ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

X.521
(10/2012)

SERIES X: DATA NETWORKS, OPEN SYSTEM COMMUNICATIONS AND SECURITY

Directory

Information technology – Open Systems Interconnection – The Directory: Selected object classes

Recommendation ITU-T X.521

ITU-T X-SERIES RECOMMENDATIONS

DATA NETWORKS, OPEN SYSTEM COMMUNICATIONS AND SECURITY

PUBLIC DATA NETWORKS	
Services and facilities	X.1–X.19
Interfaces	X.20–X.49
Transmission, signalling and switching	X.50–X.89
Network aspects	X.90–X.149
Maintenance	X.150–X.179
Administrative arrangements	X.180–X.199
OPEN SYSTEMS INTERCONNECTION	
Model and notation	X.200–X.209
Service definitions	X.210–X.219
Connection-mode protocol specifications	X.220–X.229
Connectionless-mode protocol specifications	X.230–X.239
PICS proformas	X.240–X.259
Protocol Identification	X.260–X.269
Security Protocols	X.270–X.279
Layer Managed Objects	X.280–X.289
Conformance testing	X.290–X.299
INTERWORKING BETWEEN NETWORKS	
General	X.300–X.349
Satellite data transmission systems	X.350–X.369
IP-based networks	X.370–X.379
MESSAGE HANDLING SYSTEMS	X.400–X.499
DIRECTORY	**X.500–X.599**
OSI NETWORKING AND SYSTEM ASPECTS	
Networking	X.600–X.629
Efficiency	X.630–X.639
Quality of service	X.640–X.649
Naming, Addressing and Registration	X.650–X.679
Abstract Syntax Notation One (ASN.1)	X.680–X.699
OSI MANAGEMENT	
Systems management framework and architecture	X.700–X.709
Management communication service and protocol	X.710–X.719
Structure of management information	X.720–X.729
Management functions and ODMA functions	X.730–X.799
SECURITY	X.800–X.849
OSI APPLICATIONS	
Commitment, concurrency and recovery	X.850–X.859
Transaction processing	X.860–X.879
Remote operations	X.880–X.889
Generic applications of ASN.1	X.890–X.899
OPEN DISTRIBUTED PROCESSING	X.900–X.999
INFORMATION AND NETWORK SECURITY	X.1000–X.1099
SECURE APPLICATIONS AND SERVICES	X.1100–X.1199
CYBERSPACE SECURITY	X.1200–X.1299
SECURE APPLICATIONS AND SERVICES	X.1300–X.1399
CYBERSECURITY INFORMATION EXCHANGE	X.1500–X.1599

For further details, please refer to the list of ITU-T Recommendations.

INTERNATIONAL STANDARD ISO/IEC 9594-7
RECOMMENDATION ITU-T X.521

Information technology – Open Systems Interconnection –
The Directory: Selected object classes

Summary

Recommendation ITU-T X.521 | ISO/IEC 9594-7 defines a number of selected object classes and name forms which may be found useful across a range of applications of the Directory. An object class definition specifies the attribute types which are relevant to the objects of that class. A name form definition specifies the attributes to be used in forming names for the objects of a given class.

History

Edition	Recommendation	Approval	Study Group
1.0	ITU-T X.521	1988-11-25	
2.0	ITU-T X.521	1993-11-16	7
3.0	ITU-T X.521	1997-08-09	7
3.1	ITU-T X.521 (1997) Amd. 1	2000-03-31	7
3.2	ITU-T X.521 (1997) Technical Cor. 1	2001-02-02	7
4.0	ITU-T X.521	2001-02-02	7
5.0	ITU-T X.521	2005-08-29	17
5.1	ITU-T X.521 (2005) Cor. 1	2012-04-13	17
6.0	ITU-T X.521	2008-11-13	17
6.1	ITU-T X.521 (2008) Cor. 1	2012-04-13	17
7.0	ITU-T X.521	2012-10-14	17

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at http://www.itu.int/ITU-T/ipr/.

CONTENTS

Introduction

This Recommendation | International Standard, together with other Recommendations | International Standards, has been produced to facilitate the interconnection of information processing systems to provide directory services. A set of such systems, together with the directory information that they hold, can be viewed as an integrated whole, called the *Directory*. The information held by the Directory, collectively known as the Directory Information Base (DIB), is typically used to facilitate communication between, with or about objects such as application entities, people, terminals, and distribution lists.

The Directory plays a significant role in Open Systems Interconnection, whose aim is to allow, with a minimum of technical agreement outside of the interconnection standards themselves, the interconnection of information processing systems:

– from different manufacturers;

– under different managements;

– of different levels of complexity; and

– of different ages.

This Recommendation | International Standard defines a number of attribute sets and object classes which may be found useful across a range of applications of the Directory.

This Recommendation | International Standard provides the foundation frameworks upon which industry profiles can be defined by other standards groups and industry forums. Many of the features defined as optional in these frameworks may be mandated for use in certain environments through profiles. This seventh edition technically revises and enhances the sixth edition of this Recommendation | International Standard.

This seventh edition specifies versions 1 and 2 of the Directory protocols.

The first and second editions specified only version 1. Most of the services and protocols specified in this edition are designed to function under version 1. However some enhanced services and protocols, e.g., signed errors, will not function unless all Directory entities involved in the operation have negotiated version 2. Whichever version has been negotiated, differences between the services and between the protocols defined in the six editions, except for those specifically assigned to version 2, are accommodated using the rules of extensibility defined in Rec. ITU-T X.519 | ISO/IEC 9594-5.

Annex A, which is an integral part of this Recommendation | International Standard, provides an ASN.1 module containing all of the type and value definitions which appear in this Recommendation | International Standard.

Annex B, which is not an integral part of this Recommendation | International Standard, provides some common naming and structure rules which may or may not be used by administrative authorities.

Annex C, which is not an integral part of this Recommendation | International Standard, lists the amendments and defect reports that have been incorporated to form this edition of this Recommendation | International Standard.

INTERNATIONAL STANDARD
RECOMMENDATION ITU-T

<div align="center">

Information technology – Open Systems Interconnection – The Directory: Selected object classes

</div>

<div align="center">

SECTION 1 – GENERAL

</div>

1   Scope

This Recommendation | International Standard defines a number of object classes and name forms which may be found useful across a range of applications of the Directory. The definition of an object class involves listing a number of attribute types which are relevant to objects of that class. The definition of a name form involves naming the object class to which it applies and listing the attributes to be used in forming names for objects of that class. These definitions are used by the administrative authority which is responsible for the management of the directory information.

Any administrative authority can define its own object classes or subclasses and name forms for any purpose.

> NOTE 1 – Those definitions may or may not use the notation specified in Rec. ITU-T X.501 | ISO/IEC 9594-2.

> NOTE 2 – It is recommended that an object class defined in this Recommendation | International Standard, or a subclass derived from one, or a name form defined in this Recommendation | International Standard, be used in preference to the generation of a new one, whenever the semantics is appropriate for the application.

Administrative authorities may support some or all the selected object classes and name forms, and may also add additional ones.

All administrative authorities shall support the object classes which the directory uses for its own purpose (the top, alias and Directory system agent (DSA) object classes).

2   Normative references

The following Recommendations and International Standards contain provisions which, through reference in this text, constitute provisions of this Recommendation | International Standard. At the time of publication, the editions indicated were valid. All Recommendations and Standards are subject to revision, and parties to agreements based on this Recommendation | International Standard are encouraged to investigate the possibility of applying the most recent editions of the Recommendations and Standards listed below. Members of IEC and ISO maintain registers of currently valid International Standards. The Telecommunication Standardization Bureau of the ITU maintains a list of currently valid ITU-T Recommendations.

2.1   Identical Recommendations | International Standards

–   Recommendation ITU-T X.200 (1994) | ISO/IEC 7498-1:1994, *Information technology – Open Systems Interconnection – Basic Reference Model: The Basic Model.*

–   Recommendation ITU-T X.500 (2012) | ISO/IEC 9594-1:2014, *Information technology – Open Systems Interconnection – The Directory: Overview of concepts, models and services.*

–   Recommendation ITU-T X.501 (2012) | ISO/IEC 9594-2:2014, *Information technology – Open Systems Interconnection – The Directory: Models.*

–   Recommendation ITU-T X.509 (2012) | ISO/IEC 9594-8:2014, *Information technology – Open Systems Interconnection – The Directory: Public-key and attribute certificate frameworks.*

–   Recommendation ITU-T X.511 (2012) | ISO/IEC 9594-3:2014, *Information technology – Open Systems Interconnection – The Directory: Abstract service definition.*

–   Recommendation ITU-T X.518 (2012) | ISO/IEC 9594-4:2014, *Information technology – Open Systems Interconnection – The Directory: Procedures for distributed operation.*

–   Recommendation ITU-T X.519 (2012) | ISO/IEC 9594-5:2014, *Information technology – Open Systems Interconnection – The Directory: Protocol specifications.*

–   Recommendation ITU-T X.520 (2012) | ISO/IEC 9594-6:2014, *Information technology – Open Systems Interconnection – The Directory: Selected attribute types.*

–   Recommendation ITU-T X.525 (2012) | ISO/IEC 9594-9:2014, *Information technology – Open Systems Interconnection – The Directory: Replication.*

– Recommendation ITU-T X.668 (2008) | ISO/IEC 9834-9:2008, *Information technology – Open Systems Interconnection – Procedures for the operation of OSI Registration Authorities: Registration of object identifier arcs for applications and services using tag-based identification.*

– Recommendation ITU-T X.680 (2008) | ISO/IEC 8824-1:2008, *Information technology – Abstract Syntax Notation One (ASN.1): Specification of basic notation.*

– Recommendation ITU-T X.681 (2008) | ISO/IEC 8824-2:2008, *Information technology – Abstract Syntax Notation One (ASN.1): Information object specification.*

– Recommendation ITU-T X.682 (2008) | ISO/IEC 8824-3:2008, *Information technology – Abstract Syntax Notation One (ASN.1): Constraint specification.*

– Recommendation ITU-T X.683 (2008) | ISO/IEC 8824-4:2008, *Information technology – Abstract Syntax Notation One (ASN.1): Parameterization of ASN.1 specifications.*

3   Definitions

For the purposes of this Recommendation | International Standard, the following definitions apply.

