Internet Engineering Task Force
Internet-Draft
Intended status: Standards Track
Expires: 6 July 2026

C. E. Cod竪re, Ed.
Optima SC Inc.
2 January 2026

Lightweight Directory Access Protocol (LDAP): Additional Syntaxes
draft-codere-ldapsyntax-07

Abstract

This document registers additional syntax definitions for use in Lightweight Directory Access Protocol (LDAP) directory and Directoy services series X.500. This includes widely used datatypes and syntaxes.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

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."

This Internet-Draft will expire on 6 July 2026.

Copyright Notice

Copyright (c) 2026 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust’s Legal Provisions Relating to IETF Documents (https://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

1. Introduction

The Lightweight Directory Access Protocol (LDAP) directory defines several data types which specify the syntax definitions of attributes. These are identified by ASN.1 OBJECT IDENTIFIER types. Furthermore, these syntax definitions can be used to uniquely identify data types as character representations in other applications. Some widely used syntax specifications are missing from the initial LDAP specification. This document provides additional syntax definitions that have been registered and may be used by application providers.

1.1. Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

Syntax definitions are written according to the regular expressions defined in [RFC9485].

2. Syntaxes

The following additional syntaxes and their associated descriptions and OBJECT IDENTIFIER types are defined.

2.1. ASN.1 Syntax Definitions

The following additional syntaxes are defined and are based on [ASN.1].

2.1.1. Date

The Date type represents a date in the Gregorian calender. It is defined as a useful TIME type in [ASN.1] and conforms to the extended format syntax of a calendar date as defined in [ISO.8601.2004].

A Date value SHALL be written using the following syntax: YYYY-MM-DD where YYYY represents a year between 1582 and 9999, MM the month value from 01 to 12 and DD a day in the month from 01 to 31.

Examples:

* 9999-02-25

* 1583-01-31

The LDAP definition for the Date syntax is:

* (1.3.6.1.4.1.61799.5.40.31 DESC 'Date')

This syntax corresponds to the DATE ASN.1 type from [ASN.1].

2.1.2. Date-Time

The Date-time type represents a date and local time using a 24 hour clock. It is defined as a useful TIME type in [ASN.1] and conforms to the extended format syntax of a date and time without any timezone specifier as defined in [ISO.8601.2004].

A Date-Time value SHALL be written using the following syntax: YYYY-MM-DDThh:mm:ss where YYYY represents a year between 1582 and 9999, MM the month value from 01 to 12, DD a day in the month from 01 to 31, hh the hour from 00 to 24, mm the minute from 00 to 59, and ss the seconds with allowed values of 00 to 60 where 60 represents a leap second.

Examples:

* 1583-01-01T00:59:59

* 1975-01-19T23:45:34

The LDAP definition for the Date-Time syntax is:

* (1.3.6.1.4.1.61799.5.40.33 DESC 'Date-Time')

This syntax corresponds to the DATE-TIME ASN.1 type from [ASN.1].

2.1.3. Duration

The Duration type represents an elapsed time with a resolution of up to a fractions of seconds. It is defined as a useful TIME type in [ASN.1] and conforms to the extended format syntax of a time interval by duration as defined in [ISO.8601.2004].

A duration syntax value SHALL conform to the following regular expression:

P([0-9]+Y)?([0-9]+M)?([0-9]+D)?(T([0-9]+H)?([0-9]+M)?([0-9]+(\.[0-9]+)?S)?)?

Examples:

* P29M0D -- 29 months

* P29MT0S -- 29 months

* PT3445.5S -- 3445.55 seconds

The LDAP definition for the Duration syntax is:

* (1.3.6.1.4.1.61799.5.40.34 DESC 'Duration')

This syntax corresponds to a very strict subset of DURATION ASN.1 type from [ASN.1], in that the order of parameters need to be respected.

2.1.4. Real

The Real type represents the computational approximations to the mathematical "real number". The format for the Real is as defined in Section 21 of [ASN.1].

