixto**, el "Valor" en la instrucción de codificación final **DEFAULT-FOR-EMPTY** se generará aplicando la cláusula 16 al valor en la **restricción de valor** considerado como un valor en el espacio de valor de **xsd:string** con **whiteSpace preserve**.

27.12 Si existe una **restricción de valor** que tenga un valor **fijo**, entonces, se añadirá al tipo secuencia ASN.1 una restricción de subtipo interna ASN.1. La restricción de subtipo interna aplicará la palabra clave **PRESENT** al componente **content**. Aplica una de las dos subcláusulas siguientes.

27.12.1 Si el **tipo contenido** de la **definición de tipo compleja** es una **definición de tipo simple**, la restricción de subtipo interna aplicará al componente **content** una restricción de valor único ASN.1 con un "Valor" idéntico al "Valor" en la instrucción de codificación final **DEFAULT-FOR-EMPTY**.

27.12.2 Si el **tipo contenido** de la **definición de tipo compleja** es un tipo de contenido **mixto**, la restricción de subtipo interna aplicará:

- a) al componente **embed-values**, una restricción de valor único ASN.1 con un "Valor" constituido por un único episodio de un "Valor" idéntico al "Valor" de la instrucción de codificación final **DEFAULT-FOR-EMPTY**;
- b) a cada componente del componente **content** (un tipo secuencia ASN.1) que sea **OPTIONAL**, la palabra clave **ABSENT**; y
- c) a cada componente del componente **content** (un tipo secuencia ASN.1) cuyo tipo sea un tipo secuencia de, una restricción **SIZE (0)**.

28 Correspondencia de usos especiales de declaraciones de elemento (cabecera de grupo de sustitución de elemento)

28.1 Esta cláusula se aplica cuando la invoquen explícitamente otras cláusulas de la presente Recomendación | Norma Internacional para generar una definición de tipo ASN.1 correspondiente a una **declaración de elemento** de nivel superior que sea la cabecera de un grupo de sustitución de elemento y se utilice como el **término** de **partículas**.

28.2 Un uso de una **declaración de elemento** de nivel superior tendrá una correspondencia con un tipo elección ASN.1.

28.3 Se añadirá al tipo elección ASN.1 una alternativa para la propia **declaración de elemento** de nivel superior (por ejemplo, H) y una alternativa para cada **declaración de elemento** de nivel superior en el esquema XSD de origen que no sea **abstract** y cuya **afiliación de grupo de sustitución** sea H.

28.4 Para cada alternativa, el "identificador" en "NamedType" se generará aplicando 10.3 al **nombre** de la **declaración de elemento** de nivel superior correspondiente a la alternativa y "Type" en "NamedType" será la definición de tipo ASN.1 (un "DefinedType") generada aplicando 10.2 a la asignación de tipo ASN.1 generada aplicando la cláusula 14 a la **declaración de elemento** de nivel superior.

NOTA – En XSD, la pertenencia a un grupo de sustitución es transitiva, es decir, los miembros de un grupo de sustitución ESG1 cuya cabecera sea un miembro de otro grupo de sustitución ESG2 también son miembros de ESG2.

28.5 Se añadirán alternativas al tipo elección en un orden basado en el **espacio nominal de destino** y en el **nombre** de las **declaraciones de elemento** de nivel superior. Las **declaraciones de elemento** se ordenarán en primer lugar por el **espacio nominal de destino** (con un espacio nominal **absent** precediendo todos los nombres de los espacios nominales enumerados en orden lexicográfico ascendente) y posteriormente por el **nombre** (también en orden lexicográfico ascendente) dentro de cada **espacio nominal de destino**.

NOTA – La **declaración de elemento** que sea un encabezamiento del grupo de sustitución de elemento se ordena junto con las otras **declaraciones de elemento** que pertenezcan al grupo de sustitución de elemento.

28.6 Se asignará al tipo elección una instrucción de codificación final **UNTAGGED**.

29 Generación de asignaciones de tipo ASN.1 especiales para declaraciones de elemento

29.1 Esta cláusula se aplica cuando la invoquen explícitamente otras cláusulas de la presente Recomendación | Norma Internacional para generar una asignación de tipo ASN.1 correspondiente a una **definición de tipo simple** o una **definición de tipo compleja** de nivel superior definida por el usuario, utilizada como la **definición de tipo** de las **declaraciones de elemento** que tengan una **definición de tipo** sustituible (véase 14.6) o que sean **anulables**.

29.2 Esta cláusula la invocan otras cláusulas para una determinada combinación de:

- a) una **definición de tipo simple** o de una **definición de tipo compleja**;
- b) cuando las **declaraciones de elemento** tengan una **definición de tipo** sustituible (véase 14.6);
- c) cuando la **declaración de elemento** sea **anulable**; y
- d) cuando la **declaración de elemento** tenga una **restricción de valor** y el tipo y valor de la **restricción de valor**;

y generará una asignación de tipo ASN.1 [denominada una "asignación de tipo ASN.1 especial (para declaraciones de elemento)"] para una combinación de los elementos anteriores.

29.3 Se generará una y solo una asignación de tipo ASN.1 especial para cada diferente combinación de los elementos anteriores que se produzca realmente en una o más invocaciones de esta cláusula durante la correspondencia de un esquema XSD de origen.

NOTA – Por ejemplo, si dos o mas **declaraciones de elemento** en un esquema XSD amplio tienen **definiciones de tipo** idénticas, son ambas **anulables** y ambas tienen una **restricción de valor** que es un valor por **defecto** y el mismo valor, entonces, se genera una única asignación de tipo ASN.1 especial. El nombre de referencia de tipo de esta asignación de tipo se encontrará en "Type" en los "TypeAssignment" correspondientes a ambas **declaraciones de elemento**.

29.4 El término "asignación de tipo ASN.1 asociada" designa una asignación de tipo ASN.1 que se está haciendo corresponder a partir de la **definición de tipo simple** o de la **definición de tipo compleja**, que sea la **definición de tipo** de la **declaración de elemento** para la que se está generando una asignación de tipo ASN.1 especial, aplicando la cláusula 13 o la cláusula 20, respectivamente.

NOTA – Cualquier asignación de tipo ASN.1 especial tiene una asignación de tipo ASN.1 asociada, puesto que esta cláusula aplica únicamente cuando la **definición de tipo** de una **declaración de elemento** es una **definición de tipo simple** o una **definición de tipo compleja** de nivel superior definida por el usuario. Todas estas **definiciones de tipo simples** y **definiciones de tipo complejas** tienen correspondencia con asignaciones de tipo ASN.1.

29.5 Para una determinada **declaración de elemento**, se construirá la "typereference" en "TypeAssignment" para una asignación de tipo ASN.1 especial añadiendo un sufijo al nombre de referencia de tipo de la asignación de tipo ASN.1 asociada y aplicando 10.3 a la cadena de caracteres resultante, y "Type" en "TypeAssignment" debe ser la definición de tipo ASN.1 generada aplicando la cláusula 24 o la cláusula 25 a la **definición de tipo simple** o a la **definición de tipo compleja** que sea la **definición de tipo** de la **declaración de elemento**. Aplica uno de los siguientes casos:

- a) si la **declaración de elemento** no es **anulable**, tiene una **definición de tipo** sustituible y no tiene una **restricción de valor**, el sufijo será **"-derivations"** y se aplicará la cláusula 24;
- b) si la **declaración de elemento** es **anulable**, tiene una **definición de tipo** sustituible y no tiene una **restricción de valor**, el sufijo será **"-deriv-nillable"** y se aplicará la cláusula 25;
- c) si la **declaración de elemento** no es **anulable**, tiene una **definición de tipo** sustituible y tiene una **restricción de valor** que sea un valor por **defecto**, el sufijo será **"-deriv-default-"** seguido de una representación léxica canónica (véase 2.3.1 de la parte 2 del esquema XML W3C) del valor en la **restricción de valor**, y se aplicará la cláusula 24;
- d) si la **declaración de elemento** no es **anulable**, tiene una **definición de tipo** sustituible y tiene una **restricción de valor** con un valor **fijo**, el sufijo será **"-deriv-fixed-"** seguido de una representación léxica canónica (véase 2.3.1 de la parte 2 del esquema XML W3C) del valor en la **restricción de valor**, y se aplicará la cláusula 24;
- e) si la **declaración de elemento** es **anulable**, tiene una **definición de tipo** sustituible y tiene una **restricción de valor** que sea un valor por **defecto**, el sufijo será **"-deriv-nillable-default-"** seguido de una representación léxica canónica (véase 2.3.1 de la parte 2 del esquema XML W3C) del valor en la **restricción de valor**, y se aplicará la cláusula 25;
- f) si la **declaración de elemento** es **anulable**, tiene una **definición de tipo** sustituible y tiene una **restricción de valor** con un valor **fijo**, el sufijo será **"-deriv-nillable-fixed-"** seguido de una representación léxica canónica (véase 2.3.1 de la parte 2 del esquema XML W3C) del valor en la **restricción de valor**, y se aplicará la cláusula 25.

30 Generación de asignaciones de tipo ASN.1 especiales para definiciones de tipo

30.1 Esta cláusula se aplica cuando la invoquen explícitamente otras cláusulas de la presente Recomendación | Norma Internacional para generar una asignación de tipo ASN.1 correspondiente a una **definición de tipo simple** o una **definición de tipo compleja** de nivel superior definida por el usuario que pertenezca a la jerarquía de derivación de la **definición de tipo** de las **declaraciones de elemento** que tengan una **definición de tipo** sustituible (véase 14.6) y sean **anulables**.

30.2 Esta cláusula la invocan otras cláusulas para una determinada **definición de tipo simple** o **definición de tipo compleja** y genera una asignación de tipo ASN.1 [denominada una "asignación de tipo ASN.1 especial (para una definición de tipo)"].

30.3 Se generará una y solo una asignación de tipo ASN.1 especial para cada **definición de tipo simple** o cada **definición de tipo compleja** que se produzca realmente en una o más invocaciones de esta cláusula durante la correspondencia de un esquema XSD de origen.

30.4 El término "asignación de tipo ASN.1 asociada" designa la asignación de tipo ASN.1 que se está haciendo corresponder a partir de la **definición de tipo simple** o de la **definición de tipo compleja** aplicando la cláusula 13 o la cláusula 20, respectivamente.

30.5 La "typereference" en "TypeAssignment" para una asignación de tipo ASN.1 especial se construirá añadiendo el sufijo **"-nillable"** al nombre de referencia de tipo de la asignación de tipo ASN.1 asociada y aplicando 10.3 a la

cadena de caracteres resultante, y "Type" en "TypeAssignment" será la definición de tipo ASN.1 generada aplicando la cláusula 26 o la cláusula 27 a la **definición de tipo simple** o a la **definición de tipo compleja**, respectivamente.

31 Generación de asignaciones de tipo ASN.1 especial para grupos de sustitución de elemento

31.1 Esta cláusula se aplica cuando la invoquen explícitamente otras cláusulas de la presente Recomendación | Norma Internacional para generar una asignación de tipo ASN.1 correspondiente a una **partícula** cuyo **término** sea una **declaración de elemento** de nivel superior que sea la cabecera de un grupo de sustitución de elemento.

31.2 Esta cláusula la invocan otras cláusulas para una **declaración de elemento** de nivel superior que sea la cabecera de un grupo de sustitución de elemento y genere una asignación de tipo ASN.1 [denominada una "asignación de tipo ASN.1 especial (para un grupo de sustitución de elemento)"].

31.3 Se generará una y solo una asignación de tipo ASN.1 especial para cada **declaración de elemento** que se produzca realmente en una o más invocaciones de esta cláusula durante la correspondencia de un esquema XSD de origen.

31.4 El término "asignación de tipo ASN.1 asociada" designa la asignación de tipo ASN.1 que se está haciendo corresponder a partir de la **declaración de elemento** de nivel superior aplicando la cláusula 14.

31.5 La "typereference" en "TypeAssignment" para una asignación de tipo ASN.1 especial se construirá añadiendo el sufijo **"-group"** al nombre de referencia de tipo de la asignación de tipo ASN.1 asociada y aplicando 10.3 a la cadena de caracteres resultante, y "Type" en "TypeAssignment" será la definición de tipo ASN.1 generada aplicando la cláusula 28 a la **declaración de elemento** de nivel superior.

Anexo A

Definiciones de tipo ASN.1 correspondientes a
tipos de datos incorporados XSD

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

A.1 Este anexo especifica un módulo que define los tipos ASN.1 que corresponden a los tipos de datos incorporados XSD y que se utilizan para establecer la correspondencia entre el esquema XML W3C y ASN.1.

A.2 El esquema XML W3C define muchos tipos de datos de fecha y hora incorporados que representan duraciones, instantes o instantes recurrentes. Aunque todos se derivan de la ISO 8601, existen algunas ampliaciones y restricciones. Los tipos de datos de fecha y hora incorporados XSD tienen una correspondencia con **VisibleString** con una restricción definida por el usuario que hace referencia a la cláusula XSD pertinente. Se añade una limitación alfabética permitida para proporcionar una codificación más eficaz mediante las reglas de codificación compactadas (PER), puesto que las restricciones definidas por el usuario no son visibles para las PER (y, por lo tanto, no se utilizan en la optimización de las codificaciones).

