

A Summary of the Pilot X.500 Schema for use in LDAPv3
<[draft-ietf-asid-schema-pilot-00.txt](#)>

1. Status of this Memo

This document is an Internet-Draft. 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."

To learn the current status of any Internet-Draft, please check the "1id-abstracts.txt" listing contained in the Internet-Drafts Shadow Directories on ds.internic.net (US East Coast), nic.nordu.net (Europe), ftp.isi.edu (US West Coast), or munnari.oz.au (Pacific Rim).

2. Abstract

This document provides an overview of attribute types and object classes for use in piloting directory services based on X.500 and LDAP.

3. General Issues

This document references syntaxes given in [section 6](#) of this document and section 6 of [\[1\]](#). Matching rules are listed in [section 8](#) of [\[1\]](#).

The attribute type and object class definitions are written using the BNF form of AttributeTypeDescription and ObjectClassDescription given in [\[1\]](#). Lines have been folded for readability.

4. Source

The majority of attributes and object classes are based on those defined in [RFC 1274](#) [\[2\]](#). In addition, there are new schema elements defined in this document, based on current work in schema definition in the IETF and other organizations.

5. User Attributes

Servers SHOULD recognize all the attributes of these sections.

5.1. Definitions from [RFC 1274](#)

5.1.1. uid

The uid attribute type specifies a computer system login name. This attribute is also known as userid.

```
( 0.9.2342.19200300.100.1.1 NAME 'uid' EQUALITY caseIgnoreMatch  
  SUBSTR caseIgnoreSubstringsMatch SYNTAX 'DirectoryString{256}' )
```

5.1.2. mail

The mail attribute type specifies an electronic mailbox attribute following the syntax specified in [RFC 822](#). Note that this attribute should not be used for non-Internet-format mailboxes. This attribute is also known as rfc822Mailbox.

```
( 0.9.2342.19200300.100.1.3 NAME 'mail' EQUALITY caseIgnoreIA5Match  
  SUBSTR caseIgnoreIA5SubstringsMatch SYNTAX 'IA5String{256}' )
```

5.1.3. drink

The drink attribute type specifies the favourite drink of an object (or person).

```
( 0.9.2342.19200300.100.1.5 NAME 'drink' EQUALITY caseIgnoreMatch  
  SUBSTR caseIgnoreSubstringsMatch SYNTAX 'DirectoryString{256}' )
```

5.1.4. roomNumber

The roomNumber attribute type specifies the room number of an object.

```
( 0.9.2342.19200300.100.1.6 NAME 'roomNumber' EQUALITY caseIgnoreMatch  
  SUBSTR caseIgnoreSubstringsMatch SYNTAX 'DirectoryString{256}' )
```

5.1.5. userClass

This attribute is used to hold a descriptive category name of which the object is a member. Examples might be "faculty" and "student".

```
( 0.9.2342.19200300.100.1.8 NAME 'userClass' EQUALITY caseIgnoreMatch
```

SUBSTR caseIgnoreSubstringsMatch SYNTAX 'DirectoryString{256}')

5.1.6. host

The host attribute type specifies the (domain) name of a host computer.

```
( 0.9.2342.19200300.100.1.9 NAME 'host' EQUALITY caseIgnoreMatch  
  SUBSTR caseIgnoreSubstringsMatch SYNTAX 'DirectoryString{256}' )
```

Wahl

Page 2

INTERNET-DRAFT A Summary of the Pilot X.500 Schema For LDAP March 1997

5.1.7. manager

The manager attribute type specifies the manager of an object represented by an entry.

```
( 0.9.2342.19200300.100.1.10 NAME 'manager'  
  EQUALITY distinguishedNameMatch SYNTAX 'DN' )
```

5.1.8. homePhone

The homePhone attribute type specifies a home telephone number associated with a person. Attribute values should follow the agreed format for international telephone numbers: i.e., "+44 171 123 4567".

```
( 0.9.2342.19200300.100.1.20 NAME 'homePhone' EQUALITY telephoneNumberMatch  
  SUBSTR telephoneNumberSubstringsMatch SYNTAX 'TelephoneNumber{32}' )
```

5.1.9. secretary

The secretary attribute type specifies the secretary of a person. The attribute value for secretary is a distinguished name.

```
( 0.9.2342.19200300.100.1.21 NAME 'secretary'  
  EQUALITY distinguishedNameMatch SYNTAX 'DN' )
```

