
**Information technology — Open Systems
Interconnection — The Directory:
Selected object classes**

*Technologies de l'information — Interconnexion de systèmes ouverts
(OSI) — L'annuaire: Classes d'objets sélectionnées*

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 9594-7:2008

PDF disclaimer

This PDF file may contain embedded typefaces. In accordance with Adobe's licensing policy, this file may be printed or viewed but shall not be edited unless the typefaces which are embedded are licensed to and installed on the computer performing the editing. In downloading this file, parties accept therein the responsibility of not infringing Adobe's licensing policy. The ISO Central Secretariat accepts no liability in this area.

Adobe is a trademark of Adobe Systems Incorporated.

Details of the software products used to create this PDF file can be found in the General Info relative to the file; the PDF-creation parameters were optimized for printing. Every care has been taken to ensure that the file is suitable for use by ISO member bodies. In the unlikely event that a problem relating to it is found, please inform the Central Secretariat at the address given below.

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 9594-7:2008



COPYRIGHT PROTECTED DOCUMENT

© ISO/IEC 2008

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published by ISO in 2009

Published in Switzerland

CONTENTS

	<i>Page</i>
Foreword	v
Introduction	vi
SECTION 1 – GENERAL	1
1 Scope	1
2 Normative references	1
2.1 Identical Recommendations International Standards	1
3 Definitions	2
3.1 Communication Model definitions.....	2
3.2 Directory Model definitions	2
4 Conventions	2
SECTION 2 – SELECTED OBJECT CLASSES	4
5 Definition of useful attribute sets	4
5.1 Telecommunication attribute set.....	4
5.2 Postal attribute set	4
5.3 Locale attribute set	4
5.4 Organizational attribute set	4
6 Definition of selected object classes	5
6.1 Country	5
6.2 Locality.....	5
6.3 Organization	5
6.4 Organizational Unit	5
6.5 Person.....	5
6.6 Organizational Person	6
6.7 Organizational Role.....	6
6.8 Group of Names	6
6.9 Group of Unique Names.....	6
6.10 Residential Person.....	7
6.11 Application Process	7
6.12 Application Entity	7
6.13 DSA	7
6.14 Device	8
6.15 Strong Authentication User	8
6.16 User Security Information.....	8
6.17 Certification Authority	8
6.18 Certification Authority-V2	9
6.19 DMD	9
6.20 OID Obj1	9
6.21 OID Obj2	9
6.22 OID ObjC.....	9
6.23 UII to URN.....	9
6.24 URN to URL	10
SECTION 3 – SELECTED NAME FORMS	11
7 Definition of selected name forms.....	11
7.1 Country name form	11
7.2 Locality name form	11
7.3 State or Province name form	11
7.4 Organization name form.....	11
7.5 Organizational Unit name form.....	11

	<i>Page</i>
7.6 Person name form	11
7.7 Organizational Person name form.....	12
7.8 Organizational Role name form	12
7.9 Group of Names name form	12
7.10 Residential Person name form	12
7.11 Application Process name form.....	12
7.12 Application Entity name form.....	12
7.13 DSA name form.....	12
7.14 Device name form.....	13
7.15 DMD name form.....	13
7.16 OIDC1 name form.....	13
7.17 OIDC2 name form.....	13
7.18 OIDC name form	13
Annex A – Selected object classes and name forms in ASN.1	14
Annex B – Suggested name forms and DIT structures	21
B.1 Country.....	21
B.2 Organization	22
B.3 Locality.....	22
B.4 Organizational Unit	22
B.5 Organizational Person	23
B.6 Organizational Role.....	23
B.7 Group of Names	23
B.8 Residential Person.....	24
B.9 Application Entity	24
B.10 Device	24
B.11 Application Process.....	24
B.12 Alternative Structure Rule for Locality	24
Annex C – Amendments and corrigenda.....	26

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

ISO/IEC 9594-7:2008 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 6, *Telecommunications and information exchange between systems*, in collaboration with ITU-T. The identical text is published as ITU-T Rec. X.521 (11/2008).