A.3 El módulo **XSD** es:

```
XSD {joint-iso-itu-t asn1(1) specification(0) modules(0) xsd-module(2)}
DEFINITIONS
AUTOMATIC TAGS ::=
BEGIN

/* xsd:anySimpleType */

AnySimpleType ::= XMLCompatibleString

/* xsd:anyType */

AnyType ::= SEQUENCE {
    embed-values SEQUENCE OF String,
    attr SEQUENCE
        (CONSTRAINED BY {
            /* Each item shall conform to the "AnyAttributeFormat" specified
             in ITU-T Rec. X.693 | ISO/IEC 8825-4, clause 18 */ } ) OF String,
    elem-list SEQUENCE OF elem String
        (CONSTRAINED BY {
            /* Shall conform to the "AnyElementFormat" specified
             in ITU-T Rec. X.693 | ISO/IEC 8825-4, clause 19 */ } ) }
        (CONSTRAINED BY {
            /* Shall conform to ITU-T Rec. X.693 | ISO/IEC 8825-4, clause 25 */ } )

/* xsd:anyUri */

AnyURI ::= XMLStringWithNoCRLFHT
        (CONSTRAINED BY {
            /* The XMLStringWithNoCRLFHT shall be a valid URI as defined in IETF RFC
             2396 */ } )

/* xsd:base64Binary */

Base64Binary ::= OCTET STRING

/* xsd:boolean */

Boolean ::= BOOLEAN

/* xsd:byte */

Byte ::= INTEGER (-128..127)

/* xsd:date */

Date ::= DATE-TIME (DateOnly)

/* xsd:dateTime */

DateTime ::= DATE-TIME

/* xsd:decimal */

Decimal ::= REAL (WITH COMPONENTS {..., base(10)})
```

```

        (ALL EXCEPT(-0 | MINUS-INFINITY | PLUS-INFINITY | NOT-A-NUMBER))

/* xsd:double */
Double ::= REAL (WITH COMPONENTS {
    mantissa(-9007199254740991..9007199254740991),
    base(2),
    exponent(-1075..970)})

/* xsd:duration */
Duration ::= DURATION

/* xsd:ENTITIES */
ENTITIES ::= SEQUENCE (SIZE(1..MAX)) OF ENTITY

/* xsd:ENTITY */
ENTITY ::= NCName

/* xsd:float */
Float ::= REAL (WITH COMPONENTS {
    mantissa(-16777215..16777215),
    base(2),
    exponent(-149..104)})

/* xsd:gDay */
GDay ::= DATE-TIME (Day)

/* xsd:gMonth */
GMonth ::= DATE-TIME (Month)

/* xsd:gMonthDay */
GMonthDay ::= DATE-TIME (MonthDay)

/* xsd:gYear */
GYear ::= DATE-TIME (Year)

/* xsd:gYearMonth */
GYearMonth ::= DATE-TIME (YearMonth)

/* xsd:hexBinary */
HexBinary ::= OCTET STRING

/* xsd:ID */
ID ::= NCName

/* xsd:IDREF */
IDREF ::= NCName

/* xsd:IDREFS */
IDREFS ::= SEQUENCE (SIZE(1..MAX)) OF IDREF

/* xsd:int */
Int ::= INTEGER (-2147483648..2147483647)

/* xsd:integer */
Integer ::= INTEGER

/* xsd:language */
Language ::= VisibleString (FROM ("a".."z" | "A".."Z" | "-" | "0".."9"))
    (PATTERN
        "[a-zA-Z]#(1,8)(-[a-zA-Z0-9]#(1,8))*")
    /* The semantics of Language is specified in IETF RFC 3066 */

/* xsd:long */
Long ::= INTEGER (-9223372036854775808..9223372036854775807)

```

```

/* xsd:name */
Name ::= Token (XMLStringWithNoWhitespace)
        (CONSTRAINED BY {
            /* The Token shall be a Name as defined in W3C XML 1.0, 2.3 */ } )

/* xsd:NCName */
NCName ::= Name
        (CONSTRAINED BY {
            /* The Name shall be an NCName as defined in W3C XML Namespaces, 2 */ } )

/* xsd:negativeInteger */
NegativeInteger ::= INTEGER (MIN..-1)

/* xsd:NMTOKEN */
NMTOKEN ::= Token (XMLStringWithNoWhitespace)
        (CONSTRAINED BY {
            /* The Token shall be an NMTOKEN as defined in W3C XML 1.0, 2.3 */ } )

/* xsd:NMTOKENS */
NMTOKENS ::= SEQUENCE (SIZE(1..MAX)) OF NMTOKEN

/* xsd:nonNegativeInteger */
NonNegativeInteger ::= INTEGER (0..MAX)

/* xsd:nonPositiveInteger */
NonPositiveInteger ::= INTEGER (MIN..0)

/* xsd:normalizedString */
NormalizedString ::= String (XMLStringWithNoCRLFHT)
        (CONSTRAINED BY {
            /* The String shall be a normalizedString as defined in W3C XML Schema
               Part 2, 3.3.1 */})

/* xsd:NOTATION */
NOTATION ::= QName

/* xsd:positiveInteger */
PositiveInteger ::= INTEGER (1..MAX)

/* xsd:QName */
QName ::= SEQUENCE {
    uri   AnyURI OPTIONAL,
    name  NCName }

/* xsd:short */
Short ::= INTEGER (-32768..32767)

/* xsd:string */
String ::= XMLCompatibleString

/* xsd:time */
Time ::= DATE-TIME (TimeOnly)

/* xsd:token */
Token ::= NormalizedString (CONSTRAINED BY {
    /* The NormalizedString shall be a token as defined in W3C XML Schema Part 2,
       3.3.2 */})

/* xsd:unsignedByte */
UnsignedByte ::= INTEGER (0..255)

/* xsd:unsignedInt */
UnsignedInt ::= INTEGER (0..4294967295)

/* xsd:unsignedLong */

```



```
UnsignedLong ::= INTEGER (0..18446744073709551615)
```

```
/* xsd:unsignedShort */
```

```
UnsignedShort ::= INTEGER (0..65535)
```

```
/* ASN.1 type definitions supporting the mapping of W3C XML Schema built-in datatypes */
```

```
XMLCompatibleString ::= UTF8String (FROM(
    {0, 0, 0, 9} |
    {0, 0, 0, 10} |
    {0, 0, 0, 13} |
    {0, 0, 0, 32} .. {0, 0, 215, 255} |
    {0, 0, 224, 0} .. {0, 0, 255, 253} |
    {0, 1, 0, 0} .. {0, 16, 255, 253}))
```

```
XMLStringWithNoWhitespace ::= UTF8String (FROM(
    {0, 0, 0, 33} .. {0, 0, 215, 255} |
    {0, 0, 224, 0} .. {0, 0, 255, 253} |
    {0, 1, 0, 0} .. {0, 16, 255, 253}))
```

```
XMLStringWithNoCRLFHT ::= UTF8String (FROM(
    {0, 0, 0, 32} .. {0, 0, 215, 255} |
    {0, 0, 224, 0} .. {0, 0, 255, 253} |
    {0, 1, 0, 0} .. {0, 16, 255, 253}))
```

```
/* ASN.1 type definitions supporting the mapping of W3C XML Schema built-in date and time datatypes */
```

```
DURATION ::= VisibleString (FROM ("0".."9" | "DHMPSTY:.-"))
    (CONSTRAINED BY { /* W3C XML Schema Part 2, 3.2.6 */ })
```

```
DATE-TIME ::= VisibleString (FROM ("0".."9" | "TZ:.-"))
    (CONSTRAINED BY { /* W3C XML Schema Part 2, 3.2.7 */ })
```

```
DateOnly ::= DATE-TIME (FROM ("0".."9" | "Z:.-"))
    (CONSTRAINED BY { /* W3C XML Schema Part 2, 3.2.9 */ })
```

```
Day ::= DATE-TIME (FROM ("0".."9" | "Z:.-"))
    (CONSTRAINED BY { /* W3C XML Schema Part 2, 3.2.13 */ })
```

```
Month ::= DATE-TIME (FROM ("0".."9" | "Z:.-"))
    (CONSTRAINED BY { /* W3C XML Schema Part 2, 3.2.14 */ })
```

```
MonthDay ::= DATE-TIME (FROM ("0".."9" | "Z:.-"))
    (CONSTRAINED BY { /* W3C XML Schema Part 2, 3.2.12 */ })
```

```
Year ::= DATE-TIME (FROM ("0".."9" | "Z:.-"))
    (CONSTRAINED BY { /* W3C XML Schema Part 2, 3.2.11 */ })
```

```
YearMonth ::= DATE-TIME (FROM ("0".."9" | "Z:.-"))
    (CONSTRAINED BY { /* W3C XML Schema Part 2, 3.2.10 */ })
```

```
TimeOnly ::= DATE-TIME (FROM ("0".."9" | "Z:.-"))
    (CONSTRAINED BY { /* W3C XML Schema Part 2, 3.2.8 */ })
```

```
ENCODING-CONTROL XER
```

```
GLOBAL-DEFAULTS MODIFIED-ENCODINGS
```

```
GLOBAL-DEFAULTS CONTROL-NAMESPACE
```

```
"http://www.w3.org/2001/XMLSchema-instance"
```

```
PREFIX "xsi"
```

```
NAMESPACE ALL, ALL IN ALL AS
```

```
"http://www.w3.org/2001/XMLSchema"
```

```
PREFIX "xsd"
```

```
USE-QNAME QName
```

```
BASE64 Base64Binary
```

```
DECIMAL Decimal
```

```
LIST ENTITIES, IDREFS, NMTOKENS
```

```
EMBED-VALUES AnyType
```

```
ANY-ATTRIBUTES AnyType.any-attributes
```

```
ANY-ELEMENT AnyType.any-elements.*
```

```
UNTAGGED AnyType.any-elements
```

```
NAME AnySimpleType, AnyURI, Base64Binary, Boolean,
      Byte, Date, DateTime, Decimal, Double, Duration,
      Float, GDay, GMonth, GMonthDay, GYear, GYearMonth,
      HexBinary, Int, Integer, Language, Long,
      NegativeInteger, NonNegativeInteger, NonPositiveInteger,
      NormalizedString, PositiveInteger, Short,
      String, Time, Token,
      UnsignedByte, UnsignedInt, UnsignedLong, UnsignedShort
      AS UNCAPITALIZED
WHITESPACE AnyURI, Language, Token, DURATION, DATE-TIME COLLAPSE
WHITESPACE NormalizedString REPLACE

END
```

Anexo B

Asignación de valores de identificador de objeto

(Este anexo no es parte integrante de la presente Recomendación | Norma Internacional)

En esta Recomendación | Norma Internacional se asigna el siguiente identificador de objeto y descriptor de objeto:

Para el módulo que define tipos ASN.1 correspondientes a los tipos de datos incorporados XSD:

```
{ joint-iso-itu-t asn1(1) specification(0) modules(0) xsd-module(2) }  
"ASN.1 XSD Module"
```

Anexo C

Ejemplos de correspondencias

(Este anexo no es parte integrante de la presente Recomendación | Norma Internacional)

Este anexo ilustra la correspondencias especificadas en la presente Recomendación | Norma Internacional indicando el módulo ASN.1 que se corresponde con un esquema XSD.

