INTERNET-DRAFT
Intended Category: Standard Track
Expires: November 2004
Updates: RFC 2247, RFC 2798
Obsoletes: RFC 2256

K. Dally, Editor
The MITRE Corp.

May 2004

LDAP: Schema for User Applications
<draft-ietf-ldapbis-user-schema-07>

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC 2026.

This document is intended to be, after appropriate review and revision, submitted to the RFC Editor as a Standard Track document. Distribution of this memo is unlimited. Technical discussion of this document will take place on the IETF LDAP Revision Working Group (LDAPbis) mailing list <ietf-ldapbis@openldap.org>. Please send editorial comments directly to the author <kdally@mitre.org>.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts. Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt.

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

Copyright Notice

Abstract

This document is a integral part of the Lightweight Directory Access Protocol (LDAP) technical specification [ROADMAP]. It provides a technical specification of attribute types and object classes intended for use by LDAP directory clients for many directory services, such as, White Pages. These objects are widely used as a basis for the schema in many LDAP directories. This document does

not cover attributes used for the administration of directory servers, nor does it include directory objects defined for specific uses in other documents.

Table of Contents

1. Introduction

This document provides an overview of attribute types and object classes intended for use by Lightweight Directory Access Protocol directory clients for many directory services, such as, White Pages. Originally specified in the X.500 [X.500] documents, these objects are widely used as a basis for the schema in many LDAP directories. This document does not cover attributes used for the administration of directory servers, nor does it include directory objects defined for specific uses in other documents.

1.1 Situation

This document is a integral part of the LDAP technical specification [ROADMAP] which obsoletes the previously defined LDAP technical specification [RFC3377] in its entirety. In terms of RFC 2256, Sections 6 and 8 of RFC 2256 are obsoleted by [Syntaxes]. Sections 5.1, 5.2, 7.1 and 7.2 of RFC 2256 are obsoleted by [Models]. The remainder of RFC 2256 is obsoleted by this document. Section 3.4 of this document supercedes the technical specification for the 'dc' attribute type found in RFC 2247.[editor's note: Substitute replacement RFC at time of publication.] The remainder of RFC 2247 remains in force.

This document updates RFC 2798 by replacing the informative description of the 'uid' attribute type, with the definitive description provided in Section 2.39 of this document.

A number of schema elements which were included in the previous revision of the LDAP Technical Specification are not included in this revision of LDAP. PKI-related schema elements are now specified in [LDAP-PKI]. Unless reintroduced in future technical specifications, the remainder are to be considered Historic.

1.2 Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

1.3 General Issues

This document references Syntaxes given in Section 3 of [Syntaxes] and Matching Rules specified in Section 4 of [Syntaxes].

The definitions of Attribute Types and Object Classes are written using the ABNF form of AttributeTypeDescription and ObjectClassDescription given in [Models]. Lines have been folded for readability.

1.4 Source

The schema definitions in this document are based on those found in the X.500-series [X.520] and [X.521], RFC 2798 [RFC2798] and RFC 2247 [RFC2247], specifically:

Sections	Source
2.1 - 2.3	X.520 [X.520]
2.4	RFC 2247 [RFC2247]
2.5 - 2.38	X.520 [X.520]
2.39	RFC 2798 [2798]
2.40 - 2.43	X.520 [X.520]
3.1 - 3.12	X.521 [X.521]

However, the descriptions in this document SHALL be considered definitive for use in LDAP.

2. Attribute Types

The Attribute Types contained in this section hold user information.

There is no requirement that servers implement the following attribute types:

```
searchGuide
teletexTerminalIdentifier
```

In fact, their use is greatly discouraged.

An LDAP server implementation SHOULD recognize the rest of the attribute types described in this section.

2.1 businessCategory

The businessCategory attribute type describes the kinds of business performed by an organization (e.g., "banking", "transportation"). Each kind is one value of this multi-valued attribute.

```
( 2.5.4.15 NAME 'businessCategory'
   EQUALITY caseIgnoreMatch
   SUBSTR caseIgnoreSubstringsMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 )
```

1.3.6.1.4.1.1466.115.121.1.15 refers to the Directory String syntax [Syntaxes].