3.1   Communication Model definitions

The following terms are defined in Rec. ITU-T X.519 | ISO/IEC 9594-5:

a) *application-entity*;

b) *application process*.

3.2   Directory Model definitions

The following terms are defined in Rec. ITU-T X.501 | ISO/IEC 9594-2:

a) *attribute*;

b) *attribute type*;

c) *Directory information tree (DIT)*;

d) *Directory system agent (DSA)*;

e) *attribute set*;

f) *entry*;

g) *name*;

h) *object class*;

i) *subclass*;

j) *name form*;

k) *structure rule*.

4   Conventions

The term "Directory Specification" (as in "this Directory Specification") shall be taken to mean Rec. ITU-T X.521 | ISO/IEC 9594-7. The term "Directory Specifications" shall be taken to mean the ITU-T X.500-series Recommendations and all parts of ISO/IEC 9594.

This Directory Specification uses the term *first edition systems* to refer to systems conforming to the first edition of the Directory Specifications, i.e., the 1988 edition of the CCITT X.500-series Recommendations and the ISO/IEC 9594:1990 edition.

This Directory Specification uses the term *second edition systems* to refer to systems conforming to the second edition of the Directory Specifications, i.e., the 1993 edition of the ITU-T X.500-series Recommendations and the ISO/IEC 9594:1995 edition.

This Directory Specification uses the term *third edition systems* to refer to systems conforming to the third edition of the Directory Specifications, i.e., the 1997 edition of the ITU-T X.500-series Recommendations and the ISO/IEC 9594:1998 edition.

This Directory Specification uses the term *fourth edition systems* to refer to systems conforming to the fourth edition of the Directory Specifications, i.e., the 2001 editions of Recs ITU-T X.500, ITU-T X.501, ITU-T X.511, ITU-T X.518, ITU-T X.519, ITU-T X.520, ITU-T X.521, ITU-T X.525, and ITU-T X.530, the 2000 edition of Rec. ITU-T X.509, and parts 1-10 of the ISO/IEC 9594:2001 edition.

This Directory Specification uses the term *fifth edition systems* to refer to systems conforming to the fifth edition of the Directory Specifications, i.e., the 2005 edition of the ITU-T X.500-series Recommendations and the ISO/IEC 9594:2005 edition.

This Directory Specification uses the term *sixth edition systems* to refer to systems conforming to the sixth edition of the Directory Specifications, i.e., the 2008 edition of the ITU-T X.500-series Recommendations and the ISO/IEC 9594:2008 edition.

This Directory Specification uses the term *seventh edition systems* to refer to systems conforming to the seventh edition of the Directory Specifications, i.e., the 2012 edition of the ITU-T X.500-series Recommendations and the ISO/IEC 9594:2014 edition.

This Directory Specification presents ASN.1 notation in the bold Courier New typeface. When ASN.1 types and values are referenced in normal text, they are differentiated from normal text by presenting them in the bold Courier New typeface. The names of procedures, typically referenced when specifying the semantics of processing, are differentiated from normal text by displaying them in bold Times New Roman. Access control permissions are presented in italicized Times New Roman.

Object classes and name forms are defined in this Directory Specification as values of the **OBJECT-CLASS** and **NAME-FORM** information object classes defined in Rec. ITU-T X.501 | ISO/IEC 9594-2.

SECTION 2 – SELECTED OBJECT CLASSES

5 Definition of useful attribute sets

5.1 Telecommunication attribute set

This set of attributes is used to define those which are commonly used for business communications.

```
TelecommunicationAttributeSet ATTRIBUTE ::=
  {facsimileTelephoneNumber |
   internationalISDNNumber |
   telephoneNumber |
-- teletexTerminalIdentifier  (Attribute type has been deleted)
-- teletexTerminalIdentifier |  Attribute type has been deleted
   telexNumber |
   preferredDeliveryMethod |
   destinationIndicator |
   registeredAddress |
   x121Address}
```

5.2 Postal attribute set

This set of attributes is used to define those which are directly associated with postal delivery.

```
PostalAttributeSet ATTRIBUTE ::=
  {physicalDeliveryOfficeName |
   postalAddress |
   postalCode |
   postOfficeBox |
   streetAddress}
```

5.3   Locale attribute set

This set of attributes is used to define those which are commonly used for search purposes to indicate the locale of an object.

```
LocaleAttributeSet ATTRIBUTE ::=
  {localityName |
   stateOrProvinceName |
   streetAddress}
```

5.4   Organizational attribute set

This set of attributes is used to define the attributes that an organization or organizational unit may typically possess.

```
OrganizationalAttributeSet ATTRIBUTE ::=
  {description |
   LocaleAttributeSet |
   PostalAttributeSet |
   TelecommunicationAttributeSet |
   businessCategory |
   seeAlso |
   searchGuide |
   userPassword}
```

6   Definition of selected object classes

6.1   Country

A *Country* object class is used to define country entries in the DIT.

```
country OBJECT-CLASS ::= {
  SUBCLASS OF    {top}
  MUST CONTAIN   {countryName}
  MAY CONTAIN    {description |
                  searchGuide}
  ID             id-oc-country }
```

6.2   Locality

The *Locality* object class is used to define locality in the DIT.

```
locality OBJECT-CLASS ::= {
  SUBCLASS OF    {top}
  MAY CONTAIN    {description |
                  searchGuide |
                  LocaleAttributeSet |
                  seeAlso}
  ID             id-oc-locality }
```

At least one of Locality Name or State or Province Name shall be present.

6.3   Organization

The *Organization* object class is used to define organization entries in the DIT.

```
organization OBJECT-CLASS ::= {
  SUBCLASS OF    {top}
  MUST CONTAIN   {organizationName}
  MAY CONTAIN    {OrganizationalAttributeSet}
  ID             id-oc-organization }
```

6.4 Organizational Unit

The *Organizational Unit* object class is used to define entries representing subdivisions of organizations.

```
organizationalUnit OBJECT-CLASS ::= {
  SUBCLASS OF    {top}
  MUST CONTAIN   {organizationalUnitName}
  MAY CONTAIN    {OrganizationalAttributeSet}
  ID             id-oc-organizationalUnit }
```

6.5 Person

The *Person* object class is used to define entries representing people generically.

```
person OBJECT-CLASS ::= {
  SUBCLASS OF    {top}
  MUST CONTAIN   {commonName |
                  surname}
  MAY CONTAIN    {description |
                  telephoneNumber |
                  userPassword |
                  seeAlso}
  ID             id-oc-person }
```

6.6 Organizational Person

The *Organizational Person* object class is used to define entries representing people employed by, or in some other important way associated with, an organization.

```
organizationalPerson OBJECT-CLASS ::= {
  SUBCLASS OF  {person}
  MAY CONTAIN  {LocaleAttributeSet |
                PostalAttributeSet |
                TelecommunicationAttributeSet |
                organizationalUnitName |
                title}
  ID             id-oc-organizationalPerson }
```

6.7 Organizational Role

The *Organizational Role* object class is used to define entries representing an organizational role, i.e., a position or role within an organization. An organizational role is normally considered to be filled by a particular organizational person. Over its lifetime, however, an organizational role may be filled by a number of different organizational people in succession. In general, an organizational role may be filled by a person or a non-human entity.

```
organizationalRole OBJECT-CLASS ::= {
  SUBCLASS OF    {top}
  MUST CONTAIN   {commonName}
  MAY CONTAIN    {description |
                  LocaleAttributeSet |
                  organizationalUnitName |
                  PostalAttributeSet |
                  preferredDeliveryMethod |
                  roleOccupant |
                  seeAlso |
                  TelecommunicationAttributeSet}
  ID             id-oc-organizationalRole }
```

6.8 Group Of Names

The *Group Of Names* object class is used to define entries representing an unordered set of names which represent individual objects or other groups of names. The membership of a group is static, i.e., it is explicitly modified by administrative action, rather than dynamically determined each time the group is referred to.