C.1 Esquema que utiliza definiciones de tipo simples

El esquema siguiente incluye ejemplos de tipos de datos incorporados XSD (`xsd:string`, `xsd:decimal`, `xsd:integer`, `xsd:int`, `xsd:date`), y definiciones de tipo simples y definiciones de tipo complejas.

```
<?xml version="1.0" encoding="UTF-8"?>
  <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema" elementFormDefault="unqualified">
    <xsd:element name="EXAMPLES">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element ref="personnelRecord"/>
          <xsd:element name="decimal" type="xsd:decimal"/>
          <xsd:element name="daysOfTheWeek" type="ListOfDays"/>
          <xsd:element ref="namesOfMemberNations"/>
          <xsd:element ref="fileIdentifier" maxOccurs="unbounded"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="personnelRecord">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name="name" type="name"/>
          <xsd:element name="title" type="xsd:string"/>
          <xsd:element name="decimal" type="xsd:integer"/>
          <xsd:element name="dateOfHire" type="xsd:date"/>
          <xsd:element ref="nameOfSpouse"/>
          <xsd:element ref="children"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="nameOfSpouse" type="name"/>
    <xsd:complexType name="name">
      <xsd:sequence>
        <xsd:element name="givenName" type="xsd:string"/>
        <xsd:element name="initial" type="xsd:string"/>
        <xsd:element name="familyName" type="xsd:string"/>
      </xsd:sequence>
    </xsd:complexType>
    <xsd:element name="children">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element ref="ChildInformation" minOccurs="0"
maxOccurs="unbounded"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="ChildInformation">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name="name" type="name"/>
          <xsd:element name="dateOfBirth" type="xsd:date"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:simpleType name="ListOfDays">
      <xsd:list itemType="Day"/>
    </xsd:simpleType>
    <xsd:simpleType name="Day">
      <xsd:restriction base="xsd:string">
        <xsd:enumeration value="monday"/>
        <xsd:enumeration value="tuesday"/>
        <xsd:enumeration value="wednesday"/>
        <xsd:enumeration value="thursday"/>
        <xsd:enumeration value="friday"/>
        <xsd:enumeration value="saturday"/>
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:schema>
```

```

        <xsd:enumeration value="sunday"/>
      </xsd:restriction>
    </xsd:simpleType>
    <xsd:element name="namesOfMemberNations">
      <xsd:simpleType>
        <xsd:list itemType="xsd:string"/>
      </xsd:simpleType>
    </xsd:element>
    <xsd:element name="fileIdentifier">
      <xsd:complexType>
        <xsd:choice>
          <xsd:element name="serialNumber" type="xsd:int"/>
          <xsd:element name="relativeName" type="xsd:string"/>
          <xsd:element ref="unidentified"/>
        </xsd:choice>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="unidentified">
      <xsd:complexType>
        <xsd:complexContent>
          <xsd:restriction base="xsd:anyType"/>
        </xsd:complexContent>
      </xsd:complexType>
    </xsd:element>
  </xsd:schema>

```

C.2 Definiciones ASN.1 correspondientes

Lo que sigue es la especificación ASN.1 correspondiente que valida los mismos documentos XML que el esquema XSD:

```

EXAMPLES{joint-iso-itu-t asn1(1) examples(999) xml-defined-types(3)}
DEFINITIONS AUTOMATIC TAGS

XER INSTRUCTIONS ::=
BEGIN

IMPORTS String, Decimal, Int, Date, AnyType

    FROM XSD
    {joint-iso-itu-t asn1(1) specification(0) modules(0) xsd-module(1)};

EXAMPLES ::= SEQUENCE {
    personnelRecord      PersonnelRecord,
    number                Decimal,
    daysOfTheWeek         ListOfDays,
    namesOfMemberNations NamesOfMemberNations,
    fileIdentifier-list   [UNTAGGED]
        SEQUENCE (SIZE(1..MAX)) OF fileIdentifier FileIdentifier }

PersonnelRecord ::= [NAME AS UNCAPITALIZED] SEQUENCE {
    name          Name,
    title         XSD.String,
    number        INTEGER,
    dateOfHire    Date,
    nameOfSpouse  NameOfSpouse,
    children      Children }

NameOfSpouse ::= [NAME AS UNCAPITALIZED] Name

Name ::= [NAME AS UNCAPITALIZED] SEQUENCE {
    givenName      XSD.String,
    initial        XSD.String,
    familyName     XSD.String }

Children ::= [NAME AS UNCAPITALIZED] SEQUENCE {
    childInformation-list [UNTAGGED]
        SEQUENCE OF ChildInformation }

ChildInformation ::= SEQUENCE {
    name          Name,
    dateOfBirth   Date }

```

```

ListOfDays ::= [LIST] SEQUENCE OF Day
Day ::= ENUMERATED {monday, tuesday, wednesday, thursday, friday,
                    saturday, sunday}
NamesOfMemberNations ::= [NAME AS UNCAPITALIZED] [LIST] SEQUENCE OF XSD.String
FileIdentifier ::= [NAME AS UNCAPITALIZED] SEQUENCE {
    choice [UNTAGGED] CHOICE {
        serialNumber      Int,
        relativeName      XSD.String,
        unidentified      UNIDENTIFIED    } }
UNIDENTIFIED ::= [NAME AS LOWERCASED] XSD.AnyType
ENCODING-CONTROL XER
GLOBAL-DEFAULTS MODIFIED-ENCODINGS
END

```

C.3 Otros ejemplos

En esta subcláusula, todos los ejemplos parciales (los ejemplos que no contienen el elemento **esquema**) presuponen que los elementos XML que representan la sintaxis XSD se incluyen en una declaración de **espacio nominal por defecto** cuyo **nombre de espacio nominal** es el **espacio nominal de destino** del esquema.

C.3.1 Documentos de esquema con los elementos de información **import** e **include**

El esquema XSD siguiente está compuesto de dos espacios nominales que provienen de cuatro archivos del esquema:

```

<!-- file "http://example.com/xyz/schema.xsd" -->
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xyz="http://example.com/xyz"
  targetNamespace="http://example.com/xyz">
  <xsd:element name="xyz-elem" type="xsd:string"/>
  <xsd:complexType name="Xyz-type">
    <xsd:attribute name="xyz-attr" type="xsd:boolean"/>
  </xsd:complexType>
</xsd:schema>

<!-- file "http://example.com/abc/main.xsd" -->
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:abc="http://example.com/abc"
  targetNamespace="http://example.com/xyz">
  <xsd:include schemaLocation="http://example.com/abc/sub1.xsd"/>
  <xsd:import namespace="http://www.w3.org/2001/XMLSchema"
    schemaLocation="http://example.com/xyz/schema.xsd"/>
  <xsd:redefine schemaLocation="http://example.com/abc/sub2.xsd">
    <xsd:attribute name="sub2-attr" type="xsd:token"/>
  </xsd:redefine>
  <xsd:element name="abc-elem" type="Xyz-type"/>
</xsd:schema>

<!-- file "http://example.com/abc/sub1.xsd" -->
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:abc="http://example.com/abc"
  targetNamespace="http://example.com/xyz">
  <xsd:element name="sub1-elem" type="xsd:string"/>
</xsd:schema>

<!-- file "http://example.com/abc/sub2.xsd" -->
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <xsd:element name="sub2-elem" type="xsd:string"/>
  <xsd:attribute name="sub2-attr" type="xsd:string"/>
</xsd:schema>

```

Estos cuatro documentos de esquema se corresponden con los dos módulos ASN.1 siguientes:

```

XYZ  -- The module reference is not standardized
DEFINITIONS AUTOMATIC TAGS ::=
BEGIN

  Xyz-elem ::= [NAME AS UNCAPITALIZED] XSD.String
  Xyz-type ::= SEQUENCE {
    xyz-attr [ATTRIBUTE] BOOLEAN OPTIONAL }

  ENCODING-CONTROL XER
    GLOBAL-DEFAULTS MODIFIED-ENCODINGS
    GLOBAL-DEFAULTS CONTROL-NAMESPACE
    "http://www.w3.org/2001/XMLSchema-instance"
  END

ABC  -- The module reference is not standardized
DEFINITIONS AUTOMATIC TAGS ::=
BEGIN
  IMPORTS
  Xyz-type FROM XYZ
  Token, String FROM XSD;

  Abc-elem ::= [NAME AS UNCAPITALIZED] Xyz-type
  Sub1-elem ::= [NAME AS UNCAPITALIZED] XSD.String
  Sub2-elem ::= [NAME AS UNCAPITALIZED] XSD.String
  Sub2-attr ::= [NAME AS UNCAPITALIZED] [ATTRIBUTE] XSD.Token

  ENCODING-CONTROL XER
    GLOBAL-DEFAULTS MODIFIED-ENCODINGS
    GLOBAL-DEFAULTS CONTROL-NAMESPACE
    "http://www.w3.org/2001/XMLSchema-instance"
  END

```

C.3.2 Correspondencia de definiciones de tipo simples

C.3.2.1 Definición de tipo simple derivada mediante restriction

Para un conjunto completo de ejemplos de restricciones de tipo simple, véanse los ejemplos de las facetas en C.3.3.

C.3.2.2 Definición de tipo simple derivada mediante list

```

<xsd:simpleType name="Int-list">
  <xsd:list itemType="xsd:integer"/>
</xsd:simpleType>

<xsd:simpleType name="Int-10-to-100-list">
  <xsd:list>
    <xsd:simpleType>
      <xsd:restriction base="xsd:integer">
        <xsd:minInclusive value="10"/>
        <xsd:minInclusive value="100"/>
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:list>
</xsd:simpleType>

```

Estas definiciones de tipo simples se corresponden con las asignaciones de tipo ASN.1 siguientes:

```

Int-list ::= [LIST] SEQUENCE OF INTEGER
Int-10-to-100-list ::= [LIST] SEQUENCE OF INTEGER (10..100)

```

C.3.2.3 Definición de tipo simple derivada mediante union

```

<xsd:simpleType name="Int-or-boolean">
  <xsd:union itemType="xsd:integer xsd:boolean"/>
</xsd:simpleType>

<xsd:simpleType name="Time-or-int-or-boolean--or-dateRestriction">
  <xsd:union itemType="xsd:time Int-or-boolean">
    <xsd:simpleType>
      <xsd:restriction base="xsd:date">
        <xsd:minInclusive value="2003-01-01"/>
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:union>
</xsd:simpleType>

```

```

</xsd:union>
</xsd:simpleType>

```

Estas **definiciones de tipo simple** se corresponden con las asignaciones de tipo ASN.1 siguientes:

```

Int-or-boolean ::= [USE-UNION] CHOICE {
    integer [NAMESPACE "http://www.w3.org/2001/XMLSchema"] INTEGER,
    boolean [NAMESPACE "http://www.w3.org/2001/XMLSchema"] BOOLEAN }

Time-or-int-or-boolean-or-dateRestriction ::= [USE-UNION] CHOICE {
    time [NAMESPACE "http://www.w3.org/2001/XMLSchema"] XSD.Time,
    integer [NAMESPACE "http://www.w3.org/2001/XMLSchema"] INTEGER,
    boolean [NAMESPACE "http://www.w3.org/2001/XMLSchema"] BOOLEAN,
    alt [NAME AS ""] XSD.Date (CONSTRAINED BY
        { /* minInclusive="2003-01-01" */ }) }

```

C.3.2.4 Correspondencia de jerarquías de derivación de tipo para definiciones de tipo simples

```

<xsd:simpleType name="Int-10-to-50">
    <xsd:restriction base="xsd:integer">
        <xsd:minExclusive value="10"/>
        <xsd:maxExclusive value="50"/>
    </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="Ten-multiples">
    <xsd:restriction base="Int-10-to-50">
        <xsd:enumeration value="20"/>
        <xsd:enumeration value="30"/>
        <xsd:enumeration value="40"/>
    </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="Twenty-multiples">
    <xsd:restriction base="Ten-multiples">
        <xsd:pattern value=".[02468]0[0]"/>
    </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="Stock-level">
    <xsd:extension base="Int-10-to-50">
        <xsd:attribute name="procurement" type="Int-10-to-50"/>
    </xsd:extension>
</xsd:complexType>

```

Estas **definiciones de tipo simples** se corresponden con las asignaciones de tipo ASN.1 siguientes:

```

Int-10-to-50 ::= INTEGER (10<..<50)

Ten-multiples ::= ENUMERATED {int20(20), int30(30), int40(40)}

Twenty-multiples ::= ENUMERATED {int20(20), int40(40)}

Stock-level ::= SEQUENCE {
    procurement [ATTRIBUTE] Int-10-to-50 OPTIONAL,
    base Int-10-to-50-derivations }

Ten-multiples-derivations ::= [USE-TYPE] CHOICE {
    ten-multiples [NAME AS CAPITALIZED] Ten-multiples,
    twenty-multiples [NAME AS CAPITALIZED] Twenty-multiples }

Int-10-to-50-derivations ::= [USE-TYPE] CHOICE {
    int-10-to-50 [NAME AS CAPITALIZED] Int-10-to-50,
    ten-multiples [NAME AS CAPITALIZED] Ten-multiples,
    twenty-multiples [NAME AS CAPITALIZED] Twenty-multiples,
    stock-level [NAME AS CAPITALIZED] Stock-level }

```

C.3.3 Correspondencia de facetas

C.3.3.1 length, minLength y maxLength

```

<xsd:simpleType name="String-10">
    <xsd:restriction base="xsd:string">
        <xsd:length value="10"/>
    </xsd:restriction>
</xsd:simpleType>

```



```
<xsd:simpleType name="String-5-to-10">
  <xsd:restriction base="xsd:string">
    <xsd:minLength value="5"/>
    <xsd:maxLength value="10"/>
  </xsd:restriction>
</xsd:simpleType>
```

Estas dos **definiciones de tipo simples** se corresponden con las asignaciones de tipo ASN.1 siguientes:

```
String-10 ::= XSD.String (SIZE(10))
String-5-to-10 ::= XSD.String (SIZE(5..10))
```

C.3.3.2 pattern

```
<xsd:simpleType name="My-filename">
  <xsd:restriction base="xsd:string">
    <xsd:pattern value="[&#00;-&#FF;]*"/>
    <xsd:pattern value="/?([^\/*]*[^\/*]*/)" />
  </xsd:restriction>
</xsd:simpleType>
```

Esta **definición de tipo simple** se corresponde con la asignación de tipo ASN.1 siguiente:

```
My-filename ::= XSD.String
  (CONSTRAINED BY
    { /* XML representation of the XSD pattern
      "&#00;-&#FF;" | "/?([^\/*]*[^\/*]*)" */ })
```