This sixth edition cancels and replaces the fifth edition (ISO/IEC 9594-7:2005), which has been technically revised.

ISO/IEC 9594 consists of the following parts, under the general title *Information technology — Open Systems Interconnection — The Directory*:

- *Part 1: Overview of concepts, models and services*
- *Part 2: Models*
- *Part 3: Abstract service definition*
- *Part 4: Procedures for distributed operation*
- *Part 5: Protocol specifications*
- *Part 6: Selected attribute types*
- *Part 7: Selected object classes*
- *Part 8: Public-key and attribute certificate frameworks*
- *Part 9: Replication*
- *Part 10: Use of systems management for administration of the Directory*

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 sixth edition technically revises and enhances, but does not replace, the fifth edition of this Recommendation | International Standard. Implementations may still claim conformance to the fifth edition. However, at some point, the fifth edition will not be supported (i.e., reported defects will no longer be resolved). It is recommended that implementations conform to this sixth edition as soon as possible.

This sixth 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 ITU-T Rec. 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
ITU-T RECOMMENDATION**

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

SECTION 1 – GENERAL

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 ITU-T Rec. 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 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

- ITU-T Recommendation X.200 (1994) | ISO/IEC 7498-1:1994, *Information technology – Open Systems Interconnection – Basic Reference Model: The Basic Model*.
- ITU-T Recommendation X.500 (2008) | ISO/IEC 9594-1:2008, *Information technology – Open Systems Interconnection – The Directory: Overview of concepts, models and services*.
- ITU-T Recommendation X.501 (2008) | ISO/IEC 9594-2:2008, *Information technology – Open Systems Interconnection – The Directory: Models*.
- ITU-T Recommendation X.509 (2008) | ISO/IEC 9594-8:2008, *Information technology – Open Systems Interconnection – The Directory: Public-key and attribute certificate frameworks*.
- ITU-T Recommendation X.511 (2008) | ISO/IEC 9594-3:2008, *Information technology – Open Systems Interconnection – The Directory: Abstract service definition*.
- ITU-T Recommendation X.518 (2008) | ISO/IEC 9594-4:2008, *Information technology – Open Systems Interconnection – The Directory: Procedures for distributed operation*.
- ITU-T Recommendation X.519 (2008) | ISO/IEC 9594-5:2008, *Information technology – Open Systems Interconnection – The Directory: Protocol specifications*.
- ITU-T Recommendation X.520 (2008) | ISO/IEC 9594-6:2008, *Information technology – Open Systems Interconnection – The Directory: Selected attribute types*.

- ITU-T Recommendation X.525 (2008) | ISO/IEC 9594-9:2008, *Information technology – Open Systems Interconnection – The Directory: Replication*.
- ITU-T Recommendation X.530 (2008) | ISO/IEC 9594-10:2008, *Information technology – Open Systems Interconnection – The Directory: Use of systems management for administration of the Directory*.
- ITU-T Recommendation 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*.
- ITU-T Recommendation X.680 (2008) | ISO/IEC 8824-1:2008, *Information technology – Abstract Syntax Notation One (ASN.1): Specification of basic notation*.
- ITU-T Recommendation X.681 (2008) | ISO/IEC 8824-2:2008, *Information technology – Abstract Syntax Notation One (ASN.1): Information object specification*.
- ITU-T Recommendation X.682 (2008) | ISO/IEC 8824-3:2008, *Information technology – Abstract Syntax Notation One (ASN.1): Constraint specification*.
- ITU-T Recommendation 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 ITU-T Rec. X.519 | ISO/IEC 9594-5:

- a) *application-entity*;
- b) *application process*.

3.2 Directory Model definitions

The following terms are defined in ITU-T Rec. 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 ITU-T Rec. X.521 | ISO/IEC 9594-7. The term "Directory Specifications" shall be taken to mean the 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 series of CCITT X.500 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 series of ITU-T X.500 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 series of ITU-T X.500 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 ITU-T Recs X.500, X.501, X.511, X.518, X.519, X.520, X.521, X.525, and X.530, the 2000 edition of ITU-T Rec. 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 series of ITU-T X.500 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 series of ITU-T X.500 Recommendations and the ISO/IEC 9594:2008 edition.