The membership of a group can be reduced to a set of individual object's names by replacing each group with its membership. This process could be carried out recursively until all constituent group names have been eliminated, and only the names of individual objects remain.

```
groupOfNames OBJECT-CLASS ::= {
  SUBCLASS OF    {top}
  MUST CONTAIN   {commonName | member}
  MAY CONTAIN    {description |
                  organizationName |
                  organizationalUnitName |
                  owner |
                  seeAlso |
                  businessCategory}
  ID             id-oc-groupOfNames }
```

6.9   Group Of Unique Names

The *Group Of Unique Names* object class is used to define entries representing an unordered set of names whose integrity can be assured and which represent individual objects or other groups of names. The membership of a group is static, i.e., it is explicitly modified by administrative action, rather than dynamically determined each time the group is referred to.

```
groupOfUniqueNames OBJECT-CLASS ::= {
  SUBCLASS OF    {top}
  MUST CONTAIN   {commonName |
                  uniqueMember}
  MAY CONTAIN    {description |
                  organizationName |
                  organizationalUnitName |
                  owner |
                  seeAlso |
                  businessCategory}
  ID             id-oc-groupOfUniqueNames }
```

6.10   Residential Person

The *Residential Person* object class is used to define entries representing a person in the residential environment.

```
residentialPerson OBJECT-CLASS ::= {
  SUBCLASS OF    {person}
  MUST CONTAIN   {localityName}
  MAY CONTAIN    {LocaleAttributeSet |
                  PostalAttributeSet |
                  preferredDeliveryMethod |
                  TelecommunicationAttributeSet |
                  businessCategory}
  ID             id-oc-residentialPerson }
```

6.11   Application Process

The *Application Process* object class is used to define entries representing application processes. An application process is an element within a real open-system which performs the information processing for a particular application (see Rec. ITU-T X.200 | ISO/IEC 7498-1).

```
applicationProcess OBJECT-CLASS ::= {
  SUBCLASS OF    {top}
  MUST CONTAIN   {commonName}
  MAY CONTAIN    {description |
                  localityName |
                  organizationalUnitName |
                  seeAlso}
  ID             id-oc-applicationProcess }
```

6.12 Application Entity

The *Application Entity* object class is used to define entries representing application-entities. An application-entity consists of those aspects of an application process pertinent to communications.

```
applicationEntity OBJECT-CLASS ::= {
  SUBCLASS OF    {top}
  MUST CONTAIN   {commonName |
                  presentationAddress}
  MAY CONTAIN    {description |
                  localityName |
                  organizationName |
                  organizationalUnitName |
                  seeAlso |
                  supportedApplicationContext}
  ID             id-oc-applicationEntity }
```

NOTE – If an application-entity is represented as a Directory object that is distinct from an application process, the commonName attribute is used to carry the value of the Application Entity Qualifier.

6.13 DSA

The *DSA* object class is used to define entries representing DSAs. A DSA is as defined in Rec. ITU-T X.501 | ISO/IEC 9594-2.

```
dSA OBJECT-CLASS ::= {
  SUBCLASS OF    {applicationEntity}
  MAY CONTAIN    {knowledgeInformation}
  ID             id-oc-dSA }
```

6.14 Device

The *Device* object class is used to define entries representing devices. A device is a physical unit which can communicate, such as a modem, disk drive, etc.

```
device OBJECT-CLASS ::= {
  SUBCLASS OF    {top}
  MUST CONTAIN   {commonName}
  MAY CONTAIN    {description |
                  localityName |
                  organizationName |
                  organizationalUnitName |
                  owner |
                  seeAlso |
                  serialNumber}
  ID             id-oc-device }
```

NOTE – At least one of localityName, serialNumber, owner, should be included. The choice is dependent on device type.

6.15 Strong Authentication User

The *Strong Authentication User* object class is used to define entries for objects which participate in strong authentication, as defined in Rec. ITU-T X.509 | ISO/IEC 9594-8.

```
strongAuthenticationUser OBJECT-CLASS ::= {
  SUBCLASS OF    {top}
  KIND           auxiliary
  MUST CONTAIN   {userCertificate}
  ID             id-oc-strongAuthenticationUser }
```

NOTE – Use of this object class has been deprecated in favour of the pkiUser and pkiCA object classes defined in Rec. ITU-T X.509 | ISO/IEC 9594-8. Implementations that use strongAuthenticationUser, certificationAuthority and certificationAuthorityv2 object classes are still conformant to the standard, although new implementations are strongly recommended to move to the pkiUser and pkiCA object classes.

6.16   User Security Information

The *User Security Information* object class is used to define entries for objects which need to indicate security information associated with them as defined in Rec. ITU-T X.509 | ISO/IEC 9594-8.

```
userSecurityInformation OBJECT-CLASS ::= {
  SUBCLASS OF   {top}
  KIND          auxiliary
  MAY CONTAIN   {supportedAlgorithms}
  ID            id-oc-userSecurityInformation }
```

6.17   User Password

The *User Password* object class is used to define entries for objects that maintain a user password (**userPwd**).

```
userPwdClass  OBJECT-CLASS  ::=  {
  KIND          auxiliary
  MAY CONTAIN   { userPwd }
  ID            id-oc-userPwdClass }
```

6.18   Certification Authority

The *Certification Authority* object class is used to define entries for objects which act as certification authorities, as defined in Rec. ITU-T X.509 | ISO/IEC 9594-8.

```
certificationAuthority OBJECT-CLASS ::= {
  SUBCLASS OF   {top}
  KIND          auxiliary
  MUST CONTAIN  {cACertificate |
                 certificateRevocationList |
                 authorityRevocationList}
  MAY CONTAIN   {crossCertificatePair}
  ID            id-oc-certificationAuthority }
```

> NOTE – Use of this object class has been deprecated in favour of the pkiUser and pkiCA object classes defined in Rec. ITU-T X.509 | ISO/IEC 9594-8. Implementations that use strongAuthenticationUser, certificationAuthority and certificationAuthorityv2 object classes are still conformant to the standard, although new implementations are strongly recommended to move to the pkiUser and pkiCA object classes.

6.19   Certification Authority-V2

The *Certification Authority-V2* object class is used to define entries for objects which act as certification authorities and can support the delta revocation list as defined in Rec. ITU-T X.509 | ISO/IEC 9594-8.

```
certificationAuthority-V2 OBJECT-CLASS ::= {
  SUBCLASS OF   {certificationAuthority}
  KIND          auxiliary
  MAY CONTAIN   {deltaRevocationList}
  ID            id-oc-certificationAuthority-V2 }
```

> NOTE – Use of this object class has been deprecated in favour of the pkiUser and pkiCA object classes defined in Rec. ITU-T X.509 | ISO/IEC 9594-8. Implementations that use strongAuthenticationUser, certificationAuthority and certificationAuthorityv2 object classes are still conformant to the standard, although new implementations are strongly recommended to move to the pkiUser and pkiCA object classes.

6.20   DMD

The *DMD* object class is used to define DMD entries in the DIT.

```
dMD OBJECT-CLASS ::= {
  SUBCLASS OF   {top}
  MUST CONTAIN  {dmdName}
  MAY CONTAIN   {OrganizationalAttributeSet}
  ID            id-oc-dmd }
```

6.21 OID Obj1

The *OID Obj1* object class is used to define a top level object identifier component entry in the DIT.

```
oidC1obj OBJECT-CLASS ::= {
  SUBCLASS OF    {top}
  MUST CONTAIN   {oidC}
  ID             id-oc-oidC1obj }
```

6.22 OID Obj2

The *OID Obj2* object class is used to define second level object identifier component entries in the DIT.

```
oidC2obj OBJECT-CLASS ::= {
  SUBCLASS OF    {top}
  MUST CONTAIN   {oidC}
  ID             id-oc-oidC2obj }
```

6.23 OID ObjC

The *OID ObjC* object class is used to define the third- or lower-level object identifier component entries in the DIT.