C.3.3.3 whiteSpace

```
<xsd:simpleType name="My-String">
  <xsd:restriction base="xsd:string">
    <xsd:whitespace value="preserve"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="My-NormalizedString">
  <xsd:restriction base="xsd:string">
    <xsd:whitespace value="replace"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="My-TokenString">
  <xsd:restriction base="xsd:string">
    <xsd:whitespace value="collapse"/>
  </xsd:restriction>
</xsd:simpleType>
```

Estas **definiciones de tipo simple** se corresponden con las asignaciones de tipo ASN.1 siguientes:

```
My-String ::= XSD.String
My-NormalizedString ::= [WHITESPACE REPLACE] XSD.String
  (FROM ({0, 0, 0, 32} .. {0, 16, 255, 255}))
My-TokenString ::= [WHITESPACE REPLACE] XSD.String
  (FROM ({0, 0, 0, 32} .. {0, 16, 255, 255}))
  (PATTERN "([^\ ]([^\ ]| [^\ ])*)?")
```

C.3.3.4 minInclusive, minExclusive, maxInclusive y maxExclusive

```
<xsd:simpleType name="Int-10-to-100">
  <xsd:restriction base="xsd:integer">
    <xsd:minExclusive value="10"/>
    <xsd:maxInclusive value="100"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="Pi-approximation">
  <xsd:restriction base="xsd:double">
    <xsd:minExclusive value="3.14159"/>
    <xsd:maxExclusive value="3.1416"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="Morning">
  <xsd:restriction base="xsd:time">
    <xsd:minInclusive value="00:00:00"/>
    <xsd:maxExclusive value="12:00:00"/>
  </xsd:restriction>
</xsd:simpleType>
```

```

</xsd:restriction>
</xsd:simpleType>

```

Estas **definiciones de tipo simples** se corresponden con las asignaciones de tipo ASN.1 siguientes:

```

Int-10-to-100 ::= INTEGER (10<..100)

Pi-approximation ::= XSD.Double (3.14159<.. $\pi$ .1416)

Morning ::= XSD.Time (CONSTRAINED BY
  { /* minInclusive="00:00:00" maxExclusive="12:00:00" */ })

```

C.3.3.5 totalDigits y fractionDigits

```

<xsd:simpleType name="RefundableExpenses">
  <xsd:restriction base="xsd:decimal">
    <xsd:totalDigits="5"/>
    <xsd:fractionDigits value="2"/>
  </xsd:restriction>
</xsd:simpleType>

```

Esta **definición de tipo simple** se corresponde con la asignación de tipo ASN.1 siguiente:

```

RefundableExpenses ::= XSD.Decimal (CONSTRAINED BY
  { /* totalDigits="5" fractionDigits="2" */ })

```

C.3.3.6 enumeration

```

<xsd:simpleType name="FarmAnimals">
  <xsd:restriction base="xsd:normalizedString">
    <xsd:enumeration value="Horse"/>
    <xsd:enumeration value="Bull"/>
    <xsd:enumeration value="Cow"/>
    <xsd:enumeration value="Pig"/>
    <xsd:enumeration value="Duck"/>
    <xsd:enumeration value="Goose"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="PrimeNumbersBelow30">
  <xsd:restriction base="xsd:integer">
    <xsd:enumeration value="2"/>
    <xsd:enumeration value="3"/>
    <xsd:enumeration value="5"/>
    <xsd:enumeration value="7"/>
    <xsd:enumeration value="11"/>
    <xsd:enumeration value="13"/>
    <xsd:enumeration value="17"/>
    <xsd:enumeration value="19"/>
    <xsd:enumeration value="23"/>
    <xsd:enumeration value="29"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="X680-release">
  <xsd:restriction base="xsd:gYearMonth">
    <xsd:enumeration value="2002-07"/>
    <xsd:enumeration value="1997-12"/>
    <xsd:enumeration value="1994-07"/>
  </xsd:restriction>
</xsd:simpleType>

```

Estas **definiciones de tipo simples** se corresponden con las asignaciones de tipo ASN.1 siguientes:

```

FarmAnimals ::= ENUMERATED {horse, bull, cow, pig, duck, goose}

PrimeNumbersBelow30 ::= [USE-NUMBER] ENUMERATED {int2(2), int3(3), int5(5),
  int7(7), int11(11), int13(13), int17(17), int19(19), int23(23), int29(29)}

X680-release ::= XSD.GYearMonth ("2002-07" | "1997-12" | "1994-07")

```

La siguiente instrucción de codificación se incluye en la sección de control de codificación XER:

```
TEXT FarmAnimals:ALL AS CAPITALIZED
```

C.3.3.7 enumeration en conjunción con otras facetas

Los siguientes ejemplos están basados en la herencia de facetas que utilizan la restricción de algunos de los tipos definidos en C.3.3.6.

```

<xsd:simpleType name="FarmAnimals-subset">
  <xsd:restriction base="FarmAnimals">
    <xsd:minLength value="4"/>
    <xsd:pattern value="[^oe]*/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="PrimeNumbersBelow30-subset">
  <xsd:restriction base="PrimeNumbersBelow30">
    <xsd:minExclusive value="5"/>
    <xsd:pattern value=".*[23].*/>
  </xsd:restriction>
</xsd:simpleType>

```

Estas **definiciones de tipo simples** se corresponden con las asignaciones de tipo ASN.1 siguientes:

```

/* Horse and Goose do not satisfy the pattern facet
   Cow and Pig do not satisfy the minLength facet */
FarmAnimals-subset ::= ENUMERATED {bull, duck}

/* 2, 3 and 5 do not satisfy the minExclusive facet
   2, 5, 7, 11, 17 and 19 do not satisfy the pattern facet */
PrimeNumbersBelow30-subset ::= [USE-NUMBER] ENUMERATED {int13(13), int23(23),
                                                         int29(29)}

```

La siguiente instrucción de codificación se incluye en la sección de control de codificación XER:

```
TEXT FarmAnimals-subset:ALL AS CAPITALIZED
```

C.3.4 Correspondencia de declaraciones de elemento

C.3.4.1 Declaraciones de elemento cuya definición de tipo es una definición de tipo simple o una definición de tipo compleja de nivel superior definida por el usuario

```

<xsd:element name="Forename" type="xsd:token"/>
<xsd:element name="File" type="My-filename"/>
<xsd:element name="Value" type="Int-10-to-50"/>

```

Estas **declaraciones de elemento** se corresponden con las asignaciones de tipo ASN.1 siguientes:

```

Forename ::= XSD.Token
File ::= My-filename
Value ::= Int-10-to-50-derivations

```

NOTA – El tipo **"My-filename"** y su correspondencia con ASN.1 se definen en C.3.3.2.; el tipo **"Int-10-to-50"** y su correspondencia con ASN.1 se definen en C.3.2.4.

C.3.4.2 Declaraciones de elemento cuya definición de tipo es una definición de tipo simple o una definición de tipo compleja anónima

```

<xsd:element name="maxOccurs">
  <xsd:simpleType>
    <xsd:union memberTypes="xsd:nonNegativeInteger">
      <xsd:simpleType>
        <xsd:restriction base="xsd:token">
          <xsd:enumeration name="unbounded"/>
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:union>
  </xsd:simpleType>
</xsd:element>

<xsd:element name="address">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="line-1" type="xsd:token"/>
      <xsd:element name="line-2" type="xsd:token"/>
      <xsd:element name="city" type="xsd:token"/>
      <xsd:element name="state" type="xsd:token" minOccurs="0"/>
      <xsd:element name="zip" type="xsd:token"/>
    </xsd:sequence>
    <xsd:attribute name="country" type="xsd:token"/>
  </xsd:complexType>

```

```
</xsd:element>
```

Estas **declaraciones de elemento** se corresponden con las asignaciones de tipo ASN.1 siguientes:

```
MaxOccurs ::= [NAME AS UNCAPITALIZED] [USE-UNION] CHOICE {
    nonNegativeInteger [NAMESPACE AS "http://www.w3.org/2001/XMLSchema"]
        XSD.NonNegativeInteger,
    alt [NAME AS ""] ENUMERATED {unbounded} }

Address ::= [NAME AS UNCAPITALIZED] SEQUENCE {
    Country [ATTRIBUTE] XSD.Token OPTIONAL,
    line-1 XSD.Token,
    line-2 XSD.Token,
    city XSD.Token,
    state XSD.Token OPTIONAL,
    zip XSD.Token }
```

C.3.4.3 Declaraciones de elemento que son cabecera de un grupo de sustitución de elemento

```
<xsd:element name="Tic" type="xsd:integer" abstract="true"/>
<xsd:element name="Tac" type="xsd:byte" substitutionGroup="Tic"/>
<xsd:element name="Toe" substitutionGroup="Tic"/>
<xsd:element name="Foo" type="xsd:date"/>
<xsd:element name="Bar" substitutionGroup="Foo"/>
```

Estas **declaraciones de elemento** se corresponden con:

```
Tac ::= INTEGER (-128..127)
Toe ::= INTEGER
Tic-group ::= [UNTAGGED] CHOICE {
    tac [NAME AS CAPITALIZED] Tac,
    toe [NAME AS CAPITALIZED] Toe }
Foo ::= XSD.Date
Bar ::= XSD.Date
Foo-group ::= [UNTAGGED] CHOICE {
    foo [NAME AS CAPITALIZED] Foo,
    bar [NAME AS CAPITALIZED] Bar }
```

C.3.4.4 Declaraciones de elemento con una restricción de valor que es un valor por defecto

C.3.4.4.1 La siguiente es una **declaración de elemento** con una **definición de tipo simple** anónima, no utilizada como **definición de tipo básica** para ningún tipo.

```
<xsd:element name="Telephone" type="xsd:token" default="undefined"/>
```

Esta **declaración de elemento** se corresponde con la asignación de tipo ASN.1 siguiente:

```
Telephone ::= [DEFAULT-FOR-EMPTY "undefined"] XSD.Token
```

C.3.4.4.2 La siguiente es una **declaración de elemento** con una **definición de tipo compleja** anónima, con contenido simple, no utilizada como **definición de tipo básica** para ningún tipo.

```
<xsd:element name="InternationalTelephone" default="undefined">
  <xsd:simpleContent>
    <xsd:extension base="xsd:token">
      <xsd:attribute name="country-code" type="xsd:integer"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:element>
```

Esta **declaración de elemento** se corresponde con la asignación de tipo ASN.1 siguiente:

```
InternationalTelephone ::= [DEFAULT-FOR-EMPTY "undefined"] SEQUENCE {
    country-code [ATTRIBUTE] INTEGER OPTIONAL,
    base [UNTAGGED] XSD.Token }
```

C.3.4.4.3 La siguiente es una **declaración de elemento** con una **definición de tipo compleja** anónima. La **definición de tipo compleja** tiene un contenido complejo que es **mixto** y se puede vaciar, y no se utiliza como **definición de tipo básico** para ningún tipo.

```
<xsd:element name="Description" default="absent" mixed="true">
  <xsd:choice minOccurs="0" maxOccurs="unbounded">
    <xsd:element name="bold" type="xsd:string"/>
    <xsd:element name="italic" type="xsd:string"/>
  </xsd:choice>
</xsd:element>
```

Esta **declaración de elemento** se corresponde con la asignación de tipo ASN.1 siguiente:

```
Description ::= [EMBED-VALUES] [DEFAULT-FOR-EMPTY "absent"] SEQUENCE {
  embed-values SEQUENCE OF XSD.String,
  choice-list [UNTAGGED] SEQUENCE OF [UNTAGGED] CHOICE {
    bold XSD.String,
    italic XSD.String } } (CONSTRAINED BY
  { /* Shall conform to ITU-T Rec. X.693 / ISO/IEC 8825-4, clause 25 */ })
```

C.3.4.4.4 En el siguiente ejemplo, la **definición de tipo** de la **declaración de elemento** se utiliza como la **definición de tipo básica** de otro tipo.

