

UIT-T

SECTOR DE NORMALIZACIÓN
DE LAS TELECOMUNICACIONES
DE LA UIT

X.520

(08/2005)

SERIE X: REDES DE DATOS, COMUNICACIONES DE
SISTEMAS ABIERTOS Y SEGURIDAD

Directorio

**Tecnología de la información – Interconexión
de sistemas abiertos – El directorio: Tipos de
atributos seleccionados**

Recomendación UIT-T X.520

RECOMENDACIONES UIT-T DE LA SERIE X

REDES DE DATOS, COMUNICACIONES DE SISTEMAS ABIERTOS Y SEGURIDAD

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.379
SISTEMAS DE TRATAMIENTO DE MENSAJES	
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.889
Aplicaciones genéricas de la notación de sintaxis abstracta uno	X.890–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 – Interconexión de sistemas abiertos –
El directorio: Tipos de atributos seleccionados**

Resumen

Esta Recomendación | Norma Internacional especifica varios tipos de atributos y reglas de concordancia que pueden ser útiles en la gama de aplicaciones de directorio. En particular, muchos de los atributos definidos se utilizan en la formación de nombres, sobre todo para las clases de objetos definidos en la Rec. UIT-T X.521 | ISO/CEI 9594-7.

Orígenes

La Recomendación UIT-T X.520 fue aprobada el 29 de agosto de 2005 por la Comisión de Estudio 17 (2005-2008) del UIT-T por el procedimiento de la Recomendación UIT-T A.8. Se publica también un texto idéntico como Norma Internacional ISO/CEI 9594-6.

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 2006

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

SECCIÓN 1 – CUESTIONES GENERALES.....	1
1 Alcance.....	1
2 Referencias normativas	1
2.1 Recomendaciones Normas Internacionales idénticas.....	1
2.2 Otras referencias.....	2
2.3 Normas ISO/CEI.....	2
3 Definiciones	3
4 Convenios.....	3
SECCIÓN 2 – TIPOS DE ATRIBUTOS SELECCIONADOS.....	4
5 Definición de tipos de atributos seleccionados.....	4
5.1 Tipos de atributos de sistema	4
5.2 Tipos de atributos de etiquetado.....	4
5.3 Tipos de atributos geográficos	6
5.4 Tipos de atributos de la organización.....	8
5.5 Tipos de atributos explicativos.....	9
5.6 Tipos de atributos de direccionamiento postal.....	11
5.7 Tipos de atributos de direccionamiento de telecomunicaciones.....	12
5.8 Tipos de atributos de preferencias.....	15
5.9 Tipos de atributos de aplicación de OSI.....	15
5.10 Tipos de atributos relacionales.....	16
5.11 Tipos de atributos de dominio.....	17
5.12 Atributos de notificación.....	17
SECCIÓN 3 – REGLAS DE CONCORDANCIA	23
6 Preparación de cadenas.....	23
6.1 Transcodificación.....	23
6.2 Transcripción	23
6.3 Normalización.....	23
6.4 Prohibición.....	23
6.5 Verificación de restricciones bidireccionales (bidi).....	24
6.6 Supresión de caracteres irrelevantes.....	24
7 Definición de reglas de concordancia.....	24
7.1 Reglas de concordancia de cadenas.....	24
7.2 Reglas de concordancia basada en sintaxis	28
7.3 Reglas de concordancia de tiempo	31
7.4 Reglas de concordancia de primer componente	32
7.5 Reglas de concordancia de palabra	32
7.6 Reglas de concordancia aproximada	35
7.7 Reglas de concordancia especial.....	35
7.8 Concordancia zonal.....	36
SECCIÓN 4 – CONTEXTOS	40
8 Definición de tipos de contexto.....	40
8.1 Contexto de lenguaje.....	40
8.2 Contexto temporal.....	40
8.3 Contexto local	43
8.4 Contexto opción de atributo LDAP.....	44
Anexo A – Tipos de atributos seleccionados en ASN.1	45
Anexo B – Recapitulación de tipos de atributos.....	63
Anexo C – Límites superiores	64

Anexo D – Índice alfabético (en inglés) de los atributos, reglas de concordancia y contextos	65
Anexo E – Ejemplos de reglas de concordancia zonal	67
Anexo F – Enmiendas y corrigenda.....	70

Introducción

Esta Recomendación | Norma Internacional, junto con otras Recomendaciones | Normas Internacionales, ha sido elaborada para facilitar la interconexión de los sistemas de procesamiento de información con el fin de proporcionar servicios de directorio. El conjunto de todos estos sistemas, junto con la información de directorio que contienen, puede considerarse como un todo integrado, llamado el *directorio*. La información contenida por el directorio, denominada colectivamente base de información de directorio (DIB, *directory information base*), se utiliza típicamente para facilitar la comunicación entre, con o sobre objetos tales como entidades de aplicación, personas, terminales y listas de distribución.

El directorio desempeña un papel importante en la interconexión de sistemas abiertos (OSI), cuyo objetivo es permitir, con un mínimo de acuerdos técnicos fuera de las propias normas de interconexión, la interconexión de sistemas de procesamiento de información:

- de diferentes fabricantes;
- sometidos a gestiones diferentes;
- de diferentes grados de complejidad; y
- de diferentes fechas de construcción.

Esta Recomendación | Norma Internacional define un número de tipos de atributos que pueden ser útiles en una gama de aplicaciones de directorio así como también define varias sintaxis de atributo y reglas de concordancia normalizadas. Muchos de los atributos se utilizan en particular para la formación de nombres, especialmente para las clases de objetos definidos en la Rec. UIT-T X.521 | ISO/CEI 9594-7.

Esta Recomendación | Norma Internacional proporciona los marcos básicos sobre los cuales otros organismos de normalización y foros especializados pueden definir los diferentes perfiles que ha de utilizar la industria. Conforme a estos perfiles, es posible que muchas de las características que se definen con carácter facultativo en dichos marcos sean obligatorias en algunos entornos. Esta quinta edición revisa y mejora técnicamente la cuarta edición de esta Recomendación | Norma Internacional, pero no la sustituye. Aún es posible realizar implementaciones conformes a la cuarta edición, pero más o menos corto plazo no habrá soporte para la cuarta edición (los defectos informados ya no serán resueltos). Se recomienda que las implementaciones sean conformes a esta quinta edición lo antes posible.

Esta quinta edición especifica las versiones 1 y 2 de los protocolos de directorio.

Las ediciones primera y segunda especifican también la versión 1. La mayor parte de los servicios y protocolos especificados en esta edición están diseñados para funcionar según la versión 1. No obstante, algunos servicios y protocolos mejorados, por ejemplo, los errores signados, no funcionarán a menos que todas las entidades del directorio que participan en la operación hayan negociado la versión 2. Cualquiera que sea la versión negociada, las diferencias entre los servicios y entre los protocolos definidos en las cinco ediciones, excepto los asignados de manera específica a la versión 2, se acomodan utilizando las reglas de extensibilidad definidas en esta edición de la Rec. UIT-T X.519 | ISO/CEI 9594-5.

El anexo A, que es parte integrante de esta Recomendación | Norma Internacional, proporciona la notación ASN.1 para el módulo completo que define los atributos, sintaxis de atributos y reglas de concordancia.

El anexo B, que no es parte integrante de esta Recomendación | Norma Internacional, proporciona una tabla de tipos de atributos para facilitar la referencia.

El anexo C, que no es parte integrante de esta Recomendación | Norma Internacional, proporciona constricciones de valores de límites superiores sugeridos utilizados en esta Especificaciones de directorio.

El anexo D, que no es parte integrante de esta Recomendación | Norma Internacional, proporciona una lista en orden alfabético inglés de los atributos y reglas de concordancia definidos en la presente Especificación de directorio.

El anexo E, que no es parte integrante de esta Recomendación | Norma Internacional, da ejemplos pertinentes para la definición de concordancia zonal.

El anexo F, que no es parte integrante de esta Recomendación | Norma Internacional, enumera las enmiendas e informes de defectos que se han incorporado en esta edición de la presente Recomendación | Norma Internacional.

**NORMA INTERNACIONAL
RECOMENDACIÓN UIT-T**

**Tecnología de la información – Interconexión de sistemas abiertos –
El directorio: Tipos de atributos seleccionados**

SECCIÓN 1 – CUESTIONES GENERALES

1 Alcance

La presente Recomendación | Norma Internacional define varios tipos de atributos y reglas de concordancia que pueden ser útiles en la gama de aplicaciones de directorio.

Los tipos de atributos y reglas de concordancia se clasifican en tres categorías descritas a continuación.

Algunos tipos de atributos y reglas de concordancia son utilizados por una gran variedad de aplicaciones o son comprendidos y/o utilizados por el propio directorio.

NOTA – Se recomienda utilizar de preferencia un tipo de atributo o regla de concordancia definido en esta Recomendación | Norma Internacional, en vez de crear uno nuevo, siempre que sea apropiado para la aplicación de que se trate.

Algunos tipos de atributos y reglas de concordancia están normalizados internacionalmente pero son específicos de una aplicación y se definen en las normas asociadas a la aplicación correspondiente.

Cualquier autoridad administrativa puede definir sus propios tipos de atributos y reglas de concordancia para cualquier finalidad. Estos no están normalizados internacionalmente y sólo podrán ponerse a disposición de terceros, fuera de la autoridad administrativa que los creó, mediante acuerdo bilateral.

2 Referencias normativas

Las siguientes Recomendaciones y Normas Internacionales 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 y Normas 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 y las Normas 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.

2.1 Recomendaciones | Normas Internacionales idénticas

- Recomendación UIT-T X.200 (1994) | ISO/CEI 7498-1:1994, *Tecnología de la información – Interconexión de sistemas abiertos – Modelo de referencia básico: El modelo básico.*
- Recomendación UIT-T X.500 (2005) | ISO/CEI 9594-1:2005, *Tecnología de la información – Interconexión de sistemas abiertos – El directorio: Visión de conjunto de conceptos, modelos y servicios.*
- Recomendación UIT-T X.501 (2005) | ISO/CEI 9594-2:2005, *Tecnología de la información – Interconexión de sistemas abiertos – El directorio: Modelos.*
- Recomendación UIT-T X.509 (2005) | ISO/CEI 9594-8:2005, *Tecnología de la información – Interconexión de sistemas abiertos – El directorio: Marco para certificados de claves públicas y atributos.*
- Recomendación UIT-T X.511 (2005) | ISO/CEI 9594-3:2005, *Tecnología de la información – Interconexión de sistemas abiertos – El directorio: Definición de servicio abstracto.*
- Recomendación UIT-T X.518 (2005) | ISO/CEI 9594-4:2005, *Tecnología de la información – Interconexión de sistemas abiertos – El directorio: Procedimientos para operación distribuida.*

ISO/CEI 9594-6:2005 (S)

- Recomendación UIT-T X.519 (2005) | ISO/CEI 9594-5:2005, *Tecnología de la información – Interconexión de sistemas abiertos – El directorio: Especificaciones de protocolo.*
- Recomendación UIT-T X.521 (2005) | ISO/CEI 9594-7:2005, *Tecnología de la información – Interconexión de sistemas abiertos – El directorio: Clases de objetos seleccionadas.*
- Recomendación UIT-T X.525 (2005) | ISO/CEI 9594-9:2005, *Tecnología de la información – Interconexión de sistemas abiertos – El directorio: Replicación.*
- Recomendación UIT-T X.530 (2005) | ISO/CEI 9594-10:2005, *Tecnología de la información – Interconexión de sistemas abiertos – El directorio: Utilización de la gestión de sistemas para la administración del directorio.*
- Recomendación UIT-T X.667 (2004) | ISO/CEI 9834-8:2005, *Tecnología de la información – Interconexión de sistemas abiertos – Procedimientos para el funcionamiento de autoridades de registro OSI: Generación y registro de identificadores únicos universales y su utilización como componentes de identificador de objetos ASN.1.*
- 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 objeto 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.*

2.2 Otras referencias

- Recomendación UIT-T E.123 (2001), *Notación de los números telefónicos nacionales e internacionales, direcciones de correo electrónico y direcciones web.*
- Recomendación UIT-T E.164 (2005), *Plan internacional de numeración de telecomunicaciones públicas.*
- Recomendación UIT-T F.1 (1998), *Disposiciones relativas a la explotación del servicio público internacional de telegramas.*
- Recomendación CCITT F.31 (1988), *Sistema de retransmisión de telegramas.*
- Recomendación CCITT F.401 (1992), *Servicios de tratamiento de mensajes: Denominación y direccionamiento para los servicios públicos de tratamiento de mensajes.*
- Recomendación UIT-T T.30 (2005), *Procedimientos de transmisión de documentos por facsímil por la red telefónica general conmutada.*
- Recomendación UIT-T T.62 (1993), *Procedimientos de control para servicios teletex y facsímil del grupo 4.*
- Recomendación UIT-T X.121 (2000), *Plan de numeración internacional para redes públicas de datos.*
- ISO 3166 (todas las partes), *Codes for the representation of names of countries and their subdivisions.*
- ISO 639-2:1998, *Codes for the representation of names of languages – Part 2: Alpha-3 code.*
- ISO/CEI 9945-2:2003, *Information technology – Portable Operating System Interface (POSIX) – Part 2: Shell and Utilities.*
- IETF RFC 3377 (2002), *Lightweight Directory Access Protocol (v3): Technical Specification.*
- IETF RFC 3454 (2002), *Preparation of Internationalized Strings (stringprep).*
- The Unicode Consortium. *The Unicode Standard, Version 4.0*, defined by: *The Unicode Standard, Version 4.0* (Reading, MA, Addison-Wesley, 2003. ISBN 0-321-18578-1).
- *Unicode Standard Annex #15: Unicode Normalization Forms*, by Mark Davis and Martin Dürst. An integral part of *The Unicode Standard, Version 4.0*.

2.3 Normas ISO/CEI

- ISO/CEI 10646:2003, *Information technology – Universal Multiple-Octet Coded Character Set (UCS).*

3 Definiciones

A los efectos de esta Recomendación | Norma Internacional, se aplican las siguientes definiciones.

Los siguientes términos se definen en la Rec. UIT-T X.501 | ISO/CEI 9594-2:

- a) *tipo de atributo*;
- b) *clase de objeto*;
- c) *regla de concordancia*;
- d) *contextos*.

4 Convenios

Con pequeñas excepciones, esta Especificación de directorio se ha preparado con arreglo a las *Normas de presentación de textos comunes UIT-T | ISO/CEI*, noviembre de 2001.

Cuando se escribe "Especificación de directorio" (como en "esta Especificación de directorio"), se trata de la Rec. UIT-T X.520 | ISO/CEI 9594-6. Cuando se escribe "Especificaciones de directorio", se entenderá que designa a todas las Recomendaciones de la serie X.500 y todas las partes de ISO/CEI 9594.

En esta Especificación de directorio se utiliza la expresión *sistemas de la primera edición* para hacer referencia a los sistemas conformes a la primera edición de las Especificaciones de directorio, es decir, la edición de 1988 de las Recomendaciones CCITT de la serie X.500 y la edición de ISO/CEI 9594:1990. En esta Especificación de directorio se utiliza la expresión *sistemas de la segunda edición* para hacer referencia a los sistemas conformes a la segunda edición de las Especificaciones de directorio, es decir, la edición de 1993 de las Recomendaciones UIT-T de la serie X.500 y la edición de ISO/CEI 9594:1995. En esta Especificación de directorio se utiliza la expresión *sistemas de la tercera edición* para hacer referencia a los sistemas conformes a la tercera edición de las especificaciones de directorio, es decir, la edición de 1997 de las Recomendaciones UIT-T de la serie X.500 y la edición de ISO/CEI 9594:1998. En esta Especificación de directorio se utiliza el término *sistemas de la cuarta edición* para hacer referencia a los sistemas conformes a la cuarta edición de las Especificaciones de directorio, es decir, las ediciones de 2001 de las Recs. UIT-T X.500, X.501, X.511, X.518, X.519, X.520, X.521, X.525 y X.530, la edición de 2000 de la Rec. UIT-T X.509 y las partes 1-10 de la edición de 2001 de ISO/CEI 9594.

En esta Especificación de directorio se utiliza la expresión *sistemas de la quinta edición* para referirse a sistemas conformes a la quinta edición de las Especificaciones de directorio, es decir a las ediciones 2005 de las Recs. UIT-T X.500, X.501, X.509, X.511, X.518, X.519, X.520, X.521, X.525, y X.530 y a las partes 1-10 de la edición 2005 de la norma ISO/CEI 9594:2005.

Esta Especificación de directorio presenta la notación ASN.1 con caracteres del tipo Helvética en negritas. Cuando los tipos y valores ASN.1 aparecen en texto normal, se diferencian del texto normal presentándolos en el tipo Helvética en negritas. Los nombres de los procedimientos, a los que se hace referencia cuando se especifica la semántica del procesamiento, se diferencian del texto normal presentándolos en el tipo Times en negritas. Los permisos de control de acceso se presentan en el tipo Times en cursivas.

Cuando los puntos de una lista estén numerados, no marcados con guión "-" o letras, serán considerados etapas de un procedimiento.

Los tipos de atributos, reglas de concordancia y tipos de contextos se definen en esta Recomendación | Norma Internacional mediante la utilización de las clases de objeto de información **ATRIBUTO**, **REGLA DE CONCORDANCIA** y **CONTEXTO** definidos en la Rec. UIT-T X.501 | ISO/CEI 9594-2.

Se describen ejemplos de la utilización de los tipos de atributos utilizando una notación informal, en la que los pares de tipo y valor de atributo se representan mediante un acrónimo para el tipo de atributo, seguido por un signo igual ("="), seguido del valor de atributo que se da como ejemplo.

SECCIÓN 2 – TIPOS DE ATRIBUTOS SELECCIONADOS

5 Definición de tipos de atributos seleccionados

Esta Especificación de directorio define varios tipos de atributo que pueden resultar útiles para una gama de aplicaciones de directorio.

Muchos de los atributos definidos en esta Especificación se basan en una sintaxis común ASN.1:

```
DirectoryString { INTEGER : maxSize } ::= CHOICE {
    teletexString      TeletexString (SIZE (1..maxSize)),
    printableString    PrintableString (SIZE (1..maxSize)),
    bmpString          BMPString (SIZE (1..maxSize)),
    universalString    UniversalString (SIZE (1..maxSize)),
    utf8String         UTF8String (SIZE (1..maxSize)) }
```

Es posible que algunas implementaciones de directorio no soporten **UniversalString**, **BMPString** o **UTF8String** y que no generen, concuerden, sombreen o visualicen atributos que tengan esos tipos de sintaxis.

5.1 Tipos de atributos de sistema**5.1.1 Datos de filiación**

El tipo de atributo *datos de filiación* especifica una descripción acumulativa, legible por el ser humano, del conocimiento que posee un determinado agente de sistema de directorio (DSA, *directory system agent*).

NOTA – Es un atributo caducado.

```
knowledgeInformation ATTRIBUTE ::= {
    WITH SYNTAX          DirectoryString {ub-knowledge-information}
    EQUALITY MATCHING RULE caseIgnoreMatch
    ID                   id-at-knowledgeInformation }
```

5.2 Tipos de atributos de etiquetado

Estos atributos reflejan información que ha sido explícitamente asociada con los objetos mediante un proceso de etiquetado.

5.2.1 Nombre

El tipo de atributo *nombre* (*Name*) es el supertipo de atributo a partir del cual pueden formarse tipos de atributo cadena que se utilizan usualmente para denominación.

```
name ATTRIBUTE ::= {
    WITH SYNTAX          DirectoryString {ub-name}
    EQUALITY MATCHING RULE caseIgnoreMatch
    SUBSTRINGS MATCHING RULE caseIgnoreSubstringsMatch
    ID                   id-at-name }
```

5.2.2 Nombre común

El tipo de atributo *nombre común* (*Common Name*) especifica un identificador de un objeto. Un nombre común no es un nombre del directorio; es un nombre (que podría ser ambiguo) por el cual el objeto es usualmente conocido en cierto ámbito limitado (por ejemplo en una organización) y que es conforme a los convenios de denominación del país o de la cultura a que está asociado.

Un valor de atributo para nombre común es una cadena escogida por la persona u organización que ella describe, o por la organización responsable del objeto que ella describe, para dispositivos y entidades de aplicación. Por ejemplo, un nombre típico de una persona en un país de habla inglesa se compone de un título personal (por ejemplo Mr., Ms, Rd., Professor, Sir, Lord), un primer nombre, uno o más nombres intermedios, un apellido, un calificativo generacional (eventualmente, por ejemplo, Jr.) y distinciones honoríficas y condecoraciones (eventualmente, por ejemplo, QC).

Ejemplos

Nombre común = "Mr. Robin Lachlan McLeod BSc(Hons) CEng MIEE".

Nombre común = "Divisional Coordination Committee".

Nombre común = "High Speed Modem".

Cualquier variante deberá estar asociada con el objeto denominado, en forma de valores de atributo separados y alternativos.

Deberán también admitirse otras variantes usuales, tales como la utilización de un nombre intermedio como primer nombre preferido, o la utilización de "Bill" en lugar de "William", etc.

```
commonName ATTRIBUTE ::= {
    SUBTYPE OF      name
    WITH SYNTAX     DirectoryString {ub-common-name}
    ID              id-at-commonName }
```

5.2.3 Apellido

El tipo de atributo *apellido* (*Surname*) especifica la construcción lingüística que generalmente se hereda de los padres o se asume por matrimonio, y con la que suele conocerse la persona.

Un valor de atributo para apellido es una cadena, por ejemplo, "McLeod".

```
surname ATTRIBUTE ::= {
    SUBTYPE OF      name
    WITH SYNTAX     DirectoryString {ub-surname}
    ID              id-at-surname }
```

5.2.4 Nombre de pila

El tipo de atributo *nombre de pila* (*Given Name*) especifica la construcción lingüística dada normalmente a un individuo por sus padres o elegida por el individuo o por la cual el individuo es conocido.

Un valor de atributo para nombre dado es una cadena, por ejemplo, "David" o "Juan Pablo".

```
givenName ATTRIBUTE ::= {
    SUBTYPE OF      name
    WITH SYNTAX     DirectoryString {ub-name}
    ID              id-at-givenName }
```

5.2.5 Iniciales

El tipo de atributo *iniciales* (*Initials*) contiene las iniciales de algunos o todos los nombres de un individuo, pero no del apellido o apellidos.

Un valor de atributo para iniciales es una cadena, por ejemplo, "D" o "D." o "J.P.".

```
initials ATTRIBUTE ::= {
    SUBTYPE OF      name
    WITH SYNTAX     DirectoryString {ub-name}
    ID              id-at-initials }
```

5.2.6 Calificador generacional

El tipo de atributo *calificador generacional* (*Generation Qualifier*) contiene una cadena que se utiliza para proporcionar información generacional destinada a calificar el nombre de un individuo.

Un valor de atributo para el calificador generacional es una cadena, por ejemplo, "Jr." o "II".

```
generationQualifier ATTRIBUTE ::= {
    SUBTYPE OF      name
    WITH SYNTAX     DirectoryString {ub-name}
    ID              id-at-generationQualifier }
```

5.2.7 Identificador único

El tipo de atributo *identificador único* (*Unique Identifier*) especifica un identificador que se puede utilizar para distinguir entre referencias de objetos cuando se ha reutilizado un nombre distinguido. Puede ser, por ejemplo, un identificador de objeto codificado, certificado, fecha, sello de tiempo, o alguna otra forma de certificación sobre la validez del nombre distinguido.

ISO/CEI 9594-6:2005 (S)

Un valor de atributo para identificador único es una cadena de bits.

```
uniqueIdentifier ATTRIBUTE ::= {  
    WITH SYNTAX                UniqueIdentifier  
    EQUALITY MATCHING RULE     bitStringMatch  
    ID                          id-at-uniqueIdentifier }
```

UniqueIdentifier ::= BIT STRING

5.2.8 Calificador de nombre distinguido

El tipo de atributo *calificador de nombre distinguido* (*DN Qualifier*) especifica una información inequívoca para añadir al nombre distinguido relativo de una inserción. Está destinado a ser utilizado para inserciones contenidas en múltiples agentes de sistema de directorio (DSA, *directory system agent*), que de no ser así tendrían el mismo nombre, y cuyo valor es igual en un DSA dado para todas las inserciones a las cuales se ha añadido esta información.

```
dnQualifier ATTRIBUTE ::= {  
    WITH SYNTAX                PrintableString  
    EQUALITY MATCHING RULE     caseIgnoreMatch  
    ORDERING MATCHING RULE     caseIgnoreOrderingMatch  
    SUBSTRINGS MATCHING RULE   caseIgnoreSubstringsMatch  
    ID                          id-at-dnQualifier }
```

5.2.9 Número de serie

El tipo de atributo *número de serie* (*Serial Number*) especifica un identificador, el número de serie de un objeto.

Un valor de atributo para número de serie es una cadena imprimible.

```
serialNumber ATTRIBUTE ::= {  
    WITH SYNTAX                PrintableString (SIZE (1..ub-serial-number))  
    EQUALITY MATCHING RULE     caseIgnoreMatch  
    SUBSTRINGS MATCHING RULE   caseIgnoreSubstringsMatch  
    ID                          id-at-serialNumber }
```

5.2.10 Seudónimo

El tipo de atributo *seudónimo* (*Pseudonym*) especifica un seudónimo de un objeto. Se utiliza para denominar un objeto cuando hay que aclarar que un nombre es un seudónimo.

```
pseudonym ATTRIBUTE ::= {  
    SUBTYPE OF                 name  
    WITH SYNTAX                DirectoryString {ub-pseudonym}  
    ID                          id-at-pseudonym }
```

5.2.11 Par de identificadores únicos y universales

Tipo de atributo que especifica un *par de identificadores únicos y universales* (UUID, *universal unique identifier pair*), conforme a la Rec. UIT-T X.667 | ISO/CEI 9834-8. El par representa una relación emisor/sujeto, cuya naturaleza está fuera del alcance de esta Especificación de directorio. El primer UUID representa el emisor, y el segundo representa el sujeto de esa la relación emisor/sujeto. Una cuenta de usuario es un ejemplo de este tipo de relación.

```
uUIDPair ATTRIBUTE ::= {  
    WITH SYNTAX                UUIDPair  
    EQUALITY MATCHING RULE     uUIDPairMatch  
    ID                          id-at-uuidpair }
```

```
UUIDPair ::= SEQUENCE {  
    issuerUUID    UUID,  
    subjectUUID   UUID }
```

```
UUID ::= OCTET STRING (SIZE(16))    -- UUID format only
```

5.3 Tipos de atributos geográficos

Estos tipos de atributos se refieren a posiciones geográficas o regiones con las que están asociados los objetos.

5.3.1 Nombre de país

El atributo *nombre de país* (*Country Name*) especifica un país. Cuando se usa como componente de un nombre de directorio expresa el país en que el objeto denominado está ubicado físicamente o con el que está asociado de alguna otra manera importante.

Un valor de atributo para nombre de país es una cadena tomada de ISO 3166.

```
countryName ATTRIBUTE ::= {
  SUBTYPE OF      name
  WITH SYNTAX     CountryName
  SINGLE VALUE    TRUE
  ID              id-at-countryName }
```

CountryName ::= PrintableString (SIZE(2)) -- ISO 3166 codes only

5.3.2 Nombre de localidad

El tipo de atributo *nombre de localidad* (*Locality Name*) especifica una localidad. Cuando se usa como componente de un nombre de directorio expresa una zona geográfica o localidad en la que el objeto denominado está ubicado físicamente, o con la cual se encuentra asociado de alguna otra manera importante.

Un valor de atributo para el nombre de localidad es una cadena: por ejemplo, L = "Edinburgh".

```
localityName ATTRIBUTE ::= {
  SUBTYPE OF      name
  WITH SYNTAX     DirectoryString {ub-locality-name}
  ID              id-at-localityName }
```

El tipo de atributo *nombre de localidad colectivo* (*Collective Locality Name*) especifica un nombre de localidad para una colección de inserciones.

```
collectiveLocalityName ATTRIBUTE ::= {
  SUBTYPE OF      localityName
  COLLECTIVE      TRUE
  ID              id-at-collectiveLocalityName }
```

5.3.3 Nombre de estado o provincia

El tipo de atributo *nombre de estado o provincia* (*State or Province Name*) especifica un estado o una provincia. Cuando se usa como componente de un nombre de directorio, determina una subdivisión geográfica en la que el objeto denominado está ubicado físicamente, o con la que se encuentra asociado de alguna otra manera importante.

Un valor de atributo para nombre de estado o provincia es una cadena, por ejemplo, S = "Ohio".

```
stateOrProvinceName ATTRIBUTE ::= {
  SUBTYPE OF      name
  WITH SYNTAX     DirectoryString {ub-state-name}
  ID              id-at-stateOrProvinceName }
```

El tipo de atributo *nombre colectivo de estado o provincia* (*Collective State or Province Name*) especifica un nombre de estado o provincia para una colección de inserciones.

```
collectiveStateOrProvinceName ATTRIBUTE ::= {
  SUBTYPE OF      stateOrProvinceName
  COLLECTIVE      TRUE
  ID              id-at-collectiveStateOrProvinceName }
```

5.3.4 Dirección

El tipo de atributo *dirección* (*Street Address*) especifica un lugar para la distribución y la entrega física a una dirección postal, o sea, el nombre de la calle, lugar, avenida y el número del edificio. Cuando se usa como componente de un nombre de directorio, identifica la dirección en la que el objeto denominado está ubicado, o con la que se encuentra asociado de alguna otra manera importante.

Un valor de atributo para dirección es una cadena, por ejemplo: "Arnulfstraße 60".

```
streetAddress ATTRIBUTE ::= {
  WITH SYNTAX     DirectoryString {ub-street-address}
  EQUALITY MATCHING RULE    caseIgnoreMatch
  SUBSTRINGS MATCHING RULE  caseIgnoreSubstringsMatch
  ID              id-at-streetAddress }
```

ISO/CEI 9594-6:2005 (S)

El tipo de atributo *dirección colectiva* (*Collective Street Address*) especifica la dirección para una serie de inserciones.

```
collectiveStreetAddress ATTRIBUTE ::= {  
    SUBTYPE OF      streetAddress  
    COLLECTIVE      TRUE  
    ID              id-at-collectiveStreetAddress }
```

5.3.5 Identificador de casa

El tipo de atributo *identificador de casa* (*House Identifier*) especifica una construcción lingüística utilizada para identificar un determinado edificio, por ejemplo, un número de casa o un nombre de casa con respecto a una calle, avenida, pueblo o ciudad, etc.