2.2 c

The c (countryName) attribute type contains a two-letter ISO 3166 [ISO3166] country code (e.g., "DE"). (Source: X.520)

```
( 2.5.4.6 NAME 'c'
   SUP name
   SINGLE-VALUE )
```

2.3 cn

The cn (commonName) attribute type contains names of an object (e.g., "Martin K Smith", "Marty Smith", "printer12"). Each name is one value of this multi-valued attribute. If the object corresponds to a person, it is typically the person's full name. (Source: X.520)

```
( 2.5.4.3 NAME 'cn'
   SUP name )
```

2.4 dc

The dc (short for domainComponent) attribute type is a string holding one component, a <label> [RFC1034}, of a DNS domain name (e.g., "example" or "com", but not "example.com"). The encoding of IA5String for use in LDAP is simply the characters of the string

itself. The equality matching rule is case insensitive, as is today's DNS.

```
( 0.9.2342.19200300.100.1.25 NAME 'dc'
   EQUALITY caseIgnoreIA5Match
   SUBSTR caseIgnoreIA5SubstringsMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.26

   SINGLE-VALUE )
```

1.3.6.1.4.1.1466.115.121.1.26 refers to the IA5 String syntax [Syntaxes].

It is noted that the directory will not ensure that values of this attribute conform to the label production [RFC1034]. It is the application responsibility to ensure domains it stores in this attribute are appropriately represented.

It is also noted that applications supporting Internationalized Domain Names SHALL use the ToASCII method [RFC3490] to produce <label> components of the <domain> production.

2.5 description

The description attribute type contains human-readable descriptive phrases about the object (e.g., "a color printer", "Maintenance is

done every Monday, at 1pm."). Each description is one value of this multi-valued attribute.

```
( 2.5.4.13 NAME 'description'
   EQUALITY caseIgnoreMatch
   SUBSTR caseIgnoreSubstringsMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 )
```

1.3.6.1.4.1.1466.115.121.1.15 refers to the Directory String syntax [Syntaxes].

2.6 destinationIndicator

The destinationIndicator attribute type contains country and city strings, associated with the object (the addressee), needed to provide the Public Telegram Service. Each string is one value of this multi-valued attribute. The strings are composed in accordance with CCITT Recommendations F.1 [F.1] and F.31 [F.31].

```
( 2.5.4.27 NAME 'destinationIndicator'
   EQUALITY caseIgnoreMatch
   SUBSTR caseIgnoreSubstringsMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.44 )
```

1.3.6.1.4.1.1466.115.121.1.44 refers to the Printable String syntax [Syntaxes].

2.7 distinguishedName

The distinguishedName attribute type is the attribute supertype from which attribute types with DN syntax inherit, instead of containing values which name the object itself. The attribute type is multi-valued.

It is unlikely that values of this type itself will occur in an entry. LDAP server implementations which do not support attribute subtyping need not recognize this attribute in requests. Client implementations MUST NOT assume that LDAP servers are capable of performing attribute subtyping.

```
( 2.5.4.49 NAME 'distinguishedName'
   EQUALITY distinguishedNameMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.12 )
```

1.3.6.1.4.1.1466.115.121.1.12 refers to the DN syntax [Syntaxes].

2.8 dnQualifier

The dnQualifier attribute type contains disambiguating information strings to add to the relative distinguished name of an entry. The information is intended for use when merging data from multiple sources in order to prevent conflicts between entries which would otherwise have the same name. Each string is one value of this multi-valued attribute. It is recommended that a value of the dnQualifier attribute be the same for all entries from a particular source.

```
( 2.5.4.46 NAME 'dnQualifier'
   EQUALITY caseIgnoreMatch
   ORDERING caseIgnoreOrderingMatch
   SUBSTR caseIgnoreSubstringsMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.44 )
```

1.3.6.1.4.1.1466.115.121.1.44 refers to the Printable String syntax [Syntaxes].

2.9 enhancedSearchGuide

The enhancedSearchGuide attribute type contains sets of information for use by directory clients in constructing search filters. Each set is one value of this multi-valued attribute.

```
( 2.5.4.47 NAME 'enhancedSearchGuide'
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.21 )
```

1.3.6.1.4.1.1466.115.121.1.21 refers to the Enhanced Guide

syntax [Syntaxes].

2.10 facsimileTelephoneNumber

The facsimileTelephoneNumber attribute type contains telephone numbers (and, optionally, the parameters) for facsimile terrminals. Each telephone number is one value of this multi-valued attribute.

```
( 2.5.4.23 NAME 'facsimileTelephoneNumber'
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.22 )
```

1.3.6.1.4.1.1466.115.121.1.22 refers to the Facsimile Telephone Number syntax [Syntaxes].

2.11 generationQualifier

The generationQualifier attribute type contains name strings that are the part of a person's name which typically is the suffix, as in "IIIrd" or "3rd". Each string is one value of this multi-valued attribute.

```
( 2.5.4.44 NAME 'generationQualifier'
   SUP name )
```

2.12 givenName

The givenName attribute type contains name strings that are the part of a person's name which is not their surname. Each string is one value of this multi-valued attribute.

```
   ( 2.5.4.42 NAME 'givenName'
      SUP name )
```

2.13 houseIdentifier

The houseIdentifier attribute type contains identifiers for a building within a location. Each identifier is one value of this multi-valued attribute.

```
( 2.5.4.51 NAME 'houseIdentifier'
   EQUALITY caseIgnoreMatch
   SUBSTR caseIgnoreSubstringsMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 )
```

1.3.6.1.4.1.1466.115.121.1.15 refers to the Directory String syntax [Syntaxes].