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

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

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 9594-7:2008

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

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

6.17 Certification Authority

The *Certification Authority* object class is used to define entries for objects which act as certification authorities, as defined in ITU-T Rec. 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 ITU-T Rec. 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.

6.18 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 ITU-T Rec. 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 ITU-T Rec. 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 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 }
```

6.20 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 { oidC1 }
    ID          id-oc-oidC1obj }
```

6.21 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 { oidC2 }
    ID          id-oc-oidC2obj }
```

6.22 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 }
```

6.23 UII to URN

The *UII to URN* auxiliary object class may be used together with the **oidCobj** object class to provide entry support for tag-based identification (see Annex G of ITU-T Rec. X.520 | ISO/IEC 9594-6). If this auxiliary object class is not used together with the **oidCobj** object class, the **tagOid** attribute shall be present. It provides information about how a UII may be formatted to be globally unique in URN format.

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

6.24 URN to URL

The *URN to URL* auxiliary object class is used to add attributes to an entry holding a URL to associated information content related to an identifier and/or the associated information content itself.

```
urnToUrl OBJECT-CLASS ::= {  
    SUBCLASS OF { top }  
    KIND         auxiliary  
    MUST CONTAIN { uriInUrn | contentUrl }  
    MAY CONTAIN  { tagOid }  
    ID           id-oc-urnToUrl }
```

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 9594-7:2008

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.

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

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                oidC1obj
    WITH ATTRIBUTES      { oidC1 }
    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                oidC2obj
    WITH ATTRIBUTES      { oidC2 }
    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-oidNameForm }
```

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) 6}

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

authenticationFramework, certificateExtensions, id-nf, id-oc, informationFramework,
objectClass, selectedAttributeTypes

FROM UsefulDefinitions {joint-iso-itu-t ds(5) module(1) usefulDefinitions(0) 6}

alias, ATTRIBUTE, NAME-FORM, OBJECT-CLASS, top

FROM InformationFramework informationFramework

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

businessCategory, commonName, contentUri, countryName, description, destinationIndicator,
dmdName, facsimileTelephoneNumber, internationalISDNNumber, knowledgeInformation,
localityName, member, organizationalUnitName, organizationName, owner,
physicalDeliveryOfficeName, postalAddress, postalCode, postOfficeBox, preferredDeliveryMethod,
presentationAddress, registeredAddress, roleOccupant, searchGuide, seeAlso, serialNumber,
stateOrProvinceName, streetAddress, supportedApplicationContext, surname, tagOid,
telephoneNumber, telexNumber, title, uuiFormat, uuiInUrn, uniqueMember, x121Address

FROM SelectedAttributeTypes selectedAttributeTypes

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

authorityRevocationList, cACertificate, certificateRevocationList, crossCertificatePair,
deltaRevocationList, supportedAlgorithms, userCertificate, userPassword

FROM AuthenticationFramework authenticationFramework

-- from ITU-T Rec. X.660 | ISO/IEC 9834-1

oidC, oidC1, oidC2

FROM OidDirectoryNameDef {joint-iso-itu-t registration-procedures(17) module(1)

oidDirectoryNameDef(1)} ;

-- 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 }

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 { oidC1 }
 ID id-oc-oidC1obj }

oidC2obj OBJECT-CLASS ::= {
 SUBCLASS OF { top }
 MUST CONTAIN { oidC2 }
 ID id-oc-oidC2obj }

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

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

urnToUri OBJECT-CLASS ::= {
 SUBCLASS OF { top }
 KIND auxiliary
 MUST CONTAIN { uiiInUrn | contentUri }
 MAY CONTAIN { tagOid }
 ID id-oc-urnToUri }

-- 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 }

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

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                oidC1obj
    WITH ATTRIBUTES      { oidC1 }
    ID                   id-nf-oidC1NameForm }

oidC2NameForm NAME-FORM ::= {
    NAMES                oidC2obj
    WITH ATTRIBUTES      { oidC2 }
    ID                   id-nf-oidC2NameForm }

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