5.1.10. otherMailbox

The otherMailbox attribute type specifies values for electronic mailbox types other than [RFC 822](#).

```
( 0.9.2342.19200300.100.1.22 NAME 'otherMailbox' SYNTAX 'OtherMailbox' )
```

5.1.11. dc

The dc attribute type specifies one component of a domain, such as "com"

or "edu".

```
( 0.9.2342.19200300.100.1.25 NAME 'dc' EQUALITY caseIgnoreIA5Match
  SUBSTR caseIgnoreIA5SubstringsMatch SYNTAX 'IA5String' )
```

5.1.12. dnsRecord

The dnsRecord attribute type specifies the domain resources associated with an object.

```
( 0.9.2342.19200300.100.1.26 NAME 'dnsRecord'
  EQUALITY caseExactIA5Match SYNTAX 'IA5String' )
```

Wahl

Page 3

INTERNET-DRAFT A Summary of the Pilot X.500 Schema For LDAP March 1997

5.1.13. associatedDomain

The associatedDomain attribute type specifies a DNS domain which is associated with an object in the DIT. For example, the entry in the DIT with a distinguished name "O=University College London, C=GB" would have an associated domain of "UCL.AC.UK". Note that all domains should be represented in [rfc822](#) order.

```
( 0.9.2342.19200300.100.1.37 NAME 'associatedDomain'
  EQUALITY caseIgnoreIA5Match SUBSTR caseIgnoreIA5SubstringsMatch
  SYNTAX 'IA5String' )
```

5.1.14. homePostalAddress

The homePostalAddress attribute type specifies a home postal address for an object. This should be limited to up to 6 lines of 30 characters each.

```
( 0.9.2342.19200300.100.1.39 NAME 'homePostalAddress'
  EQUALITY caseIgnoreListMatch
  SUBSTR caseIgnoreListSubstringsMatch SYNTAX 'PostalAddress' )
```

5.1.15. personalTitle

The personalTitle attribute type specifies a personal title for a person. Examples of personal titles are "Ms", "Dr", "Prof" and "Rev".

```
( 0.9.2342.19200300.100.1.40 NAME 'personalTitle'
```

```
EQUALITY caseIgnoreMatch SUBSTR caseIgnoreSubstringsMatch  
SYNTAX 'DirectoryString' )
```

5.1.16. mobile

The mobile attribute type specifies a mobile telephone number associated with a person. Attribute values should follow the agreed format for international telephone numbers: i.e., "+44 171 123 4567".

```
( 0.9.2342.19200300.100.1.41 NAME 'mobile' EQUALITY telephoneNumberMatch  
  SUBSTR telephoneNumberSubstringsMatch SYNTAX 'TelephoneNumber{32}' )
```

5.1.17. pager

The pager attribute type specifies a pager telephone number for an object. Attribute values should follow the agreed format for international telephone numbers: i.e., "+44 171 123 4567".

```
( 0.9.2342.19200300.100.1.42 NAME 'pager' EQUALITY telephoneNumberMatch  
  SUBSTR telephoneNumberSubstringsMatch SYNTAX 'TelephoneNumber{32}' )
```

5.1.18. co

The co attribute type specifies names of countries in human readable format. An example is "United States of America".

```
( 0.9.2342.19200300.100.1.43 NAME 'co' EQUALITY caseIgnoreMatch  
  SUBSTR caseIgnoreSubstringsMatch SYNTAX 'DirectoryString' )
```

5.1.19. uniqueIdentifier

The uniqueIdentifier attribute type specifies a "unique identifier" for an object represented in the directory. The domain within which the identifier is unique, and the exact semantics of the identifier, are for local definition. For a person, this might be an institution-wide payroll number. For an organisational unit, it might be a department code.

```
( 0.9.2342.19200300.100.1.44 NAME 'uniqueIdentifier'  
  EQUALITY caseIgnoreMatch SUBSTR caseIgnoreSubstringsMatch  
  SYNTAX 'DirectoryString' )
```

5.1.20. organizationalStatus

The organisationalStatus attribute type specifies a category by which a person is often referred to in an organisation. Examples of usage in academia might include undergraduate student, researcher, lecturer, etc.

```
( 0.9.2342.19200300.100.1.45 NAME 'organizationalStatus'  
  EQUALITY caseIgnoreMatch SUBSTR caseIgnoreSubstringsMatch  
  SYNTAX 'DirectoryString{256}' )
```