2.14 initials

The initials attribute type contains strings of initials of some or all of an individual's names, except the surname(s) (e.g., "K. A.", "K"). Each string is one value of this multi-valued attribute.

```
( 2.5.4.43 NAME 'initials'
   SUP name )
```

2.15 internationalISDNNumber

The internationalISDNNumber attribute type contains ISDN addresses, as defined in ITU Recommendation E.164 [E.164]. Each address is one value of this multi-valued attribute.

```
( 2.5.4.25 NAME 'internationalISDNNumber'
   EQUALITY numericStringMatch
   SUBSTR numericStringSubstringsMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.36 )
```

1.3.6.1.4.1.1466.115.121.1.36 refers to the Numeric String syntax [Syntaxes].

2.16 l

The l (localityName) attribute type contains names of a locality or place, such as a city, county or other geographic region (e.g., "Geneva"). Each name is one value of this multi-valued attribute. (Source: X.520)

```
( 2.5.4.7 NAME 'l'
   SUP name )
```

2.17 member

The member attribute type contains the Distinguished Names of objects that are on a list or in a group. Each name is one value of this multi-valued attribute.

```
( 2.5.4.31 NAME 'member'
   SUP distinguishedName )
```

2.18 name

The name attribute type is the attribute supertype from which attributes with the name syntax inherit. Such attributes are typically used for naming. The attribute type is multi-valued.

It is unlikely that values of this type itself will occur in an entry. LDAP server implementations which do not support attribute subtyping need not recognize this attribute in requests. Client implementations MUST NOT assume that LDAP servers are capable of performing attribute subtyping.

```
( 2.5.4.41 NAME 'name'
   EQUALITY caseIgnoreMatch
   SUBSTR caseIgnoreSubstringsMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 )
```

1.3.6.1.4.1.1466.115.121.1.15 refers to the Directory String syntax [Syntaxes].

2.19 o

The o (organizationName) attribute type contains the names of an organization (e.g., "IETF", "Internet Engineering Task Force"). Each name is one value of this multi-valued attribute. (Source: X.520)

```
( 2.5.4.10 NAME 'o'
   SUP name )
```

2.20 ou

The ou (organizationalUnitName) attribute type contains the names of an organizational unit (e.g., "Application Area", "LDAPbis WG"). Each name is one value of this multi-valued attribute. (Source: X.520)

```
( 2.5.4.11 NAME 'ou'
   SUP name )
```

2.21 owner

The owner attribute type contains the Distinguished Names of objects that have an ownership responsibility for the object that is owned. (e.g., The list object, "cn=All Employees, ou=Mailing List, o=Widget',' Inc.", is owned by the role object, "cn=ou=Human Resources Director, ou=employee, o=Widget',' Inc.") Each name is one value of this multi-valued attribute.

```
( 2.5.4.32 NAME 'owner'
   SUP distinguishedName )
```

2.22 physicalDeliveryOfficeName

The physicalDeliveryOfficeName attribute type contains names that a Postal Service uses to identify a post office (e.g., "Bremerhaven, Main", "Bremerhaven, Bonnstrasse").

```
( 2.5.4.19 NAME 'physicalDeliveryOfficeName'
   EQUALITY caseIgnoreMatch
   SUBSTR caseIgnoreSubstringsMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 )
```

1.3.6.1.4.1.1466.115.121.1.15 refers to the Directory String syntax [Syntaxes].

2.23 postalAddress

The postalAddress attribute type contains addresses used by a Postal Service to perform services for the object (e.g., "15 Main St., Ottawa, Canada"). Each address is one value of this multi-valued attribute.

```
( 2.5.4.16 NAME 'postalAddress'
   EQUALITY caseIgnoreListMatch
   SUBSTR caseIgnoreListSubstringsMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.41 )
```

1.3.6.1.4.1.1466.115.121.1.41 refers to the Postal Address syntax [Syntaxes].

2.24 postalCode

The postalCode attribute type contains codes used by a Postal Service to identify a postal service zones, such as the southern quadrant of a city (e.g., "22180"). Each code is one value of this

multi-valued attribute.

```
( 2.5.4.17 NAME 'postalCode'
   EQUALITY caseIgnoreMatch
   SUBSTR caseIgnoreSubstringsMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 )
```

1.3.6.1.4.1.1466.115.121.1.15 refers to the Directory String syntax [Syntaxes].