Este ejemplo utiliza los tipos XSD y ASN.1 del ejemplo citado en C.3.2.4.

```
<xsd:element name="Quantity" type="Int-10-to-50" default="20"/>
```

Esta **declaración de elemento** se corresponde con la asignación de tipo ASN.1 siguiente:

```
Quantity ::= Int-10-to-50-deriv-default-20
```

Si todavía no se ha generado ningún tipo ASN.1 correspondiente a `Int-10-to-50`, con un valor por defecto de "20", se generará también el tipo siguiente:

```
Int-10-to-50-deriv-default-20 ::= [USE-TYPE] CHOICE {
  int-10-to-50 [NAME AS CAPITALIZED] [DEFAULT-FOR-EMPTY 20]
    Int-10-to-50,
  ten-multiples [NAME AS CAPITALIZED] [DEFAULT-FOR-EMPTY int20]
    Ten-multiples,
  twenty-multiples [NAME AS CAPITALIZED] [DEFAULT-FOR-EMPTY int20]
    Twenty-multiples,
  stock-level [NAME AS CAPITALIZED] [DEFAULT-FOR-EMPTY 20]
    Stock-level }
```

C.3.4.5 Declaración de elemento con una restricción de valor que tiene un valor fijo

C.3.4.5.1 La siguiente es una **declaración de elemento** con una **definición de tipo simple** anónima, no utilizada como **definición de tipo básica** para ningún tipo.

```
<xsd:element name="UnknownTelephone" type="xsd:token" fixed="undefined"/>
```

Esta **declaración de elemento** se corresponde con la asignación de tipo ASN.1 siguiente:

```
UnknownTelephone ::= [DEFAULT-FOR-EMPTY "undefined"] XSD.Token ("undefined")
```

C.3.4.5.2 La siguiente es una **declaración de elemento** con una **definición de tipo compleja** anónima. La **definición de tipo compleja** tiene un contenido simple y no se utiliza como **definición de tipo básico** para ningún tipo.

```
<xsd:element name="UnknownInternationalTelephone" fixed="undefined">
  <xsd:simpleContent>
    <xsd:extension base="xsd:token">
      <xsd:attribute name="country-code" type="xsd:integer"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:element>
```

Esta **declaración de elemento** se corresponde con la asignación de tipo ASN.1 siguiente:

```
UnknownInternationalTelephone ::= [DEFAULT-FOR-EMPTY "undefined"] SEQUENCE {
  country-code [ATTRIBUTE] INTEGER OPTIONAL,
  base [UNTAGGED] XSD.Token }
  (WITH COMPONENTS {..., base ("undefined")})
```

C.3.4.5.3 La siguiente es una **declaración de elemento** con una **definición de tipo compleja** anónima. La **definición de tipo compleja** tiene un contenido complejo que es mixto y se puede vaciar, y no se utiliza como **definición de tipo básica** para ningún tipo.

```
<xsd:element name="UnknownDescription" fixed="absent" mixed="true">
  <xsd:choice minOccurs="0" maxOccurs="unbounded">
    <xsd:element name="bold" type="xsd:string"/>
    <xsd:element name="italic" type="xsd:string"/>
  </xsd:choice>
```

```
</xsd:choice>
</xsd:element>
```

Esta **declaración de elemento** se corresponde con la asignación de tipo ASN.1 siguiente:

```
UnknownDescription ::= [EMBED-VALUES] [DEFAULT-FOR-EMPTY "absent"] SEQUENCE {
    embed-values      SEQUENCE OF XSD.String,
    choice-list       [UNTAGGED] SEQUENCE OF [UNTAGGED] CHOICE {
        bold          XSD.String,
        italic        XSD.String } }
(CONSTRAINED BY
    { /* Shall conform to ITU-T Rec. X.693 / ISO/IEC 8825-4, clause 25 */ })
(WITH COMPONENTS { embed-values ({"absent"}),
    choice-list (SIZE(0)) })
```

C.3.4.5.4 La **definición de tipo** de la siguiente **declaración de elemento** es una **definición de tipo simple** utilizada como la **definición de tipo básica** de otro tipo.

Este ejemplo utiliza los tipos XSD y ASN.1 del ejemplo citado en C.3.2.4.

```
<xsd:element name="Quantity" type="Int-10-to-50" fixed="20"/>
```

Esta **declaración de elemento** se corresponde con la asignación de tipo ASN.1 siguiente:

```
Quantity ::= Int-10-to-50-deriv-fixed-20
```

Si todavía no se ha generado ningún tipo ASN.1 correspondiente a `Int-10-to-50`, con un valor fijo de "20", se generará también el tipo siguiente:

```
Int-10-to-50-deriv-fixed-20 ::= [USE-TYPE] CHOICE {
    int-10-to-50      [NAME AS CAPITALIZED] [DEFAULT-FOR-EMPTY 20]
        Int-10-to-50,
    ten-multiples     [NAME AS CAPITALIZED] [DEFAULT-FOR-EMPTY int20]
        Ten-multiples,
    twenty-multiples  [NAME AS CAPITALIZED] [DEFAULT-FOR-EMPTY int20]
        Twenty-multiples,
    stock-level       [NAME AS CAPITALIZED] [DEFAULT-FOR-EMPTY 20]
        Stock-level }
(WITH COMPONENTS {
    int-10-to-50 (20),
    ten-multiples (int20),
    twenty-multiples (int20),
    stock-level (WITH COMPONENTS {..., base (20)}) })
```

C.3.4.6 Declaraciones de elemento que son anulables

C.3.4.6.1 El ejemplo siguiente muestra una **declaración de elemento** que es **anulable** y cuya **definición de tipo** es un tipo de datos incorporados XSD.

```
<xsd:element name="Nillable-1" type="xsd:string" nillable="true"/>
```

Esta **declaración de elemento** se corresponde con la asignación de tipo ASN.1 siguiente:

```
Nillable-1 ::= [USE-NIL] SEQUENCE {
    content XSD.String OPTIONAL }
```

C.3.4.6.2 El ejemplo siguiente muestra una **declaración de elemento** que es **anulable** y cuya **definición de tipo** es una **definición de tipo compleja** anónima.

```
<xsd:element name="Nillable-2" nillable="true">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="a" type="xsd:string"/>
            <xsd:element name="b" type="xsd:string"/>
        </xsd:sequence>
        <xsd:attribute name="b" type="xsd:boolean"/>
    </xsd:complexType>
</xsd:element>
```

Esta **declaración de elemento** se corresponde con la asignación de tipo ASN.1 siguiente:

```
Nillable-2 ::= [USE-NIL] SEQUENCE {
    b      [ATTRIBUTE] BOOLEAN OPTIONAL,
    content SEQUENCE {
```

```

a      XSD.String,
b      XSD.String } OPTIONAL }

```

C.3.4.6.3 El ejemplo siguiente muestra una **declaración de elemento** que es **anulable** y cuya **definición de tipo** es una **definición de tipo compleja** de nivel superior.

```

<xsd:complexType name="Foo">
  <xsd:sequence>
    <xsd:element name="a" type="xsd:string"/>
    <xsd:element name="b" type="xsd:string"/>
  </xsd:sequence>
  <xsd:attribute name="b" type="xsd:boolean"/>
</xsd:complexType>

<xsd:element name="Nillable-3" type="Foo" nillable="true"/>

```

Estos componentes de esquema se corresponden con las asignaciones de tipo ASN.1 siguientes:

```

Foo ::= SEQUENCE {
  b      [ATTRIBUTE] BOOLEAN OPTIONAL,
  a      XSD.String,
  b-1    [NAME AS "b"] XSD.String }

Foo-nillable ::= [USE-NIL] SEQUENCE {
  b      [ATTRIBUTE] BOOLEAN OPTIONAL,
  content SEQUENCE {
    a      XSD.String,
    b      XSD.String } OPTIONAL }

Nillable-3 ::= Foo-nillable

```

C.3.4.6.4 El ejemplo siguiente muestra una **declaración de elemento** que es **anulable** y cuya **definición de tipo** es una **definición de tipo compleja** de nivel superior y que se utiliza como la **definición de tipo básica** de otra **definición de tipo compleja**.

Además de los componentes de esquema de C.3.4.6.3 se definen los componentes de esquema siguientes:

```

<xsd:complexType name="Bar">
  <xsd:complexContent>
    <xsd:extension base="Foo">
      <xsd:sequence>
        <xsd:element name="z" type="xsd:string"/>
      </xsd:sequence>
      <xsd:attribute name="c" type="xsd:boolean"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>

<xsd:element name="Nillable-4" type="Foo" nillable="true"/>

```

Además del tipo Foo de C.3.4.6.3, se generan los tipos ASN.1 siguientes:

```

Bar ::= SEQUENCE {
  b      [ATTRIBUTE] BOOLEAN OPTIONAL,
  c      [ATTRIBUTE] BOOLEAN OPTIONAL,
  a      XSD.String,
  b-1    [NAME AS "b"] XSD.String,
  z      XSD.String }

Foo-derivations ::= [USE-TYPE] CHOICE {
  foo    [NAME AS CAPITALIZED] Foo,
  bar    [NAME AS CAPITALIZED] Bar }

Foo-nillable ::= [USE-NIL] SEQUENCE {
  b      [ATTRIBUTE] BOOLEAN OPTIONAL,
  content SEQUENCE {
    a      XSD.String,
    b      XSD.String } OPTIONAL }

Bar-nillable ::= [USE-NIL] SEQUENCE {
  b      [ATTRIBUTE] BOOLEAN OPTIONAL,
  c      [ATTRIBUTE] BOOLEAN OPTIONAL,
  content SEQUENCE {
    a      XSD.String,
    b      XSD.String,
    z      XSD.String } OPTIONAL }

```

```

Foo-deriv-nillable ::= [USE-TYPE] CHOICE {
    foo      [NAME AS CAPITALIZED] Foo-nillable,
    bar      [NAME AS CAPITALIZED] Bar-nillable }

Nillable-4 ::= Foo-deriv-nillable

```

C.3.5 Correspondencia de usos de atributo y de declaraciones de atributo

C.3.5.1 El siguiente es un ejemplo de una **declaración de atributo** de nivel superior cuya **definición de tipo** es una **definición de tipo simple** de nivel superior definida por el usuario.

```
<xsd:attribute name="name" type="NCName"/>
```

Esta **declaración de atributo** se corresponde con la asignación de tipo ASN.1 siguiente:

```
Name ::= [NAME AS UNCAPITALIZED] [ATTRIBUTE] XSD.NCName
```

C.3.5.2 El siguiente es un ejemplo de una **declaración de atributo** de nivel superior cuya **definición de tipo** es una **definición de tipo simple**.

```

<xsd:attribute name="form">
  <xsd:simpleType>
    <xsd:restriction base="xsd:token">
      <xsd:enumeration value="qualified"/>
      <xsd:enumeration value="unqualified"/>
    </xsd:restriction>
  </xsd:simpleType>
</xsd:attribute>

```

Esta **declaración de atributo** se corresponde con la asignación de tipo ASN.1 siguiente:

```
Form ::= [NAME AS UNCAPITALIZED] [ATTRIBUTE] ENUMERATED {qualified, unqualified}
```

C.3.5.3 El siguiente ejemplo es un **uso de atributo** con una **restricción de valor** que tiene un valor por **defecto**.

La **declaración de atributo** cuyo nombre es "form", que se referencia en este ejemplo, se define en C.3.5.2.

```

<xsd:complexType name="element">
  <xsd:attribute name="name" type="xsd:NCName" default="NAME"/>
  <xsd:attribute ref="form" default="qualified"/>
</xsd:complexType>

```

Esta **definición de tipo compleja** se corresponde con la asignación de tipo ASN.1 siguiente:

```

Element ::= [NAME AS UNCAPITALIZED] SEQUENCE {
    name [ATTRIBUTE] XSD.NCName DEFAULT "NAME",
    form [ATTRIBUTE] Form DEFAULT qualified }

```

C.3.5.4 Este ejemplo muestra una **declaración de atributo** de nivel superior con una **restricción de valor** que tiene un valor por **defecto** y un **uso de atributo** con esa **declaración de atributo**.

```

<xsd:attribute name="minOccurs" type="xsd:nonNegativeInteger" default="1"/>
<xsd:attribute name="maxOccurs" default="1"/>
  <xsd:simpleType>
    <xsd:union memberTypes="xsd:nonNegativeInteger" >
      <xsd:simpleType>
        <xsd:restriction base="xsd:NMTOKEN">
          <xsd:enumeration value="unbounded"/>
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:union>
  </xsd:simpleType>
</xsd:attribute>

<xsd:complexType name="Particle">
  <xsd:sequence>
    <xsd:element name="particle"/>
  </xsd:sequence>
  <xsd:attribute ref="minOccurs"/>
  <xsd:attribute ref="maxOccurs" default="unbounded"/>
</xsd:complexType>

```

Estos componentes de esquema se corresponden con las asignaciones de tipo ASN.1 siguientes:

```

MinOccurs ::= [ATTRIBUTE] [NAME AS UNCAPITALIZED] XSD.NonNegativeInteger

MaxOccurs ::= [ATTRIBUTE] [NAME AS UNCAPITALIZED] [USE-UNION] CHOICE {
    nonNegativeInteger      [NAMESPACE AS "http://www.w3.org/2001/XMLSchema"]
                           XSD.NonNegativeInteger,

```



```

alt                                     [NAME AS "" ] ENUMERATED {unbounded} }

Particle ::= SEQUENCE {
    minOccurs [ATTRIBUTE] MinOccurs DEFAULT 1,
    maxOccurs [ATTRIBUTE] MaxOccurs DEFAULT alt : unbounded,
    particle XSD.AnyType }

```

C.3.5.5 Este ejemplo muestra un uso de atributo cuya declaración de atributo tiene un espacio nominal de destino que no es ausente.

```

<xsd:complexType name="Ack">
  <xsd:attribute name="number" type="xsd:integer" form="qualified" />
</xsd:complexType>

```

Esta definición de tipo compleja se corresponde con la asignación de tipo ASN.1 siguiente:

```

Ack ::= SEQUENCE {
    number [NAMESPACE AS "http://targetnamespaceForExample"] [ATTRIBUTE]
        INTEGER OPTIONAL }

```

C.3.6 Correspondencia de definiciones de grupo de modelo

C.3.6.1 La siguiente es una definición de grupo de modelo cuyo grupo de modelo tiene un compositor de secuencia.

```

<xsd:group name="mySequence">
  <xsd:sequence>
    <xsd:element name="a" type="xsd:string"/>
    <xsd:element name="b" type="xsd:boolean"/>
  </xsd:sequence>
</xsd:group>

```

Esta definición de grupo de modelo se corresponde con la asignación de tipo ASN.1 siguiente:

```

MySequence ::= [UNTAGGED] SEQUENCE {
    a XSD.String,
    b BOOLEAN }

```

C.3.6.2 La siguiente es una definición de grupo de modelo cuyo grupo de modelo tiene un compositor de todo:

```

<xsd:group name="myAll ">
  <xsd:all>
    <xsd:element name="a" type="xsd:string"/>
    <xsd:element name="b" type="xsd:boolean"/>
  </xsd:all>
</xsd:group>

```

Esta definición de grupo de modelo no tiene correspondencia con ASN.1. Véase C.3.8.3.1 para un ejemplo de la correspondencia de una definición de tipo compleja en la que el grupo de modelo de esa definición de grupo de modelo se produce como el grupo de modelo de mayor nivel.