5.1.21. mailPreferenceOption

An attribute to allow users to indicate a preference for inclusion of their names on mailing lists (electronic or physical). The absence of such an attribute should be interpreted as if the attribute was present with value "no-list-inclusion". This attribute should be interpreted by anyone using the directory to derive mailing lists, and its value respected.

```
( 0.9.2342.19200300.100.1.47 NAME 'mailPreferenceOption'  
  SYNTAX 'INTEGER' SINGLE-VALUE )
```

5.1.22. audio

The audio attribute type allows the storing of sounds in the Directory. The attribute uses a u-law encoded sound file as used by the "play" utility on a Sun 4. This is an interim format.

```
( 0.9.2342.19200300.100.1.55 NAME 'audio' SYNTAX 'Audio{250000}' )
```

5.2. Definitions subsequent to [RFC 1274](#)

5.2.1. labeledURI

This attribute type specifies a URI and optional descriptive information. It is defined in [RFC 2079](#) [3].

```
( 1.3.6.1.4.1.250.1.57 NAME 'labeledURI'  
  DESC 'Uniform Resource Locator'  
  EQUALITY caseExactIA5Match SYNTAX 'IA5String' )
```

5.2.2. carLicense

```
( 2.16.840.1.113730.3.1.1
```

```

NAME 'carLicense'
DESC 'automobile license plate number'
EQUALITY caseIgnoreMatch
SUBSTR caseIgnoreSubstringsMatch
SYNTAX 'DirectoryString' )

```

5.2.3. departmentNumber

```

( 2.16.840.1.113730.3.1.2
  NAME 'departmentNumber'
  DESC 'numerically identifies a department within an organization'
  EQUALITY caseIgnoreMatch
  SUBSTR caseIgnoreSubstringsMatch
  SYNTAX 'DirectoryString' )

```

5.2.4. employeeNumber

```

( 2.16.840.1.113730.3.1.3
  NAME 'employeeNumber'
  DESC 'numerically identifies an employee within an organization'
  EQUALITY caseIgnoreMatch
  SUBSTR caseIgnoreSubstringsMatch
  SYNTAX 'DirectoryString' )

```

5.2.5. employeeType

```

( 2.16.840.1.113730.3.1.4
  NAME 'employeeType'
  DESC 'a person's type of employment'
  EQUALITY caseIgnoreMatch
  SUBSTR caseIgnoreSubstringsMatch
  SYNTAX 'DirectoryString' )

```

5.2.6. jpeg

This attribute type specifies a JFIF-encoded photograph of or associated with an object.

```

( 0.9.2342.19200300.100.1.60 NAME 'jpegPhoto' SYNTAX 'JPEG' )

```

5.2.7. language

This attribute type specifies the (human) languages known by the object represented by an entry, and which should be used when communicating with the object.

```
( 1.3.6.1.4.1.1466.101.120.30 NAME 'language'
  DESC 'ISO 639 codes for language'
  EQUALITY caseIgnoreMatch
  SUBSTR caseIgnoreSubstringsMatch SYNTAX 'DirectoryString' )
```

5.2.8. homeFax

```
( 1.3.6.1.4.1.1466.101.120.31 NAME 'homeFax'
  SYNTAX 'FacsimileTelephoneNumber' )
```

5.2.9. personalMobile

This attribute type specifies the telephone number of a person's mobile phone which is used for personal (non-business) communication.

```
( 1.3.6.1.4.1.1466.101.120.32
  NAME 'personalMobile' EQUALITY telephoneNumberMatch
  SUBSTR telephoneNumberSubstringsMatch SYNTAX 'TelephoneNumber{32}' )
```

5.2.10. personalPager

This attribute type specifies the telephone number of a person's pager which is used for personal (non-business) communication.

```
( 1.3.6.1.4.1.1466.101.120.33
  NAME 'personalPager' EQUALITY telephoneNumberMatch
  SUBSTR telephoneNumberSubstringsMatch SYNTAX 'TelephoneNumber{32}' )
```

5.2.11. middleName

This attribute type specifies the middle name of a person.

```
( 1.3.6.1.4.1.1466.101.120.34 NAME 'middleName' SUP name )
```

5.2.12. thumbnailPhoto

This attribute type specifies a small photograph of the object (a person).

```
( 1.3.6.1.4.1.1466.101.120.35 NAME 'thumbnailPhoto' SYNTAX 'JPEG' )
```

5.2.13. thumbnailLogo

This attribute type specifies a small image of the logo of the organization to which the object belongs.

```
( 1.3.6.1.4.1.1466.101.120.36 NAME 'thumbnailLogo' SYNTAX 'JPEG' )
```

6. Syntaxes

Servers SHOULD recognize the syntax names defined in this section.