A Real syntax value SHALL conform to the following regular expression:

```
([-]?[0-9]+\.?[0-9]+([E][-]?[0-9]+)?)|PLUS-INFINITY|MINUS-INFINITY|NOT-A-NUMBER
```

Examples:

* 3.14159

* MINUS-INFINITY

* -5.3E4 -- Equal to -53000

The LDAP definition for the Real syntax is:

* (1.3.6.1.4.1.61799.5.40.9 DESC 'Real')

This syntax corresponds to a subset of the REAL ASN.1 type from [ASN.1] where the sequence syntax is not allowed, the values are limited to base ten, where a digit before the decimal point is required, where only the character 'E' is allowed to specify the exponent and that the preceding optional "+" sign is prohibited in the exponent.

2.1.5. Time Of Day

The Time Of Day type represents a local time using a 24 hour clock. It is defined as a useful TIME type in [ASN.1] and conforms to the extended format syntax of a local time as defined in [ISO.8601.2004].

A Time Of Day value SHALL be written using the following syntax: hh:mm:ss where hh represents the hour from 00 to 24, mm represents the minute from 00 to 59, and ss represents the seconds with allowed values of 00 to 60 where 60 represents a leap second.

Examples for Time Of Day:

* 00:59:59

* 01:45:54

The LDAP definition for the Time Of Day syntax is:

* (1.3.6.1.4.1.61799.5.40.32 DESC 'Time Of Day')

This syntax corresponds to the TIME-OF-DAY ASN.1 type from [ASN.1].

2.1.6. Visible String

The Visible String type represents a character repertoire that contains the printable ASCII character set (in the range 0020-007E hexadecimal). It is defined in [ASN.1].

This syntax value SHALL conform to the following regular expression:

[0-9A-Za-z !"#$%&'()*+,-./:;<=>?@[\]^_'{|}~]

Examples:

* hello world

* (x+y)=z

The LDAP definition for the Visible String syntax is:

* (1.3.6.1.4.1.61799.5.40.26 DESC 'Visible String')

This syntax corresponds to the VisibleString ASN.1 type from [ASN.1].

2.2. Constrained ASN.1 Syntax Definitions

The following additional syntaxes are defined as constraints of basic ASN.1 types that may be used to be more precise in encoding and input validation.

2.2.1. Short String

The Short String type represents a string that is limited to 31 characters when encoded.

The length was chosen so that when using DER encoding, using the worst-case scenario of 4 octets per character in UTF-8, the string can be encoded using one length octet. It is also sufficient for labels and short titles.

Examples:

* Hello world

* Short

The LDAP definition for the Short String type syntax is:

* (1.3.6.1.4.1.61799.5.40.12.1 DESC 'Short String')

This syntax corresponds to the following ASN.1 type from [ITU.X520.2019]:

Shortstring ::= DirectoryString{31}

2.2.2. Long String

The Long String type represents a string that is limited to 250 characters when encoded.

The length was chosen so that when using CER encoding, using the worst-case scenario of 4 octets per character, the string can still be encoded using a primitive construct. This length seems sufficient for descriptive text.

Examples:

* This is a bigger sentence

* Ceci est une phrase qui est plus longue que la pr迆c竪dente

The LDAP definition for the Long String type syntax is:

* (1.3.6.1.4.1.61799.5.40.12.2 DESC 'Long String')

This syntax corresponds to the following ASN.1 type from [ITU.X520.2019]:

Longstring ::= DirectoryString{250}

2.2.3. Text

The text type represents a string that is limited to 16383 characters when encoded.

The length was defined based on historical system constraints of 65535 octets using the worst case scenario of 4 octets per character.