```

```

-- 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 ITU-T Rec. X.501
--				ISO/IEC 9594-2
-- id-oc-alias	OBJECT IDENTIFIER	::=	{id-oc 1}	Defined in ITU-T Rec. X.501
--				ISO/IEC 9594-2
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 ITU-T Rec. X.509
--				ISO/IEC 9594-8
id-oc-dmd	OBJECT IDENTIFIER	::=	{id-oc 20}	
-- id-oc-pkiUser	OBJECT IDENTIFIER	::=	{id-oc 21}	Defined in ITU-T Rec. X.509
--				ISO/IEC 9594-8
-- id-oc-pkiCA	OBJECT IDENTIFIER	::=	{id-oc 22}	Defined in ITU-T Rec. X.509
--				ISO/IEC 9594-8
-- id-oc-deltaCRL	OBJECT IDENTIFIER	::=	{id-oc 23}	Defined in ITU-T Rec. X.509
--				ISO/IEC 9594-8
-- id-oc-pmiUser	OBJECT IDENTIFIER	::=	{id-oc 24}	Defined in ITU-T Rec. X.509
--				ISO/IEC 9594-8
-- id-oc-pmiAA	OBJECT IDENTIFIER	::=	{id-oc 25}	Defined in ITU-T Rec. X.509
--				ISO/IEC 9594-8
-- id-oc-pmiSOA	OBJECT IDENTIFIER	::=	{id-oc 26}	Defined in ITU-T Rec. X.509
--				ISO/IEC 9594-8
-- id-oc-attCertCRLDistributionPts	OBJECT IDENTIFIER	::=	{id-oc 27}	Defined in ITU-T Rec. X.509
--				ISO/IEC 9594-8
-- id-oc-parent	OBJECT IDENTIFIER	::=	{id-oc 28}	Defined in ITU-T Rec. X.501
--				ISO/IEC 9594-2
-- id-oc-child	OBJECT IDENTIFIER	::=	{id-oc 29}	Defined in ITU-T Rec. X.501
--				ISO/IEC 9594-2
-- id-oc-cpCps	OBJECT IDENTIFIER	::=	{id-oc 30}	Defined in ITU-T Rec. X.509
--				ISO/IEC 9594-8