Un valor de atributo para identificador de casa es una cadena, por ejemplo, "14".

```
houseIdentifier ATTRIBUTE ::= {  
    WITH SYNTAX      DirectoryString {ub-name}  
    EQUALITY MATCHING RULE caseIgnoreMatch  
    SUBSTRINGS MATCHING RULE caseIgnoreSubstringsMatch  
    ID              id-at-houseIdentifier }
```

5.4 Tipos de atributos de la organización

Estos tipos de atributos conciernen a organizaciones y pueden utilizarse para describir objetos en términos de organizaciones con las que están asociados.

5.4.1 Nombre de organización

El tipo de atributo *nombre de organización* (*OrganizationName*) especifica una organización. Cuando se usa como componente de nombre de directorio, expresa una organización a la que pertenece el objeto denominado.

Un valor de atributo para nombre de organización **OrganizationName** es una cadena escogida por la organización (por ejemplo O = "Scottish Telecommunications plc"). Las variantes eventuales deberán estar asociadas con la organización denominada como valores de atributo separados y alternativos.

```
organizationName ATTRIBUTE ::= {  
    SUBTYPE OF      name  
    WITH SYNTAX      DirectoryString {ub-organization-name}  
    ID              id-at-organizationName }
```

El tipo de atributo *nombre de organización colectiva* (*Collective Organization Name*) especifica un nombre de organización para una colección de inserciones.

```
collectiveOrganizationName ATTRIBUTE ::= {  
    SUBTYPE OF      organizationName  
    COLLECTIVE      TRUE  
    ID              id-at-collectiveOrganizationName }
```

5.4.2 Nombre de unidad en la organización

El tipo de atributo *nombre de unidad en la organización* (*Organizational Unit Name*) especifica un determinado departamento. Cuando se utiliza como componente de nombre de directorio, identifica una unidad a la que pertenece el objeto denominado.

Se entiende que la unidad designada forma parte de una organización designada por un atributo **organizationName**. De ello se desprende que, si un atributo nombre de unidad en la organización se utiliza en un nombre de directorio, deberá estar asociado con un atributo **organizationName**.

Un valor de atributo para nombre de unidad en la organización es una cadena escogida por la organización de la que forma parte (por ejemplo, OU = "Technology Division"). Obsérvese que la abreviatura de uso corriente "TD" sería un valor de atributo separado y alternativo.

Ejemplo

O = "Scottel", OU = "TD"

```
organizationalUnitName ATTRIBUTE ::= {  
    SUBTYPE OF      name  
    WITH SYNTAX      DirectoryString {ub-organizational-unit-name}  
    ID              id-at-organizationalUnitName }
```

El tipo de atributo *nombre de unidad en la organización colectivo* (*Collective Organizational Unit Name*) especifica un nombre de unidad para una serie de inserciones.

```
collectiveOrganizationalUnitName ATTRIBUTE ::= {
    SUBTYPE OF      organizationalUnitName
    COLLECTIVE      TRUE
    ID              id-at-collectiveOrganizationalUnitName }
```

5.4.3 Título

El tipo de atributo *título* (*Title*) especifica el cargo o función designado del objeto dentro de una organización.

Un valor de atributo para título es una cadena.

Ejemplo

T = "Manager, Distributed Applications"

```
title ATTRIBUTE ::= {
    SUBTYPE OF      name
    WITH SYNTAX     DirectoryString {ub-title}
    ID              id-at-title }
```

5.5 Tipos de atributos explicativos

Estos tipos de atributos se refieren a explicaciones (por ejemplo, en un lenguaje natural) sobre algo relativo a un objeto.

5.5.1 Descripción

El tipo de atributo *descripción* (*Description*) especifica texto que describe el objeto asociado.

Por ejemplo, el objeto "Interés normativo" podría tener asociada la descripción "lista de distribución para el intercambio de informaciones acerca de la elaboración de normas internacionales de la compañía".

Un valor de atributo para descripción es una cadena.

```
description ATTRIBUTE ::= {
    WITH SYNTAX          DirectoryString {ub-description}
    EQUALITY MATCHING RULE caselgnoreMatch
    SUBSTRINGS MATCHING RULE caselgnoreSubstringsMatch
    ID                  id-at-description }
```

5.5.2 Directriz de búsqueda

El tipo de atributo *directriz de búsqueda* (*Search Guide*) especifica la información sobre los criterios de búsqueda propuestos que se podrán incluir en algunas inserciones que se espera constituirán un objeto de base útil para la operación de búsqueda, por ejemplo, un país u organización.

Los criterios de búsqueda consisten en un identificador facultativo para el tipo de objeto buscado, y en combinaciones de tipo de atributo y operadores lógicos que se utilizarán para la construcción de un filtro. Para cada criterio de búsqueda es posible especificar el nivel de concordancia, por ejemplo concordancia aproximada.

El atributo directriz de búsqueda puede repetirse para reflejar diversos tipos de petición, por ejemplo búsqueda de una persona residencial o una persona organizacional, que se pueden efectuar a partir del objeto de base dado donde se lee la directriz de búsqueda.

```
searchGuide ATTRIBUTE ::= {
    WITH SYNTAX      Guide
    ID              id-at-searchGuide }

Guide ::= SET {
    objectClass [0] OBJECT-CLASS.&id OPTIONAL,
    criteria    [1] Criteria }

Criteria ::= CHOICE {
    type      [0] Criterialtem,
    and       [1] SET OF Criteria,
    or        [2] SET OF Criteria,
    not       [3] Criteria }
```

```

Criterialtem ::= CHOICE {
    equality           [0] AttributeType,
    substrings        [1] AttributeType,
    greaterOrEqual    [2] AttributeType,
    lessOrEqual       [3] AttributeType,
    approximateMatch  [4] AttributeType }

```

Ejemplo

A continuación se presenta un valor posible del atributo directriz de búsqueda que podría almacenarse en inserciones de la clase de objeto localidad para indicar cómo podrían encontrarse las inserciones de la clase de objeto persona particular:

```

residential-person-guide Guide ::= {
    objectClass residentialPerson.&id,
    criteria and : {
        type : substrings : commonName.&id,
        type : substrings : streetAddress.&id } }

```

La construcción de un filtro a partir de este valor de la directriz es directa.

El paso (1) produce el valor Filtro (*Filter*) intermedio:

```

intermediate-filter Filter ::=
    and : {
        item : substrings {
            type commonName.&id,
            strings { any : teletexString : "Dubois" },
            item : substrings {
                type streetAddress.&id,
                strings { any : teletexString "Hugo" } } } }

```

El paso (2) produce un filtro para concordar las inserciones persona particular en el subárbol:

```

residential-person-filter Filter ::=
    and : {
        item : equality : {
            type objectClass.&id,
            assertion residentialPerson.&id },
        intermediateFilter }

```

5.5.3 Directriz de búsqueda mejorada

El atributo *directriz de búsqueda mejorada* (*Enhanced Search Guide*) proporciona una mejora del atributo **searchGuide**, añadiendo información sobre la profundidad de búsqueda recomendada para las búsquedas entre objetos subordinados de una clase de objeto dada.

```

enhancedSearchGuide ATTRIBUTE ::= {
    WITH SYNTAX EnhancedGuide
    ID id-at-enhancedSearchGuide }

```

```

EnhancedGuide ::= SEQUENCE {
    objectClass [0] OBJECT-CLASS.&id,
    criteria [1] Criteria,
    subset [2] INTEGER
    { baseObject (0), oneLevel (1), wholeSubtree (2) } DEFAULT oneLevel }

```

5.5.4 Categoría profesional

El atributo *categoría profesional* (*Business Category*) especifica información concerniente a la ocupación de algunos objetos comunes tales como personas. Por ejemplo, este atributo permite interrogar al directorio acerca de las personas que ejercen la misma profesión.

```

businessCategory ATTRIBUTE ::= {
    WITH SYNTAX DirectoryString {ub-business-category}
    EQUALITY MATCHING RULE caseIgnoreMatch
    SUBSTRINGS MATCHING RULE caseIgnoreSubstringsMatch
    ID id-at-businessCategory }

```

5.6 Tipos de atributos de direccionamiento postal

Estos tipos de atributo se refieren a la información requerida para la entrega postal física a un objeto.

5.6.1 Dirección postal

El tipo de atributo *dirección postal* (*Postal Address*) especifica la información de dirección necesaria para que la autoridad postal efectúe la entrega física de mensajes postales al objeto denominado.

Un valor de atributo para la dirección postal estará normalmente compuesto por atributos seleccionados, tomados de la versión 1 de la dirección O/R postal no formatada MHS de conformidad con la Rec. CCITT F.401 y se limitará a 6 líneas de 30 caracteres de cada una, incluido un nombre postal de país. En general, la información contenida en este tipo de dirección puede incluir un nombre de destinatario, una dirección de calle, ciudad, estado o provincia, código postal, y quizá un número de apartado postal según las exigencias específicas del objeto denominado.

```
postalAddress ATTRIBUTE ::= {
    WITH SYNTAX                PostalAddress
    EQUALITY MATCHING RULE     caselgnoreListMatch
    SUBSTRINGS MATCHING RULE   caselgnoreListSubstringsMatch
    ID                          id-at-postalAddress }
```

```
PostalAddress ::= SEQUENCE SIZE(1..ub-postal-line) OF DirectoryString {ub-postal-string}
```

El tipo de atributo *dirección postal colectiva* (*Collective Postal Address*) especifica una dirección postal para una colección de inserciones.

```
collectivePostalAddress ATTRIBUTE ::= {
    SUBTYPE OF                postalAddress
    COLLECTIVE                 TRUE
    ID                          id-at-collectivePostalAddress }
```

5.6.2 Código postal

El tipo de atributo *código postal* (*Postal Code*) especifica el código postal del objeto denominado. Cuando este valor de atributo está presente, formará parte de la dirección postal del objeto.

Un valor de atributo para código postal es una cadena.

```
postalCode ATTRIBUTE ::= {
    WITH SYNTAX                DirectoryString {ub-postal-code}
    EQUALITY MATCHING RULE     caselgnoreMatch
    SUBSTRINGS MATCHING RULE   caselgnoreSubstringsMatch
    ID                          id-at-postalCode }
```

El tipo de atributo *código postal colectivo* (*Collective Postal Code*) especifica un código postal para una colección de inserciones.

```
collectivePostalCode ATTRIBUTE ::= {
    SUBTYPE OF                postalCode
    COLLECTIVE                 TRUE
    ID                          id-at-collectivePostalCode }
```

5.6.3 Apartado de correos

El tipo de atributo *apartado de correos* (*Post Office Box*) especifica un apartado de correos a través del cual se efectuará la entrega postal física del objeto. Cuando está presente, el valor de atributo es parte de la dirección postal del objeto.

```
postOfficeBox ATTRIBUTE ::= {
    WITH SYNTAX                DirectoryString {ub-post-office-box}
    EQUALITY MATCHING RULE     caselgnoreMatch
    SUBSTRINGS MATCHING RULE   caselgnoreSubstringsMatch
    ID                          id-at-postOfficeBox }
```

El tipo de atributo *apartado de correos colectivo* (*Collective Post Office Box*) especifica un apartado de correos para una colección de inserciones.

```
collectivePostOfficeBox ATTRIBUTE ::= {
    SUBTYPE OF                postOfficeBox
    COLLECTIVE                 TRUE
    ID                          id-at-collectivePostOfficeBox }
```

5.6.4 Nombre de oficina de entrega física

El tipo de atributo *nombre de oficina de entrega física* (*Physical Delivery Office Name*) especifica el nombre de la ciudad, pueblo, etc., donde está ubicada una oficina de entrega física.

Un valor de atributo para nombre de oficina de entrega física es una cadena.

```
physicalDeliveryOfficeName ATTRIBUTE ::= {
  WITH SYNTAX          DirectoryString {ub-physical-office-name}
  EQUALITY MATCHING RULE  caseIgnoreMatch
  SUBSTRINGS MATCHING RULE caseIgnoreSubstringsMatch
  ID                    id-at-physicalDeliveryOfficeName }
```

El tipo de atributo *nombre de oficina de entrega física colectivo* (*Collective Physical Delivery Office Name*) especifica un nombre de oficina de entrega física para una serie de inserciones.

```
collectivePhysicalDeliveryOfficeName ATTRIBUTE ::= {
  SUBTYPE OF          physicalDeliveryOfficeName
  COLLECTIVE          TRUE
  ID                    id-at-collectivePhysicalDeliveryOfficeName }
```

5.7 Tipos de atributos de direccionamiento de telecomunicaciones

Estos tipos de atributo se refieren a información de direccionamiento necesaria para comunicar con el objeto utilizando medios de telecomunicación.

5.7.1 Número de teléfono

El tipo de atributo *número de teléfono* (*Telephone Number*) especifica un número de teléfono asociado con un objeto.

Un valor de atributo para número de teléfono es una cadena conforme al formato internacional convenido para indicar los números de teléfono internacionales y que figuran en la Rec. UIT-T E.123 (por ejemplo, "+44 582 10101").

```
telephoneNumber ATTRIBUTE ::= {
  WITH SYNTAX          TelephoneNumber
  EQUALITY MATCHING RULE  telephoneNumberMatch
  SUBSTRINGS MATCHING RULE  telephoneNumberSubstringsMatch
  ID                    id-at-telephoneNumber }
```

```
TelephoneNumber ::= PrintableString (SIZE(1..ub-telephone-number))
```

-- String complying with ITU-T Rec. E.123 only

El tipo de atributo *número de teléfono colectivo* (*Collective Telephone Number*) especifica un número de teléfono para una serie de inserciones.

```
collectiveTelephoneNumber ATTRIBUTE ::= {
  SUBTYPE OF          telephoneNumber
  COLLECTIVE          TRUE
  ID                    id-at-collectiveTelephoneNumber }
```

5.7.2 Número télex

El tipo de atributo *número télex* (*Telex Number*) especifica el número télex, el indicativo de país y el distintivo de un terminal télex asociado con un objeto.

```
telexNumber ATTRIBUTE ::= {
  WITH SYNTAX          TelexNumber
  ID                    id-at-telexNumber }
```

```
TelexNumber ::= SEQUENCE {
  telexNumber          PrintableString (SIZE (1..ub-telex-number)),
  countryCode          PrintableString (SIZE (1..ub-country-code)),
  answerback          PrintableString (SIZE (1..ub-answerback)) }
```

El atributo *número télex colectivo* (*Collective Telex Number*) especifica un número de télex para una serie de inserciones.

```
collectiveTelexNumber ATTRIBUTE ::= {
  SUBTYPE OF          telexNumber
  COLLECTIVE          TRUE
  ID                    id-at-collectiveTelexNumber }
```

5.7.3 Identificador de terminal teletex

Sabiendo que la Rec. CCITT F.200 ya no está en vigor y no ha sido reemplazada, se desaprueban la utilización del **teletexTerminalIdentifier** y del tipo de atributo del **collectiveTeletexTerminalIdentifier**.

El tipo de atributo *identificador de terminal teletex* (*Teletex Terminal Identifier*) especifica el identificador de terminal teletex (y, facultativamente sus parámetros) para un terminal teletex asociado con un objeto.

Un valor de atributo para identificador de terminal teletex es una cadena que cumple los requisitos de la Rec. CCITT F.200 y un conjunto facultativo cuyos componentes son conformes a la Rec. UIT-T T.62.

```
-- teletexTerminalIdentifier ATTRIBUTE ::= {
--   WITH SYNTAX   TeletexTerminalIdentifier
--   ID            id-at-teletexTerminalIdentifier }

-- TeletexTerminalIdentifier ::= SEQUENCE {
--   teletexTerminal   PrintableString (SIZE(1..ub-teletex-terminal-id)),
--   parameters       TeletexNonBasicParameters OPTIONAL }
```

El tipo de atributo *identificador de terminal de teletex colectivo* (*Collective Teletex Terminal Identifier*) especifica un identificador de terminal teletex para una serie de inserciones.

```
-- collectiveTeletexTerminalIdentifier ATTRIBUTE ::= {
--   SUBTYPE OF   teletexTerminalIdentifier
--   COLLECTIVE   TRUE
--   ID            id-at-collectiveTeletexTerminalIdentifier }
```

5.7.4 Número de teléfono facsímil

El tipo de atributo *número de teléfono facsímil* (*Facsimile Telephone Number*) especifica un número de teléfono para un terminal facsímil (y facultativamente sus parámetros) asociado con un objeto.

Un valor de atributo para el número de teléfono facsímil es una cadena conforme al formato convenido internacionalmente para la indicación de los números de teléfono internacionales de la Rec. UIT-T E.123 (por ejemplo, "+81 3 347 7418") y una cadena de bits facultativa (formatada conforme a la Rec. UIT-T T.30).

```
facsimileTelephoneNumber ATTRIBUTE ::= {
    WITH SYNTAX           FacsimileTelephoneNumber
    EQUALITY MATCHING RULE facsimileNumberMatch
    SUBSTRINGS MATCHING RULE facsimileNumberSubstringsMatch
    ID                    id-at-facsimileTelephoneNumber }
```

```
FacsimileTelephoneNumber ::= SEQUENCE {
    telephoneNumber   TelephoneNumber,
    parameters        G3FacsimileNonBasicParameters OPTIONAL }
```

El tipo de atributo *número de teléfono facsímil colectivo* (*Collective Facsimile Telephone Number*) especifica un número de teléfono facsímil para una serie de inserciones.

```
collectiveFacsimileTelephoneNumber ATTRIBUTE ::= {
    SUBTYPE OF   facsimileTelephoneNumber
    COLLECTIVE   TRUE
    ID            id-at-collectiveFacsimileTelephoneNumber }
```

5.7.5 Dirección X.121

El tipo de atributo *dirección X.121* (*X.121 Address*) especifica una dirección definida por la Rec. UIT-T X.121 asociada con un objeto.

```
x121Address ATTRIBUTE ::= {
    WITH SYNTAX           X121Address
    EQUALITY MATCHING RULE numericStringMatch
    SUBSTRINGS MATCHING RULE numericStringSubstringsMatch
    ID                    id-at-x121Address }
```

```
X121Address ::= NumericString (SIZE(1..ub-x121-address))
```

-- String as defined by ITU-T Rec. X.121

5.7.6 Número ISDN Internacional

El tipo de atributo *número ISDN internacional* (*International ISDN Number*) especifica un número ISDN internacional asociado con un objeto.

Un valor de atributo para el número ISDN internacional es una cadena conforme al formato internacional convenido para las direcciones ISDN según la Rec. UIT-T E.164.

```
internationalISDNNumber ATTRIBUTE ::= {
    WITH SYNTAX                               InternationalISDNNumber
    EQUALITY MATCHING RULE                     numericStringMatch
    SUBSTRINGS MATCHING RULE                   numericStringSubstringsMatch
    ID                                          id-at-internationalISDNNumber }
```

InternationalISDNNumber ::= NumericString (SIZE(1..ub-international-isdn-number))

-- String complying with ITU-T Rec. E.164 only

El tipo de atributo *número ISDN internacional colectivo* (*Collective International ISDN Number*) especifica un número ISDN internacional para una serie de inserciones.

```
collectiveInternationalISDNNumber ATTRIBUTE ::= {
    SUBTYPE OF                               internationalISDNNumber
    COLLECTIVE                                TRUE
    ID                                         id-at-collectiveInternationalISDNNumber }
```

5.7.7 Dirección registrada

El tipo de atributo *dirección registrada* (*Registered Address*) especifica información nemotécnica para una dirección asociada con un objeto ubicado en un lugar determinado de una ciudad. La palabra nemotécnica está registrada en el país al que pertenece la ciudad y se utiliza para el suministro del servicio público de telegramas (de acuerdo con la Rec. UIT-T F.1).

```
registeredAddress ATTRIBUTE ::= {
    SUBTYPE OF                               postalAddress
    WITH SYNTAX                               PostalAddress
    ID                                         id-at-registeredAddress }
```

5.7.8 Indicador de destino

El tipo de atributo *indicador de destino* (*Destination Indicator*) especifica (conforme a la Rec. UIT-T F.1 y Rec. CCITT F.31) el país y la ciudad asociados con el objeto (el destinatario) necesarios para suministrar el servicio público de telegramas.

Un valor de atributo para indicador de destino es una cadena.

```
destinationIndicator ATTRIBUTE ::= {
    WITH SYNTAX                               DestinationIndicator
    EQUALITY MATCHING RULE                     caseIgnoreMatch
    SUBSTRINGS MATCHING RULE                   caseIgnoreSubstringsMatch
    ID                                          id-at-destinationIndicator }
```

DestinationIndicator ::= PrintableString (SIZE(1..ub-destination-indicator))

-- alphabetical characters only

5.7.9 Servicio de comunicaciones

El tipo de atributo *servicios de comunicaciones* (*Communication Services*) especifica el tipo de servicio(s) asociado(s) con una dirección de comunicaciones.

```
communicationsService ATTRIBUTE ::= {
    WITH SYNTAX                               CommunicationsService
    EQUALITY MATCHING RULE                     objectIdentifierMatch
    ID                                         id-at-communicationsService }
```

CommunicationsService ::= OBJECT IDENTIFIER

Este atributo describe la clase de servicio al que la dirección de comunicaciones proporciona acceso, por ejemplo, telefónico (vocal), facsímil, correo electrónico, servicio de mensajes cortos (SMS), intercambio electrónico de datos (EDI), transferencia de ficheros, etc.

La asignación de identificadores de objeto para la identificación de servicios no se efectúa en esta Especificación de directorio.

5.7.10 Red de comunicaciones

El atributo *red de comunicaciones* (*Communications Network*) especifica el tipo de red para el que se utiliza una dirección de telecomunicaciones.

```
communicationsNetwork ATTRIBUTE ::= {
    WITH SYNTAX                CommunicationsNetwork
    EQUALITY MATCHING RULE     objectIdentifierMatch
    SINGLE VALUE               TRUE
    ID                         id-at-communicationsNetwork }
```

CommunicationsNetwork ::= OBJECT IDENTIFIER

Este atributo describe el tipo de red donde es asignada la dirección de comunicaciones, por ejemplo, una red telefónica pública conmutada (RTPC), una red RDSI, o una red telefónica móvil GSM. Podría ser también una red orientada a una aplicación, por ejemplo, una red bancaria.

La asignación de identificadores de objeto para la identificación de red no se efectúa en esta Especificación de directorio.

5.8 Tipos de atributos de preferencias

Estos tipos de atributos se refieren a las preferencias de un objeto.

5.8.1 Método de entrega preferido

El tipo de atributo *método de entrega preferido* (*Preferred Delivery Method*) especifica el orden de prioridad del objeto en lo que respecta al método que se utilizará para comunicar con él.

```
preferredDeliveryMethod ATTRIBUTE ::= {
    WITH SYNTAX                PreferredDeliveryMethod
    SINGLE VALUE               TRUE
    ID                         id-at-preferredDeliveryMethod }
```

```
PreferredDeliveryMethod ::= SEQUENCE OF INTEGER {
    any-delivery-method        (0),
    mhs-delivery               (1),
    physical-delivery          (2),
    telex-delivery             (3),
    teletex-delivery           (4),
    g3-facsimile-delivery      (5),
    g4-facsimile-delivery      (6),
    ia5-terminal-delivery      (7),
    videotex-delivery          (8),
    telephone-delivery         (9) }
```

5.9 Tipos de atributos de aplicación de OSI

Estos tipos de atributos conciernen a la información relacionada con objetos en la capa de aplicación de OSI.

5.9.1 Dirección de presentación

El tipo de atributo *dirección de presentación* (*Presentation Address*) especifica una dirección asociada con un objeto que representa una entidad de aplicación de OSI.

Un valor de atributo para dirección de presentación es una dirección de presentación como la definida en la Rec. UIT-T X.200 | ISO/CEI 7498-1.

```
presentationAddress ATTRIBUTE ::= {
    WITH SYNTAX                PresentationAddress
    EQUALITY MATCHING RULE     presentationAddressMatch
    SINGLE VALUE               TRUE
    ID                         id-at-presentationAddress }
```

```

PresentationAddress ::= SEQUENCE {
    pSelector      [0] OCTET STRING OPTIONAL,
    sSelector      [1] OCTET STRING OPTIONAL,
    tSelector      [2] OCTET STRING OPTIONAL,
    nAddresses     [3] SET SIZE (1..MAX) OF OCTET STRING }

```

5.9.2 Contexto de aplicación soportado

El tipo de atributo *contexto de aplicación soportado* (*Supported Application Context*) especifica el identificador o los identificadores de objeto de uno o más contextos de aplicación que son soportados por el objeto (una entidad de aplicación de OSI).

```

supportedApplicationContext ATTRIBUTE ::= {
    WITH SYNTAX          OBJECT IDENTIFIER
    EQUALITY MATCHING RULE objectIdentifierMatch
    ID                   id-at-supportedApplicationContext }

```

5.9.3 Información de protocolo

El tipo de atributo *información de protocolo* (*Protocol Information*) asocia información de protocolo con cada dirección de red en el atributo dirección de presentación.

Para cada **nAddress**, el componente de protocolo identifica el protocolo o perfil para las capas de red y de transporte.

```

protocolInformation ATTRIBUTE ::= {
    WITH SYNTAX          ProtocolInformation
    EQUALITY MATCHING RULE protocolInformationMatch
    ID                   id-at-protocolInformation }

```

```

ProtocolInformation ::= SEQUENCE {
    nAddress      OCTET STRING,
    profiles     SET OF OBJECT IDENTIFIER }

```

5.10 Tipos de atributos relacionales

Estos tipos de atributo se refieren a información concerniente a los objetos que están relacionados de cierta manera con un determinado objeto.

NOTA – La sintaxis **DistinguishedName** utilizada en estos tipos de atributos permite la utilización del nombre distinguido primario o un nombre distinguido alternativo. La utilización del nombre distinguido primario, si se conoce, asegura la consistencia y el interfuncionamiento con los DSA anteriores a 1997. La utilización específica puede requerir que se emplee un determinado nombre alternativo. La información de contexto y los valores distinguidos alternativos pueden mantenerse además como parte del componente **valuesWithContext** de cualquier nombre distinguido relativo (RDN, *relative distinguished name*), descrito en 9.3 de la Rec. UIT-T X.501 | ISO/CEI 9594-2.

5.10.1 Nombre distinguido

El tipo de atributo *nombre distinguido* (*Distinguished Name*) es un atributo para especificar el nombre de un objeto.

```

distinguishedName ATTRIBUTE ::= {
    WITH SYNTAX          DistinguishedName
    EQUALITY MATCHING RULE distinguishedNameMatch
    ID                   id-at-distinguishedName }

```

5.10.2 Miembro

El tipo de atributo *miembro* (*Member*) especifica un grupo de nombres asociados con el objeto.

Un valor de atributo para miembro es un nombre distinguido.

```

member ATTRIBUTE ::= {
    SUBTYPE OF distinguishedName
    ID                   id-at-member }

```

5.10.3 Miembro único

El tipo de atributo *miembro único* (*Unique Member*) especifica un grupo de nombres únicos asociados con un objeto. Un nombre único es un nombre que es facultativamente inequívoco mediante la inclusión de su identificador único.

Un valor de atributo para miembro único es un nombre distinguido acompañado por un identificador único facultativo.

```
uniqueMember ATTRIBUTE ::= {
    WITH SYNTAX          NameAndOptionalUID
    EQUALITY MATCHING RULE uniqueMemberMatch
    ID                   id-at-uniqueMember }
```

```
NameAndOptionalUID ::= SEQUENCE {
    dn      DistinguishedName,
    uid     UniqueIdentifier OPTIONAL }
```

5.10.4 Propietario

El tipo de atributo *propietario* (*Owner*) especifica el nombre de cierto objeto que tiene alguna responsabilidad sobre el objeto asociado.

Un valor de atributo para propietario es un nombre distinguido (que podría representar un grupo de nombres) y puede repetirse (recurrencia).

```
owner ATTRIBUTE ::= {
    SUBTYPE OF distinguishedName
    ID         id-at-owner }
```

5.10.5 Encargado de la función

El tipo de atributo *encargado de la función* (*Role Occupant*) especifica el nombre del objeto que desempeña una función en la organización.

Un valor de atributo para encargado de la función es un nombre distinguido.

```
RoleOccupant ATTRIBUTE ::= {
    SUBTYPE OF distinguishedName
    ID         id-at-roleOccupant }
```

5.10.6 Véase también

El tipo de atributo *véase también* (*See Also*) especifica nombres de otros objetos de directorio que pueden constituir (en cierto sentido) otros aspectos del mismo objeto del mundo real.

Un valor de atributo para véase también es un nombre distinguido.

```
seeAlso ATTRIBUTE ::= {
    SUBTYPE OF distinguishedName
    ID         id-at-seeAlso }
```

5.11 Tipos de atributos de dominio

5.11.1 Nombre DMD

El tipo de atributo *DMD Name* especifica un dominio de gestión del directorio (DMD, *directory management domain*). Cuando se utiliza como un componente de un nombre de directorio identifica un DMD que gestiona el objeto denominado.

Un valor de atributo para DMD Name es una cadena elegida por el DMD.

```
dmdName ATTRIBUTE ::= {
    SUBTYPE OF name
    WITH SYNTAX DirectoryString{ub-common-name}
    ID         id-at-dmdName }
```

5.12 Atributos de notificación

Los atributos de notificación tienen la sintaxis de los atributos pero están definidos para transportar información adicional en los elementos **CommonResults** (o **CommonResultsSeq**) y **PartialOutcomeQualifier** (descritos en 7.4 y 10.1 de la Rec. UIT-T X.511 | ISO/CEI 9594-3). Suelen definirse con reglas de concordancia para que los valores retornados puedan confrontarse con valores localmente conocidos.