6.1. DSAQualitySyntax

Values with this syntax are encoded according to the following BNF:

```
<DsaQualitySyntax> ::= <DSAKeyword> [ '#' <description> ]  
  
<DSAKeyword> ::= 'DEFUNCT' | 'EXPERIMENTAL' | 'BEST-EFFORT' |  
                 'PILOT-SERVICE' | 'FULL-SERVICE'  
  
<description> ::= encoded as a PrintableString
```

6.2. DataQualitySyntax

Values with this syntax are encoded according to the following BNF:

```
<DataQualitySyntax> ::= <compKeyword> '#' <attrQuality> '#'  
                      <listQuality> [ '#' <description> ]  
  
<attrQuality> ::= <levelKeyword> '+' <compKeyword>  
  
<listQuality> ::= <list> '$' <list><listQuality>  
  
<list> ::= <attribute> '+' <attrQuality>  
  
<compKeyword> ::= 'NONE' | 'SAMPLE' | 'SELECTED' |  
                 'SUBSTANTIAL' | 'FULL'  
  
<levelKeyword> ::= 'UNKNOWN' | 'EXTERNAL' | 'SYSTEM-MAINTAINED' |  
                 'USER-SUPPLIED'
```

6.3. MailPreference

Values with MailPreference syntax are encoded according to the following BNF:

```
<mail-preference> ::= "NO-LISTS" | "ANY-LIST" | "PROFESSIONAL-LISTS"
```

6.4. DLSubmitPermission

Values of type DLSubmitPermission are encoded as strings, according to the following BNF:

```
<dlsubmit-perm> ::= <dlgroup_label> ':' <dlgroup-value>  
                  | <dl-label> ':' <dl-value>  
  
<dlgroup-label> ::= 'group_member'  
  
<dlgroup-value> ::= <name>
```

<name> ::= an encoded Distinguished Name

Wahl

Page 8

INTERNET-DRAFT A Summary of the Pilot X.500 Schema For LDAP March 1997

<dl-label> ::= 'individual' | 'dl_member' | 'pattern'

<dl-value> ::= <orname>

<orname> ::= <address> '#' <dn>
 | <address>

<address> ::= <add-label> ':' <oraddress>

<dn> ::= <dn-label> ':' <name>

<add-label> = 'X400'

<dn-label> = 'X500'

where <oraddress> is as defined in [11].

7. Object Classes

Servers SHOULD recognize these object class names.

7.1. Definitions based on [RFC 1274](#)

```
( 0.9.2342.19200300.100.4.5 NAME 'account' SUP top STRUCTURAL
  MUST uid MAY ( host $ ou $ o $ l $ seeAlso $ description ) )

( 0.9.2342.19200300.100.4.7 NAME 'room' SUP top STRUCTURAL MUST cn
  MAY ( telephoneNumber $ seeAlso $ description $ roomNumber ) )

( 0.9.2342.19200300.100.4.17 NAME 'domainRelatedObject' SUP top
  STRUCTURAL MUST associatedDomain )

( 0.9.2342.19200300.100.4.18 NAME 'friendlyCountry' SUP country
  STRUCTURAL MUST co )

( 0.9.2342.19200300.100.4.19 NAME 'simpleSecurityObject' SUP top
  STRUCTURAL MUST userPassword )
```

7.2. Other Definitions

The labeledURIObject class is a subclass of top and may contain the labeledURI attribute. The intent is that this object class can be added to existing directory objects to allow for inclusion of URI values. This approach does not preclude including the labeledURI attribute type directly in other object classes as appropriate.

(1.3.6.1.4.1.250.3.15 NAME 'labeledURIObject' SUP top MAY labeledURI)

8. Other Schema

The following schema from [RFC 1274](#) MAY be recognized by servers.