NOTE – A tag-based object identifier typically has only three levels, where the first two values are { 2 27 }.

```
oidCobj OBJECT-CLASS ::= {
  SUBCLASS OF    {top}
  MUST CONTAIN   {oidC}
  ID             id-oc-oidCobj }
```

6.24 OID root

The *OID root* object class is moved to here from Rec. ITU-T X.660 | ISO/IEC 9834-1.

```
oidRoot OBJECT-CLASS ::= {
  SUBCLASS OF    {alias}
  MUST CONTAIN   { oidC1 | oidC2 | oidC}
  ID             id-oidRoot }
```

6.25 OID arc

The *OID arc* object class is moved to here from Rec. ITU-T X.660 | ISO/IEC 9834-1.

```
oidArc OBJECT-CLASS ::= {
  SUBCLASS OF    {alias}
  MUST CONTAIN   {oidC}
  ID             id-oidArc }
```

6.26 URN ObjC

The *URN ObjC* object class is used to define the entries of an URN subtree as determined

```
urnCobj OBJECT-CLASS ::= {
  SUBCLASS OF    {top}
  MUST CONTAIN   { urnC }
  ID             id-oc-urnCobj }
```

6.27 ISO Tag Information

The *ISO Tag Information* auxiliary object class may be used to add tag-based attribute types to an entry holding information associated with a specific ISO tag.

```
isoTagInfo OBJECT-CLASS ::= {
  SUBCLASS OF  { top }
  KIND         auxiliary
  MAY CONTAIN  { tagOid |
                 tagAfi |
                 uii |
                 uiiInUrn |
```

```
                  contentUrl |
                  tagLocation }
  ID          id-oc-isoTagInfo }
```

6.28   ISO Tag Type

The *ISO Tag Type* auxiliary object class may be used to add tag-based attribute types to an entry holding information about a specific type of tag (see Annex G of Rec. ITU-T X.520 | ISO/IEC 9594-6).

```
isoTagType  OBJECT-CLASS ::= {
  SUBCLASS OF  { top }
  KIND         auxiliary
  MAY CONTAIN  { tagOid |
                 tagAfi |
                 uiiFormat }
  ID           id-oc-isoTagType }
```

6.29   EPC Tag Information object class

The *EPC Tag Information Obj* auxiliary object class may be used to add tag-based attribute types to an entry holding information associated with a specific EPC tag.

```
epcTagInfoObj OBJECT-CLASS ::= {
  SUBCLASS OF  { top }
  KIND         auxiliary
  MAY CONTAIN  { epcHeader |
                 epcPartition |
                 epc |
                 epcInUrn |
                 contentUrl |
                 tagLocation }
  ID           id-oc-epcTagInfoObj }
```

6.30   EPC Tag Type Object Class

The *EPC Tag Type Obj* auxiliary object class may be used to add tag-based attribute types to an entry holding information about a specific type of EPC tag.

```
epcTagTypeObj  OBJECT-CLASS ::= {
  SUBCLASS OF  { top }
  KIND         auxiliary
  MAY CONTAIN  { uiiFormat }
  ID           id-oc-epcTagTypeObj }
```

SECTION 3 – SELECTED NAME FORMS

7   Definition of selected name forms

7.1   Country name form

The *Country* name form specifies how entries of object class `country` may be named.

```
countryNameForm NAME-FORM ::= {
  NAMES            country
  WITH ATTRIBUTES  {countryName}
  ID               id-nf-countryNameForm }
```

7.2   Locality name form

The *Locality* name form specifies how entries of object class `locality` may be named.

```
locNameForm NAME-FORM ::= {
  NAMES           locality
  WITH ATTRIBUTES {localityName}
  ID              id-nf-locNameForm }
```

7.3   State Or Province name form

The *State Or Province* name form specifies how entries of object class `locality` may be named.

```
sOPNameForm NAME-FORM ::= {
  NAMES           locality
  WITH ATTRIBUTES {stateOrProvinceName}
  ID              id-nf-sOPNameForm }
```

7.4   Organization name form

The *Organization* name form specifies how entries of object class `organization` may be named.

```
orgNameForm NAME-FORM ::= {
  NAMES           organization
  WITH ATTRIBUTES {organizationName}
  ID              id-nf-orgNameForm }
```

7.5   Organizational Unit name form

The *Organizational Unit* name form specifies how entries of object class `organizationalUnit` may be named.

```
orgUnitNameForm NAME-FORM ::= {
  NAMES           organizationalUnit
  WITH ATTRIBUTES {organizationalUnitName}
  ID              id-nf-orgUnitNameForm }
```

7.6   Person name form

The *Person* name form specifies how entries of object class `person` may be named.

```
personNameForm NAME-FORM ::= {
  NAMES           person
  WITH ATTRIBUTES {commonName}
  ID              id-nf-personNameForm }
```

7.7   Organizational Person name form

The *Organizational Person* name form specifies how entries of object class `organizationalPerson` may be named.

```
orgPersonNameForm NAME-FORM ::= {
  NAMES           organizationalPerson
  WITH ATTRIBUTES {commonName}
  AND OPTIONALLY  {organizationalUnitName}
  ID              id-nf-orgPersonNameForm }
```

7.8   Organizational Role name form

The *Organizational Role* name form specifies how entries of object class `organizationalRole` may be named.

```
orgRoleNameForm NAME-FORM ::= {
  NAMES           organizationalRole
  WITH ATTRIBUTES {commonName}
  ID              id-nf-orgRoleNameForm }
```

7.9   Group Of Names name form

The *Group Of Names* name form specifies how entries of object class `groupOfNames` may be named.

```
gONNameForm NAME-FORM ::= {
  NAMES            groupOfNames
  WITH ATTRIBUTES  {commonName}
  ID               id-nf-gONNameForm }
```

7.10   Residential Person name form

The *Residential Person* name form specifies how entries of object class `residentialPerson` may be named.

```
resPersonNameForm NAME-FORM ::= {
  NAMES            residentialPerson
  WITH ATTRIBUTES  {commonName}
  AND OPTIONALLY   {streetAddress}
  ID               id-nf-resPersonNameForm }
```

7.11   Application Process name form

The *Application Process* name form specifies how entries of object class `applicationProcess` may be named.

```
applProcessNameForm NAME-FORM ::= {
  NAMES            applicationProcess
  WITH ATTRIBUTES  {commonName}
  ID               id-nf-applProcessNameForm }
```

7.12   Application Entity name form

The *Application Entity* name form specifies how entries of object class `applicationEntity` may be named.

```
applEntityNameForm NAME-FORM ::= {
  NAMES            applicationEntity
  WITH ATTRIBUTES  {commonName}
  ID               id-nf-applEntityNameForm }
```

7.13   DSA name form

The *DSA* name form specifies how entries of object class `dSA` may be named.

```
dSANameForm NAME-FORM ::= {
  NAMES            dSA
  WITH ATTRIBUTES  {commonName}
  ID               id-nf-dSANameForm }
```

7.14   Device name form

The *Device* name form specifies how entries of object class `device` may be named.

```
deviceNameForm NAME-FORM ::= {
  NAMES            device
  WITH ATTRIBUTES  {commonName}
  ID               id-nf-deviceNameForm }
```

7.15   DMD name form

The *DMD* name form specifies how entries of object class `dMD` may be named.

```
dMDNameForm NAME-FORM ::= {
  NAMES            dMD
  WITH ATTRIBUTES  {dmdName}
  ID               id-nf-dMDNameForm }
```

7.16   OIDC1 name form

The *OIDC1* name form specifies how the entry of object class `oidObj1` shall be named.

```
oidC1NameForm NAME-FORM ::= {
  NAMES           oidCobj
  WITH ATTRIBUTES {oidC}
  ID              id-nf-oidC1NameForm }
```

7.17   OIDC2 name form

The *OIDC2* name form specifies how entries of object class `oidObj2` shall be named.

```
oidC2NameForm NAME-FORM ::= {
  NAMES           oidCobj
  WITH ATTRIBUTES {oidC}
  ID              id-nf-oidC2NameForm }
```

7.18   OIDC name form

The *OIDC* name form specifies how entries of object class `oidObjC` shall be named.

```
oidCNameForm NAME-FORM ::= {
  NAMES           oidCobj
  WITH ATTRIBUTES {oidC}
  ID              id-nf-oidCNameForm }
```

7.19   URNC name form

The `urnCNameForm` specifies how entries of object class `urnCobj` shall be named.

```
urnCNameForm NAME-FORM ::= {
  NAMES           urnCobj
  WITH ATTRIBUTES {urnC}
  ID              id-nf-urnCNameForm }
```

7.20   OID root name form

The *OID root* name form is moved to here from Rec. ITU-T X.660 | ISO/IEC 9834-1.

```
oidRootNf NAME-FORM ::= {
  NAMES           oidRoot
  WITH ATTRIBUTES {oidC1 | oidC2 | oidC}
  ID              id-oidRootNf }
```

7.21   OID arc name form

The *OID arc* name form is moved to here from Rec. ITU-T X.660 | ISO/IEC 9834-1.

```
oidArcNf NAME-FORM ::= {
  NAMES           oidArc
  WITH ATTRIBUTES {oidC}
  ID              id-oidArcNf }
```

Annex A

Selected object classes and name forms in ASN.1

(This annex forms an integral part of this Recommendation | International Standard.)