5.12.1 Problema de DSA

El mensaje *problema de DSA (DSA Problem)* se utiliza en unión de un **serviceError** o un **PartialOutcomeQualifier**, y se define como sigue:

```
dSAProblem ATTRIBUTE ::= {
    WITH SYNTAX
    EQUALITY MATCHING RULE
    ID
    OBJECT IDENTIFIER
    objectIdentifierMatch
    id-not-dSAProblem }
```

Los valores definidos para **dsaProblem** son:

- a) **id-pr-targetDsaUnavailable** – Una petición tiene que concatenarse a otro DSA durante la resolución de nombre, pero no puede establecerse ninguna asociación con este DSA.
- b) **id-pr-dataSourceUnavailable** – Un DSA no puede completar una operación ya que la parte de la DIB no está disponible.
- c) **id-pr-administratorImposedLimit** – Una operación ha rebasado algún límite fijado por el administrador.
- d) **id-pr-permanentRestriction** – Una operación ha hecho que el DSA rebase algún límite que provoca la parada del proceso y se considera que una operación repetida encuentra el mismo problema.
- e) **id-pr-temporaryRestriction** – Una operación ha hecho que el DSA rebase algún límite que provoca la parada del proceso, pero se considera que se trata de un problema temporal, por ejemplo, agotamiento de recurso.

5.12.2 Problema del servicio de búsqueda

El atributo de mensaje *problema del servicio de búsqueda (Search Service Problem)* describe problemas al aplicar los criterios de búsqueda; se utiliza combinado con errores de servicio o con el **PartialOutcomeQualifier**. Se define como sigue:

```
searchServiceProblem ATTRIBUTE ::= {
    WITH SYNTAX
    EQUALITY MATCHING RULE
    SINGLE VALUE
    ID
    OBJECT IDENTIFIER
    objectIdentifierMatch
    TRUE
    id-not-searchServiceProblem }
```

Los valores definidos para **searchServiceProblem** son:

- a) **id-pr-unidentifiedOperation** – La operación intentada no corresponde a una de las identificadas para este servicio.
- b) **id-pr-unavailableOperation** – La operación intentada cumple una regla de búsqueda no disponible para el solicitante.
- c) **id-pr-searchAttributeViolation** – Uno o más tipos de atributo que tenían que estar en el filtro no estaban presentes.
- d) **id-pr-searchAttributeCombinationViolation** – El filtro de la petición **search** no incluye la combinación requerida de tipos de atributos.
- e) **id-pr-searchValueNotAllowed** – Se especificaron valores de atributos para los tipos de atributos en los que sólo pueden especificarse los tipos de atributos y los tipos de elementos de filtro **present** y **contextPresent**.
- f) **id-pr-missingSearchAttribute** – Los atributos identificados, que no estaban presentes en la búsqueda solicitada, se requieren para la regla de búsqueda pertinente.
- g) **id-pr-searchValueViolation** – Los valores de atributo identificados para los tipos de atributo identificados no están permitidos cuando se busca utilizando la regla de búsqueda pertinente.
- h) **id-pr-attributeNegationViolation** – El tipo de atributo identificado no está permitido en la forma negada en el filtro de búsqueda.
- i) **id-pr-searchValueRequired** – El tipo de atributo no está permitido en el elemento de filtro que no requiere concordancia de valor.
- j) **id-pr-invalidSearchValue** – Los valores de atributo identificados no son válidos para los tipos de atributo identificados para la regla de búsqueda pertinente.
- k) **id-pr-searchContextViolation** – Los tipos de contexto identificados en la búsqueda intentada no están permitidos para el tipo de atributo.

- l) **id-pr-searchContextCombinationViolation** – Las combinaciones identificadas de tipos de contexto, que no estaban presentes en la búsqueda solicitada, se requieren para la regla de búsqueda pertinente.
- m) **id-pr-missingSearchContext** – Los tipos de contexto identificados, que no estaban presentes en la búsqueda solicitada, se requieren para el tipo de atributo.
- n) **id-pr-searchContextValueViolation** – Los valores de contexto identificados para los tipos de contexto identificados no están permitidos para el tipo de atributo.
- o) **id-pr-searchContextValueRequired** – El tipo de atributo no está permitido en los elementos de filtro que no requieren concordancia de valor.
- p) **id-pr-invalidContextSearchValue** – Los valores de atributo identificados no son válidos para los tipos de atributo identificados para la regla de búsqueda pertinente.
- q) **id-pr-unsupportedMatchingRule** – No se soporta la regla de concordancia solicitada.
- r) **id-pr-attributeMatchingViolation** – La regla de concordancia solicitada identificada, o su uso particular, no está permitida para los atributos identificados para la regla de búsqueda pertinente.
- s) **id-pr-unsupportedMatchingUse** – No se soporta la forma en que se sugiere que se utilice una regla de concordancia en un filtro de búsqueda.
- t) **id-pr-matchingUseViolation** – No está permitida la forma en que se sugiere que se utilice una regla de concordancia en un filtro de búsqueda, por ejemplo, la que se especifica en una regla de búsqueda.
- u) **id-pr-hierarchySelectForbidden** – La selección de jerarquía, excepto para **self**, no está permitida para el tipo de petición.
- v) **id-pr-invalidHierarchySelect** – Se especificaron en la petición una o más opciones de selección de jerarquía no válidas.
- w) **id-pr-unavailableHierarchySelect** – Una o más opciones de jerarquía no son soportadas por la implementación.
- x) **id-pr-invalidSearchControlOptions** – Se especificaron en la petición una o más opciones de búsqueda no válidas.
- y) **id-pr-invalidServiceControlOptions** – Se especificaron en la petición una o más opciones de control de servicio no válidas.
- z) **id-pr-searchSubsetViolation** – El subconjunto de búsqueda solicitado no está permitido para la regla de búsqueda pertinente.
- aa) **id-pr-unmatchedKeyAttributes** – Se seleccionó una regla de concordancia basada en correspondencia pero los elementos del filtro susceptibles de correspondencia no presentaron ninguna concordancia en relación con el cuadro de correspondencias pertinente.
- bb) **id-pr-ambiguousKeyAttributes** – Se seleccionó una regla de concordancia basada en correspondencia, pero los elementos del filtro susceptibles de correspondencia presentaron varias coincidencias con respecto al cuadro de correspondencias pertinente.
- cc) **id-pr-unavailableRelaxationLevel** – El DSA no soporta un nivel de ampliación de relajación solicitado.
- dd) **id-pr-emptyHierarchySelection** – Se especificó una selección de jerarquía que dio como resultado el que no se devolviera ninguna inserción aunque hubiere una o más inserciones que concordaran con el filtro de búsqueda.
- ee) **id-pr-relaxationNotSupported** – Se especificó una relajación en la petición de usuario, pero no es soportada.

5.12.3 Tipo de servicio

El atributo de notificación *tipo de servicio* (*Service-type*) especifica el tipo de servicio para la búsqueda fallida.

```

serviceType ATTRIBUTE ::= {
    WITH SYNTAX
    EQUALITY MATCHING RULE
    SINGLE VALUE
    ID
    OBJECT IDENTIFIER
    objectIdentifierMatch
    TRUE
    id-not-serviceType }

```

5.12.4 Lista de tipos de atributo

El atributo de notificación *lista de tipos de atributo (Attribute Type List)* da una lista de tipos de atributo para calificar aún más un problema de servicio de búsqueda.

```
attributeTypeList ATTRIBUTE ::= {
    WITH SYNTAX          OBJECT IDENTIFIER
    EQUALITY MATCHING RULE objectIdentifierMatch
    ID                   id-not-attributeTypeList }
```

5.12.5 Lista de reglas de concordancia

El atributo de notificación *lista de reglas de concordancia (Matching Rule List)* da una lista de reglas de concordancia para calificar aún más un problema de servicio de búsqueda.

```
matchingRuleList ATTRIBUTE ::= {
    WITH SYNTAX          OBJECT IDENTIFIER
    EQUALITY MATCHING RULE objectIdentifierMatch
    ID                   id-not-matchingRuleList }
```

5.12.6 Elemento de filtro

El atributo de notificación *elemento de filtro (Filter Item)* da una lista de elementos de filtro no válidos en un filtro de búsqueda.

```
filterItem ATTRIBUTE ::= {
    WITH SYNTAX          FilterItem
    ID                   id-not-filterItem }
```

5.12.7 Combinaciones de atributos

El atributo de notificación *combinaciones de atributos (Attribute Combinations)* da una lista de combinaciones de atributos que había que presentar en un filtro, pero que no se proporcionaron.

```
attributeCombinations ATTRIBUTE ::= {
    WITH SYNTAX          AttributeCombination
    ID                   id-not-attributeCombinations }
```

5.12.8 Lista de tipos de contexto

El atributo de notificación *lista de tipos de contexto (Context Type List)* da una lista de tipos de contexto para calificar mejor un problema del servicio de búsqueda.

```
contextTypeList ATTRIBUTE ::= {
    WITH SYNTAX          OBJECT IDENTIFIER
    EQUALITY MATCHING RULE objectIdentifierMatch
    ID                   id-not-contextTypeList }
```

5.12.9 Lista de contextos

El atributo de notificación *lista de contextos (Context List)* da una lista de contextos para calificar aún más un problema del servicio de búsqueda.

```
contextList ATTRIBUTE ::= {
    WITH SYNTAX          ContextAssertion
    ID                   id-not-contextList }
```

Un valor de este tipo de atributo representa un tipo de contexto y algunos valores de contexto de este tipo no permitidos en la situación particular que dio lugar a la generación de este atributo.

5.12.10 Combinaciones de contextos

El atributo de notificación *combinaciones de contextos (Context Combinations)* da una lista de combinaciones de contextos que había que presentar en un filtro, pero que no se proporcionaron.

```
contextCombinations ATTRIBUTE ::= {
    WITH SYNTAX          ContextCombination
    ID                   id-not-contextCombinations }
```

5.12.11 Lista de selecciones de jerarquía

El atributo de notificación *lista de selecciones de jerarquía (Hierarchy Select List)* da una cadena de bits que identifica una o más opciones de selecciones de jerarquía definidas en la serie de instrucciones **HierarchySelections** en 10.2.1 de la Rec. UIT-T X.511 | ISO/CEI 9594-3.

```

hierarchySelectList ATTRIBUTE ::= {
    WITH SYNTAX    HierarchySelections
    SINGLE VALUE  TRUE
    ID            id-not-hierarchySelectList }

```

Un bit puesto a 1 en la cadena de bits **HierarchySelection** indica que la selección de jerarquía correspondiente no es válida. Se ha solicitado una selección prohibida o no soportada, o bien la selección no se ha solicitado cuando es necesario.

5.12.12 Lista de opciones de control de búsqueda

El atributo de notificación *lista de opciones de control de búsqueda (Search Control Options List)* da una cadena de bits que identifica una o más opciones de control de búsqueda definidas por el tipo de datos ASN.1 **SearchControlOptions** en 10.2.1 de la Rec. UIT-T X.511 | ISO/CEI 9594-3.

```

searchControlOptionsList ATTRIBUTE ::= {
    WITH SYNTAX    SearchControlOptions
    SINGLE VALUE  TRUE
    ID            id-not-searchControlOptionsList }

```

Un bit puesto a 1 en las **SearchControlOptions** indica que la selección de opciones de búsqueda correspondiente no es válida. Se ha solicitado una selección prohibida o no soportada, o bien la selección no se ha solicitado cuando es necesario.

5.12.13 Lista de opciones de control del servicio

El atributo de notificación *lista de opciones de control del servicio (Service Control Options List)* da una cadena de bits que identifica una o más opciones de control del servicio definidas por el tipo de datos ASN.1 en **ServiceControlOptions** definido en 7.5 de la Rec. UIT-T X.511 | ISO/CEI 9594-3.

```

serviceControlOptionsList ATTRIBUTE ::= {
    WITH SYNTAX    ServiceControlOptions
    SINGLE VALUE  TRUE
    ID            id-not-serviceControlOptionsList }

```

Un bit puesto a 1 en las **ServiceControlOptions** indica que la selección de opciones de control de servicio correspondiente no es válida. Se ha solicitado una selección prohibida o no soportada, o bien la selección no se ha solicitado cuando es necesario.

5.12.14 Localidades de concordancia múltiple

El atributo de notificación *múltiples localidades de concordancia (Multiple Matching Localities)* especifica en cada valor un conjunto de aserciones de atributo que si se confrontan con el nomenclátor geográfico darán una concordancia única.

```

multipleMatchingLocalities ATTRIBUTE ::= {
    WITH SYNTAX    MultipleMatchingLocalities
    ID            id-not-multipleMatchingLocalities }

```

```

MultipleMatchingLocalities ::= SEQUENCE {
    matchingRuleUsed  MATCHING-RULE.&id OPTIONAL,
    attributeList    SEQUENCE OF AttributeValueAssertion }

```

El elemento **matchingRuleUsed** está opcionalmente presente, y puede utilizarse para indicar la regla de concordancia por correspondencia que se utilizó.

No se define para este atributo ninguna regla de concordancia; se toleran múltiples valores idénticos o casi idénticos.

5.12.15 Relajación propuesta

El atributo de notificación *relajación propuesta* (*Proposed Relaxation*) da una secuencia de elementos **MRMapping** que pueden suministrarse como parte de la **RelaxationPolicy** aplicada en el componente **relaxation** de una petición **search** subsiguiente.

```
proposedRelaxation ATTRIBUTE ::= {  
    WITH SYNTAX          MRMappings  
    ID                   id-not-proposedRelaxation }
```

MRMappings ::= SEQUENCE OF MRMapping

La secuencia de **MRMapping** no es significativa.

5.12.16 Relajación aplicada

El atributo de notificación *relajación aplicada* (*applied relaxation*) se utiliza para enumerar los atributos del filtro que han sido sometidos a relajación o a rigorización, además de los que lo han sido por el elemento **basic** de la política de relajación.

```
appliedRelaxation ATTRIBUTE ::= {  
    WITH SYNTAX          OBJECT IDENTIFIER  
    EQUALITY MATCHING RULE objectIdentifierMatch  
    ID                   id-not-appliedRelaxation }
```

SECCIÓN 3 – REGLAS DE CONCORDANCIA

6 Preparación de cadenas

Se aplicará el siguiente proceso de 6 pasos a cada valor presentado y de atributo cuando se va a realizar una evaluación de las reglas de concordancia de cadenas.

- 1) Transcodificación
- 2) Transcripción
- 3) Normalización
- 4) Prohibición
- 5) Verificación de restricciones bidireccionales (*bidi*)
- 6) Supresión de caracteres irrelevantes

Si falla cualquiera de estos pasos, se considera que la aserción es INDEFINIDA (UNDEFINED).

Los valores de comparación que se crean durante el proceso de preparación de cadenas son efímeros, y no han de afectar el valor de atributo almacenado en el Directorio.

6.1 Transcodificación

Se transcodifica a Unicode todo valor de cadena que no sea Unicode.

Los valores **TeletexString** se transcodifican a Unicode conforme al anexo B.

El valor **PrintableString** se transcodifica directamente a Unicode.

No es necesario transcodificar los valores **UniversalString**, **UTF8String**, y **BMPString**, puesto que son cadenas basadas en Unicode (en el caso de **BMPString**, restringida a un subconjunto de Unicode).

Si en la implementación no se puede o no se quiere realizar la transcodificación descrita, o si ésta es infructuosa, este paso no se completa y la aserción se considera INDEFINIDA.

El resultado es la cadena transcodificada.

6.2 Transcripción

No se transcriben los puntos de código SOFT HYPHEN (guion virtual) (U+00AD) y MONGOLIAN TODO SOFT HYPHEN (U+1806). COMBINING GRAPHEME JOINER (U+034F) y VARIATION SELECTOR (selector de variación) (U+180B-180D, FF00-FE0F) también son puntos de código y tampoco se transcriben. Tampoco se transcribe OBJECT REPLACEMENT CHARACTER (carácter de reemplazo de objeto) (U+FFFC).

CHARACTER TABULATION (tabulación de carácter) (U+0009), LINE FEED (LF) (salto de línea) (U+000A), LINE TABULATION (tabulación de línea) (U+000B), FORM FEED (FF) (salto de página) (U+000C), CARRIAGE RETURN (CR) (retorno de carro) (U+000D) y NEXT LINE (próxima línea) (NEL) (U+0085) se transcriben a SPACE (espacio) (U+0020).

No se transcribe ninguno de los otros puntos de control de código (por ejemplo, Cc) ni los puntos de código que tienen una función de control (por ejemplo, Cf).

ZERO WIDTH SPACE (espacio de ancho cero) (U+200B) no se transcribe. Todos los otros puntos de código con propiedad Separador (espacio, línea, o párrafo) (por ejemplo, Zs, Zl o Zp) se transcriben a SPACE (U+0020).

En lo referente a las reglas de concordancia de insensibilidad a mayúsculas/minúsculas, numéricas y de prefijo almacenado, se aplica B.2 del RFC 3454.

6.3 Normalización

Se debe normalizar la cadena con arreglo a la Forma KC (compuesta por compatibilidad) de Unicode, como se describe en el anexo #15 a la Norma Unicode.

6.4 Prohibición

Se prohíben todos los puntos de código no atribuidos, de uso privado y que no estén compuestos por caracteres. No son permitidos los caracteres sustitutos (U+D800-DFFFF).

ISO/CEI 9594-6:2005 (S)

No está permitido emplear el código REPLACEMENT CHARACTER (U+FFFD). El primer punto de código de una cadena no podrá ser un carácter combinable. No se pueden utilizar cadenas vacías. Si la cadena en cuestión (cadena de entrada) contiene un punto de código prohibido, no aprueba este paso del proceso y se concluye que la aserción es INDEFINIDA. La cadena que resulta es la misma cadena de entrada

6.5 Verificación de restricciones bidireccionales (bidi)

No hay restricciones bidireccionales. La cadena que resulta es la cadena de entrada.

6.6 Supresión de caracteres irrelevantes

En este paso se han de suprimir los caracteres que no sean importantes para la regla de concordancia, algo que varía de una regla a otra. La cláusula 6.6.1 se aplica a la correspondencia de cadenas insensible a mayúscula/minúscula o a la correspondencia exacta de cadenas. La cláusula 6.6.2 vale para la correspondencia de **numericString** (cadena numérica), y la 6.6.3 para la de **telephoneNumber** (número telefónico).

6.6.1 Supresión de espacios irrelevantes

A los efectos de esta cláusula, un espacio es un punto de código SPACE (U+0020) que no va seguido de marcas combinables.

NOTA – Los pasos anteriores garantizan que la cadena no contiene ningún punto de código de la clase separador, salvo SPACE (U+0020).

Los espacios que se consideran no significativos y deben suprimirse son:

- los iniciales (es decir, los que preceden al primer carácter que no es un espacio);
- los finales (es decir, los que siguen al último carácter que no es un espacio);
- los consecutivos múltiples (estos espacios se consideran equivalentes a un solo carácter de espacio). (Una cadena formada enteramente por espacios equivale a una cadena que contiene exactamente un espacio). Por ejemplo, la supresión de espacios en la cadena Forma KC: "<SPACE><SPACE>foo<SPACE><SPACE>bar<SPACE><SPACE>" produciría la cadena: "foo<SPACE>bar", y la supresión de espacios en la cadena Forma KC: "<SPACE><SPACE><SPACE>", daría como resultado: "<SPACE>".

6.6.2 Supresión de caracteres irrelevantes en una cadena numérica (NumericString)

A los efectos de esta cláusula, un espacio es un punto de código SPACE (U+0020) que no va seguido de marcas combinables. Todos los espacios son no significativos y han de suprimirse. Por ejemplo, la supresión de espacios en la cadena Forma KC: "<SPACE><SPACE>123<SPACE><SPACE>456<SPACE><SPACE>" daría como resultado: "123456"; mientras que la cadena Forma KC: "<SPACE><SPACE><SPACE>" se convertiría en una cadena vacía.

6.6.3 Supresión de caracteres irrelevantes en un número telefónico (TelephoneNumber)

A los efectos de esta subcláusula, un guión (*hyphen*) es el punto de código HYPHEN-MINUS (U+002D), ARMENIAN HYPHEN (U+058A), HYPHEN (U+2010), NON-BREAKING HYPHEN (U+2011), MINUS SIGN (U+2212), SMALL HYPHEN-MINUS (U+FE63), o FULLWIDTH HYPHEN-MINUS (U+FF0D) que no va seguido por marcas combinables, y un espacio es el punto de código SPACE (U+0020) que tampoco va seguido por dichas marcas. Todos los guiones y espacios se consideran irrelevantes y, por tanto, han de suprimirse.

7 Definición de reglas de concordancia

NOTA – Para las definiciones de **objectIdentifierMatch** y **distinguishedNameMatch** véase la Rec. UIT-T X.501 | ISO/CEI 9594-2.

7.1 Reglas de concordancia de cadenas

En las reglas de concordancia especificadas en 7.1.1 a 7.1.9, todos los valores de cadena presentados y almacenados deben someterse al proceso de preparación para concordancia de la cláusula 6. De este preparación resultan cadenas adecuadas para la correspondencia carácter por carácter.

7.1.1 Correspondencia de mayúsculas/minúsculas (*Case Exact Match*) o indiferencia (*Case Ignore Match*)

La regla de *concordancia exacta de mayúsculas/minúsculas (Case Exact Match)* se aplica al comparar una cadena presentada con valores de atributo de tipo **DirectoryString** o uno de los tipos de datos presentes en el tipo de selección **DirectoryString**, por ejemplo **UTF8String**, sin tener en cuenta los espacios irrelevantes (véase la subcláusula 6.6).

Todos los caracteres tratados como espacios blancos en Unicode deberían considerarse como equivalentes y no deberían tenerse en cuenta por no ser significativos, siempre que así lo indiquen las reglas de concordancia del inicio de 6.1. Después de haber tenido en cuenta los espacios blancos, se debería efectuar la correspondencia sin distinguir entre mayúsculas y minúsculas, como se describe en la Norma Unicode, y aplicar la normalización de Forma D o Forma KC, conforme al Tecnical Report 15 de Unicode, dependiendo del repertorio de caracteres que se está analizando y de los requisitos de desempeño.

```
caseExactMatch MATCHING-RULE ::= {
    SYNTAX DirectoryString {ub-match}
    ID      id-mr-caseExactMatch }
```

La regla de *concordancia insensible a mayúsculas/minúsculas (Case Ignore Match)* se aplica al comparar una cadena presentada con valores de atributo de tipo **DirectoryString** o uno de los tipos de datos existentes en el tipo de selección **DirectoryString**, por ejemplo **UTF8String** sin distinguir entre mayúsculas/minúsculas de las cadenas (por ejemplo, "Dundee" y "DUNDEE" concuerdan) ni los espacios irrelevantes (véase 6.6). Esta regla es idéntica a la regla **caseExactMatch**, salvo que durante la preparación los caracteres en mayúsculas se ponen en minúsculas, conforme a 6.2.

```
caseIgnoreMatch MATCHING-RULE ::= {
    SYNTAX DirectoryString {ub-match}
    ID      id-mr-caseIgnoreMatch }
```

El resultado de ambas reglas será TRUE (VERDADERO) si las dos cadenas preparadas tienen la misma longitud y los caracteres correspondientes en las cadenas preparadas son idénticos.

7.1.2 Concordancia de orden sensible a mayúsculas/minúsculas (*Case Exact Ordering Match*) e insensible a mayúsculas/minúsculas (*Case Ignore Ordering Match*)

La regla de *concordancia de orden sensible a mayúsculas/minúsculas (Case Exact Ordering Match)* se aplica al comparar el orden de cotejo de una cadena presentada con un atributo de tipo **DirectoryString** o cualquiera de los tipos de datos presentes en el tipo de selección **DirectoryString**, por ejemplo **UTF8String**, sin tener en cuenta los espacios irrelevantes (véase 6.6).

```
caseExactOrderingMatch MATCHING-RULE ::= {
    SYNTAX DirectoryString {ub-match}
    ID      id-mr-caseExactOrderingMatch }
```

La regla de *concordancia de orden insensible a mayúsculas/minúsculas (Case Ignore Ordering Match)* compara el orden de cotejo de una cadena presentada con valores de atributo de tipo **DirectoryString** o cualquiera de los tipos de datos presentes en el tipo de selección **DirectoryString**, por ejemplo **UTF8String**, sin tener en cuenta la diferencia mayúsculas/minúsculas de las cadenas ni los espacios irrelevantes (véase cláusula 6.6). Es idéntica a la regla **caseExactOrderingMatch**, salvo que durante la preparación los caracteres en mayúsculas se ponen en minúsculas, conforme a 6.2.

```
caseIgnoreOrderingMatch MATCHING-RULE ::= {
    SYNTAX DirectoryString {ub-match}
    ID      id-mr-caseIgnoreOrderingMatch }
```

El resultado de ambas reglas será TRUE si el valor del atributo es "menos" o aparece antes que el valor presentado, cuando las cadenas son comparadas utilizando el orden de cotejo de punto de código Unicode.

NOTA – El orden de cotejo proporciona información específica (lingüística y cultural) acerca de cómo se ordenan los diferentes caracteres en un idioma determinado. Un sistema de Directorio puede soportar varios órdenes de cotejo configurables. La implementación de esta capacidad está fuera del alcance de esta Especificación de directorio.

7.1.3 Concordancia de subcadenas exacta (*Case Exact Substrings Match*) y concordancia insensible a mayúsculas/minúsculas (*Case Ignore Substrings Match*)

La regla de *concordancia exacta de subcadenas sensible a minúsculas/mayúsculas (Case Exact Substrings Match)* determina si un valor presentado es una subcadena de valores de atributo de tipo **DirectoryString**, o uno de los tipos de datos presentes en el tipo de selección **DirectoryString**, por ejemplo **UTF8String**, sin tener en cuenta los espacios irrelevantes (véase 6.6).

```
caseExactSubstringsMatch MATCHING-RULE ::= {
    SYNTAX SubstringAssertion      -- only the PrintableString choice
    ID      id-mr-caseExactSubstringsMatch }
```

ISO/CEI 9594-6:2005 (S)

La regla de *concordancia de subcadenas insensible a mayúsculas/minúsculas* (*Case Ignore Substrings Match*) determina si un valor presentado es una subcadena de valores de atributo de tipo **DirectoryString** o uno de los tipos de datos presentes en el tipo de selección **DirectoryString**, por ejemplo **UTF8String**, sin distinguir entre mayúsculas/minúsculas de las cadenas ni tener en cuenta los espacios irrelevantes (véase 6.6). Es idéntica a la regla **caseExactSubstringMatch**, salvo que durante la preparación los caracteres en mayúsculas se ponen en minúsculas, conforme a 6.2.

```
caselgnoreSubstringsMatch MATCHING-RULE ::= {  
    SYNTAX SubstringAssertion  
    ID      id-mr-caselgnoreSubstringsMatch }
```

```
SubstringAssertion ::= SEQUENCE OF CHOICE {  
    initial [0] DirectoryString {ub-match},  
    any     [1] DirectoryString {ub-match},  
    final   [2] DirectoryString {ub-match},  
    control Attribute } -- Used to specify interpretation of the following items  
                        -- at most one initial and one final component
```

El resultado de ambas reglas será TRUE si hay una división del valor de atributo (en porciones) de modo que:

- las subcadenas específicas (**initial**, **any**, **final**) concuerdan con diferentes porciones del valor en el orden de la secuencia de **strings** (cadenas);
- **initial**, si está presente concuerda con la primera porción del valor;
- **final**, si está presente, concuerda con la última porción del valor;
- **any**, si está presente, concuerda con alguna porción arbitraria del valor.
- **control** no se utiliza para el **caselgnoreSubstringsMatch**, **telephoneNumberSubstringsMatch**, ni cualquier otra forma de concordancia de subcadenas para la cual sólo se utilicen en el algoritmo de concordancia elementos iniciales, cualesquiera o finales; si se encuentra un elemento **control**, se ignora. Si el elemento de control sólo se utiliza para reglas de concordancia que especifiquen explícitamente su uso en el algoritmo de codificación. Dicha regla de uso puede también redefinir la semántica de las subcadenas **initial**, **any** y **final**.

NOTA – La regla de concordancia **generalWordMatch** es un ejemplo de dicha regla de concordancia.

Habrán como máximo un **initial** y un **final** en la **SubstringAssertion**. Si **initial** está presente, será el primer elemento. Si **final** está presente, será el último elemento. Podrá haber cualquier número de **any**, o ninguno.

Para que un componente de subcadenas concuerde con una porción del valor de atributo, los caracteres correspondientes deberán ser idénticos (incluyendo todos los caracteres y secuencias de caracteres combinables).

7.1.4 Concordancia de cadena numérica

La regla de *concordancia de cadena numérica* (*Numeric String Match*) compara, para determinar la igualdad, una cadena numérica presentada con valores de atributo de tipo **NumericString**.

```
numericStringMatch MATCHING-RULE ::= {  
    SYNTAX NumericString  
    ID      id-mr-numericStringMatch }
```

Esta regla es idéntica a la regla de concordancia **caselgnoreMatch**, (la diferencia mayúsculas/minúsculas no produce efecto alguno cuando los caracteres son numéricos), salvo que durante la preparación se suprimen los caracteres espacio, conforme a 6.6.2.

7.1.5 Concordancia de orden de cadena numérica

La regla de *concordancia de orden de cadena numérica* (*Numeric String Ordering Match*) compara el orden de cotejo de una cadena presentada con valores de atributo de tipo **NumericString**.

```
numericStringOrderingMatch MATCHING-RULE ::= {  
    SYNTAX NumericString  
    ID      id-mr-numericStringOrderingMatch }
```

Esta regla es idéntica a la regla de concordancia **caselgnoreOrderingMatch**, (la diferencia mayúsculas/minúsculas no produce efecto alguno cuando los caracteres son numéricos), salvo que durante la preparación se suprimen los caracteres espacio, conforme a 6.6.2.

7.1.6 Concordancia de subcadenas de cadena numérica

La regla de *concordancia de subcadenas de cadenas numéricas* (*Numeric String Substrings Match*) determina si un valor presentado es una subcadena de un valor de atributo de tipo **NumericString**.

```
numericStringSubstringsMatch MATCHING-RULE ::= {
  SYNTAX SubstringAssertion
  ID      id-mr-numericStringSubstringsMatch }
```

Esta regla es idéntica a la regla de concordancia **caselgnoreSubstringsMatch** (la diferencia mayúsculas/minúsculas no produce efecto alguno cuando los caracteres son numéricos), salvo que durante la preparación se suprimen los caracteres espacio, conforme a 6.6.2.