2.25 postOfficeBox

The postOfficeBox attribute type contains numbers that a Postal Service uses when a customer arranges to receive mail at a box on premises of the Postal Service (e.g., "Box 45"). Each number is one value of this multi-valued attribute.

```
( 2.5.4.18 NAME 'postOfficeBox'
   EQUALITY caseIgnoreMatch
   SUBSTR caseIgnoreSubstringsMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 )
```

1.3.6.1.4.1.1466.115.121.1.15 refers to the Directory String syntax [Syntaxes].

2.26 preferredDeliveryMethod

The preferredDeliveryMethod attribute type contains an indication of the preferred method of getting a message to the object. For example, if mhs-delivery is preferred over telephone-delivery, which is preferred over all other methods, the value of the value would be {1, 9}.

```
( 2.5.4.28 NAME 'preferredDeliveryMethod'
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.14
   SINGLE-VALUE )
```

1.3.6.1.4.1.1466.115.121.1.14 refers to the Delivery Method syntax [Syntaxes].

2.27 registeredAddress

The registeredAddress attribute type contains postal addresses suitable for reception of telegrams or expedited documents, where it is necessary to have the recipient accept delivery (e.g., "Receptionist, Widget Inc., 15 Main St., Ottawa, Canada"). Each address is one value of this multi-valued attribute.

```
( 2.5.4.26 NAME 'registeredAddress'
   SUP postalAddress
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.41 )
```

1.3.6.1.4.1.1466.115.121.1.41 refers to the Postal Address syntax [Syntaxes].

2.28 roleOccupant

The roleOccupant attribute type contains the Distinguished Names of objects(normally people) that fulfill the responsibilities of a role

object. For example, the role object, "cn=Human Resources Director, ou=Position, o=Widget',' Inc.", is fulfilled by two people whose object names are "cn=Mary Smith, ou=employee, Widget',' Inc." and

"cn=James Brown, ou=employee, o=Widget',' Inc." Each name is one value of this multi-valued attribute.

```
( 2.5.4.33 NAME 'roleOccupant'
   SUP distinguishedName )
```

2.29 searchGuide

The searchGuide attribute type contains sets of information for use by clients in constructing search filters. It is superseded by enhancedSearchGuide, described above in section 2.9.

```
( 2.5.4.14 NAME 'searchGuide'
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.25 )
```

1.3.6.1.4.1.1466.115.121.1.25 refers to the Guide syntax [Syntaxes].

2.30 seeAlso

The seeAlso attribute type contains Distinguished Names of objects that are related to the subject object. For example, the person object, "cn=James Brown, ou=employee, o=Widget Inc." is related to the role objects, "cn=Football Team Captain, ou=sponsored activities, o=Widget Inc." and "cn=Chess Team, ou=sponsored activities, o=Widget Inc.". Each name is one value of this multi-valued attribute.

```
( 2.5.4.34 NAME 'seeAlso'
   SUP distinguishedName )
```

2.31 serialNumber

The serialNumber attribute type contains the serial numbers of devices (e.g., "WI-3005". Each number is one value of this multi-valued attribute.

```
( 2.5.4.5 NAME 'serialNumber'
   EQUALITY caseIgnoreMatch
   SUBSTR caseIgnoreSubstringsMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.44 )
```

1.3.6.1.4.1.1466.115.121.1.44 refers to the Printable String syntax [Syntaxes].

2.32 sn

The sn (surname)attribute type contains name strings for the family names of a person (e.g., "Smith"). Each string is one value of this multi-valued attribute. (Source: X.520)

```
( 2.5.4.4 NAME 'sn'
   SUP name )
```

2.33 st

The st (stateOrProvinceName) attribute type contains the full names of states or provinces, (e.g. "California"). Each name is one value of this multi-valued attribute.

```
( 2.5.4.8 NAME 'st'
   SUP name )
```

2.34 street

The street (streetAddress) attribute type contains physical addresses of the object to which the entry corresponds, such as an address for package delivery. Each address is one value of this multi-valued attribute.

```
( 2.5.4.9 NAME 'street'
   EQUALITY caseIgnoreMatch
   SUBSTR caseIgnoreSubstringsMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 )
```

1.3.6.1.4.1.1466.115.121.1.15 refers to the Directory String syntax [Syntaxes].

2.35 telephoneNumber

The telephoneNumber attribute type contains telephone numbers complying with ITU Recommendation E.123 [E.123] (e.g., 1 234 567 8901) Each number is one value of this multi-valued attribute.

```
( 2.5.4.20 NAME 'telephoneNumber'
   EQUALITY telephoneNumberMatch
   SUBSTR telephoneNumberSubstringsMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.50 )
```

1.3.6.1.4.1.1466.115.121.1.50 refers to the Telephone Number syntax [Syntaxes].