Examples:

* Hello world

* Ceci est une phrase qui est encore plus longue que la pr迆c竪dente

The LDAP definition for the Text type syntax is:

* (1.3.6.1.4.1.61799.5.40.12.3 DESC 'Text')

This syntax corresponds to the following ASN.1 type from [ITU.X520.2019]:

Text ::= DirectoryString{16383}

2.2.4. Float32

The Float32 type represents a real number which fits in the range of a [IEEE_754_2019] single precision floating point value.

The Float32 syntax follows the syntax of the real type (See Section 2.1.4) with a constrained range.

Examples:

* 3.14159

* MINUS-INFINITY

* -5.3E4 -- Equal to -53000

The LDAP definition for the Float32 type syntax is:

* (1.3.6.1.4.1.61799.5.40.9.4 DESC 'Float32')

This syntax corresponds to the following ASN.1 type from [ASN.1]:

```
Float32 ::= REAL (WITH COMPONENTS {
  mantissa (-16777215..16777215),
  base (2),
  exponent (-149..104) })
```

2.2.5. Float64

The Float64 type represents a real number which fits in the range of a [IEEE_754_2019] double precision floating point value.

The Float64 syntax follows the syntax of the real type (See Section 2.1.4) with a constrained range.

Examples:

* 3.1415926535897932

* NOT-A-NUMBER

* -5.3E4 -- Equal to -53000

The LDAP definition for the Float64 type syntax is:

* (1.3.6.1.4.1.61799.5.40.9.8 DESC 'Float64')

This syntax corresponds to the following ASN.1 type from [ASN.1]:

```
Float64 ::= REAL (WITH COMPONENTS {
 mantissa (-9007199254740991..9007199254740991),
 base (2),
 exponent (-1074..971) })
```

2.2.6. UInt8

The UInt8 type represents an unsigned integer value within the range 0 to 255 inclusive.

Examples:

* 0

* 34

The LDAP definition for the UInt8 type syntax is:

* (1.3.6.1.4.1.61799.5.40.2.21 DESC 'UInt8')

This syntax corresponds to the following ASN.1 type from [ASN.1]:

UInt8 ::= INTEGER(0..255)

2.2.7. UInt16

The UInt16 type represents an unsigned integer value within the range 0 to 65535 inclusive.

Examples:

* 0

* 64991

The LDAP definition for the UInt16 type syntax is:

* (1.3.6.1.4.1.61799.5.40.2.22 DESC 'UInt16')

This syntax corresponds to the following ASN.1 type from [ASN.1]:

UInt16 ::= INTEGER(0..65535)

2.2.8. UInt32

The UInt32 type represents an unsigned integer value within the range 0 to 4294967295 inclusive.

Examples:

* 0

* 40000000

The LDAP definition for the UInt32 type syntax is:

* (1.3.6.1.4.1.61799.5.40.2.24 DESC 'UInt32')

This syntax corresponds to the following ASN.1 type from [ASN.1]:

UInt32 ::= INTEGER(0..4294967295)

2.2.9. UInt64

The UInt64 type represents an unsigned integer value within the range 0 to 18446744073709551615 inclusive.

Examples:

* 0

* 844674407370955

The LDAP definition for the UInt64 type syntax is:

* (1.3.6.1.4.1.61799.5.40.2.28 DESC 'UInt64')

This syntax corresponds to the following ASN.1 type from [ASN.1]:

```
UInt64 ::= INTEGER(0..18446744073709551615)
```

2.2.10. Int8

The Int8 type represents a signed integer value within the range -128 to 127 inclusive.

Examples:

* 0

* -123

The LDAP definition for the Int8 type syntax is:

* (1.3.6.1.4.1.61799.5.40.2.1 DESC 'Int8')

This syntax corresponds to the following ASN.1 type from [ASN.1]:

```
Int8 ::= INTEGER(-128..127)
```

2.2.11. Int16

The Int16 type represents a signed integer value within the range -32768 to 32767 inclusive.