7.1.7 Concordancia de lista insensible a mayúsculas/minúsculas

La regla de *concordancia de lista insensible a mayúsculas/minúsculas* (*Case Ignore List Match*) compara, para determinar la igualdad, una secuencia presentada de cadenas con un valor de atributo que es una secuencia de **DirectoryString**, sin tener en cuenta la diferencia entre mayúsculas/minúsculas de las cadenas ni los espacios significativos (véase 6.6).

```
caselgnoreListMatch MATCHING-RULE ::= {
  SYNTAX CaselgnoreList
  ID      id-mr-caselgnoreListMatch }
```

CaselgnoreList ::= SEQUENCE OF DirectoryString {ub-match}

Esta regla devuelve TRUE únicamente si cada lista contiene el mismo número de cadenas, y si las cadenas correspondientes concuerdan. Para establecer esta última concordancia se sigue la regla de concordancia sin tener en cuenta la posición mayúsculas/minúsculas (**caselgnoreMatch**).

7.1.8 Concordancia de subcadenas de lista insensible a mayúsculas/minúsculas

La regla de concordancia de subcadenas de lista insensible a mayúsculas/minúsculas (*Case Ignore Substring Match*) compara una subcadena presentada con un valor de atributo que es una secuencia de **DirectoryStrings**, pero sin tener en cuenta la diferencia mayúsculas/minúsculas de las cadenas ni los espacios irrelevantes (véase 6.6).

```
caselgnoreListSubstringsMatch MATCHING-RULE ::= {
  SYNTAX SubstringAssertion
  ID      id-mr-caselgnoreListSubstringsMatch }
```

Un valor presentado concuerda con un valor almacenado únicamente si el valor presentado concuerda con la cadena formada al concatenar las cadenas del valor almacenado. Para esta concordancia se utiliza la regla **caselgnoreSubstringsMatch**; sin embargo, se considera que ninguno de los valores **initial**, **any**, o **final** del valor presentado concuerda con una subcadena, de la cadena formada por concatenación, que abarca más de una de las cadenas del valor almacenado.

7.1.9 Concordancia de prefijo almacenado

La regla de *concordancia de prefijo almacenado* determina si un valor de atributo, cuya sintaxis es **DirectoryString**, es un prefijo (es decir, una subcadena inicial) del valor presentado sin tener en cuenta la diferencia mayúsculas/minúsculas de las cadenas ni los espacios irrelevantes (véase 6.6).

NOTA – Puede utilizarse, por ejemplo, para comparar valores en el directorio que son indicativos interurbanos telefónicos con un valor que supuestamente es un número telefónico.

```
storedPrefixMatch MATCHING-RULE ::= {
  SYNTAX DirectoryString {ub-match}
  ID      id-mr-storedPrefixMatch }
```

La regla devuelve TRUE si el valor de atributo es una subcadena inicial del valor presentado con caracteres correspondientes idénticos excepto en lo que tiene que ver con la diferencia mayúsculas/minúsculas.

7.2 Reglas de concordancia basada en sintaxis

7.2.1 Concordancia de booleanos

La regla de *concordancia de booleanos* (*Boolean Match*) compara para determinar la igualdad de un valor booleano con valores de atributo de tipo **BOOLEAN**.

```
booleanMatch MATCHING-RULE ::= {
    SYNTAX BOOLEAN
    ID      id-mr-booleanMatch }
```

Esta regla devuelve TRUE si ambos son **TRUE** o si ambos son **FALSE**.

7.2.2 Concordancia de enteros

La regla de *concordancia de enteros* (*Integer Match*) compara para determinar la igualdad de un valor entero o un valor enumerado presentado con valores de atributo de tipo **INTEGER** o **ENUMERATED**, respectivamente.

```
integerMatch MATCHING-RULE ::= {
    SYNTAX INTEGER
    ID      id-mr-integerMatch }
```

Esta regla devuelve TRUE si el valor entero presentado o el valor enumerado presentado es idéntico al valor de atributo.

7.2.3 Concordancia de ordenamiento de enteros

La regla de *concordancia de ordenamiento de enteros* (*Integer Ordering Match*) compara un valor entero presentado con valores de atributo de tipo **INTEGER**.

```
integerOrderingMatch MATCHING-RULE ::= {
    SYNTAX INTEGER
    ID      id-mr-integerOrderingMatch }
```

Esta regla devuelve TRUE si cualquier valor de atributo es menor que el valor presentado.

7.2.4 Concordancia de cadena de bits

La regla de *concordancia de cadenas de bits* (*Bit String Match*) compara una cadena de bits presentada como un valor de atributo del tipo **BIT STRING**.

```
bitStringMatch MATCHING-RULE ::= {
    SYNTAX BIT STRING
    ID      id-mr-bitStringMatch }
```

La regla devuelve TRUE si el valor de atributo tiene el mismo número de bits que el valor presentado y los bits concuerdan bit por bit. Si la sintaxis de atributo se define con una *NamedBitList*, no se tienen en cuenta los bits cero finales en los valores de atributo y presentado.

7.2.5 Concordancia de cadena de octetos

La regla de *concordancia de cadena de octetos* (*Octet String Match*) compara para establecer la igualdad de una cadena de octetos presentada con un valor de atributo de tipo **OCTET STRING**.

```
octetStringMatch MATCHING-RULE ::= {
    SYNTAX OCTET STRING
    ID      id-mr-octetStringMatch }
```

Esta regla devuelve TRUE únicamente si las cadenas tienen la misma longitud y los octetos correspondientes son idénticos.

7.2.6 Concordancia de ordenamiento de cadena de octetos

La regla de *concordancia de ordenamiento de cadena de octetos* (*Octet String Ordering Match*) compara el orden de cotejo de una cadena de octetos presentada con un valor de atributo de tipo **OCTET STRING**.

```
octetStringOrderingMatch MATCHING-RULE ::= {
    SYNTAX OCTET STRING
    ID      id-mr-octetStringOrderingMatch }
```

Esta regla compara cadenas de octetos desde el primer octeto hasta el último, y desde el bit más significativo hasta el bit menos significativo en cada octeto. La primera aparición de un bit diferente determina el ordenamiento de las cadenas. Un bit cero precede a un bit uno. Si las cadenas son idénticas pero contienen diferentes números de octetos, la cadena más corta precede a la más larga.

7.2.7 Concordancia de subcadenas de cadena de octetos

La regla de *concordancia de subcadenas de cadena de octetos* (*Octet String Substrings Match*) determina si una cadena de octetos presentada es una subcadena de un valor de atributo de tipo **OCTET STRING**.

```
octetStringSubstringsMatch MATCHING-RULE ::= {
    SYNTAX OctetSubstringAssertion
    ID      id-mr-octetStringSubstringsMatch }
```

```
OctetSubstringAssertion ::= SEQUENCE OF CHOICE {
    initial [0] OCTET STRING,
    any     [1] OCTET STRING,
    final   [2] OCTET STRING }
-- at most one initial and one final component
```

La regla devuelve TRUE si el valor de atributo contiene la secuencia de octetos en la cadena presentada, como se describe para **caseIgnoreSubstringsMatch**.

7.2.8 Concordancia de número telefónico

La regla de *concordancia de número telefónico* (*Telephone Number Match*) compara un valor presentado con un valor de atributo de tipo **PhoneNumber** (véase 5.7.1).

```
telephoneNumberMatch MATCHING-RULE ::= {
    SYNTAX PhoneNumber
    ID      id-mr-telephoneNumberMatch }
```

Las reglas para la concordancia son idénticas a las de **caseIgnoreMatch**, con la salvedad de que los guiones y espacios son irrelevantes (véase 6.6.3) y, por tanto, se suprimen durante la etapa correspondiente del proceso.

7.2.9 Concordancia de subcadenas de número telefónico

La regla de *concordancia de subcadenas de número telefónico* (*Telephone Number Substrings Match*) determina si una subcadena presentada es una subcadena de un valor de atributo de tipo **PrintableString** que es un número telefónico.

```
telephoneNumberSubstringsMatch MATCHING-RULE ::= {
    SYNTAX SubstringAssertion
    ID      id-mr-telephoneNumberSubstringsMatch }
```

Las reglas para la concordancia son idénticas a las de **caseExactSubstringsMatch**, salvo que los guiones y espacios son irrelevantes (véase 6.6.3) y, por tanto, se suprimen durante la etapa correspondiente del proceso.

7.2.10 Concordancia de direcciones de presentación

La regla de *concordancia de direcciones de presentación* (*Presentation Address Match*) compara la igualdad de una dirección de presentación presentada con un valor de atributo del tipo **PresentationAddress**.

```
presentationAddressMatch MATCHING-RULE ::= {
    SYNTAX PresentationAddress
    ID      id-mr-presentationAddressMatch }
```

La regla devuelve TRUE solamente si los selectores de las direcciones de presentación presentadas y almacenadas son iguales y las **nAddresses** presentadas son un subconjunto de las almacenadas.

7.2.11 Concordancia de miembro único

La regla de *concordancia de miembro único* (*Unique Member Match*) compara un valor de miembro único presentado con un valor de atributo del tipo **NameAndOptionalUID**.

```
uniqueMemberMatch MATCHING-RULE ::= {
    SYNTAX NameAndOptionalUID
    ID      id-mr-uniqueMemberMatch }
```

La regla devuelve TRUE solamente si los componentes **dn** del valor de atributo y el valor presentado concuerdan de conformidad con la regla **distinguishedNameMatch**, y el componente **uid** está ausente del valor de atributo o concuerda con el componente correspondiente del valor presentado de conformidad con la regla **bitStringMatch**.

7.2.12 Concordancia de información de protocolo

La regla de *concordancia de información de protocolo* (*Protocol Information Match*) compara valores presentados de **ProtocolInformation** con valores del mismo tipo.

```
protocolInformationMatch MATCHING-RULE ::= {
    SYNTAX OCTET STRING
    ID      id-mr-protocolInformationMatch }
```

Un valor de aserción se deriva de un valor de la sintaxis de atributo utilizando el componente **nAddress**.

La regla devuelve TRUE si el valor presentado y el componente **nAddress** del valor almacenado concuerdan de conformidad con la regla **octetStringMatch**.

7.2.13 Concordancia de número de facsímil

La regla *concordancia de número facsímil* (*Facsimile Number Match*) compara, para determinar la igualdad, un valor presentado con el primer elemento de la secuencia de valor de atributo. Dicho elemento, **telephoneNumber**, es de tipo **TelephoneNumber** (véase 5.7.1). El elemento **parameters** de la secuencia de número de facsímil no se evalúa.

```
facsimileNumberMatch MATCHING-RULE ::= {
    SYNTAX TelephoneNumber
    ID      id-mr-facsimileNumberMatch }
```

Las reglas de concordancia son idénticas a las aplicables a **telephoneNumberMatch**.

7.2.14 Concordancia de subcadenas de número facsímil

La regla *concordancia de subcadenas de número facsímil* (*Facsimile Number Substrings Match*) determina si una subcadena presentada es una subcadena del primer elemento de la secuencia de valor de atributo. Dicho elemento, **telephoneNumber**, es de tipo **TelephoneNumber** y es un número telefónico. El elemento **parameters** de la secuencia de número de facsímil no se evalúa.

```
facsimileNumberSubstringsMatch MATCHING-RULE ::= {
    SYNTAX SubstringAssertion
    ID      id-mr-facsimileNumberSubstringsMatch }
```

Las reglas de correspondencia son idénticas a las aplicables a **telephoneNumberMatch**.

7.2.15 Concordancia de par UUID

La regla *concordancia de par UUID* (*UUID Pair Match rule*) compara, para determinar la igualdad, valores presentados de **UUIDPair**, y se define así:

```
uUIDPairMatch MATCHING-RULE ::= {
    SYNTAX UUIDPair
    ID      id-mr-uuidpairmatch }
```

Un valor presentado del tipo **UUIDPair** corresponderá a un valor objetivo del tipo **UUIDPair** sólo si cada componente de aquél es igual al correspondiente componente de éste, si los componentes correspondientes tienen la misma longitud y si los octetos correspondientes son iguales.

7.2.16 Concordancia de componentes

En un sistema de directorio hay sintaxis de atributos que van desde tipos de datos simples, por ejemplo cadena de texto, enteros o booleanos, hasta tipos de datos con estructura compleja, como las sintaxis de atributos de funcionamiento de esquema de directorio. Las reglas de concordancia definidas para sintaxis complejas no suelen proporcionar más que la capacidad de correspondencia que es inmediatamente útil. En el RFC 3687 se especifican reglas genéricas de concordancia aplicables a cualesquiera partes de componente escogidas por el usuario en un valor de atributo de cualquier sintaxis de atributo arbitrariamente compleja. En el RFC 3727 se especifica un módulo ASN.1 útil como referencia en otras especificaciones. Esta regla de concordancia se importa en **SelectedAttributeTypes** con esta Especificación de directorio y puede ser escogida mediante la componente **extensibleMatch** de **FilterItem**, conforme a la Rec. UIT.T X.511 | ISO/CEI 9594-3.

7.3 Reglas de concordancia de tiempo

7.3.1 Concordancia de hora universal coordinada (UTC)

La regla de *concordancia de hora UTC (UTC Time Match)* compara la igualdad de un valor presentado con un valor de atributo del tipo **UTCTime**.

```
utCTimeMatch MATCHING-RULE ::= {
  SYNTAX UTCTime
  ID id-mr-utCTimeMatch }
```

La regla devuelve TRUE si el valor de atributo representa la misma hora que el valor presentado. Si la hora UTC se indica sin segundos, se supone que el número de segundos es cero.

7.3.2 Concordancia de ordenación de hora universal coordinada (UTC)

La regla de *concordancia de ordenación de hora UTC (UTC Time Ordering Match)* compara el ordenamiento cronológico de un valor presentado con un valor de atributo del tipo **UTCTime**.

```
utCTimeOrderingMatch MATCHING-RULE ::= {
  SYNTAX UTCTime
  ID id-mr-utCTimeOrderingMatch }
```

La regla devuelve TRUE si el valor de atributo representa una hora que es anterior a la hora presentada. Se considerará que una hora UTC con valores de año 50 a 99 es anterior a una hora UTC con valores de año 00 a 49. Si se especifica una hora UTC sin segundos, se supone que el número de segundos es cero.

El valor de año del campo de dos dígitos será racionalizado en un valor de año de cuatro dígitos como sigue:

- si el valor de dos dígitos es de 00 a 49 inclusive, se le añadirá 2000; y
- si el valor de dos dígitos es de 50 a 99 inclusive, se le añadirá 1900.

7.3.3 Concordancia de hora generalizada

La regla de *concordancia de hora generalizada (Generalized Time Match)* compara un valor presentado con un valor de atributo del tipo **GeneralizedTime** (según 42.3 b) o c) de la Rec. UIT-T X.680 | ISO/CEI 8824-1).

```
generalizedTimeMatch MATCHING-RULE ::= {
  SYNTAX GeneralizedTime
  -- as per 42.3 b) or c) of ITU-T Rec. X.680 | ISO/IEC 8824-1
  ID id-mr-generalizedTimeMatch }
```

La regla devuelve TRUE si el valor de atributo representa la misma hora que el valor presentado. Si la hora se indica sin minutos ni segundos, se supone el número de minutos o segundos es cero.

7.3.4 Concordancia de ordenamiento de hora generalizada

La regla de *concordancia ordenamiento de hora generalizada (Generalized Time Ordering Match)* compara el ordenamiento de hora de un valor presentado con un valor de atributo del tipo **GeneralizedTime** (según 42.3 b) y c) de la Rec. UIT-T X.680 | ISO/CEI 8824-1).

```
generalizedTimeOrderingMatch MATCHING-RULE ::= {
  SYNTAX GeneralizedTime
  -- as per 42.3 b) or c) of ITU-T Rec. X.680 | ISO/IEC 8824-1
  ID id-mr-generalizedTimeOrderingMatch }
```

La regla devuelve TRUE si el valor de atributo representa una hora que es anterior a la hora presentada. Si la hora se especifica sin minutos ni segundos, se supone que el número de minutos o segundos es cero.

7.3.5 Concordancia propuesta de sistemas

La regla de *concordancia propuesta de sistemas (System Proposed Match)* es una regla de concordancia simulada, definida como sigue:

```
systemProposedMatch MATCHING-RULE ::= {
  ID id-mr-systemProposedMatch }
```

La regla de concordancia puede ser incluida por un solicitante en la RelaxationPolicy en una petición **search** para indicar que el directorio debe determinar qué regla de concordancia debe utilizarse en una regla de concordancia de sustitución.

7.4 Reglas de concordancia de primer componente

7.4.1 Concordancia de primer componente de entero

La regla de *concordancia de primer componente de entero* (*Integer First Component Match*) compara un valor de entero presentado con un valor de atributo del tipo **SEQUENCE** cuyo primer componente es obligatorio y del tipo **INTEGER**.

```
integerFirstComponentMatch MATCHING-RULE ::= {
    SYNTAX INTEGER
    ID      id-mr-integerFirstComponentMatch }
```

La regla devuelve TRUE si el valor de atributo tiene un primer componente cuyo valor es igual al entero presentado.

Se deriva un valor de la sintaxis de aserción a partir del valor de la sintaxis de atributo utilizando el valor del primer componente de **SEQUENCE**.

7.4.2 Concordancia de primer componente de identificador de objeto

La regla de *concordancia de primer componente de identificador de objeto* (*Object Identifier First Component Match*) compara un valor de identificador de objeto presentado con valores de atributo del tipo **SEQUENCE** cuyo primer componente es obligatorio y del tipo **OBJECT IDENTIFIER**.

```
objectIdentifierFirstComponentMatch MATCHING-RULE ::= {
    SYNTAX OBJECT IDENTIFIER
    ID      id-mr-objectIdentifierFirstComponentMatch }
```

La regla devuelve TRUE si el valor de atributo tiene un primer componente cuyo valor concuerda con el identificador de objeto presentado utilizando las reglas de **objectIdentifierMatch**.

Un valor de la sintaxis de aserción se deriva de un valor de la sintaxis de atributo utilizando el valor del primer componente de **SEQUENCE**.

7.4.3 Concordancia de primer componente de cadena de directorio

La regla de *concordancia de primer componente de cadena de directorio* (*Directory String First Component Match*) compara un valor **DirectoryString** presentado con un valor de atributo del tipo **SEQUENCE** cuyo primer componente es obligatorio y del tipo **DirectoryString**.

```
directoryStringFirstComponentMatch MATCHING-RULE ::= {
    SYNTAX DirectoryString {ub-directory-string-first-component-match}
    ID      id-mr-directoryStringFirstComponentMatch }
```

La regla devuelve TRUE si el valor de atributo tiene un primer componente cuyo valor concuerda con **DirectoryString** presentado utilizando las reglas de **caseIgnoreMatch**.

Un valor de la sintaxis de aserción se deriva de un valor de la sintaxis de atributo utilizando el valor del primer componente de **SEQUENCE**.

7.5 Reglas de concordancia de palabra

7.5.1 Concordancia de palabra

La regla de *concordancia de palabra* (*Word Match*) compara una cadena presentada con palabras en un valor de atributo del tipo **DirectoryString**.

```
wordMatch MATCHING-RULE ::= {
    SYNTAX DirectoryString {ub-match}
    ID      id-mr-wordMatch }
```

La regla devuelve TRUE si una palabra presentada concuerda con cualquier palabra en el valor del atributo. La concordancia de palabras individuales es igual que para la regla de concordancia **caseIgnoreMatch**. La definición precisa de una "palabra" es un asunto local.

7.5.2 Concordancia de palabra clave

La regla de *concordancia de palabra clave* (*Keyword Match*) compara una cadena presentada con palabras clave en un valor de atributo de tipo **DirectoryString**.

```
keywordMatch MATCHING-RULE ::= {
    SYNTAX DirectoryString {ub-match}
    ID      id-mr-keywordMatch }
```

La regla devuelve TRUE si un valor presentado concuerda con cualquier *palabra clave* en el valor de atributo. La identificación de palabras clave en un valor de atributo y la exactitud de la concordancia son asuntos locales.

7.5.3 Concordancia de palabra general

La regla de concordancia *concordancia de palabra general* (*General Word Match*) compara palabras de una cadena presentada con palabras de un valor de atributo de tipo **DirectoryString**. La regla de concordancia se puede utilizar también para valores de atributo de un tipo que especifica explícitamente una de las opciones de **DirectoryString** como su sintaxis.

```
generalWordMatch MATCHING-RULE ::= {
    SYNTAX      SubstringAssertion
    ID          id-mr-generalWordMatch }
```

Esta regla de concordancia se diferencia de una regla de concordancia de subcadena normal por la interposición de atributos de control antes de o entre los elementos **initial**, **any**, o **final**. Si no hay atributos de control en el elemento de filtro, la concordancia se efectuará como para la regla de concordancia **caseExactSubstringsMatch** con la semántica de los elementos **initial**, **any** y **final** definidos por esa regla de concordancia. Sin embargo, si la regla de concordancia de igualdad (si existe alguna) para el tipo de atributo sometido a la concordancia es **caseIgnoreMatch**, deberá utilizarse en su lugar la **caseIgnoreSubstringsMatch**.

Se definen cuatro tipos de atributo de control para la concordancia general (las restricciones a su colocación se definen más adelante); deberán ignorarse cualesquiera otros atributos de control:

```
sequenceMatchType ATTRIBUTE ::= {
    WITH SYNTAX      SequenceMatchType
    SINGLE VALUE    TRUE
    ID               id-cat-sequenceMatchType }      -- defaulting to sequenceExact
```

```
SequenceMatchType ::= ENUMERATED {
    sequenceExact           (0),
    sequenceDeletion       (1),
    sequenceRestrictedDeletion (2),
    sequencePermutation     (3),
    sequencePermutationAndDeletion (4),
    sequenceProviderDefined (5) }
```

```
wordMatchTypes ATTRIBUTE ::= {
    WITH SYNTAX      WordMatchTypes
    SINGLE VALUE    TRUE
    ID               id-cat-wordMatchType }      -- defaulting to wordExact
```

```
WordMatchTypes ::= ENUMERATED {
    wordExact           (0),
    wordTruncated       (1),
    wordPhonetic        (2),
    wordProviderDefined (3) }
```

```
characterMatchTypes ATTRIBUTE ::= {
    WITH SYNTAX      CharacterMatchTypes
    SINGLE VALUE    TRUE
    ID               id-cat-characterMatchTypes }
```

```
CharacterMatchTypes ::= ENUMERATED {
    characterExact           (0),
    characterCaseIgnore     (1),
    characterMapped         (2) }
```

```
selectedContexts ATTRIBUTE ::= {
    WITH SYNTAX      ContextAssertion
    ID               id-cat-selectedContexts }
```

Cada atributo afecta a todos los elementos **initial**, **any** o **final** siguientes, y los valores que proporciona, anulan y reemplazan a los que eran anteriormente aplicables.

Antes del primer atributo **sequenceMatchType**, si existe alguno, se tomará **sequenceExact** como valor aplicable para el atributo **sequenceMatchType**. El atributo no afecta a la evaluación de los elementos **initial** y **final**, que se tomarán siempre para que concuerden las palabras iniciales y finales; sólo afecta a las restantes palabras no concordadas. La palabra **initial**, si está presente, concordará la primera palabra del texto almacenado. Si ambas son palabras de ruido, las dos palabras se considerarán concordantes. El posicionamiento de los atributos de **sequenceMatchType** define las palabras a las que se aplica la forma de concordancia.

NOTA 1 – En muchos casos prácticos, bastará colocar el **sequenceMatchType** antes del elemento **initial**; determinadas implementaciones pueden no sustentar la generalidad total de la definición.

Antes del primer atributo **wordMatchType**, si existe alguno, se tomará **wordExact** como valor aplicable para el atributo **wordMatchType**. Antes del primer atributo **characterMatchType** se tomará **characterExact** como valor aplicable para el atributo **characterMatchType**. Sin embargo, si la regla de concordancia de igualdad (si existe alguna) para el tipo de atributo sometido a la concordancia es **caseIgnoreMatch**, se tomará en su lugar **characterCaseIgnore**.

Si está presente el atributo de control **selectedContexts**, será el primer elemento; sólo habrá uno de dichos atributos de control; se tomará como una restricción del valor almacenado (véase más adelante).

La regla retorna TRUE si el valor presentado contiene una secuencia no vacía de palabras en la que concuerdan las palabras iniciales y finales, y además la secuencia de palabras no concordantes restantes en el valor de atributo según el **sequenceMatchType** especificado, en el que las palabras correspondientes concuerdan según los **wordMatchTypes** especificados y los caracteres correspondientes dentro de las palabras concuerdan de acuerdo con los **characterMatchTypes** especificados, salvo que si el componente **selectedContexts** está presente en el valor presentado, todos los elementos **ContextAssertion** también necesitan evaluarse a TRUE (como se especifica en la Rec. UIT-T X.501 | ISO/CEI 9594-2). La regla retorna FALSE para un determinado atributo almacenado cuando las palabras no concuerdan, o cuando no concuerda algún elemento **ContextAssertion**.

Una palabra es una secuencia no vacía de caracteres distintos de espacio limitados por el comienzo o el final de la cadena o por espacio o caracteres de puntuación. Se denominan caracteres de puntuación los que no afectan a la semántica de los testigos de palabra, y normalmente son comas, comillas, puntos al final de frases, paréntesis, etc. La determinación de los caracteres que serán caracteres de puntuación será de competencia local.

NOTA 2 – Por ejemplo, el carácter '!' se utiliza a veces en el texto para indicar un chasquido, que se utiliza en algunos idiomas africanos, por lo que a veces forma parte de una palabra en lugar de ser un signo de exclamación (que sería un carácter de puntuación).

Análogamente, la palabra **final** si está presente, concordará con la última palabra del texto almacenado; si ambas son palabras de ruido, las dos palabras se considerarán concordantes.

Las palabras de ruido, que son palabras que concuerdan con una de las palabras de la lista, definida por la implementación, de las palabras semánticamente débiles (por ejemplo, artículos y preposiciones) de acuerdo con los **characterMatchTypes** especificados se descartan de la secuencia de palabras antes de la concordancia, excepto para concordar las palabras **initial** y **final**, y la regla correspondiente en los **wordMatchTypes** se descarta de la secuencia de reglas proporcionadas y no es la última de dichas reglas.

La secuencia de palabras en el valor presentado concuerda con la secuencia de palabras del valor de atributo si este último puede transformarse de acuerdo con el **sequenceMatchType** especificado en una secuencia que contenga el mismo número de palabras que la primera secuencia y cuyas palabras correspondientes concuerden. Si **sequenceMatchType** es **sequenceExact**, la transformada deja la secuencia invariable. Si es **sequenceDeletion**, borra cero o más palabras de la secuencia. Si es **sequenceRestrictedDeletion**, borra cero o más palabras, pero no la primera palabra de la secuencia. Si es **sequencePermutation**, permuta cero o más palabras en la secuencia. Si es **sequencePermutationAndDeletion**, borra cero o más palabras en la secuencia y permuta cero o más de las palabras restantes. Si es **sequenceProviderDefined**, borra, permuta o inserta palabras de acuerdo con una regla definida por la implementación.

Una palabra en el valor presentado hace concordar una palabra en el valor de atributo si la última palabra puede transformarse de acuerdo con la regla correspondiente desde los **wordMatchTypes** especificados en una secuencia de caracteres que concuerda a su vez con los caracteres de la palabra del valor presentado. Cada palabra se hace concordar utilizando la regla correspondiente en **wordMatchTypes** donde la correspondencia es determinada previamente a la aplicación de cualesquiera supresiones o permutaciones de la concordancia de secuencia; cualesquiera palabras en exceso del número de reglas de **wordMatchTypes** se concuerda utilizando la última regla. Si la regla es exacta, la transformación deja la palabra invariable. Si es **wordTruncated**, se suprimen entonces cero más caracteres del final de la palabra, hasta una longitud de palabra mínima definida por la implementación. Si es **wordPhonetic**, la palabra se sustituye por una palabra que la concuerda de acuerdo con un algoritmo de concordancia fonética definido por la implementación. Si **wordProviderDefined**, la palabra se concuerda de acuerdo con una regla definida por la implementación.

Los caracteres de cada palabra se comparan utilizando la regla correspondiente en **characterMatchTypes**, donde la correspondencia se determina antes de aplicar cualesquiera supresiones o permutaciones de la concordancia de secuencia; los caracteres de cualesquiera palabras en exceso del número de reglas en **characterMatchTypes** se concuerdan utilizando esta última regla. Si **characterMatchTypes** es **characterExact**, los caracteres correspondientes dentro de las palabras se concuerdan si son los mismos. Si es **characterCaseIgnore**, los caracteres correspondientes dentro de las palabras se concuerdan si son los mismos cuando se ignoran las diferencias entre mayúsculas y minúsculas. Si es **characterMapped**, los caracteres concuerdan si se corresponden al mismo carácter de acuerdo con una tabla de correspondencia definida por la implementación. Esta tabla será tal que permita a los caracteres nacionales enumerados en la figura A.2/T.51 ser concordados utilizando solamente los caracteres A-Z y 0-9 en los valores presentados, y pueden hacer corresponder cortas secuencias de caracteres para que sean un solo carácter, por ejemplo ae a diptongo a-e o ue a u con diéresis (umlaut).

7.6 Reglas de concordancia aproximada

7.6.1 Concordancia de cadenas aproximada

La regla de *concordancia de cadenas aproximada* (*Approximate String Match*) compara un valor presentado con un valor de atributo de acuerdo con un algoritmo de concordancia aproximada localmente definido (por ejemplo, variaciones de deletreo, concordancia fonética, etc.). El algoritmo será el mismo que el invocado en respuesta a procesar un elemento filtro del tipo **approximateMatch** (véase la Rec. UIT-T X.511 | ISO/CEI 9594-3).

```
approximateStringMatch MATCHING-RULE ::= {  
  ID          id-mr-approximateStringMatch }
```

La sintaxis de aserción para esta regla de concordancia es la misma que la sintaxis de aserción de la regla de concordancia de igualdad para el atributo al que se aplica. Si no se define ninguna regla de concordancia de igualdad para este atributo, se permite cualquier sintaxis de aserción, pero la regla siempre evalúa a indefinido.