2.36 teletexTerminalIdentifier

The withdrawal of Rec. F.200 has resulted in the withdrawal of this attribute.

```
( 2.5.4.22 NAME 'teletexTerminalIdentifier'
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.51 )
```

2.37 telexNumber

The telexNumber attribute type contains sets of strings which are a telex number, country code, and answerback code of a telex terminal. Each set is one value of this multi-valued attribute.

```
( 2.5.4.21 NAME 'telexNumber'
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.52 )
```

1.3.6.1.4.1.1466.115.121.1.52 refers to the Telex Number syntax [Syntaxes].

2.38 title

This attribute contains the title, such as "Vice President", of a person in their organizational context.

```
( 2.5.4.12 NAME 'title'
   SUP name )
```

2.39 uid

The uid attribute type contains computer system login names associated with the object. (Source: RFC 1274, RFC 2798). Each name is one value of this multi-valued attribute.

```
( 0.9.2342.19200300.100.1.1
   NAME 'uid'
   EQUALITY caseIgnoreMatch
   SUBSTR caseIgnoreSubstringsMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 )
```

1.3.6.1.4.1.1466.115.121.1.15 refers to the Directory String syntax [Syntaxes].

2.40 uniqueMember

The uniqueMember attribute type contains the Distinguished Names of an object that is on a list or in a group, where the Relative Distinguished Names of the object include a value that distinguishs between objects when a distinguished name has been reused. For example, if "ou=1st Battalion, o=Defense, c=US" is a battalion that was disbanded, establishing a new battalion with the "same" name would have a uid value added, resulting in "ou=1st Battalion#'010101', o=Defense, c=US".

```
   ( 2.5.4.50 NAME 'uniqueMember'
      EQUALITY uniqueMemberMatch
      SYNTAX 1.3.6.1.4.1.1466.115.121.1.34 )
```

1.3.6.1.4.1.1466.115.121.1.34 refers to the Name and Optional UID syntax [Syntaxes].

2.41 userPassword

The userPassword attribute type contains character strings that are known only to the user and the system to which the user has access. Each string is one value of this multi-valued attribute.

The application SHOULD prepare textual strings used as passwords by transcoding them to Unicode, applying SASLprep [SASLprep], and encoding as UTF-8.

```
( 2.5.4.35 NAME 'userPassword'
   EQUALITY octetStringMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.40 )
```

1.3.6.1.4.1.1466.115.121.1.40 refers to the Octet String syntax [Syntaxes].

Passwords are stored using an Octet String syntax and are not encrypted. Transfer of cleartext passwords is strongly discouraged where the underlying transport service cannot guarantee confidentiality and may result in disclosure of the password to unauthorized parties.

An example of a need for multiple values in the userPassword attribute is an environment where every month the user was expected to use a different password generated by some automated system. During transitional periods, like say the last and first day of the periods, it may be necessary to allow two passwords for the two consecutive periods to be valid in the system.

2.42 x121Address

The x121Address attribute type contains data network addresses (e.g., 36111222333444555) as defined by ITU Recommendation X.121

[X.121]. Each address is one value of this multi-valued attribute.

```
( 2.5.4.24 NAME 'x121Address'
   EQUALITY numericStringMatch
   SUBSTR numericStringSubstringsMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.36 )
```

1.3.6.1.4.1.1466.115.121.1.36 refers to the Numeric String syntax [Syntaxes].

2.43 x500UniqueIdentifier

The x500UniqueIdentifier attribute type contains binary strings that are used to distinguish between objects when a distinguished name has been reused. Each string is one value of this multi-valued

attribute. In X.520 [X.520], this attribute type is called uniqueIdentifier. This is a different attribute type from both the "uid" and "uniqueIdentifier" attribute types.

```
( 2.5.4.45 NAME 'x500UniqueIdentifier'
   EQUALITY bitStringMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.6 )
```

1.3.6.1.4.1.1466.115.121.1.6 refers to the Bit String syntax [Syntaxes].

3. Object Classes

LDAP servers SHOULD recognize all the Object Classes listed here as values of the objectClass attribute (see [Models]).