Examples:

* 15667

* -32000

The LDAP definition for the Int16 type syntax is:

* (1.3.6.1.4.1.61799.5.40.2.2 DESC 'Int16')

This syntax corresponds to the following ASN.1 type from [ASN.1]:

```
Int16 ::= INTEGER(-32768 .. 32767)
```

2.2.12. Int32

The Int32 type represents a signed integer value within the range -2147483648 to 2147483647 inclusive.

Examples:

* 15667

* -3200000

The LDAP definition for the Int32 type syntax is:

* (1.3.6.1.4.1.61799.5.40.2.4 DESC 'Int32')

This syntax corresponds to the following ASN.1 type from [ASN.1]:

```
Int32 ::= INTEGER(-2147483648..2147483647)
```

2.2.13. Int64

The Int64 type represents a signed integer value within the range -9223372036854775808 to 9223372036854775807 inclusive.

Examples:

* -2337203685477580

* 3372036854775807

The LDAP definition for the Int64 type syntax is:

* (1.3.6.1.4.1.61799.5.40.2.8 DESC 'Int64')

This syntax corresponds to the following ASN.1 type from [ASN.1]:

```
Int64 ::= INTEGER(-9223372036854775808..9223372036854775807)
```

2.2.14. Percentage

The Percentage type represents a percentage value, that is an unsigned integer in the range 0 to 100 inclusive.

Examples:

* 0

* 99

The LDAP definition for the Percentage type syntax is:

* (1.3.6.1.4.1.61799.5.40.2.20 DESC 'Percentage')

This syntax corresponds to the following ASN.1 type from [ASN.1]:

```
Percentage ::= INTEGER(0..100)
```

2.3. Other Syntax Definitions

The following additional syntaxes are defined and are based on IETF RFC's, or other international standards.

2.3.1. DCMIType

DCMIType is a controlled vocabulary to describe the type of a resource. It is specified in [DCMIType].

Examples:

* Text

* Moving Image

The LDAP definition for the DCMIType syntax is:

* (1.3.6.1.4.1.61799.5.40.19.2 DESC 'DCMIType')

This syntax corresponds to the following ASN.1 type from [ASN.1]:

```
DCMIType ::= PrintableString ("Collection" | "Dataset" |
             "Event" | "Image" | "Interactive Resource" |
             "Moving Image" |
             "Physical Object" |
             "Service" | "Software" |
             "Sound" |
             "Still Image" |
             "Text")
```

2.3.2. Language

A language provides a representation of a spoken or written language as well as an optional locale specifier. The exact syntax allowed is defined in Section 2 of [RFC5646].

A Language syntax value SHALL conform to the following regular expression:

[a-zA-Z]{1,8}(-[a-zA-Z0-9]{1,8})*

Examples:

* en

* fr-CA

The LDAP definition for the Language syntax is:

* (1.3.6.1.4.1.61799.5.40.19.1 DESC 'Language')

This syntax corresponds to the following ASN.1 type from [ASN.1]:

```
Language ::= PrintableString (PATTERN "[a-zA-Z]#(1,8)(-[a-zA-Z0-9]#(1,8))*")
  -- ISO 639 code minimally
```

2.3.3. Media type

The Media Type syntax identifies values that represent an IANA registered Media type [IANAREG]. The format for the MIME Media type is defined in Section 5.1 of [RFC6838].

This syntax value SHALL conform to the following regular expression:

[A-Za-z0-9]([A-Za-z0-9!#$&^_.+-]){0,126}/[A-Za-z0-9]([A-Za-z0-9!#$&^_.+-]){0,126}

Examples:

* text/xhtml

* application/alto-costmap+json

The LDAP definition for the MIME Media type syntax is:

* (1.3.6.1.4.1.61799.5.40.26.5 DESC 'Media Type')

This syntax corresponds to the following ASN.1 type from [ITU.X520.2019]:

```
MediaType ::= DirectoryString{255}
```

2.3.4. OpenDate

An OpenDate represents either part of a Date or a Date and Time in extended format as specified in ISO 8601. The exact syntax allowed is defined by W3C Date and Time formats [W3C.NOTE-datetime-19980827] with a 3 digit fraction. The time component, when present, always contains timezone information.

