

<Working Group Name>
Internet Draft
Intended status: Informational
Expires: December 2009

S. Gryphon
Gryphon Technology
June 8, 2009

LDAP Schema for vCard v4.0
draft-gryphon-ldap-schema-vcard4-00.txt

Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

This document may contain material from IETF Documents or IETF Contributions published or made publicly available before November 10, 2008. The person(s) controlling the copyright in some of this material may not have granted the IETF Trust the right to allow modifications of such material outside the IETF Standards Process. Without obtaining an adequate license from the person(s) controlling the copyright in such materials, this document may not be modified outside the IETF Standards Process, and derivative works of it may not be created outside the IETF Standards Process, except to format it for publication as an RFC or to translate it into languages other than English.

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>

This Internet-Draft will expire on December 8, 2009.

Copyright Notice

Copyright (c) 2009 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 in effect on the date of publication of this document (<http://trustee.ietf.org/license-info>). Please review these documents carefully, as they describe your rights and restrictions with respect to this document.

Abstract

This document works to harmonize the vCard directory information card and Lightweight Directory Access Protocol (LDAP) standards by extending both standards to support a common directory card entity.

Additional LDAP attributes and object classes, and additional properties for vCard are defined. A standard mapping process between the two designed to support vCard's goal of being a transport format between directories (not just LDAP) is defined.

Table of Contents

1.	Introduction	4
1.1.	Background	4
2.	Conventions used in this document	4
3.	LDAP Attribute Types	4
3.1.	'anniversary'	5
3.2.	'birthDate'	5
3.3.	'category'	5
3.4.	'childName'	6
3.5.	'gender'	6
3.6.	'homeCountry'	6
3.7.	'homeCountryName'	7
3.8.	'homeHouseIdentifier'	7
3.9.	'homeLocalityName'	8
3.10.	'homePostalCode'	8
3.11.	'homePostOfficeBox'	8
3.12.	'homeStateName'	9
3.13.	'homeStreet'	9
3.14.	'latLong'	9
3.15.	'logo'	10
3.16.	'otherFacsimileTelephoneNumber'	10
3.17.	'otherHomePhone'	11
3.18.	'otherMailbox'	11

3.19.	'otherMobile'	11
3.20.	'otherPager'	12
3.21.	'otherTelephone'	12
3.22.	'sortName'	12
3.23.	'spouseName'	12
3.24.	'tzid'	13
3.25.	'tzOffset'	13
4.	LDAP Object Classes	14
4.1.	'basicDirectoryCard'	14
4.2.	'directoryCard'	14
4.3.	'extendedDirectoryCard'	15
5.	vCard Schema Extensions	15
5.1.	AN Type Definition	15
5.2.	CAR Type Definition	16
5.3.	DRINK Type Definition	16
6.	Format Conversion	17
6.1.	Roundtrip support	17
6.2.	Attribute to Property Mapping	18
6.3.	Converting from LDAP to vCard	19
6.4.	Converting from vCard to LDAP	21
7.	Calendar LDAP Schema Update	23
7.1.	Calendar Attributes	24
7.1.1.	'calCalAdrURI'	24
7.1.2.	'calCalURI'	24
7.1.3.	'calCAPURI'	24
7.1.4.	'calFBURL'	24
7.1.5.	'calOtherCalAdrURIs'	24
7.1.6.	'calOtherCalURIs'	24
7.1.7.	'calOtherCAPURIs'	25
7.1.8.	'calOtherFBURLs'	25
7.2.	Calendar Object Classes	25
7.2.1.	'calEntry'	25
8.	Security Considerations	25
9.	IANA Considerations	26
10.	References	28
10.1.	Normative References	28
10.2.	Informative References	28
11.	Acknowledgments	29
Appendix A.	Schema Summary	30
A.1.	Attribute source specifications	30
A.1.1.	'basicDirectoryCard' Summary	30
A.1.2.	'directoryCard' Summary	31
A.1.3.	'extendedDirectoryCard' Summary	31
Appendix B.	Changes (to be removed before publication)	32
B.1.	Changes from draft-gryphon-ldap-schema-vcard-00	32

1. Introduction

1.1. Background

vCard is intended to be a format for representing directory information about people in a MIME message, including LDAP directories (see [[VCARD4](#)]).

Lightweight Directory Access Protocol (LDAP) is a common standard used to store and access directory information, including information about people (see [[RFC2798](#)], [[RFC4512](#)], [[RFC4517](#)], and [[RFC4519](#)]).

Although both intended to represent contact information about people, the two standards have significant differences in the attributes they support, and no clear method for mapping between the two.

This standard documents:

- o New LDAP attribute types to support additional fields commonly used in vCard.
- o New LDAP object classes to represent collections of attributes similar to those used in vCard.
- o Extensions to vCard for some relevant attributes from LDAP or otherwise commonly used.
- o Definition of a mapping process between LDAP and vCard.
- o Updates to the Calendar Attributes from [[RFC2739](#)].

2. Conventions used in this document

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 [[RFC2119](#)].

3. LDAP Attribute Types

Current attributes defined in LDAP standards are not sufficient for support of the usual fields in vCard information cards.

These attributes provide the additional support needed to implement the 'directoryCard' object class (detailed below).

In addition, four new attributes ('anniversary', 'childName', 'gender' and 'spouseName') are provided for the vCard extensions detailed in this document (AN, GENDER, RELATED).