3.1 applicationProcess

The applicationProcess object class definition is the basis of an entry which represents an application executing in a computer system.

```
( 2.5.6.11 NAME 'applicationProcess'
   SUP top
   STRUCTURAL
   MUST cn
   MAY ( seeAlso $
         ou $
         l $
         description ) )
```

3.2 country

The country object class definition is the basis of an entry which represents a country.

```
( 2.5.6.2 NAME 'country'
   SUP top
   STRUCTURAL
   MUST c
   MAY ( searchGuide $
         description ) )
```

3.3 device

The device object class is the basis of an entry which represents an appliance or computer or network element.

```
( 2.5.6.14 NAME 'device'
```

```
      SUP top
      STRUCTURAL
      MUST cn
```

```
      MAY ( serialNumber $
            seeAlso $
            owner $
            ou $
            o $
            l $
            description ) )
```

3.4 groupOfNames

The groupOfNames object class is the basis of an entry which represents a set of named objects including information related to the purpose or maintenance of the set.

```
( 2.5.6.9 NAME 'groupOfNames'
   SUP top
   STRUCTURAL
   MUST ( member $
          cn )
   MAY ( businessCategory $
         seeAlso $
         owner $
         ou $
         o $
         description ) )
```

3.5 groupOfUniqueNames

The groupOfUniqueNames object class is the same as the groupOfNames object class except that the object names are not repeated or reassigned within a set scope.

```
( 2.5.6.17 NAME 'groupOfUniqueNames'
   SUP top
   STRUCTURAL
   MUST ( uniqueMember $
          cn )
   MAY ( businessCategory $
         seeAlso $
         owner $
         ou $
         o $
         description ) )
```

3.6 locality

The locality object class is the basis of an entry which represents a place in the physical world.

```
   ( 2.5.6.3 NAME 'locality'
      SUP top
      STRUCTURAL
```

```
      MAY ( street $
            seeAlso $
            searchGuide $
            st $
            l $
            description ) )
```

3.7 organization

The organization object class is the basis of an entry which represents a structured group of people.

```
( 2.5.6.4 NAME 'organization'
   SUP top
   STRUCTURAL
   MUST o
   MAY ( userPassword $ searchGuide $ seeAlso $
         businessCategory $ x121Address $ registeredAddress $
         destinationIndicator $ preferredDeliveryMethod $
         telexNumber $ teletexTerminalIdentifier $ telephoneNumber $
         internationaliSDNNumber $ facsimileTelephoneNumber $
         street $ postOfficeBox $ postalCode $
         postalAddress $ physicalDeliveryOfficeName $ st $
         l $ description ) )
```

3.8 organizationalPerson

The organizationalPerson object class is the basis of an entry which represents a person in relation to an organization.

```
( 2.5.6.7 NAME 'organizationalPerson'
   SUP person
   STRUCTURAL
   MAY ( title $ x121Address $ registeredAddress $
         destinationIndicator $ preferredDeliveryMethod $
         telexNumber $ teletexTerminalIdentifier $ telephoneNumber $
         internationaliSDNNumber $ facsimileTelephoneNumber $
         street $ postOfficeBox $ postalCode $ postalAddress $
         physicalDeliveryOfficeName $ ou $ st $ l ) )
```

3.9 organizationalRole

The organizationalRole object class is the basis of an entry which represents a job or function or position in an organization.

```
( 2.5.6.8 NAME 'organizationalRole'
   SUP top
   STRUCTURAL
   MUST cn
```

```
      MAY ( x121Address $ registeredAddress $ destinationIndicator $
            preferredDeliveryMethod $ telexNumber $
            teletexTerminalIdentifier $ telephoneNumber $
```

```
        internationaliSDNNumber $ facsimileTelephoneNumber $
        seeAlso $ roleOccupant $ preferredDeliveryMethod $
        street $ postOfficeBox $ postalCode $ postalAddress $
        physicalDeliveryOfficeName $ ou $ st $ l $ description ) )
```

3.10 organizationalUnit

The organizationalUnit object class is the basis of an entry which represents a piece of an organization.

```
( 2.5.6.5 NAME 'organizationalUnit'
   SUP top
   STRUCTURAL
   MUST ou
   MAY ( businessCategory $ description $ destinationIndicator $
         facsimileTelephoneNumber $ internationaliSDNNumber $ l $
         physicalDeliveryOfficeName $ postalAddress $ postalCode $
         postOfficeBox $ preferredDeliveryMethod $
         registeredAddress $ searchGuide $ seeAlso $ st $ street $
         telephoneNumber $ teletexTerminalIdentifier $ telexNumber $
         userPassword $ x121Address ) )
```

3.11 person

The person object class is the basis of an entry which represents a human being.

```
( 2.5.6.6 NAME 'person'
   SUP top
   STRUCTURAL
   MUST ( sn $
         cn )
   MAY ( userPassword $
         telephoneNumber $
         seeAlso $ description ) )
```