8.1. Other Attribute Types

```
( 0.9.2342.19200300.100.1.2 NAME 'textEncoded0Raddress'
  EQUALITY caseIgnoreMatch SUBSTR caseIgnoreSubstringsMatch
  SYNTAX 'DirectoryString{256}' )

( 0.9.2342.19200300.100.1.4 NAME 'info' EQUALITY caseIgnoreMatch
  SUBSTR caseIgnoreSubstringsMatch SYNTAX 'DirectoryString{2048}' )

( 0.9.2342.19200300.100.1.7 NAME 'photo' SYNTAX 'Fax{250000}' )

( 0.9.2342.19200300.100.1.11 NAME 'documentIdentifier'
  EQUALITY caseIgnoreMatch SUBSTR caseIgnoreSubstringsMatch
  SYNTAX 'DirectoryString{256}' )

( 0.9.2342.19200300.100.1.12 NAME 'documentTitle' EQUALITY caseIgnoreMatch
  SUBSTR caseIgnoreSubstringsMatch SYNTAX 'DirectoryString{256}' )

( 0.9.2342.19200300.100.1.13 NAME 'documentVersion'
  EQUALITY caseIgnoreMatch SUBSTR caseIgnoreSubstringsMatch
  SYNTAX 'DirectoryString{256}' )

( 0.9.2342.19200300.100.1.14 NAME 'documentAuthor'
  EQUALITY distinguishedNameMatch SYNTAX 'DN' )

( 0.9.2342.19200300.100.1.15 NAME 'documentLocation'
  EQUALITY caseIgnoreMatch SUBSTR caseIgnoreSubstringsMatch
  SYNTAX 'DirectoryString{256}' )

( 0.9.2342.19200300.100.1.23 NAME 'lastModifiedTime' OBSOLETE
  SYNTAX 'UTCTime' )

( 0.9.2342.19200300.100.1.24 NAME 'lastModifiedBy' OBSOLETE
  EQUALITY distinguishedNameMatch SYNTAX 'DN' )
```

```
( 0.9.2342.19200300.100.1.38 NAME 'associatedName'
  EQUALITY distinguishedNameMatch SYNTAX 'DN' )

( 0.9.2342.19200300.100.1.46 NAME 'janetMailbox'
  EQUALITY caseIgnoreIA5Match SUBSTR caseIgnoreIA5SubstringsMatch
  SYNTAX 'IA5String{256}' )

( 0.9.2342.19200300.100.1.48 NAME 'buildingName'
  EQUALITY caseIgnoreMatch SUBSTR caseIgnoreSubstringsMatch
  SYNTAX 'DirectoryString{256}' )

( 0.9.2342.19200300.100.1.49 NAME 'dsaQuality'
  SYNTAX 'DSAQualitySyntax' SINGLE-VALUE )

( 0.9.2342.19200300.100.1.50 NAME 'singleLevelQuality'
  SYNTAX 'DataQualitySyntax' SINGLE-VALUE )
```

Wahl

Page 10

INTERNET-DRAFT A Summary of the Pilot X.500 Schema For LDAP March 1997

```
( 0.9.2342.19200300.100.1.51 NAME 'subtreeMinimumQuality'
  SYNTAX 'DataQualitySyntax' SINGLE-VALUE )

( 0.9.2342.19200300.100.1.52 NAME 'subtreeMaximumQuality'
  SYNTAX 'DataQualitySyntax' SINGLE-VALUE )

( 0.9.2342.19200300.100.1.53 NAME 'personalSignature'
  SYNTAX 'Fax{50000}' )

( 0.9.2342.19200300.100.1.54 NAME 'dITRedirect'
  EQUALITY distinguishedNameMatch SYNTAX 'DN' )

( 0.9.2342.19200300.100.1.56 NAME 'documentPublisher'
  EQUALITY caseIgnoreMatch SUBSTR caseIgnoreSubstringsMatch
  SYNTAX 'DirectoryString' )
```