This annex includes all of the ASN.1 type and value definitions contained in this Directory Specification in the form of the ASN.1 module `SelectedObjectClasses`.

```
SelectedObjectClasses {joint-iso-itu-t ds(5) module(1) selectedObjectClasses(6) 7}
DEFINITIONS ::=
BEGIN

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

  authenticationFramework, certificateExtensions, id, id-nf, id-oc,
  informationFramework, objectClass, passwordPolicy, selectedAttributeTypes
    FROM UsefulDefinitions {joint-iso-itu-t ds(5) module(1) usefulDefinitions(0) 7}

  alias, ATTRIBUTE, NAME-FORM, OBJECT-CLASS, top
    FROM InformationFramework informationFramework

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

  businessCategory, commonName, contentUrl, countryName, description,
  destinationIndicator, dmdName, epc, epcInUrn, facsimileTelephoneNumber,
  internationalISDNNumber, knowledgeInformation, localityName, member, oidC, oidC1,
  oidC2, organizationalUnitName, organizationName, owner, physicalDeliveryOfficeName,
  postalAddress, postalCode, postOfficeBox, preferredDeliveryMethod,
  presentationAddress, registeredAddress, roleOccupant, searchGuide, seeAlso,
  serialNumber, stateOrProvinceName, streetAddress, supportedApplicationContext,
  surname, tagAfi, tagLocation, tagOid, telephoneNumber, telexNumber, title, uii,
  uiiFormat, uiiInUrn, uniqueMember, urnC, x121Address
    FROM SelectedAttributeTypes selectedAttributeTypes

-- from Rec. ITU-T X.509 | ISO/IEC 9594-8

  authorityRevocationList, cACertificate, certificateRevocationList,
  crossCertificatePair, deltaRevocationList, supportedAlgorithms,
  userCertificate, userPassword
    FROM AuthenticationFramework authenticationFramework

userPwd
    FROM PasswordPolicy passwordPolicy;

--  Attribute sets

TelecommunicationAttributeSet ATTRIBUTE ::=
  {facsimileTelephoneNumber |
   internationalISDNNumber |
   telephoneNumber |
-- teletexTerminalIdentifier  (Attribute type has been deleted)
   telexNumber |
   preferredDeliveryMethod |
   destinationIndicator |
   registeredAddress |
   x121Address}
```

```
PostalAttributeSet ATTRIBUTE ::=
  {physicalDeliveryOfficeName |
   postalAddress |
   postalCode |
   postOfficeBox |
   streetAddress}

LocaleAttributeSet ATTRIBUTE ::=
  {localityName |
   stateOrProvinceName |
   streetAddress}

OrganizationalAttributeSet ATTRIBUTE ::=
  {description |
   LocaleAttributeSet |
   PostalAttributeSet |
   TelecommunicationAttributeSet |
   businessCategory |
   seeAlso |
   searchGuide |
   userPassword}

--   Object classes

country OBJECT-CLASS ::= {
  SUBCLASS OF    {top}
  MUST CONTAIN   {countryName}
  MAY CONTAIN    {description |
                  searchGuide}
  ID             id-oc-country }

locality OBJECT-CLASS ::= {
  SUBCLASS OF  {top}
  MAY CONTAIN  {description |
                searchGuide |
                LocaleAttributeSet |
                seeAlso}
  ID           id-oc-locality }

organization OBJECT-CLASS ::= {
  SUBCLASS OF    {top}
  MUST CONTAIN   {organizationName}
  MAY CONTAIN    {OrganizationalAttributeSet}
  ID             id-oc-organization }

organizationalUnit OBJECT-CLASS ::= {
  SUBCLASS OF    {top}
  MUST CONTAIN   {organizationalUnitName}
  MAY CONTAIN    {OrganizationalAttributeSet}
  ID             id-oc-organizationalUnit }

person OBJECT-CLASS ::= {
  SUBCLASS OF    {top}
  MUST CONTAIN   {commonName |
                  surname}
  MAY CONTAIN    {description |
                  telephoneNumber |
                  userPassword |
                  seeAlso}
  ID             id-oc-person }

organizationalPerson OBJECT-CLASS ::= {
  SUBCLASS OF  {person}
  MAY CONTAIN  {LocaleAttributeSet |
                PostalAttributeSet |
                TelecommunicationAttributeSet |
                organizationalUnitName |
                title}
  ID           id-oc-organizationalPerson }

organizationalRole OBJECT-CLASS ::= {
```

```
   SUBCLASS OF    {top}
   MUST CONTAIN   {commonName}
   MAY CONTAIN    {description |
                   LocaleAttributeSet |
                   organizationalUnitName |
                   PostalAttributeSet |
                   preferredDeliveryMethod |
                   roleOccupant |
                   seeAlso |
                   TelecommunicationAttributeSet}
   ID             id-oc-organizationalRole }

groupOfNames OBJECT-CLASS ::= {
   SUBCLASS OF    {top}
   MUST CONTAIN   {commonName | member}
   MAY CONTAIN    {description |
                   organizationName |
                   organizationalUnitName |
                   owner |
                   seeAlso |
                   businessCategory}
   ID             id-oc-groupOfNames }

groupOfUniqueNames OBJECT-CLASS ::= {
   SUBCLASS OF    {top}
   MUST CONTAIN   {commonName |
                   uniqueMember}
   MAY CONTAIN    {description |
                   organizationName |
                   organizationalUnitName |
                   owner |
                   seeAlso |
                   businessCategory}
   ID             id-oc-groupOfUniqueNames }

residentialPerson OBJECT-CLASS ::= {
   SUBCLASS OF    {person}
   MUST CONTAIN   {localityName}
   MAY CONTAIN    {LocaleAttributeSet |
                   PostalAttributeSet |
                   preferredDeliveryMethod |
                   TelecommunicationAttributeSet |
                   businessCategory}
   ID             id-oc-residentialPerson }

applicationProcess OBJECT-CLASS ::= {
   SUBCLASS OF    {top}
   MUST CONTAIN   {commonName}
   MAY CONTAIN    {description |
                   localityName |
                   organizationalUnitName |
                   seeAlso}
   ID             id-oc-applicationProcess }

applicationEntity OBJECT-CLASS ::= {
   SUBCLASS OF    {top}
   MUST CONTAIN   {commonName |
                   presentationAddress}
   MAY CONTAIN    {description |
                   localityName |
                   organizationName |
                   organizationalUnitName |
                   seeAlso |
                   supportedApplicationContext}
   ID             id-oc-applicationEntity }

dSA OBJECT-CLASS ::= {
   SUBCLASS OF    {applicationEntity}
   MAY CONTAIN    {knowledgeInformation}
   ID             id-oc-dSA }
```

```
device OBJECT-CLASS ::= {
  SUBCLASS OF    {top}
  MUST CONTAIN   {commonName}
  MAY CONTAIN    {description |
                  localityName |
                  organizationName |
                  organizationalUnitName |
                  owner |
                  seeAlso |
                  serialNumber}
  ID             id-oc-device }

strongAuthenticationUser OBJECT-CLASS ::= {
  SUBCLASS OF    {top}
  KIND           auxiliary
  MUST CONTAIN   {userCertificate}
  ID             id-oc-strongAuthenticationUser }

userSecurityInformation OBJECT-CLASS ::= {
  SUBCLASS OF    {top}
  KIND           auxiliary
  MAY CONTAIN    {supportedAlgorithms}
  ID             id-oc-userSecurityInformation }

userPwdClass  OBJECT-CLASS  ::=  {
  KIND           auxiliary
  MAY CONTAIN    { userPwd }
  ID             id-oc-userPwdClass }

certificationAuthority OBJECT-CLASS ::= {
  SUBCLASS OF    {top}
  KIND           auxiliary
  MUST CONTAIN   {cACertificate |
                  certificateRevocationList |
                  authorityRevocationList}
  MAY CONTAIN    {crossCertificatePair}
  ID             id-oc-certificationAuthority }

certificationAuthority-V2 OBJECT-CLASS ::= {
  SUBCLASS OF    {certificationAuthority}
  KIND           auxiliary
  MAY CONTAIN    {deltaRevocationList}
  ID             id-oc-certificationAuthority-V2 }

dMD OBJECT-CLASS ::= {
  SUBCLASS OF    {top}
  MUST CONTAIN   {dmdName}
  MAY CONTAIN    {OrganizationalAttributeSet}
  ID             id-oc-dmd }

oidC1obj OBJECT-CLASS ::= {
  SUBCLASS OF    {top}
  MUST CONTAIN   {oidC}
  ID             id-oc-oidC1obj }

oidC2obj OBJECT-CLASS ::= {
  SUBCLASS OF    {top}
  MUST CONTAIN   {oidC}
  ID             id-oc-oidC2obj }

oidCobj OBJECT-CLASS ::= {
  SUBCLASS OF    {top}
  MUST CONTAIN   {oidC}
  ID             id-oc-oidCobj }

oidRoot OBJECT-CLASS ::= {
  SUBCLASS OF    {alias}
  MUST CONTAIN   { oidC1 | oidC2 | oidC}
  ID             id-oidRoot }

oidArc OBJECT-CLASS ::= {
```

```
  SUBCLASS OF    {alias}
  MUST CONTAIN   {oidC}
  ID             id-oidArc }

urnCobj OBJECT-CLASS ::= {
  SUBCLASS OF    {top}
  MUST CONTAIN   { urnC }
  ID             id-oc-urnCobj }

isoTagInfo OBJECT-CLASS ::= {
  SUBCLASS OF    { top }
  KIND           auxiliary
  MAY CONTAIN    { tagOid |
                   tagAfi |
                   uii |
                   uiiInUrn |
                   contentUrl |
                   tagLocation }
  ID             id-oc-isoTagInfo }

isoTagType  OBJECT-CLASS ::= {
  SUBCLASS OF    { top }
  KIND           auxiliary
  MAY CONTAIN    { tagOid |