7.7 Reglas de concordancia especial

7.7.1 El atributo se tiene en cuenta si está ausente

La regla de *concordancia no tener en cuenta si está ausente* (*Ignore if Absent Match*) compara un valor para cualquier fin y cualquier atributo.

```
ignoreIfAbsentMatch MATCHING-RULE ::= {  
  ID          id-mr-ignoreIfAbsentMatch }
```

El resultado de la comparación será:

- Si el atributo está ausente, el valor TRUE.
- Si el atributo está presente, el valor indefinido (undefined).

Esta concordancia sólo puede utilizarse como una regla de concordancia progenitora. Se utiliza entonces en unión de una regla de concordancia que concuerda valores cuando el atributo está presente. Véase también 13.5.2 de la Rec. UIT-T X.501 | ISO/CEI 9594-2.

NOTA – Dentro de un área administrativa específica puede obtenerse el mismo efecto especificando un subcomponente **defaultValues** vacío de la petición-perfil-atributo apropiada.

7.7.2 Concordancia nula

La regla *concordancia nula* (*Null Match*) compara un valor para cualquier fin y cualquier atributo, con la regla especial.

```
nullMatch MATCHING-RULE ::= {  
  ID          id-mr-nullMatch }
```

El resultado de la comparación será:

- si el elemento de filtro es no negado, el valor TRUE; y
- si el ítem de filtro es negado, el valor FALSE.

Esta concordancia puede utilizarse formalmente para hacer que se ignore un elemento de filtro. Un elemento de filtro que utiliza concordancia nula se considerará ausente cuando se evalúa la compatibilidad con reglas de búsqueda.

7.8 Concordancia zonal

Una *concordancia zonal* es ante todo aplicable a peticiones **search** que utilizan elementos de filtro correspondientes geográficamente relacionados. Estos elementos de filtro podrían ser aserciones para **localityName**, **stateOrProvinceName**, **postalCode**, etc.

La concordancia zonal utiliza elementos de filtro combinables para la concordancia con la tabla de correspondencia.

La concordancia zonal puede tener en cuenta que la percepción de las localidades por los usuarios puede ser diferente del modelo de localidad utilizado dentro de un dominio de gestión del directorio (DMD). La correspondencia entre la percepción de los usuarios y el modelo utilizado dentro de un DMD debe tener en cuenta que un usuario puede utilizar localidades que no estén directamente recogidas en inserciones del directorio o sus nombres. Dichas localidades pueden ser confusas al corresponder exactamente a localidades más oficiales. Además, un usuario puede hacer averiguaciones ligeramente incorrectas de nombres de localidad cuando efectúa una búsqueda si el objeto buscado vive cerca de la frontera de una localidad vecina. A tal fin, una región, por ejemplo un país, se divide en *zonas*. Las zonas son áreas totalmente contenidas en una localidad referenciada en una petición **search**. El resultado de una correspondencia de los elementos de filtro correspondientes es una lista de zonas. En el anexo E figura una explicación más detallada de la correspondencia zonal.

Cuando se utiliza correspondencia zonal, la tabla de correspondencia se denomina *nomenclátor geográfico* (es decir, un diccionario geográfico). Dentro del filtro, un conjunto de elementos de filtro de localidad combinables pueden juntos definir un único *lugar denominado* (es decir, un área local única ordinariamente contigua) o, cuando está permitido, un pequeño número de lugares denominados que concuerdan con los elementos de filtro. Un lugar denominado es un lugar del mundo real concreto, como es una ciudad, un pueblo, una comarca, etc.

Un nomenclátor abarcará en general (es decir, proporcionará una base de datos geográfica correspondiente) un dominio que comprende un solo país o región. Una indagación de búsqueda geográfica deberá interpretarse en términos de un nomenclátor específico. Cómo se determina el alcance de una búsqueda y se selecciona un nomenclátor apropiado, es asunto de carácter local, pero la selección puede efectuarse utilizando un nomenclátor por defecto para el DSA, o basarse en uno o más de los atributos, por ejemplo, **countryName**, **stateOrProvinceName** o **localityName**, asociado(s) con la operación de búsqueda (por ejemplo, presente como parte del nombre distinguido del **baseObject**, o como parte del filtro).

El primer paso de una concordancia zonal es utilizar uno o más elementos de filtro para identificar uno o más lugares denominados. A tal fin, se utilizan juntos elementos de filtro de localidad combinables (es decir, todos los elementos de filtro de localidad contenidos en un solo subfiltro).

En otro caso, el procedimiento hasta ahora identifica uno o más nombres denominados. En esta fase, no se ha hecho en absoluto referencia a información contenida en el árbol de información de directorio (DIT). El resto del filtro puede utilizarse para identificar todas las inserciones contenidas en el ámbito de búsqueda que tienen posiciones correspondientes a esos lugares denominados, que se definen más adelante. La relajación puede ser aplicable para que los lugares denominados concuerden con más posiciones de inserciones si de otra forma se retornasen resultados inadecuados.

La concordancia zonal no soporta rigorización.

Cada entrada que haya de considerarse elegible para la concordancia deberá tener una posición identificada ya sea por un lugar denominado único, que utilice quizás más de un valor de nombre de lugar, por ejemplo ("Newton" "Chester" "Cheshire"), o por una o más *zonas* (véase el párrafo siguiente), representado por valores situados en un atributo de zona. Si una inserción tiene zonas para definir su posición, puede también tener valores de localidad, pero estos últimos, en este caso, son informativos. La autoridad administrativa es responsable de asegurar que la información de localidad identifica con exactitud un lugar denominado.

Las zonas son componentes geográficos primitivos no superpuestos, de distinto carácter que los lugares, por lo que un lugar se compone precisamente de una o más zonas, como se indica en el nomenclátor geográfico. Las zonas se identifican por valores de cadena únicos en una región de índice geográfico. Así, dos lugares superpuestos compartirían una o más zonas que corresponden al área de superposición. Las zonas se representan dentro de las inserciones como atributos, posiblemente como un atributo operacional. En este caso, la información zonal nunca retornaría como atributo valores, a menos que el atributo que representa la zona sea solicitado específicamente como atributo operacional. Otra posibilidad es que una zona pueda ser un atributo estándar (por ejemplo, **postalCode**). Los valores de localidad se retornan como es habitual, sujetos a control de acceso.

NOTA 1 – La naturaleza exacta de una zona, y su correspondencia hacia un atributo específico, es un asunto de carácter local, y probablemente dependería de las capacidades de una implementación específica. En el Reino Unido, un buen candidato para una zona sería un código postal, como "RG12 2JL", que a menudo define una pequeña área tal como un lado de una calle. Las zonas en las áreas urbanas serían entonces pequeñas; las de las zonas campestres serían en cambio grandes. En las áreas no pobladas y sin características de relieve (por ejemplo, desiertos), una zona podría ser ciertamente muy grande.

La posición de una inserción (definida por zonas) concuerda con un lugar denominado, definido por el nomenclátor, si existe superposición entre las zonas definidas para el lugar denominado y las zonas definidas para la inserción (es decir, se utiliza una regla de concordancia con superposición). Si la posición de la inserción se define como un lugar denominado, la posición se considera compuesta por las zonas que constituyen el lugar denominado.

La concordancia zonal permite la concordancia ampliada (es decir, relajada), donde el nivel 0 corresponde a la definición básica de los objetos en el nomenclátor. Los niveles 1 y superiores corresponden a una ampliación gradual y sistemática de las zonas que comprenden un lugar, por lo que concuerdan ubicaciones con más inserciones.

Sigue a continuación una declaración más formal de la concordancia zonal modelo subyacente:

- a) La concordancia zonal se basa en la existencia de uno o más *nomenclátor* que son soportados a tal fin por los DSA. Un nomenclátor es un diccionario geográfico que comprende, como dominio suyo, un país o una *región* denominada, soportada por una base de datos adecuada. La selección del dominio para la búsqueda específica se lleva a cabo por medios locales. El nomenclátor contiene nombres de lugares y sus características, incluidas listas de lugares denominados concordantes. Es soportado por mecanismos para encontrar y cotejar las características de los nombres de lugares indicados por atributos de localidad combinables, y es completamente independiente del DIT.
- b) La región comprendida por un nomenclátor contiene *lugares*. Un lugar es un área geográfica denominada reconocible; los lugares pueden superponerse, y pueden incluso ampliarse algo más allá del límite de la región. Los lugares que son identificables por referencia al nomenclátor se llaman *lugares denominados*.
- c) El propio índice geográfico se basa en cadenas que son *nombres de lugar*. Éstos se utilizan para identificar (o nombrar) nombres denominados. El nombre de un lugar denominado puede ser:
 - un nombre de lugar único, posiblemente en más de una palabra;
 - un conjunto de nombres de lugar, donde en general un nombre de lugar corresponde a un área más grande y califica un nombre de lugar que corresponde (en el contexto) a un área más pequeña.
- d) El concepto de áreas mayores y menores puede a veces representarse ventajosamente en la característica de escala aplicada a un lugar. Ejemplos informales de lugares de escala variable son las parcelas, puntos, pueblos, ciudades, ciudades grandes, condados, provincias, países. En general, un lugar denominado debe asociarse en el nomenclátor con los nombres de lugares de mayor escala, que lo contienen aun si éstos no son necesarios para identificación inequívoca.
- e) Los nombres de lugar pueden también tener sinónimos asociados con un lugar determinado, que podría (por ejemplo) representar abreviaturas o nombres alternativos. Es conveniente definir un nombre canónico para cada lugar con el que puedan hacerse corresponder los sinónimos de los nombres de lugar componentes.
- f) Los nombres de lugar pueden a veces derivarse de nombres de lugar más simples utilizando componentes semánticos tales como "Near" ("cerca de") (por ejemplo, "Near Tenterden"). Podría razonablemente considerarse que define un lugar en forma de anillo en torno a la ciudad de Tenterden en Kent (Inglaterra), pero probablemente sería mejor considerarlo un nombre de lugar que no define en sí un lugar.
- g) Todos los lugares comprendidos en el nomenclátor deberán tener un nombre canónico único compuesto por un conjunto distintivo de nombres de lugares, donde estos nombres pueden ordenarse en términos de la escala que cada nombre de lugar implica en el contexto.
- h) Los lugares se descomponen en zonas de manera que las zonas estén siempre insertas en cada lugar, y cada parte de un lugar tiene una zona correspondiente. Una zona es el bloque constituyente de lugares en un nomenclátor; cada punto de una región tiene una única zona en la que está contenido.
- i) Las zonas pueden tener zonas vecinas (por ejemplo, a menos que estén efectivamente delimitadas por un accidente geográfico o político de importancia, tal como un lago, río, mar, montaña o frontera de país). Por tanto, el área definida por un lugar puede ordinariamente ampliarse con la inclusión de zonas vecinas a las zonas que la contienen; la ampliación puede llevarse a cabo indefinidamente paso a paso. La inclusión de un único nivel de ampliación de vecindad se denomina ampliación 1 de un lugar. Otro nivel de ampliación es la llamada ampliación 2, y así sucesivamente. El ámbito de una ampliación puede ser localmente ajustable (ampliarse o reducirse) para representar una situación práctica, pero tales ajustes deben ser relativamente escasos.
- j) Una inserción que represente un objeto físico puede definirse que tiene una *ubicación (location)*. Una ubicación puede definirse en forma de un conjunto de zonas en un atributo de zona apropiado, o identificándolo como un lugar denominado mediante el uso de uno o más nombres de lugar utilizando un atributo de localidad tal como **locationName**, que puede también representarse como un conjunto de zonas. Una inserción concordará con un lugar si el conjunto de zonas que comprenden su localidad se

- superpone con el conjunto de zonas que representan el lugar (posiblemente con ampliación n) que es el resultado de consultar el nomenclátor, descrito más arriba.
- k) La selección de zonas, lugares, nombres de lugar y la compilación de sus relaciones es un asunto de carácter local.
 - l) Las inserciones que concordarían por concordancia de igualdad sobre la base de las cadenas que contienen continuarán concordando (en efecto, soslayando la concordancia zonal).

Para calificar aún más la concordancia zonal, se define la clase de objeto no genérica de información **ZONAL-MATCHING** como una especialización de la clase de objeto de información no genérico **MAPPING-BASED-MATCHING**. Un ejemplar de esta clase de objeto determina las características de la concordancia zonal.

ZONAL-MATCHING ::= MAPPING-BASED-MATCHING { ZonalSelect, TRUE, ZonalResult, zonalMatch.&id }

Un ejemplar de esta clase de objeto de información se caracteriza por:

- a) La referencia ficticia **&selectBy**, si está presente, es en esta clase de objeto de información sustituida por un conjunto de tipos de atributo. La selección de un ejemplar de esta clase de información se basa en los atributos y en los tipos de atributo representados en el filtro de búsqueda. Puede seleccionarse un ejemplar de objeto de información si todos los tipos de atributo representados por este componente están representados en el filtro. Los subtipos de atributo no se consideran (es decir, la selección deberá basarse en atributos denominados explícitamente). Sin embargo, pueden también tenerse en cuenta para seleccionar un ejemplar de criterios locales no definidos en esta Especificación de directorio. Por ejemplo, la selección puede ser determinada parcialmente por el **baseObject** del argumento de búsqueda. Si este componente está ausente, la selección se basa totalmente en la toma de decisión local.
- b) El **&ApplicableTo** especificará un conjunto de tipos de atributo relacionados con la localidad determinados por necesidades locales, tales como **localityName**, **stateOrProvinceName**, **streetName**, **postalCode**, etc.
- c) El componente **&subtypeIncluded** se fija de acuerdo con las necesidades locales.
- d) La referencia de valor ficticio **&combinable** es sustituida incondicionalmente por **TRUE**.
- e) La referencia de tipo ficticia **&mappingResults** es en esta clase de objeto de información sustituida por el tipo de datos **ZonalResult**.
- f) El **&userControl** se fija de acuerdo con las necesidades locales.
NOTA 2 – Este campo debe tomar en la mayoría de los casos el valor **TRUE**.
- g) El **&exclusive** se fija de acuerdo con las necesidades locales.
NOTA 3 – Un ejemplar de objeto de información de esta clase de objeto de información es un candidato a relajación exclusiva.
- h) El **&matching-rule** es para esta clase de objeto de información derivada fijado a **zonalMatch**.
- i) El **&id** da una identificación única del ejemplar del algoritmo de concordancia zonal.

El tipo de datos **ZonalSelect** es:

ZonalSelect ::= SEQUENCE OF AttributeType

El tipo de datos **ZonalResult** se utiliza para indicar condiciones de excepción para la concordancia zonal.

ZonalResult ::= ENUMERATED {
cannot-select-mapping (0),
zero-mappings (2),
multiple-mappings (3) }

Los valores:

- a) **cannot-select-mapping** es el resultado cuando la información proporcionada en el nombre de objeto de base y el subfiltro son insuficientes para identificar la correspondencia que ha de utilizarse en la regla de concordancia zonal. La concordancia correspondiente produce un resultado de indefinido. Ninguno de los subfiltros que tienen elementos de filtro correspondibles de acuerdo con la especificación **&applicableTo** evaluará consecuentemente a **TRUE**.
NOTA 4 – Dentro del área administrativa específica del servicio y para reglas de búsqueda adecuadamente designadas, el análisis del argumento de búsqueda debe haber detectado información insuficiente en el argumento de búsqueda.
- b) **zero-mappings** es el resultado cuando la información proporcionada en el elemento (o elementos) de filtro que ha de hacerse corresponder no puede hacerlo, ya sea porque no existe ningún elemento correspondiente en la tabla de correspondencia, o porque el proceso de correspondencia produjo cero

elementos de filtro para su correspondencia con inserciones. En esta situación deberá retornarse un **serviceError** con el problema **requestedServiceNotAvailable**. La notificación **component** de **CommonResults** contendrá:

- i) un atributo de notificación **searchServiceProblem** con el valor **id-pr-unmatchedKeyAttributes**; y
 - ii) un atributo de notificación **filterItem** que indica los elementos de filtro correspondientes incapaces de ofrecer una concordancia.
- c) **multiple-mappings** es el resultado cuando la información proporcionada en el elemento (o elementos) de filtro puede hacerse corresponder con múltiples inserciones del nomenclátor. La concordancia correspondiente proporciona un valor TRUE, pero puede, no obstante, hacer que la búsqueda sea abandonada con un error. En esta situación deberá retornarse un **serviceError** con el problema **requestedServiceNotAvailable**. La notificación **component** de **CommonResults** contendrá:
- i) un atributo de notificación **searchServiceProblem** con el valor **id-pr-ambiguousKeyAttributes**; y
 - ii) un atributo de notificación **multipleMatchingLocalities** indicado por la regla de concordancia **zonalMatch**.

La regla de concordancia **zonalMatch** es la regla de concordancia por correspondencia asociada con un ejemplar de la clase de objeto de información **ZONAL-MATCHING**.

```
zonalMatch MATCHING-RULE ::= {
    UNIQUE-MATCH-INDICATOR      multipleMatchingLocalities
    ID                            id-mr-zonalMatch }
```

Esta regla de concordancia por correspondencia incluye el campo **UNIQUE-MATCH-INDICATOR**, que implica que la concordancia con el nomenclátor dará un resultado inequívoco. Si varias inserciones en la tabla concuerdan en el proceso de correspondencia, se retorna un **serviceError** con el problema **ambiguousKeyAttributes**. El componente **notificación** de **CommonResults** debe contener el atributo de notificación **multipleMatchingLocalities** (véase 5.12.14). Se incluye un valor del atributo de notificación **multipleMatchingLocalities** para cada inserción en la tabla con concordancia con el nomenclátor. Cada uno de estos valores será una especificación de conjunto de **AttributeValueAssertion** que si es suministrada en elementos de filtro **equality** seguidos por AND producirían una concordancia única con la inserción en la tabla correspondiente. Esto permitirá al usuario en una petición de **búsqueda** posterior seleccionar uno de los valores de atributo de notificación retornados que han de reflejarse en el filtro.

SECCIÓN 4 – CONTEXTOS

8 Definición de tipos de contexto

Esta Especificación de directorio define un número de tipos de contexto que pueden ser útiles en toda una gama de aplicaciones de directorio.

8.1 Contexto de lenguaje

El *contexto de lenguaje* asocia un valor de atributo con un determinado lenguaje:

```
languageContext CONTEXT ::= {
    WITH SYNTAX      LanguageContextSyntax
    ID                id-avc-language }
```

LanguageContextSyntax ::= PrintableString (SIZE(2..3)) -- ISO 639-2 codes only

Se considera que un valor presentado concuerda con un valor almacenado si la secuencia de caracteres en el valor presentado es idéntica a la del valor almacenado.

8.2 Contexto temporal

El *contexto temporal* asocia un valor de atributo con un conjunto de tiempos. Son posibles diversas expresiones de tiempo, entre ellas:

- a) tiempo absoluto de inicio o fin (por ejemplo, 24:00, 14 de diciembre de 1994);
- b) bandas de tiempo específicas en el día (por ejemplo, 09:00 a 17:00);
- c) días de la semana (por ejemplo, lunes);
- d) días del mes (por ejemplo, el décimo, el penúltimo día, etc.);
- e) meses del año (por ejemplo, marzo);
- f) un año determinado (por ejemplo, 1995);
- g) semanas del mes (por ejemplo, la segunda semana);
- h) periodicidad diaria o semanal (por ejemplo, en semanas alternas);
- i) negativos lógicos (por ejemplo, lunes, no).

```
temporalContext CONTEXT ::= {
    WITH SYNTAX      TimeSpecification
    ASSERTED AS     TimeAssertion
    ID                id-avc-temporal }
```

```
TimeSpecification ::= SEQUENCE {
    time              CHOICE {
        absolute      SEQUENCE {
            startTime [0] GeneralizedTime OPTIONAL,
            endTime   [1] GeneralizedTime OPTIONAL },
        periodic      SET OF Period },
    notThisTime      BOOLEAN DEFAULT FALSE,
    timeZone         TimeZone OPTIONAL }
```

```
Period ::= SEQUENCE {
    timesOfDay [0] SET SIZE (1..MAX) OF DayTimeBand OPTIONAL,
    days       [1] CHOICE {
        intDay SET OF INTEGER,
        bitDay BIT STRING { sunday (0), monday (1), tuesday (2),
            wednesday (3), thursday (4), friday (5), saturday (6) },
        dayOf XDayOf } OPTIONAL,
    weeks     [2] CHOICE {
        allWeeks NULL,
        intWeek  SET OF INTEGER,
        bitWeek  BIT STRING { week1 (0), week2 (1), week3 (2), week4 (3),
            week5 (4) } } OPTIONAL,
```

```

months      [3] CHOICE {
              allMonths  NULL,
              intMonth   SET OF INTEGER,
              bitMonth   BIT STRING { january (0), february (1), march (2), april (3),
                                   may (4), june (5), july (6), august (7), september (8),
                                   october (9), november (10), december (11) }
            } OPTIONAL,
years       [4] SET OF INTEGER (1000 .. MAX) OPTIONAL }

```

```

XDayOf ::= CHOICE {
  first   [1] NamedDay,
  second  [2] NamedDay,
  third   [3] NamedDay,
  fourth  [4] NamedDay,
  fifth   [5] NamedDay }

```

```

NamedDay ::= CHOICE {
  intNamedDays  ENUMERATED {
    sunday (1),
    monday (2),
    tuesday (3),
    wednesday (4),
    thursday (5),
    friday (6),
    saturday (7) },
  bitNamedDays  BIT STRING { sunday (0), monday (1), tuesday (2),
                             wednesday (3), thursday (4), friday (5), saturday (6) } }

```

```

DayTimeBand ::= SEQUENCE {
  startDayTime [0] DayTime DEFAULT { hour 0 },
  endDayTime  [1] DayTime DEFAULT { hour 23, minute 59, second 59 } }

```

```

DayTime ::= SEQUENCE {
  hour   [0] INTEGER (0..23),
  minute [1] INTEGER (0..59) DEFAULT 0,
  second [2] INTEGER (0..59) DEFAULT 0 }

```

```

TimeZone ::= INTEGER (-12..12)

```

```

TimeAssertion ::= CHOICE {
  now      NULL,
  at       GeneralizedTime,
  between SEQUENCE {
    startTime [0] GeneralizedTime,
    endTime  [1] GeneralizedTime OPTIONAL,
    entirely  BOOLEAN DEFAULT FALSE } }

```

La elección de **tiempo absoluto** (**absolute time**) expresa un tiempo o banda temporal determinados que utilizan anotaciones de tiempo absolutas (**GeneralizedTime**). Un tiempo específico se expresa fijando el **tiempo de inicio** (**startTime**) igual al **tiempo de fin** (**endTime**). En otros casos el **startTime** es anterior en el tiempo al **endTime** y se expresa un periodo de tiempo. Si falta el **endTime**, el periodo de tiempo incluye todos los tiempos después del **startTime**.

La expresión **periodicidad** (**periodic**) permite la especificación de tiempo como un conjunto de periodos. El efecto combinado es una O lógica del conjunto.

NOTA 1 – Podría ocurrir también que un valor de atributo esté asociado con el contexto temporal con múltiples valores de contexto, uno para cada periodo, ya que esto actúa también como una O lógica. No obstante, se incluye aquí el CONJUNTO DE (SET OF) para que el **esta vez no** (**notThisTime**) abarque el conjunto y, por consiguiente, efectúe un "ninguno" lógico. Cuando **notThisTime** es FALSE, la elección del método que se debe utilizar para especificar un conjunto de periodos depende del especificador.

Dentro de cada **periodo** (**Period**), cada elemento en la SECUENCIA DE (SEQUENCE OF) se considera "dentro" del elemento siguiente en la SEQUENCE OF. La SEQUENCE OF está en orden creciente de granularidad de periodo de tiempo, aunque no todos los niveles puedan estar presentes.

Se supone que el elemento final en un **periodo** (**Period**) es válido para todos los periodos de tiempo de granularidad más alta.

NOTA 2 – Por ejemplo, si una SEQUENCE OF **periodo** (**Period**) termina con **horas del día** (**timesOfDay**), se considera válido para todos los días.

Un **timesOfDay** indica las bandas temporales válidas durante los días especificados en el elemento siguiente de **periodo (Period)**. Si **días (days)** no es el elemento siguiente, las bandas temporales son válidas para todos los días posibles dentro del elemento siguiente. Si no se incluye **timesOfDay** todas las horas del día son válidas dentro del elemento siguiente. Pueden especificarse distintas bandas temporales para días distintos, teniendo múltiples apariciones de **periodo (Period)**.

El elemento **días (days)** expresa días determinados de una semana, mes o año según el elemento siguiente de **periodo (Period)**. Si **días (days)** precede **semanas (weeks)** en un **periodo (Period)**, expresa entonces los días de la semana y los ENTEROS (**INTEGER**) están limitados a los valores 1 a 7, donde 1 es domingo. Si **días (days)** precede **meses (months)** en un **periodo (Period)**, expresa entonces los días del mes y los **INTEGER** están limitados a los valores 1 a 31, donde 1 es el primer día del mes. Si **días (days)** precede **años (years)** en un **periodo (Period)**, expresa entonces los días del año y los **INTEGER** están limitados a los valores 1 a 366, donde 1 es el primer día del año.

La anotación **día de (dayOf)** se utiliza para indicar la primera, segunda, tercera y quinta aparición del **día señalado (NamedDay)** en un mes, (por ejemplo, el primer lunes del mes, o el segundo martes y viernes de agosto). El uso de **quinto (fifth)** indicará siempre el último **NamedDay** de ese mes (por ejemplo, el último martes de julio). Si se especifica la elección **DayOf** para **días (days)**, el elemento **semanas (weeks)** de **periodo (Period)** no es significativo si está presente y se ignora.

Si no se especifica **días (days)**, todos los días son válidos dentro del elemento siguiente del **periodo (Period)**.

El elemento **semanas (weeks)** expresa determinadas semanas de un mes o de un año, según el elemento siguiente de **periodo (Period)**. Si **semanas (weeks)** precede a meses en un **periodo (Period)**, expresa entonces las semanas del mes y los **INTEGER** están limitados a los valores 1 a 5, donde 1 es la primera semana del mes. Se supone que la primera semana del mes es la primera semana que contiene por lo menos cuatro días de ese mes. La quinta semana significa siempre la última semana del mes.

Si **semanas (weeks)** precede a años en un **periodo (Period)**, expresa entonces las semanas del año y los **INTEGER** están limitados a los valores 1 a 53, donde 1 es la primera semana del año. Se supone que la primera semana del año es la primera semana que contiene por lo menos cuatro días de ese año. La semana quincuagésima tercera es siempre la última semana del año.

Si se especifica **todas las semanas (allWeeks)**, todas las semanas son válidas dentro del elemento siguiente del **periodo (Period)** (de este modo **días (days)** puede expresar los días de la semana para todas las semanas).

Si no se especifica **semanas (weeks)**, todas las semanas son válidas dentro del elemento siguiente del **periodo (Period)**.

El elemento **meses (months)** expresa determinados meses del año. Cuando se expresa **meses (months)** con **INTEGER**, éstos están limitados a los valores 1 a 12 donde 1 es el primer mes del año (es decir, enero).

Si se especifica **todos los meses (allMonths)**, todos los meses del año son válidos (de este modo, **semanas (weeks)** puede expresar las semanas del mes para todos los meses, o si no se especifica **semanas (weeks)**, **días (days)** puede expresar los días del mes para todos los meses).

Si no se especifica **meses (months)**, todos los meses del año son válidos.

El componente **años (years)** expresa uno o más años. Si no se especifica **años (years)**, todos los años son válidos.

La anotación **zona de tiempo (timeZone)** expresa la zona de tiempo, en delta horas de GMT, en las cuales se expresa **tiempo (time)**. Si **timeZone** no está presente, un agente de sistema de directorio (DSA) que procesa el contexto temporal interpretará el **tiempo (time)** pertinente en la zona de tiempo del DSA.

Si **notThisTime** es **FALSE**, el valor de contexto temporal es el tiempo expresado en **tiempo (time)** en la **especificación de tiempo (TimeSpecification)**. Si **notThisTime** es **TRUE**, se considera que el valor de contexto temporal es todo el tiempo excepto el expresado en **tiempo (time)** en **TimeSpecification** (es decir, se efectúa un NO lógico).

Se considera que una aserción de tiempo concuerda con una especificación de tiempo si existe una superposición en los tiempos especificados. Si la aserción de tiempo contiene **ahora (now)**, se utiliza el tiempo actual en la evaluación. Si se especifica **ahora o a (now o at)**, la aserción se considera verdadera y el tiempo determinado cae dentro de los tiempos abarcados por la **TimeSpecification** almacenada. Si la aserción de tiempo utiliza **entre y totalmente (between y entirely)** es **FALSE**, la aserción se considera verdadera si alguna parte de la banda temporal **entre (between)** cae dentro de los tiempos abarcados por la **TimeSpecification** almacenada (no es necesario que la superposición sea completa: se considera que concuerdan mientras haya un periodo de superposición dentro de las dos especificaciones de tiempo). Si

la aserción de tiempo utiliza **entre** y **totalmente (between y entirely)** es **TRUE**, la aserción se considera verdadera sólo si la banda temporal **entre (between)** completa cae dentro de los tiempos abarcados por la **TimeSpecification** almacenada.

Ejemplos

NOTA 3 – Los siguientes ejemplos utilizan los formatos **INTEGER** para elementos donde se puede elegir entre **INTEGER** o **BIT STRING**.

- a) De 09.00 a 17.00 todos los días podría expresarse como:

```
periodic {
  timesOfDay { {
    startDayTime hour 9,
    endDayTime hour 17 } } }
```

- b) Todos los lunes podría expresarse como:

```
periodic {
  days intDay : {2} }
```

- c) De 09.00 a 12.00 de la mañana de lunes a viernes y todos los sábados durante enero, y todos los días martes en febrero y marzo podría expresarse como:

```
periodic {
  timesOfDay { {
    startDayTime hour 9,
    endDayTime hour 12 } }
  days intDay : {2,3,4,5,6},
  weeks allWeeks : NULL,
  months intMonth : {1} },
  { days {7},
    weeks {1,2,3,4,5},
    months {1} },
  { days {3},
    weeks {1,2,3,4,5},
    months {2,3} } }
```

- d) Todo el mes de agosto de 1996 podría expresarse como:

```
periodic {
  { months {8},
    years {1996} } }
```

- e) El primer día de cada mes podría expresarse como:

```
periodic {
  { days {1},
    months NULL } }
```

8.3 Contexto local

El *contexto local* asocia un valor de atributo con una localización determinada como se define en POSIX:

```
localeContext CONTEXT ::= {
  WITH SYNTAX LocaleContextSyntax
  ID id-avc-locale }
```

```
LocaleContextSyntax ::= CHOICE {
  localeID1OBJECT IDENTIFIER,
  localeID2DirectoryString {ub-localeContextSyntax} }
```

Se considera que un valor presentado concuerda con un valor almacenado si ambos son identificadores de objeto y si éstos son iguales, o si ambos son cadenas y éstas son las mismas.