3.12 residentialPerson

The residentialPerson object class is the basis of an entry which includes a person's residence in the representation of the person.

```
( 2.5.6.10 NAME 'residentialPerson'
   SUP person
   STRUCTURAL
   MUST l
   MAY ( businessCategory $ x121Address $ registeredAddress $
         destinationIndicator $ preferredDeliveryMethod $
         telexNumber $ teletexTerminalIdentifier $ telephoneNumber $
         internationaliSDNNumber $ facsimileTelephoneNumber $
```

```
        preferredDeliveryMethod $ street $ postOfficeBox $
        postalCode $ postalAddress $ physicalDeliveryOfficeName $
        st $ l ) )
```

4. IANA Considerations

It is requested that the Internet Assigned Numbers Authority (IANA) update the LDAP descriptors registry as indicated in the following template:

Subject: Request for LDAP Descriptor Registration Update
Descriptor (short name): see comment
Object Identifier: see comment
Person & email address to contact for further information:
Kathy Dally <kdally@mitre.org>
Usage: (A = attribute type, O = Object Class) see comment
Specification: RFC XXXX [editor's note: The RFC number will be the one assigned to this document.
Author/Change Controller: IESG

Comments
In the LDAP descriptors registry, the following descriptors (short names) should be updated to refer to RFC XXXX [editor's note: This document].

NAME	Type	OID
applicationProcess	O	2.5.6.11
businessCategory	A	2.5.4.15
c	A	2.5.4.6
cn	A	2.5.4.3
country	O	2.5.6.2
dc	A	0.9.2342.19200300.100.1.25
description	A	2.5.4.13
destinationIndicator	A	2.5.4.27
device	O	2.5.6.14
distinguishedName	A	2.5.4.49
dnQualifier	A	2.5.4.46
enhancedSearchGuide	A	2.5.4.47
facsimileTelephoneNumber	A	2.5.4.23
generationQualifier	A	2.5.4.44
givenName	A	2.5.4.42
groupOfNames	O	2.5.6.9
groupOfUniqueNames	O	2.5.6.17
houseIdentifier	A	2.5.4.51
initials	A	2.5.4.43
internationalISDNNumber	A	2.5.4.25
l	A	2.5.4.7
locality	O	2.5.6.3
member	A	2.5.4.31
name	A	2.5.4.41
o	A	2.5.4.10

organization	O	2.5.6.4
organizationalPerson	O	2.5.6.7
organizationalRole	O	2.5.6.8
organizationalUnit	O	2.5.6.5

ou	A	2.5.4.11
owner	A	2.5.4.32
person	O	2.5.6.6
physicalDeliveryOfficeName	A	2.5.4.19
postalAddress	A	2.5.4.16
postalCode	A	2.5.4.17
postOfficeBox	A	2.5.4.18
preferredDeliveryMethod	A	2.5.4.28
registeredAddress	A	2.5.4.26
residentialPerson	O	2.5.6.10
roleOccupant	A	2.5.4.33
searchGuide	A	2.5.4.14
seeAlso	A	2.5.4.34
serialNumber	A	2.5.4.5
sn	A	2.5.4.4
st	A	2.5.4.8
street	A	2.5.4.9
telephoneNumber	A	2.5.4.20
teletexTerminalIdentifier	A	2.5.4.22
telexNumber	A	2.5.4.21
title	A	2.5.4.12
uid	A	0.9.2342.19200300.100.1.1
uniqueMember	A	2.5.4.50
userPassword	A	2.5.4.35
x121Address	A	2.5.4.24
x500UniqueIdentifier	A	2.5.4.45

5. Security Considerations

Attributes of directory entries are used to provide descriptive information about the real-world objects they represent, which can be people, organizations or devices. Most countries have privacy laws regarding the publication of information about people.

Transfer of cleartext passwords is strongly discouraged where the underlying transport service cannot guarantee confidentiality and may result in disclosure of the password to unauthorized parties.

Multiple attribute values for the userPassword needs to be used with care. Especially reset/deletion of a password by an admin without knowing the old user password gets tricky or impossible if multiple values for different applications are present.

Certainly, applications which intend to replace the userPassword value(s) with new value(s) should use modify/replaceValues (or modify/deleteAttribute+addAttribute). Additionally, server implementations are encouraged to provide administrative controls

which, if enabled, restrict the userPassword attributer to one value.

Note that when used for authentication purposes [AuthMeth], the user need only prove knowledge of one of the values, not all of the values.

6. Acknowledgements

The definitions, on which this document is based, have been developed by committees for telecommunications and international standards.

This document is an update of RFC 2256 by Mark Wahl. RFC 2256 was a product of the IETF ASID Working Group.

The dc attribute type definition in this document supercedes the

specification in RFC 2247 by S. Kille, M. Wahl, A. Grimstad, R. Huber, and S. Sataluri.

The uid attribute type definition in this document supercedes the specification of the userid in RFC 1274 by P. Barker and S. Kille and of the uid in RFC 2798 by M. Smith.