                   tagAfi |
                   uiiFormat }
  ID             id-oc-isoTagType }

epcTagInfoObj OBJECT-CLASS ::= {
  SUBCLASS OF    { top }
  KIND           auxiliary
  MAY CONTAIN    { epc |
                   epcInUrn |
                   contentUrl |
                   tagLocation }
  ID             id-oc-epcTagInfoObj }

epcTagTypeObj OBJECT-CLASS ::= {
  SUBCLASS OF    { top }
  KIND           auxiliary
  MAY CONTAIN    { uiiFormat }
  ID             id-oc-epcTagTypeObj }

--  Name forms

countryNameForm NAME-FORM ::= {
  NAMES             country
  WITH ATTRIBUTES   {countryName}
  ID                id-nf-countryNameForm }

locNameForm NAME-FORM ::= {
  NAMES             locality
  WITH ATTRIBUTES   {localityName}
  ID                id-nf-locNameForm }

sOPNameForm NAME-FORM ::= {
  NAMES             locality
  WITH ATTRIBUTES   {stateOrProvinceName}
  ID                id-nf-sOPNameForm }

orgNameForm NAME-FORM ::= {
  NAMES             organization
  WITH ATTRIBUTES   {organizationName}
  ID                id-nf-orgNameForm }

orgUnitNameForm NAME-FORM ::= {
  NAMES             organizationalUnit
  WITH ATTRIBUTES   {organizationalUnitName}
  ID                id-nf-orgUnitNameForm }

personNameForm NAME-FORM ::= {
```

```
  NAMES             person
  WITH ATTRIBUTES   {commonName}
  ID                id-nf-personNameForm }

orgPersonNameForm NAME-FORM ::= {
  NAMES             organizationalPerson
  WITH ATTRIBUTES   {commonName}
  AND OPTIONALLY    {organizationalUnitName}
  ID                id-nf-orgPersonNameForm }

orgRoleNameForm NAME-FORM ::= {
  NAMES             organizationalRole
  WITH ATTRIBUTES   {commonName}
  ID                id-nf-orgRoleNameForm }

gONNameForm NAME-FORM ::= {
  NAMES             groupOfNames
  WITH ATTRIBUTES   {commonName}
  ID                id-nf-gONNameForm }

resPersonNameForm NAME-FORM ::= {
  NAMES             residentialPerson
  WITH ATTRIBUTES   {commonName}
  AND OPTIONALLY    {streetAddress}
  ID                id-nf-resPersonNameForm }

applProcessNameForm NAME-FORM ::= {
  NAMES             applicationProcess
  WITH ATTRIBUTES   {commonName}
  ID                id-nf-applProcessNameForm }

applEntityNameForm NAME-FORM ::= {
  NAMES             applicationEntity
  WITH ATTRIBUTES   {commonName}
  ID                id-nf-applEntityNameForm }

dSANameForm NAME-FORM ::= {
  NAMES             dSA
  WITH ATTRIBUTES   {commonName}
  ID                id-nf-dSANameForm }

deviceNameForm NAME-FORM ::= {
  NAMES             device
  WITH ATTRIBUTES   {commonName}
  ID                id-nf-deviceNameForm }

dMDNameForm NAME-FORM ::= {
  NAMES             dMD
  WITH ATTRIBUTES   {dmdName}
  ID                id-nf-dMDNameForm }

oidC1NameForm NAME-FORM ::= {
  NAMES             oidCobj
  WITH ATTRIBUTES   {oidC}
  ID                id-nf-oidC1NameForm }

oidC2NameForm NAME-FORM ::= {
  NAMES             oidCobj
  WITH ATTRIBUTES   {oidC}
  ID                id-nf-oidC2NameForm }

oidCNameForm NAME-FORM ::= {
  NAMES             oidCobj
  WITH ATTRIBUTES   {oidC}
  ID                id-nf-oidCNameForm }

urnCNameForm NAME-FORM ::= {
  NAMES             urnCobj
  WITH ATTRIBUTES   {urnC}
  ID                id-nf-urnCNameForm }
```

```
oidRootNf NAME-FORM ::= {
  NAMES           oidRoot
  WITH ATTRIBUTES {oidC1 | oidC2 | oidC}
  ID              id-oidRootNf }

oidArcNf NAME-FORM ::= {
  NAMES           oidArc
  WITH ATTRIBUTES {oidC}
  ID              id-oidArcNf }

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

--  Object classes

--  id-oc-top                   OBJECT IDENTIFIER ::= {id-oc 0} Defined in X.501 | Part 2
--  id-oc-alias                 OBJECT IDENTIFIER ::= {id-oc 1} Defined in X.501 | Part 2
id-oc-country                   OBJECT IDENTIFIER ::= {id-oc 2}
id-oc-locality                  OBJECT IDENTIFIER ::= {id-oc 3}
id-oc-organization              OBJECT IDENTIFIER ::= {id-oc 4}
id-oc-organizationalUnit        OBJECT IDENTIFIER ::= {id-oc 5}
id-oc-person                    OBJECT IDENTIFIER ::= {id-oc 6}
id-oc-organizationalPerson      OBJECT IDENTIFIER ::= {id-oc 7}
id-oc-organizationalRole        OBJECT IDENTIFIER ::= {id-oc 8}
id-oc-groupOfNames              OBJECT IDENTIFIER ::= {id-oc 9}
id-oc-residentialPerson         OBJECT IDENTIFIER ::= {id-oc 10}
id-oc-applicationProcess        OBJECT IDENTIFIER ::= {id-oc 11}
id-oc-applicationEntity         OBJECT IDENTIFIER ::= {id-oc 12}
id-oc-dSA                       OBJECT IDENTIFIER ::= {id-oc 13}
id-oc-device                    OBJECT IDENTIFIER ::= {id-oc 14}
id-oc-strongAuthenticationUser  OBJECT IDENTIFIER ::= {id-oc 15} -- Deprecated, see 6.15
id-oc-certificationAuthority    OBJECT IDENTIFIER ::= {id-oc 16} -- Deprecated, see 6.17
id-oc-certificationAuthority-V2
                                OBJECT IDENTIFIER ::= {id-oc 16 2} -- Deprecated, see 6.18
id-oc-groupOfUniqueNames        OBJECT IDENTIFIER ::= {id-oc 17}
id-oc-userSecurityInformation   OBJECT IDENTIFIER ::= {id-oc 18}
--  id-oc-cRLDistributionPoint  OBJECT IDENTIFIER ::= {id-oc 19} Defined in X.509 | Part 8
id-oc-dmd                       OBJECT IDENTIFIER ::= {id-oc 20}
--  id-oc-pkiUser               OBJECT IDENTIFIER ::= {id-oc 21} Defined in X.509 | Part 8
--  id-oc-pkiCA                 OBJECT IDENTIFIER ::= {id-oc 22} Defined in X.509 | Part 8
--  id-oc-deltaCRL              OBJECT IDENTIFIER ::= {id-oc 23} Defined in X.509 | Part 8
--  id-oc-pmiUser               OBJECT IDENTIFIER ::= {id-oc 24} Defined in X.509 | Part 8
--  id-oc-pmiAA                 OBJECT IDENTIFIER ::= {id-oc 25} Defined in X.509 | Part 8
--  id-oc-pmiSOA                OBJECT IDENTIFIER ::= {id-oc 26} Defined in X.509 | Part 8
--  id-oc-attCertCRLDistributionPts
--                              OBJECT IDENTIFIER ::= {id-oc 27} Defined in X.509 | Part 8
--  id-oc-parent                OBJECT IDENTIFIER ::= {id-oc 28} Defined in X.501 | Part 2
--  id-oc-child                 OBJECT IDENTIFIER ::= {id-oc 29} Defined in X.501 | Part 2
--  id-oc-cpCps                 OBJECT IDENTIFIER ::= {id-oc 30} Defined in X.509 | Part 8
--  id-oc-pkiCertPath           OBJECT IDENTIFIER ::= {id-oc 31} Defined in X.509 | Part 8
--  id-oc-privilegePolicy       OBJECT IDENTIFIER ::= {id-oc 32} Defined in X.509 | Part 8
--  id-oc-pmiDelegationPath     OBJECT IDENTIFIER ::= {id-oc 33} Defined in X.509 | Part 8
--  id-oc-protectedPrivilegePolicy
--                              OBJECT IDENTIFIER ::= {id-oc 34} Defined in X.509 | Part 8
id-oc-oidC1obj                  OBJECT IDENTIFIER ::= {id-oc 35}
id-oc-oidC2obj                  OBJECT IDENTIFIER ::= {id-oc 36}
id-oc-oidCobj                   OBJECT IDENTIFIER ::= {id-oc 37}
id-oc-isoTagInfo                OBJECT IDENTIFIER ::= {id-oc 38}
id-oc-isoTagType                OBJECT IDENTIFIER ::= {id-oc 39}
--  id-oc-integrityInfo         OBJECT IDENTIFIER ::= {id-oc 40} Defined in X.501 | Part 2
id-oc-userPwdClass              OBJECT IDENTIFIER ::= {id-oc 41}
id-oc-urnCobj                   OBJECT IDENTIFIER ::= {id-oc 42}
id-oc-epcTagInfoObj             OBJECT IDENTIFIER ::= {id-oc 43}
id-oc-epcTagTypeObj             OBJECT IDENTIFIER ::= {id-oc 44}

id-oidRoot                      OBJECT IDENTIFIER ::= {id 3}
id-oidArc                       OBJECT IDENTIFIER ::= {id 5}

--  Name forms

id-nf-countryNameForm           OBJECT IDENTIFIER ::= {id-nf 0}
```

```
id-nf-locNameForm             OBJECT IDENTIFIER ::= {id-nf 1}
id-nf-sOPNameForm             OBJECT IDENTIFIER ::= {id-nf 2}
id-nf-orgNameForm             OBJECT IDENTIFIER ::= {id-nf 3}
id-nf-orgUnitNameForm         OBJECT IDENTIFIER ::= {id-nf 4}
id-nf-personNameForm          OBJECT IDENTIFIER ::= {id-nf 5}
id-nf-orgPersonNameForm       OBJECT IDENTIFIER ::= {id-nf 6}
id-nf-orgRoleNameForm         OBJECT IDENTIFIER ::= {id-nf 7}
id-nf-gONNameForm             OBJECT IDENTIFIER ::= {id-nf 8}
id-nf-resPersonNameForm       OBJECT IDENTIFIER ::= {id-nf 9}
id-nf-applProcessNameForm     OBJECT IDENTIFIER ::= {id-nf 10}
id-nf-applEntityNameForm      OBJECT IDENTIFIER ::= {id-nf 11}
id-nf-dSANameForm             OBJECT IDENTIFIER ::= {id-nf 12}
id-nf-deviceNameForm          OBJECT IDENTIFIER ::= {id-nf 13}
-- id-nf-cRLDistPtNameForm    OBJECT IDENTIFIER ::= {id-nf 14}
id-nf-dMDNameForm             OBJECT IDENTIFIER ::= {id-nf 15}
-- id-nf-subentryNameForm     OBJECT IDENTIFIER ::= {id-nf 16}
id-nf-oidC1NameForm           OBJECT IDENTIFIER ::= {id-nf 17}
id-nf-oidC2NameForm           OBJECT IDENTIFIER ::= {id-nf 18}
id-nf-oidCNameForm            OBJECT IDENTIFIER ::= {id-nf 19}
id-nf-urnCNameForm            OBJECT IDENTIFIER ::= {id-nf 20}

id-oidRootNf                  OBJECT IDENTIFIER ::= {id 4}
id-oidArcNf                   OBJECT IDENTIFIER ::= {id 6}

END -- SelectedObjectClasses
```