C.3.6.3 La siguiente es una definición de grupo de modelo cuyo grupo de modelo tiene un compositor de elección.

```

<xsd:group name="myChoice">
  <xsd:choice>
    <xsd:element name="am" type="xsd:string"/>
    <xsd:element name="bm" type="xsd:boolean"/>
  </xsd:choice>
</xsd:group>

```

Esta definición de grupo de modelo se corresponde con la asignación de tipo ASN.1 siguiente:

```

MyChoice ::= [UNTAGGED] CHOICE {
    am XSD.String,
    bm BOOLEAN }

```

C.3.7 Correspondencia de partículas

La definición de grupo de modelo de C.3.6.3 y su tipo ASN.1 correspondiente se utilizan en algunos de los ejemplos de partículas.

C.3.7.1 El siguiente ejemplo muestra partículas de un grupo de modelo con un compositor de secuencia.

```

<xsd:complexType name="ElementSequence">
  <xsd:sequence>
    <xsd:element name="elem1" type="xsd:boolean"/>
    <xsd:element name="elem2" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="elem3" type="xsd:boolean" minOccurs="2" maxOccurs="5"/>
    <xsd:element name="elem4" type="xsd:boolean" minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>

```

```

        <xsd:element name="elem5" type="xsd:boolean" minOccurs="5" maxOccurs="unbounded"/>
    </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="ModelGroupSequence">
    <xsd:sequence>
        <xsd:group ref="myChoice"/>
        <xsd:choice>
            <xsd:element name="a" type="xsd:string"/>
            <xsd:element name="b" type="xsd:string"/>
        </xsd:choice>
        <xsd:sequence>
            <xsd:element name="c" type="xsd:string"/>
            <xsd:element name="d" type="xsd:string"/>
        </xsd:sequence>
        <xsd:choice minOccurs="3" maxOccurs="12">
            <xsd:element name="e" type="xsd:string"/>
            <xsd:element name="f" type="xsd:string"/>
        </xsd:choice>
    </xsd:sequence>
</xsd:complexType>

```

Estas **definiciones de tipo complejo** se corresponden con las asignaciones de tipo ASN.1 siguientes:

```

ElementSequence ::= SEQUENCE {
    elem1          BOOLEAN,
    elem2          BOOLEAN OPTIONAL,
    elem3-list     [UNTAGGED] SEQUENCE (SIZE(2..5)) OF elem3 BOOLEAN,
    elem4-list     [UNTAGGED] SEQUENCE OF elem4 BOOLEAN,
    elem5-list     [UNTAGGED] SEQUENCE (SIZE(1..MAX)) OF elem5 BOOLEAN }

ModelGroupSequence ::= SEQUENCE {
    myChoice       MyChoice,
    choice         [UNTAGGED] CHOICE {
        a          XSD.String,
        b          XSD.String },
    sequence       [UNTAGGED] SEQUENCE {
        c          XSD.String,
        d          XSD.String },
    choice-list    [UNTAGGED] SEQUENCE (SIZE(3..12)) OF [UNTAGGED] CHOICE {
        e          XSD.String,
        f          XSD.String } }

```

C.3.7.2 El siguiente ejemplo muestra **partículas** de un **grupo de modelo** con un **compositor de todo**.

```

<xsd:complexType name="ElementAll">
    <xsd:all>
        <xsd:element name="elem1" type="xsd:boolean"/>
        <xsd:element name="elem2" type="xsd:boolean" minOccurs="0"/>
    </xsd:all>
</xsd:complexType>

```

Esta **definición de tipo compleja** se corresponde con la asignación de tipo ASN.1 siguiente:

```

ElementAll ::= [USE-ORDER] SEQUENCE {
    order SEQUENCE OF ENUMERATED {elem1, elem2},
    elem1 XSD.String,
    elem2 XSD.String OPTIONAL }
(CONSTRAINED BY
    { /* Shall conform to ITU-T Rec. X.693 / ISO/IEC 8825-4, clause 35 */ })

```

C.3.7.3 El siguiente ejemplo muestra **partículas** de un **grupo de modelo** con un **compositor de elección**.

```

<xsd:complexType name="ElementSequence">
    <xsd:choice>
        <xsd:element name="elem1" type="xsd:boolean"/>
        <xsd:element name="elem2" type="xsd:boolean" minOccurs="0"/>
        <xsd:element name="elem3" type="xsd:boolean" minOccurs="2" maxOccurs="5"/>
        <xsd:element name="elem4" type="xsd:boolean" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="elem5" type="xsd:boolean" minOccurs="5" maxOccurs="unbounded"/>
    </xsd:choice>
</xsd:complexType>

<xsd:complexType name="ModelGroupChoice">
    <xsd:choice>
        <xsd:group ref="myChoice"/>
    </xsd:choice>

```

```

        <xsd:element name="a" type="xsd:string"/>
        <xsd:element name="b" type="xsd:string"/>
    </xsd:choice>
    <xsd:sequence>
        <xsd:element name="c" type="xsd:string"/>
        <xsd:element name="d" type="xsd:string"/>
    </xsd:sequence>
    <xsd:choice minOccurs="3" maxOccurs="12">
        <xsd:element name="e" type="xsd:string"/>
        <xsd:element name="f" type="xsd:string"/>
    </xsd:choice>
</xsd:choice>
</xsd:complexType>

```

Estas **definiciones de tipo complejo** se corresponden con las asignaciones de tipo ASN.1 siguientes:

```

ElementSequence ::= SEQUENCE {
    choice [UNTAGGED] CHOICE {
        elem1      BOOLEAN,
        elem2-list [UNTAGGED] SEQUENCE (SIZE(0..1)) OF elem2 BOOLEAN,
        elem3-list [UNTAGGED] SEQUENCE (SIZE(2..5)) OF elem3 BOOLEAN,
        elem4-list [UNTAGGED] SEQUENCE OF elem4 BOOLEAN,
        elem5-list [UNTAGGED] SEQUENCE (SIZE(5..MAX)) OF elem5 BOOLEAN } }

ModelGroupChoice ::= SEQUENCE {
    choice [UNTAGGED] CHOICE {
        myChoice    MyChoice,
        choice      [UNTAGGED] CHOICE {
            a        XSD.String,
            b        XSD.String },
        sequence    [UNTAGGED] SEQUENCE {
            c        XSD.String,
            d        XSD.String }
        choice-list [UNTAGGED] SEQUENCE (SIZE(3..12)) OF [UNTAGGED] CHOICE {
            e        XSD.String,
            f        XSD.String } }

```

C.3.8 Correspondencia de definiciones de tipo complejas

C.3.8.1 El siguiente ejemplo es una **definición de tipo compleja** cuyo **tipo contenido** está **vacío**.

```

<xsd:complexType name="Null"/>

<xsd:complexType name="Ack">
    <xsd:sequence/>
    <xsd:attribute name="packetNumber" type="xsd:integer"/>
</xsd:complexType>

```

Estas **definiciones de tipo complejas** se corresponden con las asignaciones de tipo ASN.1 siguientes:

```

Null ::= SEQUENCE {}

Ack ::= SEQUENCE {
    packetNumber [ATTRIBUTE] INTEGER OPTIONAL }

```

C.3.8.2 El siguiente ejemplo es una **definición de tipo complejo** cuyo **tipo contenido** es una **definición de tipo simple**.

```

<xsd:complexType name="Formatted">
    <xsd:simpleContent>
        <xsd:extension base="xsd:token">
            <xsd:attribute name="format">
                <xsd:simpleType>
                    <xsd:restriction base="xsd:token">
                        <xsd:enumeration value="bold"/>
                        <xsd:enumeration value="italic"/>
                    </xsd:restriction>
                </xsd:simpleType>
            </xsd:attribute>
        </xsd:extension>
    </xsd:simpleContent>
</xsd:complexType>

```

Esta **definición de tipo compleja** se corresponde con la asignación de tipo ASN.1 siguiente:

```

Formatted ::= SEQUENCE {
    Format    [ATTRIBUTE] ENUMERATED {bold, italic} OPTIONAL,
    Base     [UNTAGGED] XSD.Token }

```

C.3.8.3 Los ejemplos siguientes son **definiciones de tipo complejas** cuyo **tipo contenido** es un modelo de contenido de un **único elemento**.

C.3.8.3.1 En el ejemplo siguiente, el **tipo contenido** es el **grupo de modelo** de una **definición de grupo de modelo**.

Este ejemplo utiliza los tipos definidos en C.3.6.

```
<xsd:complexType name="MyComplexType-1">
  <xsd:group ref="myAll"/>
</xsd:complexType>

<xsd:complexType name="MyComplexType-2">
  <xsd:group ref="myChoice"/>
</xsd:complexType>

<xsd:complexType name="MyComplexType-3">
  <xsd:group ref="mySequence" maxOccurs="100"/>
</xsd:complexType>
```

Estas **definiciones de tipo complejas** se corresponden con las asignaciones de tipo ASN.1 siguientes:

```
MyComplexType-1 ::= [USE-ORDER] SEQUENCE {
  order      SEQUENCE OF ENUMERATED {a,b},
  a          XSD.String,
  b          BOOLEAN }
(CONSTRAINED BY
  { /* Shall conform to ITU-T Rec. X.693 | ISO/IEC 8825-4, clause 35 */ })

MyComplexType-2 ::= SEQUENCE {
  myChoice MyChoice }

MyComplexType-3 ::= SEQUENCE {
  mySequence-list SEQUENCE (SIZE(1..100)) OF MySequence }
```

C.3.8.3.2 En el ejemplo siguiente, el **tipo contenido** es un **grupo de modelo** cuyo **compositor** es **elección**.

```
<xsd:complexType name="MyComplexType-4">
  <xsd:choice>
    <xsd:element name="a" type="xsd:string"/>
    <xsd:element name="b" type="xsd:boolean"/>
  </xsd:choice>
</xsd:complexType>

<xsd:complexType name="MyComplexType-5">
  <xsd:choice minOccurs="0">
    <xsd:element name="a" type="xsd:string"/>
    <xsd:element name="b" type="xsd:boolean"/>
  </xsd:choice>
</xsd:complexType>

<xsd:complexType name="MyComplexType-6">
  <xsd:choice maxOccurs="5">
    <xsd:element name="a" type="xsd:string"/>
    <xsd:element name="b" type="xsd:boolean"/>
  </xsd:choice>
</xsd:complexType>
```

Estas **definiciones de tipo complejo** se corresponden con las asignaciones de tipo ASN.1 siguientes:

```
MyComplexType-4 ::= SEQUENCE {
  choice [UNTAGGED] CHOICE {
    a XSD.String,
    b BOOLEAN } }

MyComplexType-5 ::= SEQUENCE {
  choice [UNTAGGED] CHOICE {
    a XSD.String,
    b BOOLEAN } OPTIONAL }

MyComplexType-6 ::= SEQUENCE {
  choice-list [UNTAGGED] SEQUENCE (SIZE(1..5)) OF [UNTAGGED] CHOICE {
    a XSD.String,
    b BOOLEAN } }
```

C.3.8.3.3 En el ejemplo siguiente, el **tipo contenido** es un **grupo de modelo** cuyo **compositor** es **todo**.

```
<xsd:complexType name="MyComplexType-7">
  <xsd:all>
    <xsd:element name="a" type="xsd:string"/>
    <xsd:element name="b" type="xsd:boolean"/>
```

```

</xsd:all>
</xsd:complexType>

<xsd:complexType name="MyComplexType-8">
  <xsd:all minOccurs="0">
    <xsd:element name="a" type="xsd:string"/>
    <xsd:element name="b" type="xsd:boolean"/>
  </xsd:all>
</xsd:complexType>

```

Estas **definiciones de tipo complejo** se corresponden con las asignaciones de tipo ASN.1 siguientes:

```

MyComplexType-7 ::= [USE-ORDER] SEQUENCE {
  order      SEQUENCE OF ENUMERATED {a,b},
  a          XSD.String,
  b          BOOLEAN }
(CONSTRAINED BY
  { /* Shall conform to ITU-T Rec. X.693 | ISO/IEC 8825-4, clause 35 */ })

MyComplexType-8 ::= [USE-ORDER] SEQUENCE {
  order      SEQUENCE OF ENUMERATED {a,b},
  a          XSD.String OPTIONAL,
  b          BOOLEAN OPTIONAL }
(CONSTRAINED BY
  { /* Shall conform to ITU-T Rec. X.693 | ISO/IEC 8825-4, clause 35 */ })