This document is based upon input of the IETF LDAPBIS working group. The author wishes to thank S. Legg and K. Zeilenga for their significant contribution to this update.

7. References

7.1 Normative

[E.123] Notation for national and international telephone numbers, ITU-T Recommendation E.123, 1988

[E.164] The international public telecommunication numbering plan, ITU-T Recommendation E.164, 1997

[ISO3166] ISO 3166, "Codes for the representation of names of countries".

[Models] K. Zeilenga, "LDAP: The Models", draft-ietf-ldapbis-models-xx (a work in progress)

[RFC1034] P. Mockapetris, " DOMAIN NAMES - CONCEPTS AND FACILITIES", RFC 1034, November 1987

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", RFC 2119, March 1997

[RFC3490] Faltstrom P., Hoffman P., Costello A.,
"Internationalizing Domain Names in Applications (IDNA)",
RFC 3490, March 2003

...[ROADMAP] Zeilenga, K., "LDAP: Technical Specification Road Map", draft-ietf-ldapbis-roadmap-xx (a work in progress)

[Syntaxes] S. Legg (editor), "LDAP: Syntaxes", draft-ietf-ldapbis-syntaxes-xx (a work in progress)

[X.121] International numbering plan for public data networks, ITU-T Recommendation X.121, 1996

[X.509] The Directory: Authentication Framework, ITU-T Recommendation X.509, 1993

[X.520] The Directory: Selected Attribute Types, ITU-T Recommendation X.520, 1993

[X.521] The Directory: Selected Object Classes. ITU-T Recommendation X.521, 1993

7.2 Informative

[AUTHMETH] Harrison R., "LDAP: Authentication Methods and Connection Level Security Mechanisms", draft-ietf-ldapbis-authmeth-xx (a work in progress)

[F.1] Operational Provisions For The International Public Telegram Service Transmission System, CCITT Recommmendation F.1, 1992

[F.31] Telegram Retransmission System, CCITT Recommendation F.31, 1988

[LDAP-CERT] Klasen, N., Gietz, P. "An LDAPv3 Schema for X.509 Certificates", Internet Draft draft-klasen-ldap-x509certificate-schema-xx (a work in progress)

[LDAP-CRL] Chadwick, D. W. and M. V. Sahalayev, "Internet X.509 Public Key Infrastructure - LDAP Schema for X.509 CRLs", Internet Draft draft-ietf-pkix-ldap-crl-schema-xx (a work in progress)

[RFC2247] Kille, S., Wahl, M., Grimstad, A., Huber, R., and Sataluri, S., "Using Domains in LDAP/X.500 Distinguished Names", RFC 2247, January 1998

[RFC3377] Hodges, J., Morgan, R., "Lightweight Directory Access Protocol (v3): Technical Specification", RFC 3377, September 2002

[SASLprep] Zeilenga K., "SASLprep: Stringprep profile for user names and passwords", draft-ietf-sasl-saslprep-xx (a work in progress)

[X.500] The Directory, ITU-T Recommendations X.501-X.525, 1993

8. Author's Address

Kathy Dally
The MITRE Corp.
7515 Colshire Dr., H300
McLean VA 22102
USA

Phone: +1 703 883 6058
Email: kdally@mitre.org

9. Full Copyright Statement

Appendix A Changes RFC 2256

This appendix lists the changes that have been made from RFC 2256 to this I-D.

1. Replaced the document title.

2. Removed the IESG Note.

3. Dependencies on RFC 1274 have been eliminated.

4. Added a Security Considerations section and an IANA considerations section.

5. Deleted the conformance requirement for subschema object classes in favor of a statement in [Syntaxes].

6. Added explanation to attribute types and to each object class.

7. Removed Section 4, Syntaxes, and Section 6, Matching Rules, (moved to [Syntaxes]).

8. Removed the certificate-related attribute types:
   authorityRevocationList,
   cACertificate,
   certificateRevocationList,
   crossCertificatePair,
   deltaRevocationList,
   supportedAlgorithms, and
   userCertificate.

   Removed the certificate-related Object Classes:
   certificationAuthority,
   certificationAuthority-V2,
   cRLDistributionPoint,
   strongAuthenticationUser, and
   userSecurityInformation

   LDAP PKI is now discussed in [LDAP-CRL] and {LDAP-CERT].

9. Removed the dmdName, knowledgeInformation, presentationAddress, protocolInformation, and supportedApplicationContext attribute types and the dmd, applicationEntity, and dSA object classes.

10. Deleted the aliasedObjectName and objectClass attribute type definitions. Deleted the alias and top object class

definitions. They are included in [Models].

11. Added the 'dc' attribute type from RFC 2247.

12. Numerous edititorial changes.