Annex B

Suggested name forms and Directory information tree (DIT) structures

(This annex does not form an integral part of this Recommendation | International Standard.)

This annex suggests a Directory information tree (DIT) structure shown in Figure B.1 and related DIT structure rules using the name forms defined in clause 3. The rules cover an unconstrained DIT structure. This example is for illustrative purposes only, and is not intended to limit the types of names that can be validly constructed in the Directory.

The integer identifiers assigned in this annex and used in Figure B.1 are arbitrary and have no global (or standardized) significance. A particular structure rule identifier only has significance within the scope of the subschema in which it applied. Each Directory management domain (DMD) is responsible for creating its own DIT structure and structure rules that may differ from this example.
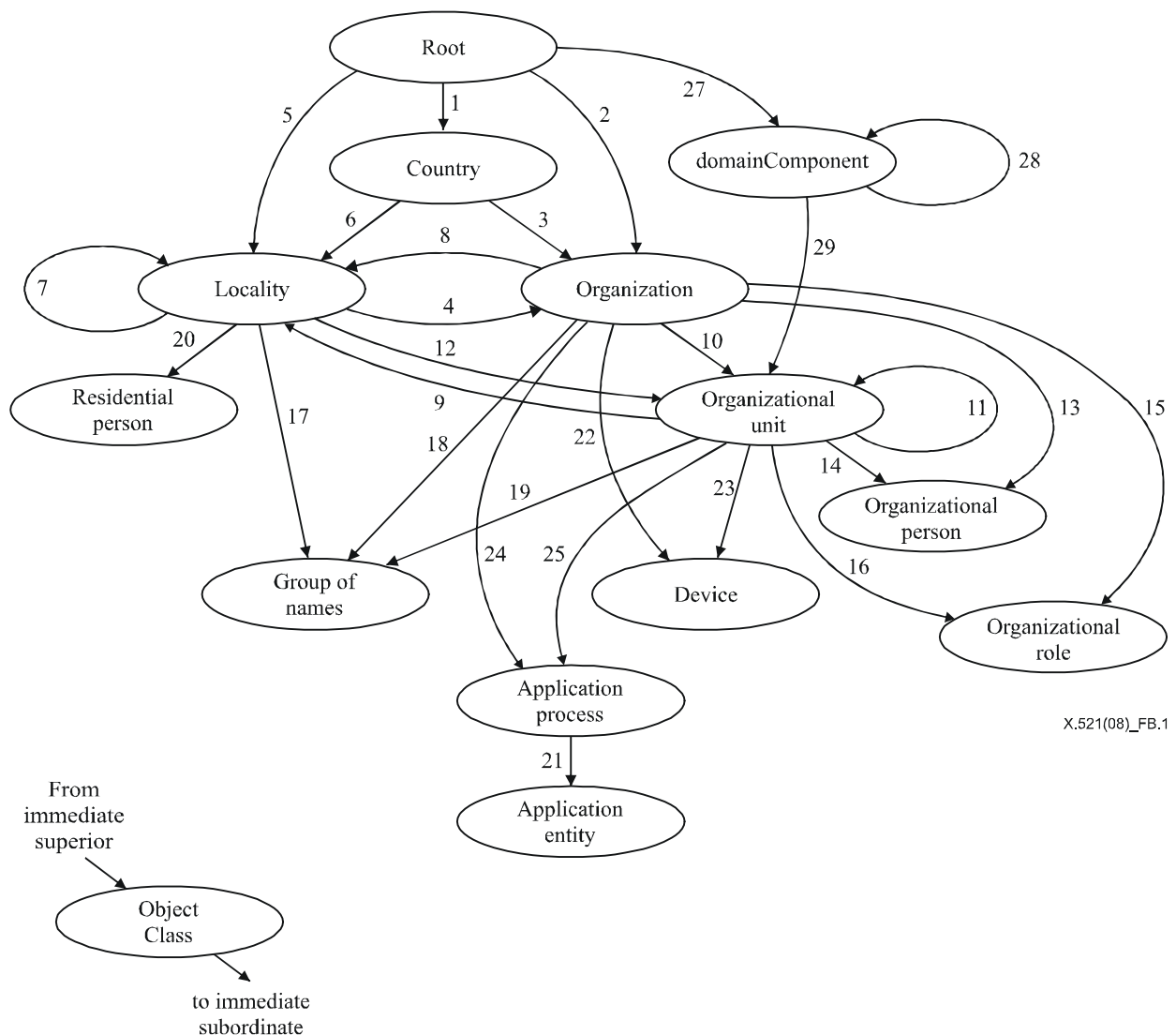

Figure B.1 – Example DIT structure

B.1   Country

Attribute type `countryName` is used for naming.

The root is the immediate superior to entries of object class `country`.

```
sr1  STRUCTURE-RULE  ::=  {
    NAME FORM        countryNameForm
    ID               1 }
```

B.2   Organization

Attribute type `organizationName` is used for naming.

The root or an entry of object class `country` or `locality` can be the immediate superior of entries of object class `organization`.

> NOTE – When the organization is directly under the root, this denotes an international organization. The naming values of the organizationName attribute for international organizations must all be distinct.

```
sr2  STRUCTURE-RULE  ::=  {
    NAME FORM        orgNameForm
    ID               2 }

sr3  STRUCTURE-RULE  ::=  {
    NAME FORM        orgNameForm
    SUPERIOR RULES  { sr1.&id }
    ID               3 }

sr4  STRUCTURE-RULE  ::=  {
    NAME FORM        orgNameForm
    SUPERIOR RULES  { sr5.&id | sr6.&id | sr7.&id | sr8.&id | sr9.&id }
    ID               4 }
```

B.3   Locality

Attribute type `localityName` or `stateOrProvinceName` is used for naming.

> NOTE – For naming locality using stateOrProvinceName, see B.12.