Examples:

* 2034
* 1975-01
* 1975-01-19
* 1975-01-19T19:20+01:00
* 1975-01-19T19:20:30+01:00
* 1975-01-19T19:20:30.451+01:00
* 1975-01-19T18:20Z
* 1975-01-19T18:20:30Z
* 1975-01-19T18:20:30.451Z

The LDAP definition for the OpenDate syntax is:

* (1.3.6.1.4.1.61799.5.40.14.1 DESC 'OpenDate')

This syntax corresponds to a subset of the TIME ASN.1 type from [ASN.1] with the specified configuration:

```
OpenDate ::=  TIME((SETTINGS "Basic=Date Date=Y Year=Basic")|
 (SETTINGS "Basic=Date Date=YM Year=Basic")|
 (SETTINGS "Basic=Date Date=YMD Year=Basic")|
 (SETTINGS "Basic=Date-Time Date=YMD Year=Basic Time=HM Local-or-UTC=LD")|
 (SETTINGS "Basic=Date-Time Date=YMD Year=Basic Time=HMS Local-or-UTC=LD")|
 (SETTINGS "Basic=Date-Time Date=YMD Year=Basic Time=HMSF3 Local-or-UTC=LD")|
 (SETTINGS "Basic=Date-Time Date=YMD Year=Basic Time=HM Local-or-UTC=Z")|
 (SETTINGS "Basic=Date-Time Date=YMD Year=Basic Time=HMS Local-or-UTC=Z")|
 (SETTINGS "Basic=Date-Time Date=YMD Year=Basic Time=HMSF3 Local-or-UTC=Z"))
```

2.3.5. URI

The URI syntax type identifies values that are referenced by a Uniform Resource Identifier (URI). The format and encoding for the URI is as defined in [RFC3986]. Even if relative URI's are allowed, it is RECOMMENDED they not be used unless the context of use is known.

Examples:

* http://www.example.com/my/picture.jpg

* ldap://ldap.example.com/cn=babs%20jensen

The LDAP definition for the URI syntax is:

* (1.3.6.1.4.1.61799.5.40.26.4 DESC 'URI')

This syntax corresponds to the following ASN.1 type from [ITU.X520.2019]:

```
URI ::= DirectoryString{ub-uri-length}
```

The value of ub-uri-length (an integer) is implementation defined but MUST be at least 2000 characters.

2.3.6. NCName

The NCName syntax type should be used to identify values that represent identifiers and local attribute names. A name is a subset of the NCName definition in [W3C.xmlschema11-2].

This syntax value SHALL conform to the following regular expression:

```
[\p{L}\p{Nl}_][\p{L}\p{Nl}\p{Nd}.-_]*
```

Examples:

* MyID

* attribte.0.subdivision

The LDAP definition for the NCName type syntax is:

* (1.3.6.1.4.1.61799.5.40.26.6 DESC 'NCName')

This syntax corresponds to the following ASN.1 type from [ITU.X520.2019]:

```
NCName ::= UnboundedDirectoryString
```

2.3.7. Normalized String

The Normalized String syntax type represents white space normalized strings. A Normalized String is a string that does not contain any control characters including the carriage return (%xD), line feed (%xA) or tab (%x9) character. This is similar to the NormalizedString datatype in [W3C.xmlschema11-2] based on the Char type defined in [W3C.xml].

Examples:

* Paragraph start with some start spaces.

* This is some other text with spaces.