ISO/IEC 9594-7:2008 (E)

-- id-oc-pkiCertPath	OBJECT IDENTIFIER	::=	{id-oc 31}	Defined in ITU-T Rec. X.509 ISO/IEC 9594-8
--				
-- id-oc-privilegePolicy	OBJECT IDENTIFIER	::=	{id-oc 32}	Defined in ITU-T Rec. X.509 ISO/IEC 9594-8
--				
-- id-oc-pmiDelegationPath	OBJECT IDENTIFIER	::=	{id-oc 33}	Defined in ITU-T Rec. X.509 ISO/IEC 9594-8
--				
-- id-oc-protectedPrivilegePolicy	OBJECT IDENTIFIER	::=	{id-oc 34}	Defined in ITU-T Rec. X.509 ISO/IEC 9594-8
--				
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-uidToUrn	OBJECT IDENTIFIER	::=	{id-oc 38}	
id-oc-urnToUri	OBJECT IDENTIFIER	::=	{id-oc 39}	
-- id-oc-integrityInfo	OBJECT IDENTIFIER	::=	{id-oc 40}	Defined in ITU-T Rec. X.501 ISO/IEC 9594-2
--				
-- 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-dSANNameForm	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}	
END -- SelectedObjectClasses				

Annex B

Suggested name forms and DIT structures

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

This annex suggests a 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 is applied. Each DMD is responsible for creating its own DIT structure and structure rules that may differ from this example.

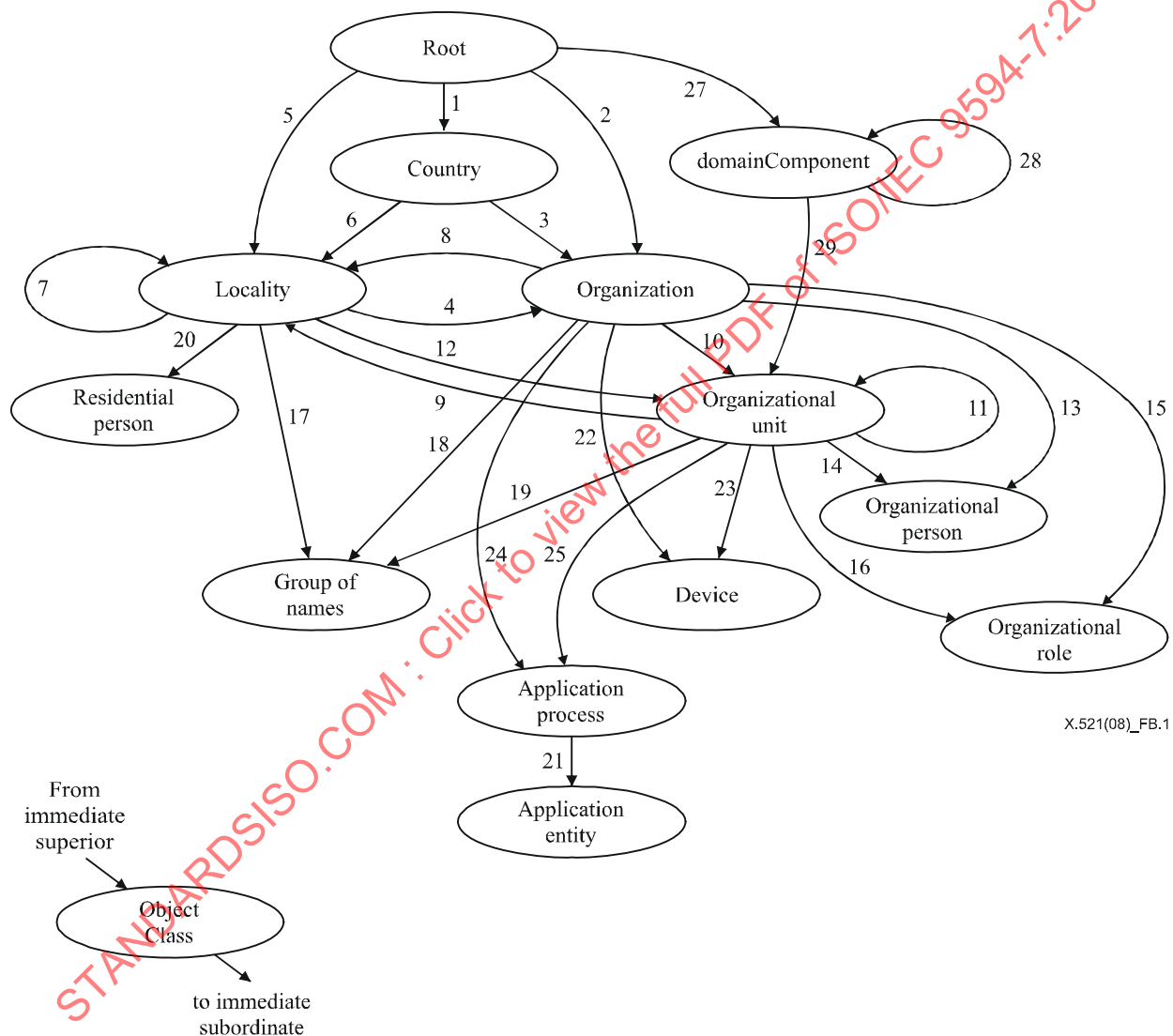


Figure B.1 – Example DIT structure

B.1 Country

Attribute **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 }
```