Sólo pueden utilizarse como valores de contexto los identificadores de objeto registrados o las cadenas para localizaciones. El concepto de localización se describe en ISO/CEI 9945-2:2003 *Information Technology – Portable Operating System Interface (POSIX) – Part 2: Shell and Utilities*.

NOTA – Se crearán autoridades de registro para asignar identificadores de objeto y/o identificadores de cadena para especificaciones de localizaciones. Por ejemplo, el Comité Europeo de Normalización (CEN, *European Committee for Standardization*) ha publicado una Norma Europea para el Registro de Información sobre Localizaciones, ENV12005:1996, *Procedimientos para el Registro Europeo de Elementos Culturales*.

8.4 Contexto opción de atributo LDAP

El *contexto opción de atributo LDAP* se utiliza para alinear los contextos X.500 con las opciones de atributo LDAP.

```

IdapAttributeOptionContext CONTEXT ::= {
    WITH SYNTAX      AttributeOptionList
    ASSERTED AS     AttributeOptionList
    ABSENT-MATCH    FALSE
    ID              id-avc-ldapAttributeOption }
    
```

AttributeOptionList ::= SEQUENCE OF UTF8String

Una lista de opciones de valor de contexto permite la correspondencia más cercana y natural del concepto de contexto definido en esta Especificación de directorio y las opciones de atributo LDAP. Se hace corresponder cada opción de atributo de subtipo LDAP con un solo valor **UTF8String** de la lista. Dos valores **IdapAttributeOptionContext** serán iguales si contienen la misma lista de cadenas, en cualquier orden, utilizando una comparación insensible a mayúsculas/minúsculas. Una **AttributeOptionList** en una **ContextAssertion** corresponde a una **AttributeOptionList** en un valor de contexto almacenado si es igual a la lista almacenada o a un subconjunto de ésta, ignorando la diferencia entre mayúsculas y minúsculas y el orden de las opciones.

NOTA 1 – La **AttributeOptionList** se implementa como una SECUENCIA DE (**SEQUENCE OF**) para simplificar la codificación DER.

NOTA 2 – Las opciones de atributo LDAP están restringidas a los caracteres 'A' a 'Z', 'a' a 'z', '0' a '9' y al guion, por lo que bastaría con emplear **PrintableString** en lugar de **UTF8String**. No obstante, el conjunto de caracteres subyacente para las opciones de atributo es UTF8 y una extensión futura de LDAP debe utilizar este repertorio más amplio. Se optó entonces por **UTF8String** con miras a adecuar la especificación a posibles cambios futuros.

Se permite específicamente que haya **AttributeOptionList** vacías. En LDAP, un determinado valor puede estar presente simultáneamente en el atributo de base y en cualquiera de sus subtipos opcionales, por ejemplo (en formato LDIF):

```

description: This is a string
description;lang-en: This is a string
description;lang-en;lang-en-us: This is a string
    
```

En la Rec. UIT-T X.500 | ISO/CEI 9594-1 este ejemplo se representaría mediante un solo valor, "This is a string", con un solo contexto que tiene **contextType id-avc-ldapAttributeOption**, y tres **contextValues**: { }, { "lang-en" } and { "lang-en", "lang-en-us" }. Es decir, una **AttributeOptionList** vacía, una **AttributeOptionList** que contiene el único valor "lang-en" y una **AttributeOptionList** en la que hay dos valores, a saber "lang-en" y "lang-en-us".

Anexo A

Tipos de atributos seleccionados en ASN.1

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

Este anexo comprende todas las definiciones de tipo y valor ASN.1 contenidas en esta Especificación de directorio en forma del módulo ASN.1 **SelectedAttributeTypes**.

SelectedAttributeTypes {joint-iso-itu-t ds(5) module(1) selectedAttributeTypes(5) 5}

DEFINITIONS ::=

BEGIN

-- EXPORTS All --

-- The types and values defined in this module are exported for use in the other ASN.1 modules contained within the Directory Specifications, and for the use of other applications which will use them to access Directory services. Other applications may use them for their own purposes, but this will not constrain extensions and modifications needed to maintain or improve the Directory service.

IMPORTS

-- from ITU-T Rec. X.501 | ISO/IEC 9594-2

directoryAbstractService, id-at, id-avc, id-cat, id-mr, id-not, id-pr, informationFramework, serviceAdministration, upperBounds

FROM UsefulDefinitions {joint-iso-itu-t ds(5) module(1) usefulDefinitions(0) 5 }

Attribute, ATTRIBUTE, AttributeType, AttributeValueAssertion, CONTEXT, ContextAssertion, DistinguishedName, distinguishedNameMatch, MAPPING-BASED-MATCHING{ }, MATCHING-RULE, OBJECT-CLASS, objectIdentifierMatch

FROM InformationFramework informationFramework

AttributeCombination, ContextCombination, MRMapping

FROM ServiceAdministration serviceAdministration

-- from ITU-T Rec. X.511 | ISO/IEC 9594-3

FilterItem, HierarchySelections, SearchControlOptions, ServiceControlOptions

FROM DirectoryAbstractService directoryAbstractService

-- from ITU-T Rec. X.520 | ISO/IEC 9594-6

ub-answerback, ub-business-category, ub-common-name, ub-country-code, ub-description, ub-destination-indicator, ub-directory-string-first-component-match, ub-international-isdn-number, ub-knowledge-information, ub-localeContextSyntax, ub-locality-name, ub-match, ub-name, ub-organization-name, ub-organizational-unit-name, ub-physical-office-name, ub-postal-code, ub-postal-line, ub-postal-string, ub-post-office-box, ub-pseudonym, ub-serial-number, ub-state-name, ub-street-address, ub-surname, ub-telephone-number, ub-telex-number, ub-teletex-terminal-id, ub-title, ub-user-password, ub-x121-address

FROM UpperBounds upperBounds

-- from ITU-T Rec. X.411 | ISO/IEC 10021-4

G3FacsimileNonBasicParameters

FROM MTSAbstractService{joint-iso-itu-t mhs(6) mts(3) modules(0) mts-abstract-service(1) version-1999(1) } ;

*/*from IETF RFC 3727*

The following import is provided for information only (see 7.2.16), it is not referenced by any ASN.1 construct within these Directory Specifications. Note that the ASN.1 module in RFC 3727 imports from the InformationFramework module of edition 4 of ITU-T Rec. X.501 | ISO/IEC 9594-2. A specification importing from both these Directory Specifications and from RFC 3727 should take corrective actions, e.g., by making a copy of the ASN.1 module of RFC 3727 and then update the IMPORT statement.

allComponentsMatch, componentFilterMatch, directoryComponentsMatch, presentMatch, rdnMatch
FROM ComponentMatching {iso(1) 2 36 79672281 xed(3) module (0) component-matching(4)} */

-- Directory string type --

```
DirectoryString { INTEGER : maxSize } ::= CHOICE {
    teletexString TeletexString (SIZE (1..maxSize)),
    printableString PrintableString (SIZE (1..maxSize)),
    bmpString BMPString (SIZE (1..maxSize)),
    universalString UniversalString (SIZE (1..maxSize)),
    uTF8String UTF8String (SIZE (1..maxSize)) }
```

-- Attribute types --

```
knowledgeInformation ATTRIBUTE ::= {
    WITH SYNTAX DirectoryString {ub-knowledge-information}
    EQUALITY MATCHING RULE caseIgnoreMatch
    ID id-at-knowledgeInformation }
```

```
name ATTRIBUTE ::= {
    WITH SYNTAX DirectoryString {ub-name}
    EQUALITY MATCHING RULE caseIgnoreMatch
    SUBSTRINGS MATCHING RULE caseIgnoreSubstringsMatch
    ID id-at-name }
```

```
commonName ATTRIBUTE ::= {
    SUBTYPE OF name
    WITH SYNTAX DirectoryString {ub-common-name}
    ID id-at-commonName }
```

```
surname ATTRIBUTE ::= {
    SUBTYPE OF name
    WITH SYNTAX DirectoryString {ub-surname}
    ID id-at-surname }
```

```
givenName ATTRIBUTE ::= {
    SUBTYPE OF name
    WITH SYNTAX DirectoryString {ub-name}
    ID id-at-givenName }
```

```
initials ATTRIBUTE ::= {
    SUBTYPE OF name
    WITH SYNTAX DirectoryString {ub-name}
    ID id-at-initials }
```

```
generationQualifier ATTRIBUTE ::= {
    SUBTYPE OF name
    WITH SYNTAX DirectoryString {ub-name}
    ID id-at-generationQualifier }
```

```
uniqueIdentifier ATTRIBUTE ::= {
    WITH SYNTAX UniqueIdentifier
    EQUALITY MATCHING RULE bitStringMatch
    ID id-at-uniqueIdentifier }
```

UniqueIdentifier ::= BIT STRING

```
dnQualifier ATTRIBUTE ::= {
    WITH SYNTAX PrintableString
    EQUALITY MATCHING RULE caseIgnoreMatch
    ORDERING MATCHING RULE caseIgnoreOrderingMatch
    SUBSTRINGS MATCHING RULE caseIgnoreSubstringsMatch
    ID id-at-dnQualifier }
```

```
serialNumber ATTRIBUTE ::= {
    WITH SYNTAX PrintableString (SIZE (1..ub-serial-number))
    EQUALITY MATCHING RULE caseIgnoreMatch
    SUBSTRINGS MATCHING RULE caseIgnoreSubstringsMatch
    ID id-at-serialNumber }
```

```

pseudonym ATTRIBUTE ::= {
    SUBTYPE OF      name
    WITH SYNTAX     DirectoryString {ub-pseudonym}
    ID              id-at-pseudonym }

uUIDPair ATTRIBUTE ::= {
    WITH SYNTAX     UUIDPair
    EQUALITY MATCHING RULE  uUIDPairMatch
    ID              id-at-uuidpair }

UUIDPair ::= SEQUENCE {
    issuerUUID      UUID,
    subjectUUID     UUID }

UUID ::= OCTET STRING (SIZE(16))      -- UUID format only

countryName ATTRIBUTE ::= {
    SUBTYPE OF      name
    WITH SYNTAX     CountryName
    SINGLE VALUE    TRUE
    ID              id-at-countryName }

CountryName ::= PrintableString (SIZE(2))      -- ISO 3166 codes only

localityName ATTRIBUTE ::= {
    SUBTYPE OF      name
    WITH SYNTAX     DirectoryString {ub-locality-name}
    ID              id-at-localityName }

collectiveLocalityName ATTRIBUTE ::= {
    SUBTYPE OF      localityName
    COLLECTIVE      TRUE
    ID              id-at-collectiveLocalityName }

stateOrProvinceName ATTRIBUTE ::= {
    SUBTYPE OF      name
    WITH SYNTAX     DirectoryString {ub-state-name}
    ID              id-at-stateOrProvinceName }

collectiveStateOrProvinceName ATTRIBUTE ::= {
    SUBTYPE OF      stateOrProvinceName
    COLLECTIVE      TRUE
    ID              id-at-collectiveStateOrProvinceName }

streetAddress ATTRIBUTE ::= {
    WITH SYNTAX     DirectoryString {ub-street-address}
    EQUALITY MATCHING RULE  caseIgnoreMatch
    SUBSTRINGS MATCHING RULE  caseIgnoreSubstringsMatch
    ID              id-at-streetAddress }

collectiveStreetAddress ATTRIBUTE ::= {
    SUBTYPE OF      streetAddress
    COLLECTIVE      TRUE
    ID              id-at-collectiveStreetAddress }

houseIdentifier ATTRIBUTE ::= {
    WITH SYNTAX     DirectoryString {ub-name}
    EQUALITY MATCHING RULE  caseIgnoreMatch
    SUBSTRINGS MATCHING RULE  caseIgnoreSubstringsMatch
    ID              id-at-houseIdentifier }

organizationName ATTRIBUTE ::= {
    SUBTYPE OF      name
    WITH SYNTAX     DirectoryString {ub-organization-name}
    ID              id-at-organizationName }

collectiveOrganizationName ATTRIBUTE ::= {
    SUBTYPE OF      organizationName
    COLLECTIVE      TRUE
    ID              id-at-collectiveOrganizationName }

```

```

organizationalUnitName ATTRIBUTE ::= {
    SUBTYPE OF      name
    WITH SYNTAX     DirectoryString {ub-organizational-unit-name}
    ID              id-at-organizationalUnitName }

collectiveOrganizationalUnitName ATTRIBUTE ::= {
    SUBTYPE OF      organizationalUnitName
    COLLECTIVE      TRUE
    ID              id-at-collectiveOrganizationalUnitName }

title ATTRIBUTE ::= {
    SUBTYPE OF      name
    WITH SYNTAX     DirectoryString {ub-title}
    ID              id-at-title }

description ATTRIBUTE ::= {
    WITH SYNTAX     DirectoryString {ub-description}
    EQUALITY MATCHING RULE      caseIgnoreMatch
    SUBSTRINGS MATCHING RULE    caseIgnoreSubstringsMatch
    ID              id-at-description }

searchGuide ATTRIBUTE ::= {
    WITH SYNTAX     Guide
    ID              id-at-searchGuide }

Guide ::= SET {
    objectClass [0] OBJECT-CLASS.&id OPTIONAL,
    criteria    [1] Criteria }

Criteria ::= CHOICE {
    type      [0] Criterialtem,
    and       [1] SET OF Criteria,
    or        [2] SET OF Criteria,
    not       [3] Criteria }

Criterialtem ::= CHOICE {
    equality      [0] AttributeType,
    substrings   [1] AttributeType,
    greaterOrEqual [2] AttributeType,
    lessOrEqual  [3] AttributeType,
    approximateMatch [4] AttributeType }

enhancedSearchGuide ATTRIBUTE ::= {
    WITH SYNTAX     EnhancedGuide
    ID              id-at-enhancedSearchGuide }

EnhancedGuide ::= SEQUENCE {
    objectClass [0] OBJECT-CLASS.&id,
    criteria    [1] Criteria,
    subset      [2] INTEGER
        { baseObject (0), oneLevel (1), wholeSubtree (2) } DEFAULT oneLevel }

businessCategory ATTRIBUTE ::= {
    WITH SYNTAX     DirectoryString {ub-business-category}
    EQUALITY MATCHING RULE      caseIgnoreMatch
    SUBSTRINGS MATCHING RULE    caseIgnoreSubstringsMatch
    ID              id-at-businessCategory }

postalAddress ATTRIBUTE ::= {
    WITH SYNTAX     PostalAddress
    EQUALITY MATCHING RULE      caseIgnoreListMatch
    SUBSTRINGS MATCHING RULE    caseIgnoreListSubstringsMatch
    ID              id-at-postalAddress }

PostalAddress ::= SEQUENCE SIZE(1..ub-postal-line) OF DirectoryString {ub-postal-string}

collectivePostalAddress ATTRIBUTE ::= {
    SUBTYPE OF      postalAddress
    COLLECTIVE      TRUE
    ID              id-at-collectivePostalAddress }

```

```

postalCode ATTRIBUTE ::= {
    WITH SYNTAX                DirectoryString {ub-postal-code}
    EQUALITY MATCHING RULE     caselgnoreMatch
    SUBSTRINGS MATCHING RULE  caselgnoreSubstringsMatch
    ID                          id-at-postalCode }

collectivePostalCode ATTRIBUTE ::= {
    SUBTYPE OF                 postalCode
    COLLECTIVE                 TRUE
    ID                          id-at-collectivePostalCode }

postOfficeBox ATTRIBUTE ::= {
    WITH SYNTAX                DirectoryString {ub-post-office-box}
    EQUALITY MATCHING RULE     caselgnoreMatch
    SUBSTRINGS MATCHING RULE  caselgnoreSubstringsMatch
    ID                          id-at-postOfficeBox }

collectivePostOfficeBox ATTRIBUTE ::= {
    SUBTYPE OF                 postOfficeBox
    COLLECTIVE                 TRUE
    ID                          id-at-collectivePostOfficeBox }

physicalDeliveryOfficeName ATTRIBUTE ::= {
    WITH SYNTAX                DirectoryString {ub-physical-office-name}
    EQUALITY MATCHING RULE     caselgnoreMatch
    SUBSTRINGS MATCHING RULE  caselgnoreSubstringsMatch
    ID                          id-at-physicalDeliveryOfficeName }

collectivePhysicalDeliveryOfficeName ATTRIBUTE ::= {
    SUBTYPE OF                 physicalDeliveryOfficeName
    COLLECTIVE                 TRUE
    ID                          id-at-collectivePhysicalDeliveryOfficeName }

telephoneNumber ATTRIBUTE ::= {
    WITH SYNTAX                TelephoneNumber
    EQUALITY MATCHING RULE     telephoneNumberMatch
    SUBSTRINGS MATCHING RULE  telephoneNumberSubstringsMatch
    ID                          id-at-telephoneNumber }

TelephoneNumber ::= PrintableString (SIZE(1..ub-telephone-number))
    -- String complying with ITU-T Rec. E.123 only

collectiveTelephoneNumber ATTRIBUTE ::= {
    SUBTYPE OF                 telephoneNumber
    COLLECTIVE                 TRUE
    ID                          id-at-collectiveTelephoneNumber }

telexNumber ATTRIBUTE ::= {
    WITH SYNTAX                TelexNumber
    ID                          id-at-telexNumber }

TelexNumber ::= SEQUENCE {
    telexNumber    PrintableString (SIZE (1..ub-telex-number)),
    countryCode   PrintableString (SIZE (1..ub-country-code)),
    answerback    PrintableString (SIZE (1..ub-answerback)) }

collectiveTelexNumber ATTRIBUTE ::= {
    SUBTYPE OF                 telexNumber
    COLLECTIVE                 TRUE
    ID                          id-at-collectiveTelexNumber }

facsimileTelephoneNumber ATTRIBUTE ::= {
    WITH SYNTAX                FacsimileTelephoneNumber
    EQUALITY MATCHING RULE     facsimileNumberMatch
    SUBSTRINGS MATCHING RULE  facsimileNumberSubstringsMatch
    ID                          id-at-facsimileTelephoneNumber }

FacsimileTelephoneNumber ::= SEQUENCE {
    telephoneNumber    TelephoneNumber,
    parameters          G3FacsimileNonBasicParameters OPTIONAL }

```

collectiveFacsimileTelephoneNumber ATTRIBUTE ::= {
 SUBTYPE OF facsimileTelephoneNumber
 COLLECTIVE TRUE
 ID id-at-collectiveFacsimileTelephoneNumber }

x121Address ATTRIBUTE ::= {
 WITH SYNTAX X121Address
 EQUALITY MATCHING RULE numericStringMatch
 SUBSTRINGS MATCHING RULE numericStringSubstringsMatch
 ID id-at-x121Address }

X121Address ::= NumericString (SIZE(1..ub-x121-address))
-- String as defined by ITU-T Rec. X.121

internationalISDNNumber ATTRIBUTE ::= {
 WITH SYNTAX InternationalISDNNumber
 EQUALITY MATCHING RULE numericStringMatch
 SUBSTRINGS MATCHING RULE numericStringSubstringsMatch
 ID id-at-internationalISDNNumber }

InternationalISDNNumber ::= NumericString (SIZE(1..ub-international-isdn-number))
-- String complying with ITU-T Rec. E.164 only

collectiveInternationalISDNNumber ATTRIBUTE ::= {
 SUBTYPE OF internationalISDNNumber
 COLLECTIVE TRUE
 ID id-at-collectiveInternationalISDNNumber }

registeredAddress ATTRIBUTE ::= {
 SUBTYPE OF postalAddress
 WITH SYNTAX PostalAddress
 ID id-at-registeredAddress }

destinationIndicator ATTRIBUTE ::= {
 WITH SYNTAX DestinationIndicator
 EQUALITY MATCHING RULE caseIgnoreMatch
 SUBSTRINGS MATCHING RULE caseIgnoreSubstringsMatch
 ID id-at-destinationIndicator }

DestinationIndicator ::= PrintableString (SIZE(1..ub-destination-indicator))
-- alphabetical characters only

communicationsService ATTRIBUTE ::= {
 WITH SYNTAX OBJECT IDENTIFIER
 EQUALITY MATCHING RULE objectIdentifierMatch
 ID id-at-communicationsService }

communicationsNetwork ATTRIBUTE ::= {
 WITH SYNTAX OBJECT IDENTIFIER
 EQUALITY MATCHING RULE objectIdentifierMatch
 SINGLE VALUE TRUE
 ID id-at-communicationsNetwork }

preferredDeliveryMethod ATTRIBUTE ::= {
 WITH SYNTAX PreferredDeliveryMethod
 SINGLE VALUE TRUE
 ID id-at-preferredDeliveryMethod }

PreferredDeliveryMethod ::= SEQUENCE OF INTEGER {
 any-delivery-method (0),
 mhs-delivery (1),
 physical-delivery (2),
 telex-delivery (3),
 teletex-delivery (4),
 g3-facsimile-delivery (5),
 g4-facsimile-delivery (6),
 ia5-terminal-delivery (7),
 videotex-delivery (8),
 telephone-delivery (9) }

```

presentationAddress ATTRIBUTE ::= {
    WITH SYNTAX                PresentationAddress
    EQUALITY MATCHING RULE     presentationAddressMatch
    SINGLE VALUE               TRUE
    ID                          id-at-presentationAddress }

PresentationAddress ::= SEQUENCE {
    pSelector      [0] OCTET STRING OPTIONAL,
    sSelector      [1] OCTET STRING OPTIONAL,
    tSelector      [2] OCTET STRING OPTIONAL,
    nAddresses     [3] SET SIZE (1..MAX) OF OCTET STRING }

supportedApplicationContext ATTRIBUTE ::= {
    WITH SYNTAX                OBJECT IDENTIFIER
    EQUALITY MATCHING RULE     objectIdentifierMatch
    ID                          id-at-supportedApplicationContext }

protocolInformation ATTRIBUTE ::= {
    WITH SYNTAX                ProtocolInformation
    EQUALITY MATCHING RULE     protocolInformationMatch
    ID                          id-at-protocolInformation }

ProtocolInformation ::= SEQUENCE {
    nAddress        OCTET STRING,
    profiles        SET OF OBJECT IDENTIFIER }

distinguishedName ATTRIBUTE ::= {
    WITH SYNTAX                DistinguishedName
    EQUALITY MATCHING RULE     distinguishedNameMatch
    ID                          id-at-distinguishedName }

member ATTRIBUTE ::= {
    SUBTYPE OF distinguishedName
    ID                          id-at-member }

uniqueMember ATTRIBUTE ::= {
    WITH SYNTAX                NameAndOptionalUID
    EQUALITY MATCHING RULE     uniqueMemberMatch
    ID                          id-at-uniqueMember }

NameAndOptionalUID ::= SEQUENCE {
    dn      DistinguishedName,
    uid     UniquelyIdentifier OPTIONAL }

owner ATTRIBUTE ::= {
    SUBTYPE OF distinguishedName
    ID                          id-at-owner }

roleOccupant ATTRIBUTE ::= {
    SUBTYPE OF distinguishedName
    ID                          id-at-roleOccupant }

seeAlso ATTRIBUTE ::= {
    SUBTYPE OF distinguishedName
    ID                          id-at-seeAlso }

dmdName ATTRIBUTE ::= {
    SUBTYPE OF name
    WITH SYNTAX                DirectoryString{ub-common-name}
    ID                          id-at-dmdName }

-- Notification attributes --

dSAPProblem ATTRIBUTE ::= {
    WITH SYNTAX                OBJECT IDENTIFIER
    EQUALITY MATCHING RULE     objectIdentifierMatch
    ID                          id-not-dSAPProblem }

```

```

searchServiceProblem ATTRIBUTE ::= {
    WITH SYNTAX                OBJECT IDENTIFIER
    EQUALITY MATCHING RULE     objectIdentifierMatch
    SINGLE VALUE               TRUE
    ID                         id-not-searchServiceProblem }

serviceType ATTRIBUTE ::= {
    WITH SYNTAX                OBJECT IDENTIFIER
    EQUALITY MATCHING RULE     objectIdentifierMatch
    SINGLE VALUE               TRUE
    ID                         id-not-serviceType }

attributeTypeList ATTRIBUTE ::= {
    WITH SYNTAX                OBJECT IDENTIFIER
    EQUALITY MATCHING RULE     objectIdentifierMatch
    ID                         id-not-attributeTypeList }

matchingRuleList ATTRIBUTE ::= {
    WITH SYNTAX                OBJECT IDENTIFIER
    EQUALITY MATCHING RULE     objectIdentifierMatch
    ID                         id-not-matchingRuleList }

filterItem ATTRIBUTE ::= {
    WITH SYNTAX                FilterItem
    ID                         id-not-filterItem }

attributeCombinations ATTRIBUTE ::= {
    WITH SYNTAX                AttributeCombination
    ID                         id-not-attributeCombinations }

contextTypeList ATTRIBUTE ::= {
    WITH SYNTAX                OBJECT IDENTIFIER
    EQUALITY MATCHING RULE     objectIdentifierMatch
    ID                         id-not-contextTypeList }

contextList ATTRIBUTE ::= {
    WITH SYNTAX                ContextAssertion
    ID                         id-not-contextList }

contextCombinations ATTRIBUTE ::= {
    WITH SYNTAX                ContextCombination
    ID                         id-not-contextCombinations }

hierarchySelectList ATTRIBUTE ::= {
    WITH SYNTAX                HierarchySelections
    SINGLE VALUE               TRUE
    ID                         id-not-hierarchySelectList }

searchControlOptionsList ATTRIBUTE ::= {
    WITH SYNTAX                SearchControlOptions
    SINGLE VALUE               TRUE
    ID                         id-not-searchControlOptionsList }

serviceControlOptionsList ATTRIBUTE ::= {
    WITH SYNTAX                ServiceControlOptions
    SINGLE VALUE               TRUE
    ID                         id-not-serviceControlOptionsList }

multipleMatchingLocalities ATTRIBUTE ::= {
    WITH SYNTAX                MultipleMatchingLocalities
    ID                         id-not-multipleMatchingLocalities }

MultipleMatchingLocalities ::= SEQUENCE {
    matchingRuleUsed           MATCHING-RULE.&id OPTIONAL,
    attributeList              SEQUENCE OF AttributeValueAssertion }

proposedRelaxation ATTRIBUTE ::= {
    WITH SYNTAX                SEQUENCE OF MRMapping
    ID                         id-not-proposedRelaxation }

```

```

appliedRelaxation ATTRIBUTE ::= {
    WITH SYNTAX
    EQUALITY MATCHING RULE
    ID
    OBJECT IDENTIFIER
    objectIdentifierMatch
    id-not-appliedRelaxation }

-- Matching rules --

caseIgnoreMatch MATCHING-RULE ::= {
    SYNTAX DirectoryString {ub-match}
    ID id-mr-caseIgnoreMatch }

caseIgnoreOrderingMatch MATCHING-RULE ::= {
    SYNTAX DirectoryString {ub-match}
    ID id-mr-caseIgnoreOrderingMatch }

caseIgnoreSubstringsMatch MATCHING-RULE ::= {
    SYNTAX SubstringAssertion
    ID id-mr-caseIgnoreSubstringsMatch }

SubstringAssertion ::= SEQUENCE OF CHOICE {
    initial [0] DirectoryString {ub-match},
    any [1] DirectoryString {ub-match},
    final [2] DirectoryString {ub-match},
    control Attribute } -- Used to specify interpretation of the following items
    -- at most one initial and one final component

caseExactMatch MATCHING-RULE ::= {
    SYNTAX DirectoryString {ub-match}
    ID id-mr-caseExactMatch }

caseExactOrderingMatch MATCHING-RULE ::= {
    SYNTAX DirectoryString {ub-match}
    ID id-mr-caseExactOrderingMatch }

caseExactSubstringsMatch MATCHING-RULE ::= {
    SYNTAX SubstringAssertion -- only the PrintableString choice
    ID id-mr-caseExactSubstringsMatch }

numericStringMatch MATCHING-RULE ::= {
    SYNTAX NumericString
    ID id-mr-numericStringMatch }

numericStringOrderingMatch MATCHING-RULE ::= {
    SYNTAX NumericString
    ID id-mr-numericStringOrderingMatch }

numericStringSubstringsMatch MATCHING-RULE ::= {
    SYNTAX SubstringAssertion
    ID id-mr-numericStringSubstringsMatch }

caseIgnoreListMatch MATCHING-RULE ::= {
    SYNTAX CaseIgnoreList
    ID id-mr-caseIgnoreListMatch }

CaseIgnoreList ::= SEQUENCE OF DirectoryString {ub-match}

caseIgnoreListSubstringsMatch MATCHING-RULE ::= {
    SYNTAX SubstringAssertion
    ID id-mr-caseIgnoreListSubstringsMatch }

storedPrefixMatch MATCHING-RULE ::= {
    SYNTAX DirectoryString {ub-match}
    ID id-mr-storedPrefixMatch }

booleanMatch MATCHING-RULE ::= {
    SYNTAX BOOLEAN
    ID id-mr-booleanMatch }