The LDAP definition for the Normalized String type syntax is:

* (1.3.6.1.4.1.61799.5.40.12.4 DESC 'Normalized String')

This syntax corresponds to the following ASN.1 type from [ASN.1] :

```
NormalizedString ::= CHOICE {
  printableString  PrintableString,
  bmpString        BMPString
    (FROM(
      {0, 0, 0, 32} .. {0, 0, 215, 255} |
      {0, 0, 224, 0} .. {0, 0, 255, 253})),
  universalString  UniversalString
    (FROM(
      {0, 0, 0, 32} .. {0, 0, 215, 255} |
      {0, 0, 224, 0} .. {0, 0, 255, 253} |
      {0, 1, 0, 0} .. {0, 16, 255, 253})),
  uTF8String       UTF8String
    (FROM(
      {0, 0, 0, 32} .. {0, 0, 215, 255} |
      {0, 0, 224, 0} .. {0, 0, 255, 253} |
      {0, 1, 0, 0} .. {0, 16, 255, 253}))
   }
```

2.3.8. Qualified Name

The Qualified Name syntax type identifies values that represent identifiers and attribute names using namespaces. This is a subset of the QName definition in [W3C.xmlschema11-2].

This syntax value SHALL conform to the following regular expression:

[\p{L}\p{Nl}_][\p{L}\p{Nl}\p{Nd}.-_]*(:)?[\p{L}\p{Nl}\p{Nd}.-_]+

Examples:

* MyID

* attribte.0:subdivision

The LDAP definition for the QualifiedName type syntax is:

* (1.3.6.1.4.1.61799.5.40.26.6 DESC 'QualifiedName')

This syntax corresponds to the following ASN.1 type from [ITU.X520.2019]:

QualifiedName ::= UnboundedDirectoryString

2.3.9. Time Of Day with Timezone

The Time Of Day with Timezone syntax type represents a time with explicit timezone information using a 24 hour clock. It is defined as a TIME type in [ASN.1] and conforms to the extended format syntax of a time either represented as a Local time and the difference from UTC or UTC of day as defined in [ISO.8601.2004].

A Time Of Day with Timezone value SHALL be written using the following syntax: hh:mm:ss where hh represents the hour from 00 to 24, mm represents the minute from 00 to 59, and ss represents the seconds with allowed values of 00 to 60 where 60 represents a leap second followed by a timezone indicator.

The timezone indicator is in the form +hh:mm where hh represents the number of hours and mm the number of minutes if the local time is ahead of or equal to UTC time. The timezone indicator is -hh:mm if the local time is behind UTC time. If the time represents an UTC time, the time shall be followed without space, by the timezone UTC designator [Z]. This standard supports time differences in the range 15 hours to +15 hours to align with [ASN.1].

Examples:

* 00:59:59Z

* 01:45:54-01:00

The LDAP definition for the Time Of Day with Timezone syntax is:

* (1.3.6.1.4.1.61799.5.40.35 DESC 'Time Of Day with Timzone')

This syntax corresponds to a subset of the TIME ASN.1 type from [ASN.1] with the specified configuration:

```
Time-with-timezone ::=
  TIME((SETTINGS "Basic=Time Time=HMS Local-or-UTC=LD")|
       (SETTINGS "Basic=Time Time=HMS Local-or-UTC=Z"))
```

2.3.10. Token

The Token syntax type represents white space normalized strings. A Normalized String is a string that does not contain any control characters including the carriage return (%xD), line feed (%xA) or tab (%x9) character, that have no leading or trailing spaces (%x20) and that have no internal sequences of two or more spaces. This is similar to the Token datatype in [W3C.xmlschema11-2] based on the Char type defined in [W3C.xml].

Examples:

* This is a token with spaces.