8.2. Other Classes

```
( 0.9.2342.19200300.100.4.3 NAME 'pilotObject' SUP top STRUCTURAL
  MAY ( jpegPhoto $ audio $ dITRedirect $ lastModifiedBy $
    lastModifiedTime $ uniqueIdentifier $ manager $ photo $ info ) )

( 0.9.2342.19200300.100.4.4 NAME 'newPilotPerson' SUP person
  STRUCTURAL MAY ( personalSignature $ mailPreferenceOption $
    organizationalStatus $ pager $ mobile $
    otherMailbox $ janetMailbox $ businessCategory $
    preferredDeliveryMethod $ personalTitle $ secretary $
    homePostalAddress $ homePhone $ userClass $ roomNumber $
    favouriteDrink $ mail $ textEncodedORaddress $ uid ) )
```

```
( 0.9.2342.19200300.100.4.6 NAME 'document' SUP ( top $ pilotObject )
  STRUCTURAL MUST documentIdentifier
  MAY ( documentPublisher $ documentStore $ documentAuthorSurName $
    documentAuthorCommonName $ abstract $ subject $ keywords $
    updatedByDocument $ updatesDocument $ obsoletedByDocument $
    obsoletesDocument $ documentLocation $ documentAuthor $
    documentVersion $ documentTitle $ ou $ o $ l $ seeAlso $ description $
    cn ) )

( 0.9.2342.19200300.100.4.9 NAME 'documentSeries' SUP top STRUCTURAL
  MUST cn MAY ( ou $ o $ l $ telephoneNumber $ seeAlso $ description ) )

( 0.9.2342.19200300.100.4.13 NAME 'domain' SUP top STRUCTURAL
  MUST dc
  MAY ( userPassword $ searchGuide $ seeAlso $ businessCategory $
    x121Address $ registeredAddress $ destinationIndicator $
    preferredDeliveryMethod $ telexNumber $ teletexTerminalIdentifier $
    telephoneNumber $ internationaliSDNNumber $ facsimileTelephoneNumber $
    street $ postOfficeBox $ postalCode $ postalAddress $
    physicalDeliveryOfficeName $ st $ l $ description $ o $
    associatedName ) )
```

Wahl

Page 11

INTERNET-DRAFT A Summary of the Pilot X.500 Schema For LDAP March 1997

```
( 0.9.2342.19200300.100.4.14 NAME 'RFC822localPart' SUP domain
  STRUCTURAL
  MAY ( x121Address $ registeredAddress $ destinationIndicator $
    preferredDeliveryMethod $ telexNumber $ teletexTerminalIdentifier $
    telephoneNumber $ internationaliSDNNumber $ facsimileTelephoneNumber $
    streetAddress $ postOfficeBox $ postalCode $ postalAddress $
    physicalDeliveryOfficeName $ telephoneNumber $ seeAlso $ description $
    sn $ cn ) )

( 0.9.2342.19200300.100.4.15 NAME 'DNSDomain' SUP domain STRUCTURAL
  MAY DNSRecord )

( 0.9.2342.19200300.100.4.20 NAME 'pilotOrganization'
  SUP ( organization $ organizationalUnit ) STRUCTURAL
  MAY buildingName )

( 0.9.2342.19200300.100.4.21 NAME 'pilotDSA' SUP dSA STRUCTURAL
  MUST dSAQuality )

( 0.9.2342.19200300.100.4.22 NAME 'oldQualityLabelledData' SUP top
  STRUCTURAL MUST dSAQuality
  MAY ( subtreeMaximumQuality $ subtreeMinimumQuality ) )
```

```
( 0.9.2342.19200300.100.4.23 NAME 'qualityLabelledData' SUP top
  STRUCTURAL MUST singleLevelQuality
  MAY ( subtreeMaximumQuality $ subtreeMinimumQuality ) )
```

9. Security Considerations

Security issues are not discussed in this memo.

10. Acknowledgements

The definitions on which this document have been developed by committees for telecommunications, international standards, the Internet community, and the Network Applications Consortium. In particular the contributions from [RFC 1274](#), by Paul Barker and Steve Kille, are gratefully acknowledged.

11. Bibliography

- [1] M. Wahl, A. Coulbeck, T. Howes, S. Kille, W. Yeong, C. Robbins, "Lightweight X.500 Directory Access Protocol Attribute Syntax Definitions", INTERNET-DRAFT [<draft-ietf-asid-ldapv3-attributes-04.txt>](#), March 1997.
- [2] P. Barker, S. Kille, "The COSINE and Internet X.500 Schema", [RFC 1274](#), November 1991.
- [3] M. Smith, "Definition of X.500 Attribute Types and an Object Class to hold Uniform Resource Identifiers (URIs)", January 1997.

Wahl

Page 12

INTERNET-DRAFT A Summary of the Pilot X.500 Schema For LDAP March 1997

12. Authors Address

Mark Wahl
Critical Angle Inc.
4815 West Braker Lane #502-385
Austin, TX 78759
USA

EEmail: M.Wahl@critical-angle.com

Expires September 1997