```

```

integerMatch MATCHING-RULE ::= {
    SYNTAX INTEGER
    ID      id-mr-integerMatch }

integerOrderingMatch MATCHING-RULE ::= {
    SYNTAX INTEGER
    ID      id-mr-integerOrderingMatch }

bitStringMatch MATCHING-RULE ::= {
    SYNTAX BIT STRING
    ID      id-mr-bitStringMatch }

octetStringMatch MATCHING-RULE ::= {
    SYNTAX OCTET STRING
    ID      id-mr-octetStringMatch }

octetStringOrderingMatch MATCHING-RULE ::= {
    SYNTAX OCTET STRING
    ID      id-mr-octetStringOrderingMatch }

octetStringSubstringsMatch MATCHING-RULE ::= {
    SYNTAX OctetSubstringAssertion
    ID      id-mr-octetStringSubstringsMatch }

OctetSubstringAssertion ::= SEQUENCE OF CHOICE {
    initial  [0] OCTET STRING,
    any     [1] OCTET STRING,
    final   [2] OCTET STRING }
-- at most one initial and one final component

telephoneNumberMatch MATCHING-RULE ::= {
    SYNTAX TelephoneNumber
    ID      id-mr-telephoneNumberMatch }

telephoneNumberSubstringsMatch MATCHING-RULE ::= {
    SYNTAX SubstringAssertion
    ID      id-mr-telephoneNumberSubstringsMatch }

presentationAddressMatch MATCHING-RULE ::= {
    SYNTAX PresentationAddress
    ID      id-mr-presentationAddressMatch }

uniqueMemberMatch MATCHING-RULE ::= {
    SYNTAX NameAndOptionalUID
    ID      id-mr-uniqueMemberMatch }

protocolInformationMatch MATCHING-RULE ::= {
    SYNTAX OCTET STRING
    ID      id-mr-protocolInformationMatch }

facsimileNumberMatch MATCHING-RULE ::= {
    SYNTAX TelephoneNumber
    ID      id-mr-facsimileNumberMatch }

facsimileNumberSubstringsMatch MATCHING-RULE ::= {
    SYNTAX SubstringAssertion
    ID      id-mr-facsimileNumberSubstringsMatch }

uUIDPairMatch MATCHING-RULE ::= {
    SYNTAX UUIDPair
    ID      id-mr-uuidpairmatch }

uTCTimeMatch MATCHING-RULE ::= {
    SYNTAX UTCTime
    ID      id-mr-utctimeMatch }

uTCTimeOrderingMatch MATCHING-RULE ::= {
    SYNTAX UTCTime
    ID      id-mr-utctimeOrderingMatch }

```

```

generalizedTimeMatch MATCHING-RULE ::= {
  SYNTAX GeneralizedTime
  -- as per 42.3 b) or c) of ITU-T Rec. X.680 | ISO/IEC 8824-1
  ID      id-mr-generalizedTimeMatch }

generalizedTimeOrderingMatch MATCHING-RULE ::= {
  SYNTAX GeneralizedTime
  -- as per 42.3 b) or c) of ITU-T Rec. X.680 | ISO/IEC 8824-1
  ID      id-mr-generalizedTimeOrderingMatch }

systemProposedMatch MATCHING-RULE ::= {
  ID      id-mr-systemProposedMatch }

integerFirstComponentMatch MATCHING-RULE ::= {
  SYNTAX INTEGER
  ID      id-mr-integerFirstComponentMatch }

objectIdentifierFirstComponentMatch MATCHING-RULE ::= {
  SYNTAX OBJECT IDENTIFIER
  ID      id-mr-objectIdentifierFirstComponentMatch }

directoryStringFirstComponentMatch MATCHING-RULE ::= {
  SYNTAX DirectoryString {ub-directory-string-first-component-match}
  ID      id-mr-directoryStringFirstComponentMatch }

wordMatch MATCHING-RULE ::= {
  SYNTAX DirectoryString {ub-match}
  ID      id-mr-wordMatch }

keywordMatch MATCHING-RULE ::= {
  SYNTAX DirectoryString {ub-match}
  ID      id-mr-keywordMatch }

generalWordMatch MATCHING-RULE ::= {
  SYNTAX      SubstringAssertion
  ID          id-mr-generalWordMatch }

sequenceMatchType ATTRIBUTE ::= {
  WITH SYNTAX      SequenceMatchType
  SINGLE VALUE     TRUE
  ID               id-cat-sequenceMatchType }      -- defaulting to sequenceExact

SequenceMatchType ::= ENUMERATED {
  sequenceExact          (0),
  sequenceDeletion       (1),
  sequenceRestrictedDeletion (2),
  sequencePermutation     (3),
  sequencePermutationAndDeletion (4),
  sequenceProviderDefined (5) }

wordMatchTypes ATTRIBUTE ::= {
  WITH SYNTAX      WordMatchTypes
  SINGLE VALUE     TRUE
  ID               id-cat-wordMatchType }      -- defaulting to wordExact

WordMatchTypes ::= ENUMERATED {
  wordExact          (0),
  wordTruncated      (1),
  wordPhonetic       (2),
  wordProviderDefined (3) }

characterMatchTypes ATTRIBUTE ::= {
  WITH SYNTAX      CharacterMatchTypes
  SINGLE VALUE     TRUE
  ID               id-cat-characterMatchTypes }

CharacterMatchTypes ::= ENUMERATED {
  characterExact          (0),
  characterCaseIgnore     (1),
  characterMapped         (2) }

```

```
selectedContexts ATTRIBUTE ::= {
    WITH SYNTAX      ContextAssertion
    ID               id-cat-selectedContexts }
```

```
approximateStringMatch MATCHING-RULE ::= {
    ID               id-mr-approximateStringMatch }
```

```
ignoreIfAbsentMatch MATCHING-RULE ::= {
    ID               id-mr-ignoreIfAbsentMatch }
```

```
nullMatch MATCHING-RULE ::= {
    ID               id-mr-nullMatch }
```

ZONAL-MATCHING ::= MAPPING-BASED-MATCHING { ZonalSelect, TRUE, ZonalResult, zonalMatch.&id }

ZonalSelect ::= SEQUENCE OF AttributeType

```
ZonalResult ::= ENUMERATED {
    cannot-select-mapping      (0),
    zero-mappings              (2),
    multiple-mappings          (3) }
```

```
zonalMatch MATCHING-RULE ::= {
    UNIQUE-MATCH-INDICATOR    multipleMatchingLocalities
    ID                         id-mr-zonalMatch }
```

-- Contexts --

```
languageContext CONTEXT ::= {
    WITH SYNTAX      LanguageContextSyntax
    ID               id-avc-language }
```

LanguageContextSyntax ::= PrintableString (SIZE(2..3)) -- ISO 639-2 codes only

```
temporalContext CONTEXT ::= {
    WITH SYNTAX      TimeSpecification
    ASSERTED AS     TimeAssertion
    ID               id-avc-temporal }
```

```
TimeSpecification ::= SEQUENCE {
    time              CHOICE {
        absolute      SEQUENCE {
            startTime [0] GeneralizedTime OPTIONAL,
            endTime   [1] GeneralizedTime OPTIONAL },
        periodic      SET OF Period },
    notThisTime      BOOLEAN DEFAULT FALSE,
    timeZone         TimeZone OPTIONAL }
```

```
Period ::= SEQUENCE {
    timesOfDay [0] SET SIZE (1..MAX) OF DayTimeBand OPTIONAL,
    days       [1] CHOICE {
        intDay SET OF INTEGER,
        bitDay BIT STRING { sunday (0), monday (1), tuesday (2),
            wednesday (3), thursday (4), friday (5), saturday (6) },
        dayOf XDayOf } OPTIONAL,
    weeks      [2] CHOICE {
        allWeeks NULL,
        intWeek  SET OF INTEGER,
        bitWeek  BIT STRING { week1 (0), week2 (1), week3 (2), week4 (3),
            week5 (4) } } OPTIONAL,
    months     [3] CHOICE {
        allMonths NULL,
        intMonth  SET OF INTEGER,
        bitMonth  BIT STRING { january (0), february (1), march (2), april (3),
            may (4), june (5), july (6), august (7), september (8),
            october (9), november (10), december (11) }
        } OPTIONAL,
    years      [4] SET OF INTEGER (1000 .. MAX) OPTIONAL }
```

```
XDayOf ::= CHOICE {
    first    [1] NamedDay,
    second  [2] NamedDay,
    third   [3] NamedDay,
    fourth  [4] NamedDay,
    fifth   [5] NamedDay }
```

```
NamedDay ::= CHOICE {
    intNamedDays    ENUMERATED {
        sunday    (1),
        monday    (2),
        tuesday    (3),
        wednesday  (4),
        thursday   (5),
        friday     (6),
        saturday   (7) },
    bitNamedDays    BIT STRING { sunday (0), monday (1) , tuesday (2),
        wednesday (3), thursday (4), friday (5), saturday (6) } }
```

```
DayTimeBand ::= SEQUENCE {
    startDayTime [0] DayTime DEFAULT { hour 0 },
    endDayTime  [1] DayTime DEFAULT { hour 23, minute 59, second 59 } }
```

```
DayTime ::= SEQUENCE {
    hour    [0] INTEGER (0..23),
    minute  [1] INTEGER (0..59) DEFAULT 0,
    second  [2] INTEGER (0..59) DEFAULT 0 }
```

```
TimeZone ::= INTEGER (-12..12)
```

```
TimeAssertion ::= CHOICE {
    now      NULL,
    at       GeneralizedTime,
    between  SEQUENCE {
        startTime [0] GeneralizedTime,
        endTime  [1] GeneralizedTime OPTIONAL,
        entirely  BOOLEAN DEFAULT FALSE } }
```

```
localeContext CONTEXT ::= {
    WITH SYNTAX  LocaleContextSyntax
    ID           id-avc-locale }
```

```
LocaleContextSyntax ::= CHOICE {
    localeID1 OBJECT IDENTIFIER,
    localeID2 DirectoryString {ub-localeContextSyntax} }
```

```
ldapAttributeOptionContext CONTEXT ::= {
    WITH SYNTAX  AttributeOptionList
    ASSERTED AS AttributeOptionList
    ABSENT-MATCH FALSE
    ID           id-avc-ldapAttributeOption }
```

```
AttributeOptionList ::= SEQUENCE OF UTF8String
```

```
-- Object identifier assignments --
-- object identifiers assigned in other modules are shown in comments
```

```
-- Attributes --
```

```
-- id-at-objectClass          OBJECT IDENTIFIER ::= {id-at 0}
-- id-at-aliasedEntryName     OBJECT IDENTIFIER ::= {id-at 1}
-- id-at-encryptedAliasedEntryName OBJECT IDENTIFIER ::= {id-at 1 2}
id-at-knowledgeInformation    OBJECT IDENTIFIER ::= {id-at 2}
id-at-commonName             OBJECT IDENTIFIER ::= {id-at 3}
-- id-at-encryptedCommonName  OBJECT IDENTIFIER ::= {id-at 3 2}
id-at-surname                OBJECT IDENTIFIER ::= {id-at 4}
-- id-at-encryptedSurname     OBJECT IDENTIFIER ::= {id-at 4 2}
id-at-serialNumber           OBJECT IDENTIFIER ::= {id-at 5}
-- id-at-encryptedSerialNumber OBJECT IDENTIFIER ::= {id-at 5 2}
id-at-countryName            OBJECT IDENTIFIER ::= {id-at 6}
-- id-at-encryptedCountryName OBJECT IDENTIFIER ::= {id-at 6 2}
```

id-at-localityName	OBJECT IDENTIFIER ::= {id-at 7}
-- <i>id-at-encryptedLocalityName</i>	<i>OBJECT IDENTIFIER ::= {id-at 7 2}</i>
id-at-collectiveLocalityName	OBJECT IDENTIFIER ::= {id-at 7 1}
-- <i>id-at-encryptedCollectiveLocalityName</i>	<i>OBJECT IDENTIFIER ::= {id-at 7 1 2}</i>
id-at-stateOrProvinceName	OBJECT IDENTIFIER ::= {id-at 8}
-- <i>id-at-encryptedStateOrProvinceName</i>	<i>OBJECT IDENTIFIER ::= {id-at 8 2}</i>
id-at-collectiveStateOrProvinceName	OBJECT IDENTIFIER ::= {id-at 8 1}
-- <i>id-at-encryptedCollectiveStateOrProvinceName</i>	<i>OBJECT IDENTIFIER ::= {id-at 8 1 2}</i>
id-at-streetAddress	OBJECT IDENTIFIER ::= {id-at 9}
-- <i>id-at-encryptedStreetAddress</i>	<i>OBJECT IDENTIFIER ::= {id-at 9 2}</i>
id-at-collectiveStreetAddress	OBJECT IDENTIFIER ::= {id-at 9 1}
-- <i>id-at-encryptedCollectiveStreetAddress</i>	<i>OBJECT IDENTIFIER ::= {id-at 9 1 2}</i>
id-at-organizationName	OBJECT IDENTIFIER ::= {id-at 10}
-- <i>id-at-encryptedOrganizationName</i>	<i>OBJECT IDENTIFIER ::= {id-at 10 2}</i>
id-at-collectiveOrganizationName	OBJECT IDENTIFIER ::= {id-at 10 1}
-- <i>id-at-encryptedCollectiveOrganizationName</i>	<i>OBJECT IDENTIFIER ::= {id-at 10 1 2}</i>
id-at-organizationalUnitName	OBJECT IDENTIFIER ::= {id-at 11}
-- <i>id-at-encryptedOrganizationalUnitName</i>	<i>OBJECT IDENTIFIER ::= {id-at 11 2}</i>
id-at-collectiveOrganizationalUnitName	OBJECT IDENTIFIER ::= {id-at 11 1}
-- <i>id-at-encryptedCollectiveOrganizationalUnitName</i>	<i>OBJECT IDENTIFIER ::= {id-at 11 1 2}</i>
id-at-title	OBJECT IDENTIFIER ::= {id-at 12}
-- <i>id-at-encryptedTitle</i>	<i>OBJECT IDENTIFIER ::= {id-at 12 2}</i>
id-at-description	OBJECT IDENTIFIER ::= {id-at 13}
-- <i>id-at-encryptedDescription</i>	<i>OBJECT IDENTIFIER ::= {id-at 13 2}</i>
id-at-searchGuide	OBJECT IDENTIFIER ::= {id-at 14}
-- <i>id-at-encryptedSearchGuide</i>	<i>OBJECT IDENTIFIER ::= {id-at 14 2}</i>
id-at-businessCategory	OBJECT IDENTIFIER ::= {id-at 15}
-- <i>id-at-encryptedBusinessCategory</i>	<i>OBJECT IDENTIFIER ::= {id-at 15 2}</i>
id-at-postalAddress	OBJECT IDENTIFIER ::= {id-at 16}
-- <i>id-at-encryptedPostalAddress</i>	<i>OBJECT IDENTIFIER ::= {id-at 16 2}</i>
id-at-collectivePostalAddress	OBJECT IDENTIFIER ::= {id-at 16 1}
-- <i>id-at-encryptedCollectivePostalAddress</i>	<i>OBJECT IDENTIFIER ::= {id-at 16 1 2}</i>
id-at-postalCode	OBJECT IDENTIFIER ::= {id-at 17}
-- <i>id-at-encryptedPostalCode</i>	<i>OBJECT IDENTIFIER ::= {id-at 17 2}</i>
id-at-collectivePostalCode	OBJECT IDENTIFIER ::= {id-at 17 1}
-- <i>id-at-encryptedCollectivePostalCode</i>	<i>OBJECT IDENTIFIER ::= {id-at 17 1 2}</i>
id-at-postOfficeBox	OBJECT IDENTIFIER ::= {id-at 18}
-- <i>id-at-encryptedPostOfficeBox</i>	<i>OBJECT IDENTIFIER ::= {id-at 18 2}</i>
-- <i>id-at-encryptedCollectivePostOfficeBox</i>	<i>OBJECT IDENTIFIER ::= {id-at 18 1 2}</i>
id-at-physicalDeliveryOfficeName	OBJECT IDENTIFIER ::= {id-at 19}
-- <i>id-at-encryptedPhysicalDeliveryOfficeName</i>	<i>OBJECT IDENTIFIER ::= {id-at 19 2}</i>
-- <i>id-at-encryptedCollectivePhysicalDeliveryOfficeName</i>	<i>OBJECT IDENTIFIER ::= {id-at 19 1 2}</i>
id-at-telephoneNumber	OBJECT IDENTIFIER ::= {id-at 20}
-- <i>id-at-encryptedTelephoneNumber</i>	<i>OBJECT IDENTIFIER ::= {id-at 20 2}</i>
id-at-collectiveTelephoneNumber	OBJECT IDENTIFIER ::= {id-at 20 1}
-- <i>id-at-encryptedCollectiveTelephoneNumber</i>	<i>OBJECT IDENTIFIER ::= {id-at 20 1 2}</i>
id-at-telexNumber	OBJECT IDENTIFIER ::= {id-at 21}
-- <i>id-at-encryptedTelexNumber</i>	<i>OBJECT IDENTIFIER ::= {id-at 21 2}</i>
id-at-collectiveTelexNumber	OBJECT IDENTIFIER ::= {id-at 21 1}
-- <i>id-at-encryptedCollectiveTelexNumber</i>	<i>OBJECT IDENTIFIER ::= {id-at 21 1 2}</i>
-- <i>id-at-teletexTerminalIdentifier</i>	<i>OBJECT IDENTIFIER ::= {id-at 22}</i>
-- <i>id-at-encryptedTeletexTerminalIdentifier</i>	<i>OBJECT IDENTIFIER ::= {id-at 22 2}</i>
-- <i>id-at-collectiveTeletexTerminalIdentifier</i>	<i>OBJECT IDENTIFIER ::= {id-at 22 1}</i>
-- <i>id-at-encryptedCollectiveTeletexTerminalIdentifier</i>	<i>OBJECT IDENTIFIER ::= {id-at 22 1 2}</i>
id-at-facsimileTelephoneNumber	OBJECT IDENTIFIER ::= {id-at 23}
-- <i>id-at-encryptedFacsimileTelephoneNumber</i>	<i>OBJECT IDENTIFIER ::= {id-at 23 2}</i>
id-at-collectiveFacsimileTelephoneNumber	OBJECT IDENTIFIER ::= {id-at 23 1}
-- <i>id-at-encryptedCollectiveFacsimileTelephoneNumber</i>	<i>OBJECT IDENTIFIER ::= {id-at 23 1 2}</i>
id-at-x121Address	OBJECT IDENTIFIER ::= {id-at 24}
-- <i>id-at-encryptedX121Address</i>	<i>OBJECT IDENTIFIER ::= {id-at 24 2}</i>
id-at-internationalISDNNumber	OBJECT IDENTIFIER ::= {id-at 25}
-- <i>id-at-encryptedInternationalISDNNumber</i>	<i>OBJECT IDENTIFIER ::= {id-at 25 2}</i>
id-at-collectiveInternationalISDNNumber	OBJECT IDENTIFIER ::= {id-at 25 1}
-- <i>id-at-encryptedCollectiveInternationalISDNNumber</i>	<i>OBJECT IDENTIFIER ::= {id-at 25 1 2}</i>
id-at-registeredAddress	OBJECT IDENTIFIER ::= {id-at 26}
-- <i>id-at-encryptedRegisteredAddress</i>	<i>OBJECT IDENTIFIER ::= {id-at 26 2}</i>
id-at-destinationIndicator	OBJECT IDENTIFIER ::= {id-at 27}
-- <i>id-at-encryptedDestinationIndicator</i>	<i>OBJECT IDENTIFIER ::= {id-at 27 2}</i>

id-at-preferredDeliveryMethod	OBJECT IDENTIFIER ::= {id-at 28}
-- id-at-encryptedPreferredDeliveryMethod	OBJECT IDENTIFIER ::= {id-at 28 2}
id-at-presentationAddress	OBJECT IDENTIFIER ::= {id-at 29}
-- id-at-encryptedPresentationAddress	OBJECT IDENTIFIER ::= {id-at 29 2}
id-at-supportedApplicationContext	OBJECT IDENTIFIER ::= {id-at 30}
-- id-at-encryptedSupportedApplicationContext	OBJECT IDENTIFIER ::= {id-at 30 2}
id-at-member	OBJECT IDENTIFIER ::= {id-at 31}
-- id-at-encryptedMember	OBJECT IDENTIFIER ::= {id-at 31 2}
id-at-owner	OBJECT IDENTIFIER ::= {id-at 32}
-- id-at-encryptedOwner	OBJECT IDENTIFIER ::= {id-at 32 2}
id-at-roleOccupant	OBJECT IDENTIFIER ::= {id-at 33}
-- id-at-encryptedRoleOccupant	OBJECT IDENTIFIER ::= {id-at 33 2}
id-at-seeAlso	OBJECT IDENTIFIER ::= {id-at 34}
-- id-at-encryptedSeeAlso	OBJECT IDENTIFIER ::= {id-at 34 2}
-- id-at-userPassword	OBJECT IDENTIFIER ::= {id-at 35}
-- id-at-encryptedUserPassword	OBJECT IDENTIFIER ::= {id-at 35 2}
-- id-at-userCertificate	OBJECT IDENTIFIER ::= {id-at 36}
-- id-at-encryptedUserCertificate	OBJECT IDENTIFIER ::= {id-at 36 2}
-- id-at-cACertificate	OBJECT IDENTIFIER ::= {id-at 37}
-- id-at-encryptedCACertificate	OBJECT IDENTIFIER ::= {id-at 37 2}
-- id-at-authorityRevocationList	OBJECT IDENTIFIER ::= {id-at 38}
-- id-at-encryptedAuthorityRevocationList	OBJECT IDENTIFIER ::= {id-at 38 2}
-- id-at-certificateRevocationList	OBJECT IDENTIFIER ::= {id-at 39}
-- id-at-encryptedCertificateRevocationList	OBJECT IDENTIFIER ::= {id-at 39 2}
-- id-at-crossCertificatePair	OBJECT IDENTIFIER ::= {id-at 40}
-- id-at-encryptedCrossCertificatePair	OBJECT IDENTIFIER ::= {id-at 40 2}
id-at-name	OBJECT IDENTIFIER ::= {id-at 41}
id-at-givenName	OBJECT IDENTIFIER ::= {id-at 42}
-- id-at-encryptedGivenName	OBJECT IDENTIFIER ::= {id-at 42 2}
id-at-initials	OBJECT IDENTIFIER ::= {id-at 43}
-- id-at-encryptedInitials	OBJECT IDENTIFIER ::= {id-at 43 2}
id-at-generationQualifier	OBJECT IDENTIFIER ::= {id-at 44}
-- id-at-encryptedGenerationQualifier	OBJECT IDENTIFIER ::= {id-at 44 2}
id-at-uniqueIdentifier	OBJECT IDENTIFIER ::= {id-at 45}
-- id-at-encryptedUniqueIdentifier	OBJECT IDENTIFIER ::= {id-at 45 2}
id-at-dnQualifier	OBJECT IDENTIFIER ::= {id-at 46}
-- id-at-encryptedDnQualifier	OBJECT IDENTIFIER ::= {id-at 46 2}
id-at-enhancedSearchGuide	OBJECT IDENTIFIER ::= {id-at 47}
-- id-at-encryptedEnhancedSearchGuide	OBJECT IDENTIFIER ::= {id-at 47 2}
id-at-protocolInformation	OBJECT IDENTIFIER ::= {id-at 48}
-- id-at-encryptedProtocolInformation	OBJECT IDENTIFIER ::= {id-at 48 2}
id-at-distinguishedName	OBJECT IDENTIFIER ::= {id-at 49}
-- id-at-encryptedDistinguishedName	OBJECT IDENTIFIER ::= {id-at 49 2}
id-at-uniqueMember	OBJECT IDENTIFIER ::= {id-at 50}
-- id-at-encryptedUniqueMember	OBJECT IDENTIFIER ::= {id-at 50 2}
id-at-houseIdentifier	OBJECT IDENTIFIER ::= {id-at 51}
-- id-at-encryptedHouseIdentifier	OBJECT IDENTIFIER ::= {id-at 51 2}
-- id-at-supportedAlgorithms	OBJECT IDENTIFIER ::= {id-at 52}
-- id-at-encryptedSupportedAlgorithms	OBJECT IDENTIFIER ::= {id-at 52 2}
-- id-at-deltaRevocationList	OBJECT IDENTIFIER ::= {id-at 53}
-- id-at-encryptedDeltaRevocationList	OBJECT IDENTIFIER ::= {id-at 53 2}
id-at-dmdName	OBJECT IDENTIFIER ::= {id-at 54}
-- id-at-encryptedDmdName	OBJECT IDENTIFIER ::= {id-at 54 2}
-- id-at-clearance	OBJECT IDENTIFIER ::= {id-at 55}
-- id-at-encryptedClearance	OBJECT IDENTIFIER ::= {id-at 55 2}
-- id-at-defaultDirQop	OBJECT IDENTIFIER ::= {id-at 56}
-- id-at-encryptedDefaultDirQop	OBJECT IDENTIFIER ::= {id-at 56 2}
-- id-at-attributeIntegrityInfo	OBJECT IDENTIFIER ::= {id-at 57}
-- id-at-encryptedAttributeIntegrityInfo	OBJECT IDENTIFIER ::= {id-at 57 2}
-- id-at-attributeCertificate	OBJECT IDENTIFIER ::= {id-at 58}
-- id-at-encryptedAttributeCertificate	OBJECT IDENTIFIER ::= {id-at 58 2}
-- id-at-attributeCertificateRevocationList	OBJECT IDENTIFIER ::= {id-at 59}
-- id-at-encryptedAttributeCertificateRevocationList	OBJECT IDENTIFIER ::= {id-at 59 2}
-- id-at-confKeyInfo	OBJECT IDENTIFIER ::= {id-at 60}
-- id-at-encryptedConfKeyInfo	OBJECT IDENTIFIER ::= {id-at 60 2}
-- id-at-aACertificate	OBJECT IDENTIFIER ::= {id-at 61}
-- id-at-attributeDescriptorCertificate	OBJECT IDENTIFIER ::= {id-at 62}
-- id-at-attributeAuthorityRevocationList	OBJECT IDENTIFIER ::= {id-at 63}
-- id-at-family-information	OBJECT IDENTIFIER ::= {id-at 64}

id-at-pseudonym	OBJECT IDENTIFIER ::= {id-at 65}
id-at-communicationsService	OBJECT IDENTIFIER ::= {id-at 66}
id-at-communicationsNetwork	OBJECT IDENTIFIER ::= {id-at 67}
-- id-at-certificationPracticeStmnt	OBJECT IDENTIFIER ::= {id-at 68}
-- id-at-certificatePolicy	OBJECT IDENTIFIER ::= {id-at 69}
-- id-at-pkiPath	OBJECT IDENTIFIER ::= {id-at 70}
-- id-at-privPolicy	OBJECT IDENTIFIER ::= {id-at 71}
-- id-at-role	OBJECT IDENTIFIER ::= {id-at 72}
-- id-at-delegationPath	OBJECT IDENTIFIER ::= {id-at 73}
-- id-at-protPrivPolicy	OBJECT IDENTIFIER ::= {id-at 74}
-- id-at-xMLPrivilegeInfo	OBJECT IDENTIFIER ::= {id-at 75}
id-at-uuidpair	OBJECT IDENTIFIER ::= {id-at 76}

-- Control attributes --

id-cat-sequenceMatchType	OBJECT IDENTIFIER ::= {id-cat 1}
id-cat-wordMatchType	OBJECT IDENTIFIER ::= {id-cat 2}
id-cat-characterMatchTypes	OBJECT IDENTIFIER ::= {id-cat 3}
id-cat-selectedContexts	OBJECT IDENTIFIER ::= {id-cat 4}

-- Notification attributes --

id-not-dSAPProblem	OBJECT IDENTIFIER ::= {id-not 0}
id-not-searchServiceProblem	OBJECT IDENTIFIER ::= {id-not 1}
id-not-serviceType	OBJECT IDENTIFIER ::= {id-not 2}
id-not-attributeTypeList	OBJECT IDENTIFIER ::= {id-not 3}
id-not-matchingRuleList	OBJECT IDENTIFIER ::= {id-not 4}
id-not-filterItem	OBJECT IDENTIFIER ::= {id-not 5}
id-not-attributeCombinations	OBJECT IDENTIFIER ::= {id-not 6}
id-not-contextTypeList	OBJECT IDENTIFIER ::= {id-not 7}
id-not-contextList	OBJECT IDENTIFIER ::= {id-not 8}
id-not-contextCombinations	OBJECT IDENTIFIER ::= {id-not 9}
id-not-hierarchySelectList	OBJECT IDENTIFIER ::= {id-not 10}
id-not-searchControlOptionsList	OBJECT IDENTIFIER ::= {id-not 11}
id-not-serviceControlOptionsList	OBJECT IDENTIFIER ::= {id-not 12}
id-not-multipleMatchingLocalities	OBJECT IDENTIFIER ::= {id-not 13}
id-not-proposedRelaxation	OBJECT IDENTIFIER ::= {id-not 14}
id-not-appliedRelaxation	OBJECT IDENTIFIER ::= {id-not 15}