* _Identifier_

The LDAP definition for the Token type syntax is:

* (1.3.6.1.4.1.61799.5.40.12.5 DESC 'Token')

This syntax corresponds to the following ASN.1 type from [ASN.1] :

```
Token ::= CHOICE {
  printableString  PrintableString(PATTERN "[^ ]+( [^ ]+)*"),
  bmpString        BMPString
    (FROM(
      {0, 0, 0, 32} .. {0, 0, 215, 255} |
      {0, 0, 224, 0} .. {0, 0, 255, 253})) (PATTERN "[^ ]+( [^ ]+)*"),
  universalString  UniversalString
    (FROM(
      {0, 0, 0, 32} .. {0, 0, 215, 255} |
      {0, 0, 224, 0} .. {0, 0, 255, 253} |
      {0, 1, 0, 0} .. {0, 16, 255, 253})) (PATTERN "[^ ]+( [^ ]+)*"),
  uTF8String       UTF8String
    (FROM(
      {0, 0, 0, 32} .. {0, 0, 215, 255} |
      {0, 0, 224, 0} .. {0, 0, 255, 253} |
      {0, 1, 0, 0} .. {0, 16, 255, 253})) (PATTERN "[^ ]+( [^ ]+)*")
  }
```

3. IANA Considerations

IANA is requested to assign the LDAP values [RFC4520] specified in this document to https://www.iana.org/assignments/ldap-parameters/ldap-parameters.xhtml#ldap-parameters-8.

3.1. Syntax registration

Subject: Request for LDAP Syntax Registration

Object Identifier: See table below

Description: List of additional useful LDAP syntaxes

Person & email address to contact for further information:
carl.codere@optimasc.com

Specification/Reference: [RFC-to-be]

Author/Change Controller/Owner: IESG

Comments: See table for list of additional syntaxes

Object Identifier	Syntax
1.3.6.1.4.1.61799.5.40.31	Date
1.3.6.1.4.1.61799.5.40.33	Date-Time
1.3.6.1.4.1.61799.5.40.19.2	DCMIType
1.3.6.1.4.1.61799.5.40.34	Duration
1.3.6.1.4.1.61799.5.40.9.4	Float32
1.3.6.1.4.1.61799.5.40.9.8	Float64
1.3.6.1.4.1.61799.5.40.2.1	Int8
1.3.6.1.4.1.61799.5.40.2.2	Int16
1.3.6.1.4.1.61799.5.40.2.4	Int32
1.3.6.1.4.1.61799.5.40.2.8	Int64
1.3.6.1.4.1.61799.5.40.19.1	Language

1.3.6.1.4.1.61799.5.40.26.6	NCName
1.3.6.1.4.1.61799.5.40.12.4	Normalized String
1.3.6.1.4.1.61799.5.40.12.1	Short String
1.3.6.1.4.1.61799.5.40.12.2	Long String
1.3.6.1.4.1.61799.5.40.26.5	Media Type
1.3.6.1.4.1.61799.5.40.14.1	OpenDate
1.3.6.1.4.1.61799.5.40.2.20	Percentage
1.3.6.1.4.1.61799.5.40.26.7	QualifiedName
1.3.6.1.4.1.61799.5.40.9	Real
1.3.6.1.4.1.61799.5.40.12.3	Text
1.3.6.1.4.1.61799.5.40.12.5	Token
1.3.6.1.4.1.61799.5.40.32	Time Of Day
1.3.6.1.4.1.61799.5.40.35	Time Of Day with Timezone
1.3.6.1.4.1.61799.5.40.2.21	UInt8
1.3.6.1.4.1.61799.5.40.2.22	UInt16
1.3.6.1.4.1.61799.5.40.2.24	UInt32
1.3.6.1.4.1.61799.5.40.2.28	UInt64
1.3.6.1.4.1.61799.5.40.26.4	URI
1.3.6.1.4.1.61799.5.40.26	Visible String

Table 1: List of additional LDAP syntaxes

4. Security Considerations

When interpreting security-sensitive fields (in particular, fields used to grant or deny access), implementations MUST ensure that any matching rule comparisons are done on the underlying abstract value, regardless of the particular encoding used.

5. References

5.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC9485] Bormann, C. and T. Bray, "I-Regexp: An Interoperable Regular Expression Format", RFC 9485, DOI 10.17487/RFC9485, October 2023, <https://www.rfc-editor.org/info/rfc9485>.