```

C.3.8.3.4 En el ejemplo siguiente, el **tipo contenido** es un **grupo de modelo** cuyo **compositor** es **secuencia**.

```

<xsd:complexType name="MyComplexType-9">
  <xsd:sequence>
    <xsd:element name="a" type="xsd:string"/>
    <xsd:element name="b" type="xsd:boolean"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="MyComplexType-10">
  <xsd:sequence minOccurs="0">
    <xsd:element name="a" type="xsd:string"/>
    <xsd:element name="b" type="xsd:boolean"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="MyComplexType-11">
  <xsd:sequence maxOccurs="5">
    <xsd:element name="a" type="xsd:string"/>
    <xsd:element name="b" type="xsd:boolean"/>
  </xsd:sequence>
</xsd:complexType>

```

Estas **definiciones de tipo complejas** se corresponden con las asignaciones de tipo ASN.1 siguientes:

```

MyComplexType-9 ::= SEQUENCE {
  a XSD.String,
  b BOOLEAN }

MyComplexType-10 ::= SEQUENCE {
  sequence [UNTAGGED] SEQUENCE {
    a XSD.String,
    b BOOLEAN } OPTIONAL }

MyComplexType-11 ::= SEQUENCE {
  sequence-list [UNTAGGED] SEQUENCE (SIZE(1..5)) OF [UNTAGGED] SEQUENCE {
    a XSD.String,
    b BOOLEAN } }

```

C.3.8.4 El ejemplo siguiente muestra una **definición de tipo compleja** cuyo **tipo contenido** es un **modelo de contenido mixto**.

```

<xsd:complexType name="MyComplexType-12" mixed="true">
  <xsd:sequence>
    <xsd:element name="a" type="xsd:string"/>
    <xsd:element name="b" type="xsd:boolean"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="MyComplexType-13" mixed="true">
  <xsd:all>
    <xsd:element name="a" type="xsd:string"/>
    <xsd:element name="b" type="xsd:boolean"/>
  </xsd:all>
</xsd:complexType>

```

```

</xsd:all>
</xsd:complexType>

<xsd:complexType name="MyComplexType-14" mixed="true">
  <xsd:choice>
    <xsd:element name="a" type="xsd:string"/>
    <xsd:element name="b" type="xsd:boolean"/>
  </xsd:choice>
</xsd:complexType>

<xsd:complexType name="MyComplexType-15" mixed="true">
  <xsd:all minOccurs="0">
    <xsd:element name="a" type="xsd:string"/>
    <xsd:element name="b" type="xsd:boolean"/>
  </xsd:all>
</xsd:complexType>

<xsd:complexType name="MyComplexType-16">
  <xsd:sequence maxOccurs="unbounded" minOccurs="0">
    <xsd:element name="a" type="xsd:string"/>
    <xsd:element name="b" type="xsd:boolean"/>
  </xsd:sequence>
</xsd:complexType>

```

Estas **definiciones de tipo complejas** se corresponden con las asignaciones de tipo ASN.1 siguientes:

```

MyComplexType-12 ::= [EMBED-VALUES] SEQUENCE {
  embed-values      SEQUENCE OF XSD.String,
  a                 XSD.String,
  b                 BOOLEAN }
(CONSTRAINED BY
  { /* Shall conform to ITU-T Rec. X.693 | ISO/IEC 8825-4, clause 25 */ })

MyComplexType-13 ::= [EMBED-VALUES] [USE-ORDER] SEQUENCE {
  embed-values      SEQUENCE OF XSD.String,
  order             SEQUENCE OF ENUMERATED {a,b},
  a                 XSD.String,
  b                 BOOLEAN }
(CONSTRAINED BY
  { /* Shall conform to ITU-T Rec. X.693 | ISO/IEC 8825-4, clause 25 */ })
(CONSTRAINED BY
  { /* Shall conform to ITU-T Rec. X.693 | ISO/IEC 8825-4, clause 35 */ })

MyComplexType-14 ::= [EMBED-VALUES] SEQUENCE {
  embed-values      SEQUENCE OF XSD.String,
  choice            [UNTAGGED] CHOICE {
    a                 XSD.String,
    b                 BOOLEAN } }
(CONSTRAINED BY
  { /* Shall conform to ITU-T Rec. X.693 | ISO/IEC 8825-4, clause 25 */ })

MyComplexType-15 ::= [EMBED-VALUES] [USE-ORDER] SEQUENCE {
  embed-values      SEQUENCE OF XSD.String,
  order             SEQUENCE OF ENUMERATED {a,b},
  a                 XSD.String OPTIONAL,
  b                 BOOLEAN OPTIONAL }
(CONSTRAINED BY
  { /* Shall conform to ITU-T Rec. X.693 | ISO/IEC 8825-4, clause 35 */ })
(CONSTRAINED BY
  { /* Shall conform to ITU-T Rec. X.693 | ISO/IEC 8825-4, clause 25 */ })

MyComplexType-16 ::= [EMBED-VALUES] SEQUENCE {
  embed-values      SEQUENCE OF XSD.String,
  sequence-list     [UNTAGGED] SEQUENCE OF [UNTAGGED] SEQUENCE {
    a                 XSD.String,
    b                 BOOLEAN } }
(CONSTRAINED BY
  { /* Shall conform to ITU-T Rec. X.693 | ISO/IEC 8825-4, clause 25 */ })

```

C.3.8.5 El ejemplo siguiente muestra **usos de atributo** para una **definición de tipo compleja** construidos utilizando una definición de grupo de atributo.

```

<xs:attributeGroup name="AG1">
  <xs:attribute name="a1" type="xs:string"/>
  <xs:attribute name="a2" type="xs:string"/>
  <xs:attribute name="a3" type="xs:decimal"/>
</xs:attributeGroup>

```

```

<xs:attributeGroup name="AG2">
  <xs:attribute name="a1" use="prohibited"/>
  <xs:attribute name="a3" type="xs:integer"/>
</xs:attributeGroup>

<xs:complexType name="MyComplexType-17">
  <xs:attribute name="a4" type="xs:boolean"/>
  <xs:attribute name="a5" type="xs:boolean"/>
  <xs:attributeGroup ref="AG1"/>
</xs:complexType>

<xs:complexType name="MyComplexType-18">
  <xs:complexContent>
    <xs:restriction base="MyComplexType-17">
      <xs:attributeGroup ref="AG2"/>
      <xs:attribute name="a4" use="prohibited"/>
    </xs:restriction>
  </xs:complexContent>
</xs:complexType>

```

Estas **definiciones de tipo complejas** se corresponden con las asignaciones de tipo ASN.1 siguientes:

```

MyComplexType-17 ::= SEQUENCE {
  a1 [ATTRIBUTE] XSD.String OPTIONAL,
  a2 [ATTRIBUTE] XSD.String OPTIONAL,
  a3 [ATTRIBUTE] XSD.Decimal OPTIONAL,
  a4 [ATTRIBUTE] BOOLEAN OPTIONAL,
  a5 [ATTRIBUTE] BOOLEAN OPTIONAL }

MyComplexType-18 ::= SEQUENCE {
  a2 [ATTRIBUTE] XSD.String OPTIONAL,
  a3 [ATTRIBUTE] INTEGER OPTIONAL,
  a5 [ATTRIBUTE] BOOLEAN OPTIONAL }

MyComplexType-17-derivations ::= [USE-TYPE] CHOICE {
  myComplexType-17 [NAME AS CAPITALIZED] MyComplexType-17,
  myComplexType-18 [NAME AS CAPITALIZED] MyComplexType-18 }

```

C.3.8.6 Derivación de definiciones de tipo complejas.

```

<xsd:complexType name="MyComplexType-19">
  <xsd:sequence minOccurs="0" maxOccurs="unbounded">
    <xsd:element name="a" type="xsd:string"/>
    <xsd:element name="b" type="xsd:boolean"/>
    <xsd:element name="c" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
  <xsd:attribute name="attr1" type="xsd:short" use="required"/>
  <xsd:attribute name="attr2" type="xsd:short"/>
</xsd:complexType>

<xsd:complexType name="MyComplexType-20">
  <xsd:complexContent>
    <xsd:restriction base="MyComplexType-19">
      <xsd:sequence>
        <xsd:element name="a" type="xsd:token"/>
        <xsd:element name="b" type="xsd:boolean"/>
      </xsd:sequence>
      <xsd:attribute name="attr2" type="xsd:short" use="prohibited"/>
    </xsd:restriction>
  </xsd:complexContent>
</xsd:complexType>

<xsd:complexType name="MyComplexType-21">
  <xsd:complexContent>
    <xsd:extension base="MyComplexType-20">
      <xsd:sequence>
        <xsd:element name="d" type="xsd:string"/>
      </xsd:sequence>
      <xsd:attribute name="attr3" type="xsd:boolean"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>

```

Estas **definiciones de tipo complejas** se corresponden con las asignaciones de tipo ASN.1 siguientes:

```

MyComplexType-19 ::= SEQUENCE {
  attr1 [ATTRIBUTE] XSD.Short,
  attr2 [ATTRIBUTE] XSD.Short OPTIONAL,
  sequence-list [UNTAGGED] SEQUENCE OF [UNTAGGED] SEQUENCE {

```

```

        a          XSD.String,
        b          BOOLEAN,
        c          BOOLEAN OPTIONAL } }

MyComplexType-20 ::= SEQUENCE {
    attr1          [ATTRIBUTE] XSD.Short,
    a              XSD.Token,
    b              BOOLEAN }

MyComplexType-21 ::= SEQUENCE {
    attr1          [ATTRIBUTE] XSD.Short,
    attr3          [ATTRIBUTE] BOOLEAN OPTIONAL,
    a              XSD.String,
    b              BOOLEAN,
    d              XSD.String }

MyComplexType-20-derivations ::= [USE-TYPE] CHOICE {
    myComplexType-20 [NAME AS CAPITALIZED] MyComplexType-20,
    myComplexType-21 [NAME AS CAPITALIZED] MyComplexType-21 }

MyComplexType-19-derivations ::= [USE-TYPE] CHOICE {
    myComplexType-19 [NAME AS CAPITALIZED] MyComplexType-19,
    myComplexType-20 [NAME AS CAPITALIZED] MyComplexType-20,
    myComplexType-21 [NAME AS CAPITALIZED] MyComplexType-21 }

```

C.3.9 Correspondencia de comodines

Para estos ejemplos se supone que el espacio nominal de destino es el URI siguiente: "http://www.asn1.org/X694/wildcard".

C.3.9.1 Comodin de atributo

```

<xsd:complexType name="AnyAttribute-1">
  <xsd:anyAttribute namespace="##any"/>
</xsd:complexType>

<xsd:complexType name="AnyAttribute-2">
  <xsd:anyAttribute namespace="##other"/>
</xsd:complexType>

<xsd:complexType name="AnyAttribute-3">
  <xsd:anyAttribute namespace="##targetNamespace"/>
</xsd:complexType>

<xsd:complexType name="AnyAttribute-4">
  <xsd:anyAttribute namespace="##local http://www.asn1.org/X694/attribute"/>
</xsd:complexType>

<xsd:complexType name="AnyAttribute-5">
  <xsd:complexContent>
    <xsd:extension base="AnyAttribute-4">
      <xsd:anyAttribute namespace="##targetNamespace"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>

```

Estas definiciones de tipo complejas se corresponden con las asignaciones de tipo ASN.1 siguientes:

```

AnyAttribute-1 ::= SEQUENCE {
    attr [ANY-ATTRIBUTES] SEQUENCE (CONSTRAINED BY {
        /* Each item shall conform to the "AnyAttributeFormat" specified in
           ITU-T Rec. X.693 | ISO/IEC 8825-4, clause 18 */})
    OF XSD.String }

AnyAttribute-2 ::= SEQUENCE {
    attr [ANY-ATTRIBUTES EXCEPT "http://www.asn1.org/X694/wildcard"]
    SEQUENCE (CONSTRAINED BY {
        /* Each item shall conform to the "AnyAttributeFormat" specified in
           ITU-T Rec. X.693 | ISO/IEC 8825-4, clause 18 */})
    OF XSD.String }

AnyAttribute-3 ::= SEQUENCE {
    attr [ANY-ATTRIBUTES FROM "http://www.asn1.org/X694/wildcard"]
    SEQUENCE (CONSTRAINED BY {
        /* Each item shall conform to the "AnyAttributeFormat" specified in
           ITU-T Rec. X.693 | ISO/IEC 8825-4, clause 18 */})
    OF XSD.String }