-- Problem definitions --

id-pr-targetDsaUnavailable	OBJECT IDENTIFIER ::= {id-pr 1}
id-pr-dataSourceUnavailable	OBJECT IDENTIFIER ::= {id-pr 2}
id-pr-unidentifiedOperation	OBJECT IDENTIFIER ::= {id-pr 3}
id-pr-unavailableOperation	OBJECT IDENTIFIER ::= {id-pr 4}
id-pr-searchAttributeViolation	OBJECT IDENTIFIER ::= {id-pr 5}
id-pr-searchAttributeCombinationViolation	OBJECT IDENTIFIER ::= {id-pr 6}
id-pr-searchValueNotAllowed	OBJECT IDENTIFIER ::= {id-pr 7}
id-pr-missingSearchAttribute	OBJECT IDENTIFIER ::= {id-pr 8}
id-pr-searchValueViolation	OBJECT IDENTIFIER ::= {id-pr 9}
id-pr-attributeNegationViolation	OBJECT IDENTIFIER ::= {id-pr 10}
id-pr-searchValueRequired	OBJECT IDENTIFIER ::= {id-pr 11}
id-pr-invalidSearchValue	OBJECT IDENTIFIER ::= {id-pr 12}
id-pr-searchContextViolation	OBJECT IDENTIFIER ::= {id-pr 13}
id-pr-searchContextCombinationViolation	OBJECT IDENTIFIER ::= {id-pr 14}
id-pr-missingSearchContext	OBJECT IDENTIFIER ::= {id-pr 15}
id-pr-searchContextValueViolation	OBJECT IDENTIFIER ::= {id-pr 16}
id-pr-searchContextValueRequired	OBJECT IDENTIFIER ::= {id-pr 17}
id-pr-invalidContextSearchValue	OBJECT IDENTIFIER ::= {id-pr 18}
id-pr-unsupportedMatchingRule	OBJECT IDENTIFIER ::= {id-pr 19}
id-pr-attributeMatchingViolation	OBJECT IDENTIFIER ::= {id-pr 20}
id-pr-unsupportedMatchingUse	OBJECT IDENTIFIER ::= {id-pr 21}
id-pr-matchingUseViolation	OBJECT IDENTIFIER ::= {id-pr 22}
id-pr-hierarchySelectForbidden	OBJECT IDENTIFIER ::= {id-pr 23}
id-pr-invalidHierarchySelect	OBJECT IDENTIFIER ::= {id-pr 24}
id-pr-unavailableHierarchySelect	OBJECT IDENTIFIER ::= {id-pr 25}
id-pr-invalidSearchControlOptions	OBJECT IDENTIFIER ::= {id-pr 26}
id-pr-invalidServiceControlOptions	OBJECT IDENTIFIER ::= {id-pr 27}
id-pr-searchSubsetViolation	OBJECT IDENTIFIER ::= {id-pr 28}
id-pr-unmatchedKeyAttributes	OBJECT IDENTIFIER ::= {id-pr 29}

id-pr-ambiguousKeyAttributes	OBJECT IDENTIFIER ::= {id-pr 30}	
id-pr-unavailableRelaxationLevel	OBJECT IDENTIFIER ::= {id-pr 31}	
id-pr-emptyHierarchySelection	OBJECT IDENTIFIER ::= {id-pr 32}	
id-pr-administratorImposedLimit	OBJECT IDENTIFIER ::= {id-pr 33}	
id-pr-permanentRestriction	OBJECT IDENTIFIER ::= {id-pr 34}	
id-pr-temporaryRestriction	OBJECT IDENTIFIER ::= {id-pr 35}	
id-pr-relaxationNotSupported	OBJECT IDENTIFIER ::= {id-pr 36}	
<i>-- Matching rules --</i>		
<i>-- id-mr-objectIdentifierMatch</i>	<i>OBJECT IDENTIFIER ::= {id-mr 0}</i>	
<i>-- id-mr-distinguishedNameMatch</i>	<i>OBJECT IDENTIFIER ::= {id-mr 1}</i>	
id-mr-caseIgnoreMatch	OBJECT IDENTIFIER ::= {id-mr 2}	
id-mr-caseIgnoreOrderingMatch	OBJECT IDENTIFIER ::= {id-mr 3}	
id-mr-caseIgnoreSubstringsMatch	OBJECT IDENTIFIER ::= {id-mr 4}	
id-mr-caseExactMatch	OBJECT IDENTIFIER ::= {id-mr 5}	
id-mr-caseExactOrderingMatch	OBJECT IDENTIFIER ::= {id-mr 6}	
id-mr-caseExactSubstringsMatch	OBJECT IDENTIFIER ::= {id-mr 7}	
id-mr-numericStringMatch	OBJECT IDENTIFIER ::= {id-mr 8}	
id-mr-numericStringOrderingMatch	OBJECT IDENTIFIER ::= {id-mr 9}	
id-mr-numericStringSubstringsMatch	OBJECT IDENTIFIER ::= {id-mr 10}	
id-mr-caseIgnoreListMatch	OBJECT IDENTIFIER ::= {id-mr 11}	
id-mr-caseIgnoreListSubstringsMatch	OBJECT IDENTIFIER ::= {id-mr 12}	
id-mr-booleanMatch	OBJECT IDENTIFIER ::= {id-mr 13}	
id-mr-integerMatch	OBJECT IDENTIFIER ::= {id-mr 14}	
id-mr-integerOrderingMatch	OBJECT IDENTIFIER ::= {id-mr 15}	
id-mr-bitStringMatch	OBJECT IDENTIFIER ::= {id-mr 16}	
id-mr-octetStringMatch	OBJECT IDENTIFIER ::= {id-mr 17}	
id-mr-octetStringOrderingMatch	OBJECT IDENTIFIER ::= {id-mr 18}	
id-mr-octetStringSubstringsMatch	OBJECT IDENTIFIER ::= {id-mr 19}	
id-mr-telephoneNumberMatch	OBJECT IDENTIFIER ::= {id-mr 20}	
id-mr-telephoneNumberSubstringsMatch	OBJECT IDENTIFIER ::= {id-mr 21}	
id-mr-presentationAddressMatch	OBJECT IDENTIFIER ::= {id-mr 22}	
id-mr-uniqueMemberMatch	OBJECT IDENTIFIER ::= {id-mr 23}	
id-mr-protocolInformationMatch	OBJECT IDENTIFIER ::= {id-mr 24}	
id-mr-uTCTimeMatch	OBJECT IDENTIFIER ::= {id-mr 25}	
id-mr-uTCTimeOrderingMatch	OBJECT IDENTIFIER ::= {id-mr 26}	
id-mr-generalizedTimeMatch	OBJECT IDENTIFIER ::= {id-mr 27}	
id-mr-generalizedTimeOrderingMatch	OBJECT IDENTIFIER ::= {id-mr 28}	
id-mr-integerFirstComponentMatch	OBJECT IDENTIFIER ::= {id-mr 29}	
id-mr-objectIdentifierFirstComponentMatch	OBJECT IDENTIFIER ::= {id-mr 30}	
id-mr-directoryStringFirstComponentMatch	OBJECT IDENTIFIER ::= {id-mr 31}	
id-mr-wordMatch	OBJECT IDENTIFIER ::= {id-mr 32}	
id-mr-keywordMatch	OBJECT IDENTIFIER ::= {id-mr 33}	
<i>-- id-mr-certificateExactMatch</i>	<i>OBJECT IDENTIFIER ::= {id-mr 34}</i>	
<i>-- id-mr-certificateMatch</i>	<i>OBJECT IDENTIFIER ::= {id-mr 35}</i>	
<i>-- id-mr-certificatePairExactMatch</i>	<i>OBJECT IDENTIFIER ::= {id-mr 36}</i>	
<i>-- id-mr-certificatePairMatch</i>	<i>OBJECT IDENTIFIER ::= {id-mr 37}</i>	
<i>-- id-mr-certificateListExactMatch</i>	<i>OBJECT IDENTIFIER ::= {id-mr 38}</i>	
<i>-- id-mr-certificateListMatch</i>	<i>OBJECT IDENTIFIER ::= {id-mr 39}</i>	
<i>-- id-mr-algorithmIdentifierMatch</i>	<i>OBJECT IDENTIFIER ::= {id-mr 40}</i>	
id-mr-storedPrefixMatch	OBJECT IDENTIFIER ::= {id-mr 41}	
<i>-- id-mr-attributeCertificateMatch</i>	<i>OBJECT IDENTIFIER ::= {id-mr 42}</i>	
<i>-- id-mr-readerAndKeyIDMatch</i>	<i>OBJECT IDENTIFIER ::= {id-mr 43}</i>	
<i>-- id-mr-attributeIntegrityMatch</i>	<i>OBJECT IDENTIFIER ::= {id-mr 44}</i>	obsolete
<i>-- id-mr-attributeCertificateExactMatch</i>	<i>OBJECT IDENTIFIER ::= {id-mr 45}</i>	
<i>-- id-mr-holderIssuerMatch</i>	<i>OBJECT IDENTIFIER ::= {id-mr 46}</i>	
id-mr-systemProposedMatch	OBJECT IDENTIFIER ::= {id-mr 47}	
id-mr-generalWordMatch	OBJECT IDENTIFIER ::= {id-mr 48}	
id-mr-approximateStringMatch	OBJECT IDENTIFIER ::= {id-mr 49}	
id-mr-ignoreIfAbsentMatch	OBJECT IDENTIFIER ::= {id-mr 50}	
id-mr-nullMatch	OBJECT IDENTIFIER ::= {id-mr 51}	
id-mr-zonalMatch	OBJECT IDENTIFIER ::= {id-mr 52}	
<i>-- id-mr-authAttIdMatch</i>	<i>OBJECT IDENTIFIER ::= {id-mr 53}</i>	
<i>-- id-mr-roleSpecCertIdMatch</i>	<i>OBJECT IDENTIFIER ::= {id-mr 54}</i>	
<i>-- id-mr-basicAttConstraintsMatch</i>	<i>OBJECT IDENTIFIER ::= {id-mr 55}</i>	
<i>-- id-mr-delegatedNameConstraintsMatch</i>	<i>OBJECT IDENTIFIER ::= {id-mr 56}</i>	
<i>-- id-mr-timeSpecMatch</i>	<i>OBJECT IDENTIFIER ::= {id-mr 57}</i>	
<i>-- id-mr-attDescriptorMatch</i>	<i>OBJECT IDENTIFIER ::= {id-mr 58}</i>	
<i>-- id-mr-acceptableCertPoliciesMatch</i>	<i>OBJECT IDENTIFIER ::= {id-mr 59}</i>	

ISO/CEI 9594-6:2005 (S)

```
-- id-mr-policyMatch          OBJECT IDENTIFIER ::= {id-mr 60}
-- id-mr-delegationPathMatch  OBJECT IDENTIFIER ::= {id-mr 61}
-- id-mr-pkiPathMatch         OBJECT IDENTIFIER ::= {id-mr 62}
id-mr-facsimileNumberMatch    OBJECT IDENTIFIER ::= {id-mr 63}
id-mr-facsimileNumberSubstringsMatch OBJECT IDENTIFIER ::= {id-mr 64}
-- id-mr-enhancedCertificateMatch OBJECT IDENTIFIER ::= {id-mr 65}
-- id-mr-sOAIentifierMatch    OBJECT IDENTIFIER ::= {id-mr 66}
-- id-mr-indirectIssuerMatch  OBJECT IDENTIFIER ::= {id-mr 67}
id-mr-uidpairmatch           OBJECT IDENTIFIER ::= {id-mr 68}

-- contexts --

id-avc-language              OBJECT IDENTIFIER ::= {id-avc 0}
id-avc-temporal              OBJECT IDENTIFIER ::= {id-avc 1}
id-avc-locale                OBJECT IDENTIFIER ::= {id-avc 2}
-- id-avc-attributeValueSecurityLabelContext OBJECT IDENTIFIER ::= {id-avc 3}
-- id-avc-attributeValueIntegrityInfoContext OBJECT IDENTIFIER ::= {id-avc 4}
id-avc-ldapAttributeOption   OBJECT IDENTIFIER ::= {id-avc 5}

END -- SelectedAttributeTypes
```

Anexo B

Recapitulación de tipos de atributos

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

Este anexo recapitula los tipos de atributos seleccionados referenciados o definidos en esta Especificación de directorio y muestra su relación jerárquica. Los atributos que comparten una sintaxis ASN.1 común aparecen sangrados bajo esa sintaxis, y los atributos que son subtipos de otros atributos aparecen sangrados bajo su supertipo. No se muestran los atributos colectivos que son subtipos de un atributo no colectivo conexo, pero el atributo conexo se señala con un asterisco (*). Los atributos de notificación se marcan con un signo de número (#).

<i>DirectoryString</i>	owner
name	roleOccupant
commonName	seeAlso
surname	
givenName	<i>FilterItem</i>
initials	filterItem #
generationQualifier	<i>AttributeCombination</i>
countryName	attributeCombinations #
localityName *	
stateOrProvinceName *	<i>ContextAssertion</i>
organizationName *	contextList #
organizationalUnitName *	
pseudonym	<i>ContextCombination</i>
title	contextCombinations #
dmdName	
streetAddress *	<i>HierarchySelections</i>
houseIdentifier	hierarchySelectList #
description	
businessCategory	<i>SearchControlOptions</i>
postalCode *	searchControlOptionsList #
postOfficeBox *	
physicalDeliveryOfficeName *	<i>ServiceControlOptions</i>
knowledgeInformation	serviceControlOptionsList #
<i>PrintableString</i>	<i>MultipleMatchingLocalities</i>
serialNumber	multipleMatchingLocalities
dnQualifier	
destinationIndicator	<i>MRMappings</i>
telephoneNumber *	proposedRelaxation
<i>NumericString</i>	<i>Guide</i>
x121Address	searchGuide
internationalISDNNumber *	
OBJECT IDENTIFIER	<i>EnhancedGuide</i>
communicationsService	enhancedSearchGuide
communicationsNetwork	
supportedApplicationContext	<i>PostalAddress</i>
dSAProblem #	postalAddress *
searchServiceProblem #	registeredAddress
serviceType #	
attributeTypeList #	<i>TelexNumber</i>
matchingRuleList #	telexNumber *
contextTypeList #	
appliedRelaxation #	<i>FacsimileTelephoneNumber</i>
<i>BIT STRING</i>	facsimileTelephoneNumber *
uniqueIdentifier	<i>PresentationAddress</i>
<i>NameAndOptionalUID</i>	presentationAddress
uniqueMember	
<i>DistinguishedName</i>	<i>ProtocolInformation</i>
distinguishedName	protocolInformation
member	
	<i>PreferredDeliveryMethod</i>
	preferredDeliveryMethod
	<i>UUIDPair</i>
	uUUIDPair

Anexo C

Límites superiores

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

Este anexo incluye todas las constricciones de valores de límite superior sugeridos en estas Especificaciones de directorio, en forma del módulo ASN.1 **UpperBounds**.

UpperBounds {joint-iso-itu-t ds(5) module(1) upperBounds(10) 5}

DEFINITIONS ::=

BEGIN

-- EXPORTS All --

-- The types and values defined in this module are exported for use in the other ASN.1 modules contained

-- within the Directory Specifications, and for the use of other applications which will use them to access

-- Directory services. Other applications may use them for their own purposes, but this will not constrain

-- extensions and modifications needed to maintain or improve the Directory service.

ub-answerback	INTEGER	::= 8
ub-business-category	INTEGER	::= 128
ub-common-name	INTEGER	::= 64
ub-content	INTEGER	::= 32768
ub-country-code	INTEGER	::= 4
ub-description	INTEGER	::= 1024
ub-destination-indicator	INTEGER	::= 128
ub-directory-string-first-component-match	INTEGER	::= 32768
ub-domainLocalID	INTEGER	::= 64
ub-international-isdn-number	INTEGER	::= 16
ub-knowledge-information	INTEGER	::= 32768
ub-labeledURI	INTEGER	::= 32768
ub-localeContextSyntax	INTEGER	::= 128
ub-locality-name	INTEGER	::= 128
ub-match	INTEGER	::= 128
ub-name	INTEGER	::= 64
ub-organization-name	INTEGER	::= 64
ub-organizational-unit-name	INTEGER	::= 64
ub-physical-office-name	INTEGER	::= 128
ub-post-office-box	INTEGER	::= 40
ub-postal-code	INTEGER	::= 40
ub-postal-line	INTEGER	::= 6
ub-postal-string	INTEGER	::= 30
ub-privacy-mark-length	INTEGER	::= 128
ub-pseudonym	INTEGER	::= 128
ub-saslMechanism	INTEGER	::= 64
ub-schema	INTEGER	::= 1024
ub-search	INTEGER	::= 32768
ub-serial-number	INTEGER	::= 64
ub-state-name	INTEGER	::= 128
ub-street-address	INTEGER	::= 128
ub-surname	INTEGER	::= 64
ub-tag	INTEGER	::= 64
ub-telephone-number	INTEGER	::= 32
ub-teletex-terminal-id	INTEGER	::= 1024
ub-telex-number	INTEGER	::= 14
ub-title	INTEGER	::= 64
ub-user-password	INTEGER	::= 128
ub-x121-address	INTEGER	::= 15

END -- UpperBounds

Anexo D

Índice alfabético (en inglés) de los atributos, reglas de concordancia y contextos

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

El presente anexo proporciona una lista alfabética (en inglés) de los atributos, reglas de concordancia y contextos definidos en la presente Especificación de directorio junto con las referencias a las subcláusulas en las cuales están definidos.

Applied Relaxation	5.12.16	Given Name	5.2.4
Approximate String Match	7.6.1	Hierarchy Select List	5.12.11
Attribute Combinations	5.12.7	House Identifier	5.3.5
Attribute Type List	5.12.4	Ignore if Absent Match	7.7.1
Bit String Match	7.2.4	Initials	5.2.5
Boolean Match	7.2.1	Integer First Component Match	7.4.1
Business Category	5.5.4	Integer Match	7.2.2
Case Exact Match	7.1.1	Integer Ordering Match	7.2.3
Case Exact Ordering Match	7.1.2	International ISDN Number	5.7.6
Case Exact Substrings Match	7.1.3	Keyword Match	7.5.2
Case Ignore List Match	7.1.7	Knowledge Information	5.1.1
Case Ignore List Substrings Match	7.1.8	Language Context	8.1
Case Ignore Match	7.1.1	LDAP Attribute Option Context	8.4
Case Ignore Ordering Match	7.1.2	Locale Context	8.3
Case Ignore Substrings Match	7.1.3	Locality Name	5.3.2
Common Name	5.2.2	Matching Rule List	5.12.5
Communications Network	5.7.10	Member	5.10.2
Communications Service	5.7.9	Multiple Matching Localities	5.12.14
Component Match	7.2.16	Name	5.2.1
Context Combinations	5.12.10	Null Match	7.7.2
Context List	5.12.9	Numeric String Match	7.1.4
Context Type List	5.12.8	Numeric String Ordering Match	7.1.5
Country Name	5.3.1	Numeric String Substrings Match	7.1.6
Description	5.5.1	Object Identifier First Component Match	7.4.2
Destination Indicator	5.7.8	Octet String Match	7.2.5
Directory String First Component Match	7.4.3	Octet String Ordering Match	7.2.6
Distinguished Name	5.10.1	Octet String Substrings Match	7.2.7
DMD name	5.11.1	Organizational Unit Name	5.4.2
DN Qualifier	5.2.8	OrganizationName	5.4.1
DSA Problem	5.12.1	Owner	5.10.4
Enhanced Search Guide	5.5.3	Physical Delivery Office Name	5.6.4
Facsimile Number Match	7.2.13	Post Office Box	5.6.3
Facsimile Number Substrings Match	7.2.14	Postal Address	5.6.1
Facsimile Telephone Number	5.7.4	Postal Code	5.6.2
Filter Item	5.12.6	Preferred Delivery Method	5.8.1
General Word Match	7.5.3	Presentation Address	5.9.1
Generation Qualifier	5.2.6	Presentation Address Match	7.2.10
Generalized Time Match	7.3.3	Proposed Relaxation	5.12.15
Generalized Time Ordering Match	7.3.4	Protocol Information	5.9.3

ISO/IEC 9594-6:2005 (E)

Protocol Information Match	7.2.12	Telephone Number	5.7.1
Pseudonym	5.2.10	Telephone Number Match	7.2.8
Registered Address	5.7.7	Telephone Number Substrings Match	7.2.9
Role Occupant	5.10.5	Teletex Terminal Identifier (deleted)	5.7.3
Search Guide	5.5.2	Telex Number	5.7.2
Search Control Options List	5.12.12	Temporal Context	8.2
Search Service Problem	5.12.2	Title	5.4.3
See Also	5.10.6	Unique Identifier	5.2.7
Serial Number	5.2.9	Unique Member	5.10.3
Service Control Options List	5.12.13	Unique Member Match	7.2.11
Service type	5.12.3	Universal Unique Identifier Pair	5.2.11
State or Province Name	5.3.3	UTC Time Match	7.3.1
Stored Prefix Match	7.1.9	UTC Time Ordering Match	7.3.2
Street Address	5.3.4	UUID Pair Match	7.2.15
Supported Application Context	5.9.2	Word Match	7.5.1
Surname	5.2.3	X.121 Address	5.7.5
System Proposed Match	7.3.5	Zonal Match	7.8

Anexo E

Ejemplos de reglas de concordancia zonal

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

NOTA – Las notas que siguen contienen ejemplos correspondientes a la definición de la concordancia zonal de 7.8 de esta Especificación de directorio. Para ayudar a identificar las situaciones correspondientes a los ejemplos, se mantiene el texto definitivo, pero en cursivas.

En la concordancia zonal, el mecanismo central implementa una correspondencia a partir de aserciones en cadena o combinaciones de aserciones, que se utiliza en el **filtro (filter)** de una operación búsqueda, a un conjunto de características irreducibles que pueden ser poseídas por los objetos y descritas por atributos en inserciones correspondientes. La correspondencia se expresa como un conjunto de elementos de filtro alternativos que sustituyen a los elementos de filtro del filtro original. Los atributos utilizados para representar las aserciones en el **filter** no son necesariamente los mismos utilizados para representar las características del objeto en la inserción. He aquí como podría tener lugar una concordancia zonal específica:

- Un usuario busca un abonado telefónico, un tal Sr. Smithers que vive en Bracknell, utiliza un filtro: `{{locality=Bracknell} AND {surname=Smithers}}`.
- El directorio contiene una correspondencia geográfica (denominada *nomenclátor*) que hace corresponder Bracknell a códigos postales (por ejemplo, RG12 2JL) que sirven como zonas en el área de Bracknell, en efecto convirtiendo el filtro a `{{zone=b1} OR {zone= b2}... } AND {surname=Smithers}`. Aquí, b_1 , b_2 , ..., b_n son el conjunto de códigos postales que representan Bracknell; cada residencia individual tiene un único código postal, mientras que un gran edificio o emplazamiento podría tener más de uno. La concordancia trata de localizar una persona del apellido (surname) indicado, cuya ubicación geográfica comparte una zona común con b_1 o b_2 ...
- Si la búsqueda es infructuosa, la correspondencia puede automáticamente relajarse para incluir más zonas (es decir, códigos postales adyacentes); se podría quizá de este modo encontrar un abonado denominado Smithers que vive en el pueblo de Newell Green (inmediatamente adyacente a Bracknell).

Una regla de concordancia por correspondencia puede entender nombres alternativos e información redundante, y puede combinar múltiples predicados, por ejemplo, `{{locality=Newton} AND {locality=Cumbria}}`; puede incluso identificar múltiples componentes en un solo predicado, por ejemplo, `{locality="Newton, Cumbria"}`. Así, el ejemplo de correspondencia puede también funcionar en el caso siguiente:

- `{{locality=Bullbrook} AND {surname=Smithers}}`
(Aquí Bullbrook es un distrito de Bracknell)
- `{{locality=Bracknell} AND {locality=Bullbrook} AND {surname=Smithers}}`
- `{{locality=Bullbrook, Bracknell} AND {surname=Smithers}}`
- `{{locality=Berks} AND {locality=Bracknell} AND {locality=Bullbrook} AND {surname=Smithers}}`
(Bracknell queda dentro de los límites del viejo condado de Berkshire, abreviado Berks)
- `{{locality=Berkshire} AND {locality=Bracknell} AND {locality=Bullbrook} AND {surname=Smithers}}`
- `{{locality=East Berks} AND {locality=Bracknell Forest} AND {surname=Smithers}}`
(La nueva área administrativa regional en la que queda Bracknell se llama East Berks[hire]; el distrito administrativo local se llama Bracknell Forest)
- `{{postcode=RG12 2JL} AND {surname=Smithers}}`
(RG12 2JL es uno de los aproximadamente 20 códigos postales de Bullbrook)

Las reglas de concordancia zonal son reglas de concordancia por correspondencia que se ocupan de la concordancia geográfica. Se basan en un diccionario de nombres de localidades denominado nomenclátor. Un índice geográfico en general comprenderá (es decir, proporcionará una base de datos geográficos relativos a) un dominio que comprende un solo país o región. Una petición de búsqueda geográfica deberá interpretarse en términos de un nomenclátor específico. Un índice geográfico relaciona principalmente cadenas de nombres de lugares con lugares denominados, identificados por una o más cadenas de nombres de lugares. Ejemplos de lugares denominados en Gran Bretaña, como los identificados por cadenas de nombres de lugar son "Mongworthy" en Devon, "Offleyhoo" en Hertfordshire, "Thames Valley" y "London".

Algunas cadenas de nombres de lugares se corresponden directamente con un solo lugar denominado, pero esto no es siempre posible. Ejemplos de nombres de lugares que no identifican lugares son "Newton", "Lees", porque cada uno de estos nombres corresponde a múltiples lugares denominados. Por tanto, puede ser necesario que un lugar denominado se identifique con múltiples nombres de lugar distintos; por ejemplo, los siguientes son tres nombres denominados: ("Newton" "Tattenhall" "Cheshire"), ("Newton" "Chester" "Cheshire"), ("Newton" "Cumbria"), donde la agrupación de nombres de lugar es indicada por los paréntesis.

Un nombre de lugar puede internamente tener múltiples componentes, por ejemplo, "London Heathrow", "Newton Abbott", pero cada uno se cuenta como una sola cadena, sea porque el nombre es incompleto, incluso localmente, sin todos sus componentes, o porque un componente (por ejemplo, "Abbott") no es semánticamente un nombre de lugar (no se recoge ningún nombre de lugar "Abbott" en los nomenclátor estándar).

Un lugar denominado puede también ser identificable por un subconjunto de sus múltiples nombres, por ejemplo ("Newton" "Tattenhall") puede definir adecuadamente el lugar antes mencionado. Sin embargo, en este caso, ("Newton" "Tattenhall" "Cheshire") puede ser un agrupamiento más útil, por analogía con Newtons que sólo necesiten calificativo de condado, por ejemplo ("Newton" "Cumbria").

Sigue a continuación una declaración más formal de la concordancia zonal subyacente modelo:

- a) *La concordancia zonal se basa en la existencia de uno o más nomenclátors (gazetteers) que son soportados a tal fin por los DSA. Un nomenclátor es un diccionario geográfico que comprende, como dominio suyo, un país o región denominada, soportada por una base de datos adecuada. La selección del dominio para una búsqueda específica se lleva a cabo por medios locales. Por ejemplo, un nomenclátor podría comprender el territorio principal de Gran Bretaña (Inglaterra, Escocia, Gales) con sus islas próximas. El nomenclátor contiene nombres de lugares y sus características, incluidas listas de lugares denominados concordantes. Es soportado por mecanismos para determinar y cotejar las características de los nombres de lugares indicados por atributos de localidad combinables, y es completamente independiente del DIT. En la figura E.1 la región es la línea exterior marcada con trazo grueso.*
- b) *La región comprendida por un nomenclátor contiene lugares. En la figura E.1 la región es la línea exterior marcada por límites correspondientes a letras. Un lugar es un área geográfica denominada reconocible; los lugares pueden superponerse, e incluso pueden extenderse algo más allá del límite de la región (como F en la figura E.1). Ejemplos de lugares son England, Berkshire, Bracknell, Bullbrook (estos cuatro se contienen progresivamente entre sí), y Thames Valley (que incluye parte de Berkshire, pero se extiende más allá del mismo). Los lugares que son identificables por referencia al nomenclátor se llaman *lugares denominados*.*
- c) *El propio índice geográfico se basa en cadenas que son nombres de lugar (por ejemplo, "England", "Berkshire", "Bracknell", "Bullbrook", "Thames Valley"). Éstos se utilizan para identificar (o denominar) lugares denominados. El nombre de un lugar denominado puede ser:*
 - *Un nombre de lugar único, posiblemente en más de una palabra, por ejemplo "Newton Abbott".*
 - *Un conjunto de nombres de lugar, donde en general un nombre de lugar corresponde a un área más grande (por ejemplo, "Cumbria") y califica un nombre de lugar que corresponde (en el contexto) a un área más pequeña (por ejemplo, "Newton").*

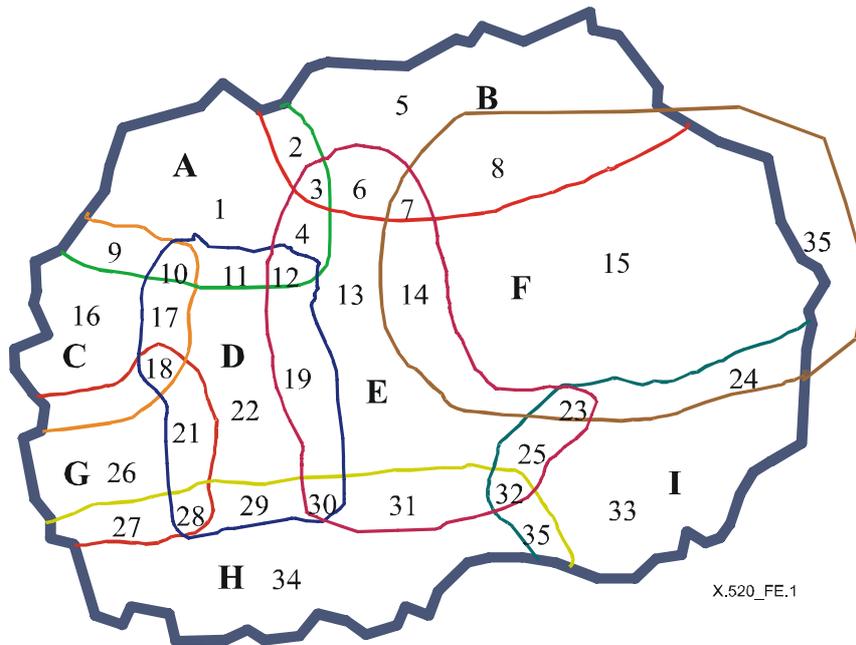


Figura E.1 – Regiones, lugares y zonas

En general, en lugar denominado debe asociarse en el nomenclátor con los nombres de lugares de mayor escala que lo contienen, aun si éstos no son necesarios para una identificación inequívoca. Por ejemplo, el nomenclátor tendría que indicar que la ciudad de Newton Abbott es accesible como "Newton Abbott" o como ("Newton Abbott" "Devon") y estaría por tanto asociada con el nombre de lugar "Devon" (que, como así ocurre, es sinónimo de "Devonshire").

Anexo F

Enmiendas y corrigenda

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

Esta edición de la presente Especificación de directorio incluye el siguiente proyecto de enmienda que ha sido votado y aprobado por ISO/CEI:

- Enmienda 3 con la que se introducen mejoras para potenciar al máximo el alineamiento entre X.500 y LDAP.

Esta edición de esta Especificación de directorio incluye los corrigenda técnicos que subsanan los defectos indicados en los siguientes informes de defectos: 287, 288, 312 y 313.

SERIES DE RECOMENDACIONES DEL UIT-T

Serie A	Organización del trabajo del UIT-T
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	Gestión de las telecomunicaciones, incluida la RGT y el mantenimiento de redes
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, comunicaciones de sistemas abiertos y seguridad
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