[RFC6838] Freed, N., Klensin, J., and T. Hansen, "Media Type Specifications and Registration Procedures", BCP 13, RFC 6838, DOI 10.17487/RFC6838, January 2013, <https://www.rfc-editor.org/info/rfc6838>.

[RFC5646] Phillips, A., Ed. and M. Davis, Ed., "Tags for Identifying Languages", BCP 47, RFC 5646, DOI 10.17487/RFC5646, September 2009, <https://www.rfc-editor.org/info/rfc5646>.

[RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, DOI 10.17487/RFC3986, January 2005, <https://www.rfc-editor.org/info/rfc3986>.

[IEEE_754_2019] IEEE, "IEEE Standard for Floating-Point Arithmetic", IEEE IEEE 754-2019, DOI 10.1109/IEEESTD.2019.8766229, 18 July 2019, <https://ieeexplore.ieee.org/document/8766229>.

[RFC4520] Zeilenga, K., "Internet Assigned Numbers Authority (IANA) Considerations for the Lightweight Directory Access Protocol (LDAP)", BCP 64, RFC 4520, DOI 10.17487/RFC4520, June 2006, <https://www.rfc-editor.org/info/rfc4520>.

[W3C.NOTE-datetime-19980827] Wicksteed, C., Ed. and M. Wolf, Ed., "Date and Time Formats", W3C NOTE NOTE-datetime-19980827, W3C NOTE-datetime-19980827, 27 August 1998, <http://www.w3.org/TR/1998/NOTE-datetime-19980827>.

[W3C.xml] "Extensible Markup Language (XML) 1.0 (Fifth Edition)", W3C REC xml, W3C xml, <https://www.w3.org/TR/xml/>.

[W3C.xmlschema11-2] "W3C XML Schema Definition Language (XSD) 1.1 Part 2: Datatypes", W3C REC xmlschema11-2, W3C xmlschema11-2, <https://www.w3.org/TR/xmlschema11-2/>.

[DCMIType] Dublincore, "DCMI Metadata Terms: DCMI Type Vocabulary", January 2020, <https://www.dublincore.org/specifications/dublin-core/dcmi-terms/2020-01-20/>.

[ASN.1] International Telecommunication Union, "Abstract Syntax Notation One (ASN.1): Specification of basic notation", ITU-T Recommendation X.680, February 2021.

[ISO.8601.2004] International Organization for Standardization, "Data elements and interchange formats - Information interchange - Representation of dates and times", ISO Standard 8601, December 2004.

[ITU.X520.2019] International Telecommunications Union, "Information Technology - Open Systems Interconnection - The Directory: Selected attribute types", ITU-T Recommendation X.520, ISO Standard 9594-7, October 2019.

5.2. Informative References

[RFC4517] Legg, S., Ed., "Lightweight Directory Access Protocol (LDAP): Syntaxes and Matching Rules", RFC 4517, DOI 10.17487/RFC4517, June 2006, <https://www.rfc-editor.org/info/rfc4517>.

[ITU.X500.2019] International Telecommunications Union, "Information Technology - Open Systems Interconnection - The Directory: Overview of Concepts, Models and Services", ITU-T Recommendation X.500, ISO Standard 9594-1, October 2019.

[IANAREG] IANA, "Media Types", <https://www.iana.org/assignments/media-types/media-types.xhtml>.

Acknowledgements

This template uses extracts from templates written by Pekka Savola, Elwyn Davies and Henrik Levkowetz.

This document was sponsored by Andy Newton of the IETF ART group.

Sean Turner acted as a shepherd for this document to help it become a proposed standard.

Contributors

This document was reviewed and improved with the help of Howard Chu.

Author’s Address

Carl Eric Codere (editor)
Optima SC Inc.
Canada
Email: carl.codere@optimasc.com
URI: http://www.optimasc.com