```



```

AnyAttribute-4 ::= SEQUENCE {
    attr [ANY-ATTRIBUTES FROM ABSENT
        "http://www.asn1.org/X694/attribute"]
    SEQUENCE (CONSTRAINED BY {
        /* Each item shall conform to the "AnyAttributeFormat" specified in
        ITU-T Rec. X.693 | ISO/IEC 8825-4, clause 18 */})
    OF XSD.String }

AnyAttribute-5 ::= SEQUENCE {
    attr [ANY-ATTRIBUTES FROM ABSENT
        "http://www.asn1.org/X694/attribute"
        "http://www.asn1.org/X694/wildcard"]
    SEQUENCE (CONSTRAINED BY {
        /* Each item shall conform to the "AnyAttributeFormat" specified in
        ITU-T Rec. X.693 | ISO/IEC 8825-4, clause 18 */})
    OF XSD.String }

```

C.3.9.2 El siguiente es un ejemplo de un comodín de modelo de contenido.

```

<xsd:complexType name="Any-1">
    <xsd:sequence>
        <xsd:any namespace="##any"/>
    </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="Any-2">
    <xsd:sequence>
        <xsd:any minOccurs="0" namespace="##other"/>
    </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="Any-3">
    <xsd:sequence>
        <xsd:any minOccurs="0" masOccurs="unbounded" namespace="##local"/>
    </xsd:sequence>
</xsd:complexType>

```

Estas definiciones de tipo complejas se corresponden con las asignaciones de tipo ASN.1 siguientes:

```

Any-1 ::= SEQUENCE {
    elem [ANY-ELEMENT] XSD.String (CONSTRAINED BY {
        /* Shall conform to the "AnyElementFormat" specified in
        ITU-T Rec. X.693 | ISO/IEC 8825-4, clause 18 */}) }

Any-2 ::= SEQUENCE {
    elem [ANY-ELEMENT EXCEPT ABSENT
        "http://www.asn1.org/X694/wildcard"]
    XSD.String (CONSTRAINED BY {
        /* Shall conform to the "AnyElementFormat" specified in
        ITU-T Rec. X.693 | ISO/IEC 8825-4, clause 18 */})
    OPTIONAL }

Any-3 ::= SEQUENCE {
    elem-list SEQUENCE OF elem
        [ANY-ELEMENT FROM ABSENT] XSD.String (CONSTRAINED BY {
            /* Shall conform to the "AnyElementFormat" specified in
            ITU-T Rec. X.693 | ISO/IEC 8825-4, clause 18 */}) }

```

NOTA – Para más ejemplos del cómputo de "NamespaceRestriction" véanse los ejemplos de comodines de atributo en C.3.9.1.

Anexo D

Uso de la correspondencia para proporcionar codificaciones binarias para el esquema XML W3C

(Este anexo no es parte integrante de la presente Recomendación | Norma Internacional)

Este anexo describe la utilización de las correspondencias especificadas en la presente Recomendación | Norma Internacional junto con las reglas de codificación ASN.1 normalizadas para proporcionar codificaciones binarias canónicas y compactas para datos definidos por un esquema XSD.

D.1 Codificación de esquemas XSD

D.1.1 Los esquemas XSD se pueden hacer corresponder con definiciones de tipo ASN.1 como se especifica en el texto de la presente Recomendación | Norma Internacional y el tipo de nivel superior puede entonces codificarse utilizando cualquiera de las reglas de codificación ASN.1 especificadas en la Rec. UIT-T X.690 | ISO/CEI 8825-1, en la Rec. UIT-T X.691 | ISO/CEI 8825-2 y en la Rec. UIT-T X.693 | ISO/CEI 8825-4.

D.1.2 Cada una de estas codificaciones tiene un valor de identificador de objeto asociado que se puede utilizar para identificar la codificación en transferencia. La forma en la que esa identificación se comunica con un decodificador está fuera del ámbito de la presente Recomendación | Norma Internacional.

D.1.3 Cuando el esquema XSD no se envía al receptor mediante el método descrito en D.3, la forma en la que el receptor obtiene el esquema se encuentra fuera del ámbito de la presente Recomendación | Norma Internacional.

D.2 Transferencia sin utilizar el esquema XSD para los esquemas

D.2.1 Este método hace la suposición de que el receptor conoce el esquema XSD utilizado por el transmisor.

D.2.2 La figura D.1 muestra cómo utilizar la correspondencia definida en esta Recomendación | Norma Internacional para codificar documentos XML mediante las reglas de codificación ASN.1.

D.2.3 El transmisor y el receptor utilizan el mismo esquema XSD (fijo) para generar un esquema ASN.1 que a su vez se entrega a un compilador ASN.1 que genera el cuadro de codificaciones BER, DER, PER o XER para documentos XML conformes a dicho esquema XSD.

D.3 Transferencia que utiliza el esquema XSD para los esquemas

D.3.1 Puesto que se dispone de un único esquema XSD para esquemas, es posible proceder en dos pasos (véase la figura D.2).

D.3.2 El transmisor y el receptor construyen un módulo ASN.1 y un codificador/decodificador a partir del esquema XSD para los esquemas.

D.3.3 En el primer paso, el transmisor codifica en BER, DER o PER el esquema XSD para el documento y envía el esquema codificado al receptor. El receptor decodifica dicho esquema y, utilizando la correspondencia del esquema XSD con ASN.1 y un compilador ASN.1, genera un módulo ASN.1 y un codificador/decodificador para documentos XML conforme a dicho esquema.

D.3.4 En el segundo paso el transmisor codifica en BER, DER, PER o XER el documento XML y envía el documento codificado al receptor.

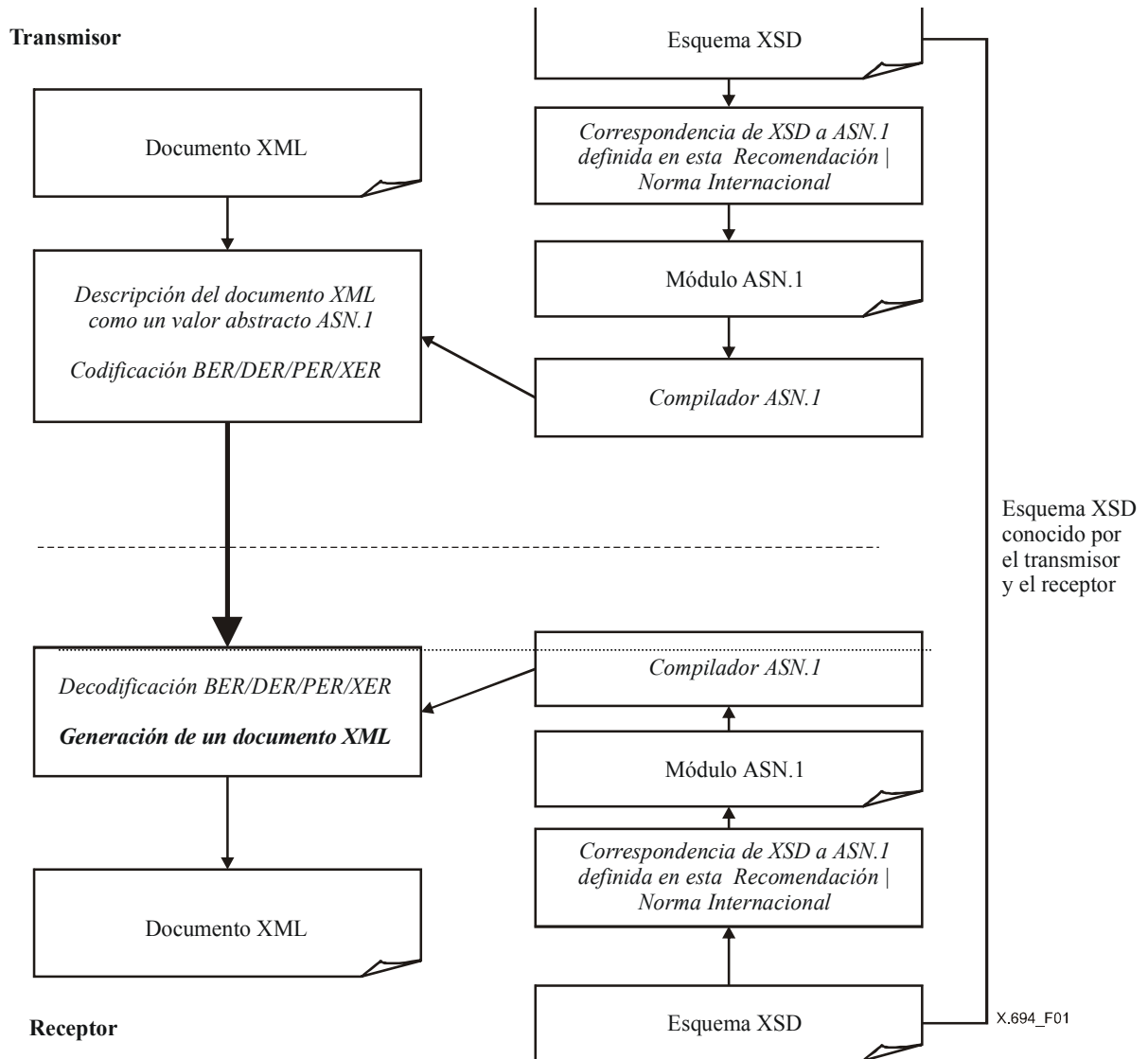


Figura D.1 – Transferencia de un documento XML utilizando la correspondencia de XSD a ASN.1

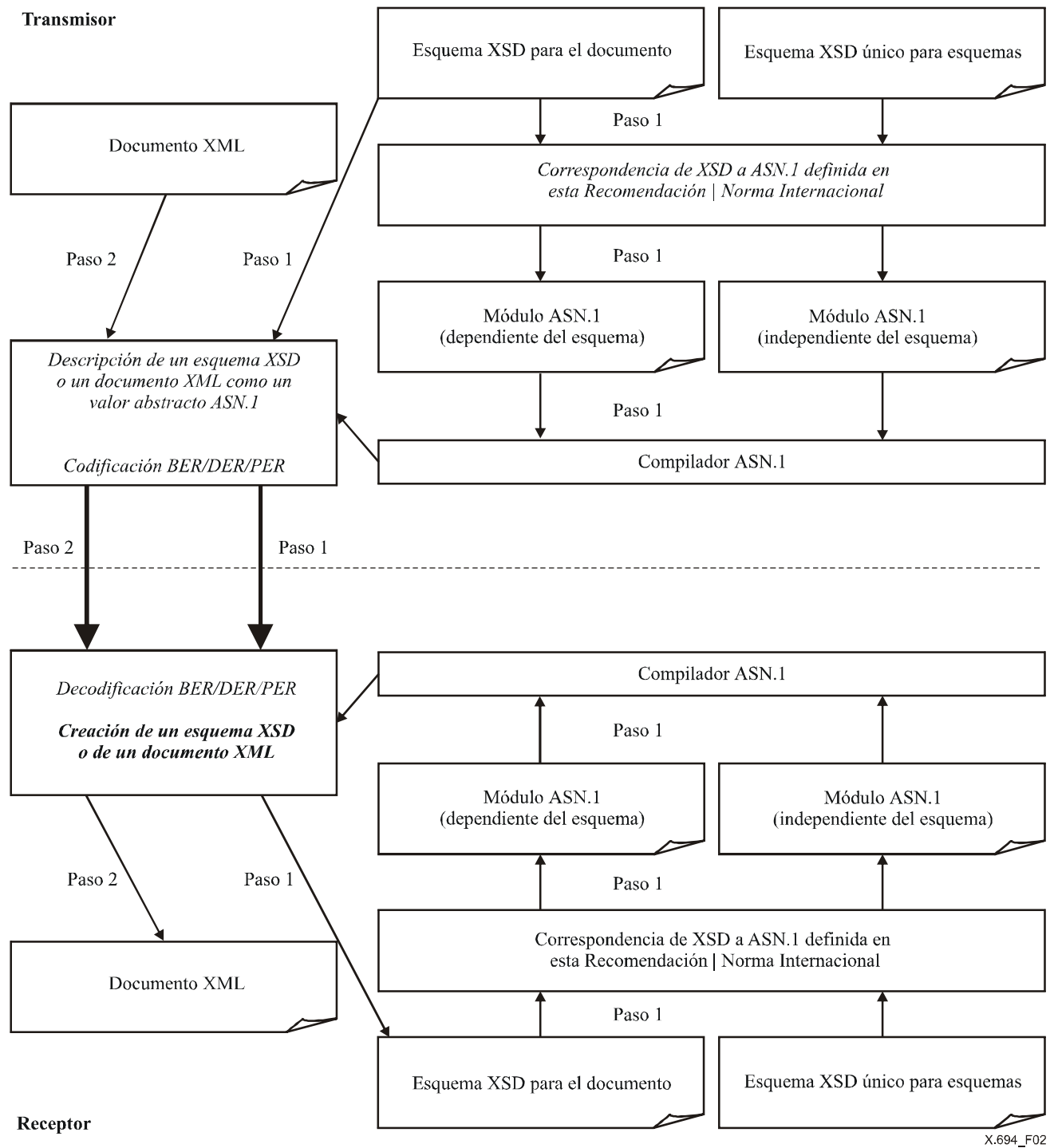


Figura D.2 – Transferencia de un esquema XSD y un documento XML utilizando la correspondencia de XSD a ASN.1

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