The root or an entry of object class `country`, `locality`, `organization` or `organizationalUnit` can be the immediate superior of entries of object class `locality`.

```
sr5  STRUCTURE-RULE  ::=  {
  NAME FORM        locNameForm
  ID               5 }

sr6  STRUCTURE-RULE  ::=  {
  NAME FORM        locNameForm
  SUPERIOR RULES  { sr1.&id }
  ID               6 }

sr7  STRUCTURE-RULE  ::=  {
  NAME FORM        locNameForm
  SUPERIOR RULES  { sr5.&id | sr6.&id | sr7.&id | sr8.&id | sr9.&id }
  ID               7 }

sr8  STRUCTURE-RULE  ::=  {
  NAME FORM        locNameForm
  SUPERIOR RULES  { sr2.&id | sr3.&id | sr4.&id }
  ID               8 }

sr9  STRUCTURE-RULE  ::=  {
  NAME FORM        locNameForm
  SUPERIOR RULES  { sr10.&id | sr11.&id | sr12.&id }
  ID               9 }
```

B.4   Organizational Unit

Attribute type `organizationalUnitName` is used for naming.

An entry of object class `organization`, `organizationalUnit`, `locality` or `domainComponent` can be the immediate superior of entries of object class `organizationalUnit`.

```
sr10  STRUCTURE-RULE  ::=  {
  NAME FORM       orgUnitNameForm
  SUPERIOR RULES  { sr2.&id | sr3.&id | sr4.&id }
  ID              10 }

sr11  STRUCTURE-RULE  ::=  {
  NAME FORM       orgUnitNameForm
  SUPERIOR RULES  { sr10.&id | sr11.&id | sr12.&id }
  ID              11 }

sr12  STRUCTURE-RULE  ::=  {
  NAME FORM       orgUnitNameForm
  SUPERIOR RULES  { sr5.&id | sr6.&id | sr7.&id | sr8.&id | sr9.&id }
  ID              12 }
```

B.5   Organizational Person

Attribute type `commonName` and optionally `organizationalUnitName` is used for naming.

An entry of object class `organization` or `organizationalUnit` can be the immediate superior of entries of object class `organizationalPerson`.

```
sr13  STRUCTURE-RULE  ::=  {
  NAME FORM       orgPersonNameForm
  SUPERIOR RULES  { sr2.&id | sr3.&id | sr4.&id }
  ID              13 }

sr14  STRUCTURE-RULE  ::=  {
  NAME FORM       orgPersonNameForm
  SUPERIOR RULES  { sr10.&id | sr11.&id | sr12.&id }
  ID              14 }
```

B.6   Organizational Role

Attribute type `commonName` is used for naming.

An entry of object class `organization` or `organizationalUnit` can be the immediate superior of entries of object class `organizationalRole`.

```
sr15  STRUCTURE-RULE  ::=  {
  NAME FORM       orgRoleNameForm
  SUPERIOR RULES  { sr2.&id | sr3.&id | sr4.&id }
  ID              15 }

sr16  STRUCTURE-RULE  ::=  {
  NAME FORM       orgRoleNameForm
  SUPERIOR RULES  { sr10.&id | sr11.&id | sr12.&id }
  ID              16 }
```

B.7   Group of Names

Attribute type `commonName` is used for naming.

An entry of object class `locality`, `organization` or `organizationalUnit` can be the immediate superior of entries of object class `groupOfNames`.

```
sr17  STRUCTURE-RULE  ::=  {
  NAME FORM       gonNameForm
  SUPERIOR RULES  { sr5.&id | sr6.&id | sr7.&id | sr8.&id | sr9.&id }
  ID              17 }
```

```
sr18  STRUCTURE-RULE  ::=  {
  NAME FORM        gonNameForm
  SUPERIOR RULES   { sr2.&id | sr3.&id | sr4.&id }
  ID               18 }

sr19  STRUCTURE-RULE  ::=  {
  NAME FORM        gonNameForm
  SUPERIOR RULES   { sr10.&id | sr11.&id | sr12.&id }
  ID               19 }
```

B.8   Residential Person

Attribute type **commonName** and optionally attribute type **streetAddress** is used for naming.

An entry of object class **locality** is the immediate superior of entries of object class **residentialPerson**.

```
sr20  STRUCTURE-RULE  ::=  {
  NAME FORM        resPersonNameForm
  SUPERIOR RULES   { sr5.&id | sr6.&id | sr7.&id | sr8.&id | sr9.&id }
  ID               20 }
```

B.9   Application Entity

Attribute type **commonName** is used for naming.

An entry of object class **applicationProcess** is the immediate superior of entries of object class **applicationEntity**.

```
sr21  STRUCTURE-RULE  ::=  {
  NAME FORM        applEntityNameForm
  SUPERIOR RULES   { sr24.&id | sr25.&id }
  ID               21 }
```

B.10   Device

Attribute type **commonName** is used for naming.

An entry of object class **organization** or **organizationalUnit** can be the immediate superior of entries of object class **device**.

```
sr22  STRUCTURE-RULE  ::=  {
  NAME FORM        deviceNameForm
  SUPERIOR RULES   { sr2.&id | sr3.&id | sr4.&id }
  ID               22 }

sr23  STRUCTURE-RULE  ::=  {
  NAME FORM        deviceNameForm
  SUPERIOR RULES   { sr10.&id | sr11.&id | sr12.&id }
  ID               23 }
```

B.11   Application Process

Attribute type **commonName** is used for naming.

An entry of object class **organization** or **organizationalUnit** can be the immediate superior of entries of object class **applicationProcess**.

```
sr24  STRUCTURE-RULE  ::=  {
  NAME FORM        applProcessNameForm
  SUPERIOR RULES   { sr2.&id | sr3.&id | sr4.&id }
  ID               24 }

sr25  STRUCTURE-RULE  ::=  {
  NAME FORM        applProcessNameForm
  SUPERIOR RULES   { sr10.&id | sr11.&id | sr12.&id }
  ID               25 }
```

B.12 Alternative Structure Rule for Locality

If the **stateOrProvinceName** attribute type is used for naming locality and locality constrained to existing only as an immediate subordinate of country, then one additional structure rule is required to define this.

```
sr26  STRUCTURE-RULE  ::=  {
  NAME FORM        sOPNameForm
  SUPERIOR RULES  { sr1.&id }
  ID              26 }
```

In addition the structure rules **sr4**, **sr7**, **sr12**, **sr17**, and **sr20** must be modified to include **sr26** within their respective list of superior structure rule as follows.

```
sr4  STRUCTURE-RULE  ::=  {
  NAME FORM        orgNameForm
  SUPERIOR RULES  { sr5.&id | sr6.&id | sr7.&id | sr8.&id | sr9.&id | sr26.&id }
  ID              4 }

sr7  STRUCTURE-RULE  ::=  {
  NAME FORM        locNameForm
  SUPERIOR RULES  { sr5.&id | sr6.&id | sr7.&id | sr8.&id | sr9.&id | sr26.&id }
  ID              7 }

sr12  STRUCTURE-RULE  ::=  {
  NAME FORM        orgUnitNameForm
  SUPERIOR RULES  { sr5.&id | sr6.&id | sr7.&id | sr8.&id | sr9.&id | sr26.&id }
  ID              12 }

sr17  STRUCTURE-RULE  ::=  {
  NAME FORM        gonNameForm
  SUPERIOR RULES  { sr5.&id | sr6.&id | sr7.&id | sr8.&id | sr9.&id | sr26.&id }
  ID              17 }

sr20  STRUCTURE-RULE  ::=  {
  NAME FORM        resPersonNameForm
  SUPERIOR RULES  { sr5.&id | sr6.&id | sr7.&id | sr8.&id | sr9.&id | sr26.&id }
  ID              20 }
```

Annex C

Amendments and corrigenda

(This annex does not form an integral part of this Recommendation | International Standard.)

This edition of this Directory Specification includes the following amendment to the previous edition that was balloted and approved by ISO/IEC:

–   Amendment 1 on Password policy support

–   Amendment 2 on Communications support enhancements

–   Amendment 3 on Directory-IdM support.

This edition of this Directory Specification includes the following technical corrigenda correcting the defects documented in Defect Reports against the 6th edition of this Directory Specification:

–   Technical Corrigendum 1 (covering Defect Reports 349 and 357).

SERIES OF ITU-T RECOMMENDATIONS

Series A   Organization of the work of ITU-T

Series D   General tariff principles

Series E   Overall network operation, telephone service, service operation and human factors

Series F   Non-telephone telecommunication services

Series G   Transmission systems and media, digital systems and networks

Series H   Audiovisual and multimedia systems

Series I   Integrated services digital network

Series J   Cable networks and transmission of television, sound programme and other multimedia signals

Series K   Protection against interference

Series L   Construction, installation and protection of cables and other elements of outside plant

Series M   Telecommunication management, including TMN and network maintenance

Series N   Maintenance: international sound programme and television transmission circuits

Series O   Specifications of measuring equipment

Series P   Terminals and subjective and objective assessment methods

Series Q   Switching and signalling

Series R   Telegraph transmission

Series S   Telegraph services terminal equipment

Series T   Terminals for telematic services

Series U   Telegraph switching

Series V   Data communication over the telephone network

Series X   **Data networks, open system communications and security**

Series Y   Global information infrastructure, Internet protocol aspects and next-generation networks

Series Z   Languages and general software aspects for telecommunication systems