3.1. 'anniversary'

The date of marriage, or equivalent, of the contact represented by the directory entry. The value is usually only specified to the calendar day, without a time component.

```
( 1.3.6.1.4.1.33592.1.3.1 NAME 'anniversary'
  EQUALITY generalizedTimeMatch
  ORDERING generalizedTimeOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.24
  SINGLE-VALUE )
```

The 'generalizedTimeMatch' and 'generalizedTimeOrderingMatch' matching rules and the GeneralizedTime (1.3.6.1.4.1.1466.115.121.1.24) syntax are defined in [[RFC4517](#)].

3.2. 'birthDate'

The date of birth of the contact represented by the directory entry. The value is usually only specified to the calendar day, without a time component.

```
( 1.3.6.1.4.1.33592.1.3.2 NAME 'birthDate'
  EQUALITY generalizedTimeMatch
  ORDERING generalizedTimeOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.24
  SINGLE-VALUE )
```

The 'generalizedTimeMatch' and 'generalizedTimeOrderingMatch' matching rules and the GeneralizedTime (1.3.6.1.4.1.1466.115.121.1.24) syntax are defined in [[RFC4517](#)].

3.3. 'category'

The 'category' attribute type is used to classify contact directory entries into categories. Each kind is one value of this multi-valued attribute.


```
( 1.3.6.1.4.1.33592.1.3.3 NAME 'category'  
  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 [\[RFC4517\]](#).

Examples: "work", "friends", and "family".

[3.4.](#) 'childName'

The 'childName' attribute type contains name strings for the children of a person. Each child is one value of this multi-valued attribute.

```
( 1.3.6.1.4.1.33592.1.3.4 NAME 'childName'  
  SUP name )
```

Examples: "Jack Smith", "Jill Smith".

[3.5.](#) 'gender'

Gender of the person represented by the directory entry, encoded as a value from [\[SEX\]](#).

The attribute should have a single digit, with 0 = Not known, 1 = Male, 2 = Female, or 9 = Not applicable. The values are not enforced by the directory, and user applications should consider any unexpected values as equivalent to 0.

```
( 1.3.6.1.4.1.33592.1.3.5 NAME 'gender'  
  EQUALITY integerMatch  
  ORDERING integerOrderingMatch  
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27  
  SINGLE-VALUE )
```

The Integer (1.3.6.1.4.1.1466.115.121.1.27) syntax and the 'integerMatch' and 'integerOrderingMatch' rules are described in [\[RFC4517\]](#).

Example: "1"

[3.6.](#) 'homeCountry'

The 'homeCountry' (Friendly Country Name) attribute specifies names of countries for the home address of the entry in human-readable

format and used in conjunction with 'homeCountryName', which is limited to 2 character ISO codes.

Although LDAP already supports multiple values for address elements (houseIdentifier, street, l, st, postalCode, c), there are no defined rules for correlating multiple entries or for specifying they type (work, home, etc).

In contrast, vCard, provides a very large number of possible options for ways to specify combinations of properties for structured addresses.

The various "home" attributes specified here allow a way to store specific home and work addresses for the same directory entry (similar to 'telephone' and 'homePhone').

```
( 1.3.6.1.4.1.33592.1.3.6 NAME 'homeCountry'
  EQUALITY caseIgnoreMatch
  SUBSTR caseIgnoreSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 )
```

The DirectoryString (1.3.6.1.4.1.1466.115.121.1.15) syntax and the 'caseIgnoreMatch' and 'caseIgnoreSubstringsMatch' rules are described in [[RFC4517](#)].

3.7. 'homeCountryName'

The 'homeCountryName' attribute type contains a two-letter ISO 3166 [[ISO3166](#)] country code.

```
( 1.3.6.1.4.1.33592.1.3.7 NAME 'homeCountryName'
  SUP name
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.11
  SINGLE-VALUE )
```

1.3.6.1.4.1.1466.115.121.1.11 refers to the Country String syntax [[RFC4517](#)].

Examples: "DE", "AU" and "FR".

3.8. 'homeHouseIdentifier'

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


```
( 1.3.6.1.4.1.33592.1.3.8 NAME 'homeHouseIdentifier'  
  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 [\[RFC4517\]](#).

Example: "Unit 20" to represent the unit number 20.

[3.9.](#) 'homeLocalityName'

The 'homeLocalityName' attribute type contains names of a locality or place, such as a city, county, or other geographic region.

```
( 1.3.6.1.4.1.33592.1.3.9 NAME 'homeLocalityName'  
  SUP name )
```

Examples: "Geneva", "Paris", and "Edinburgh".

[3.10.](#) 'homePostalCode'

The 'homePostalCode' attribute type contains codes used by a Postal Service to identify postal service zones.

```
( 1.3.6.1.4.1.33592.1.3.10 NAME 'homePostalCode'  
  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 [\[RFC4517\]](#).

Example: "22180", to identify Vienna, VA, in the USA.

[3.11.](#) 'homePostOfficeBox'

The 'homePostOfficeBox' attribute type contains postal box identifiers that a Postal Service uses when a customer arranges to receive mail at a box on the premises of the Postal Service.

```
( 1.3.6.1.4.1.33592.1.3.11 NAME 'homePostOfficeBox'  
  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 [\[RFC4517\]](#).

Example: "Box 45".

[3.12.](#) 'homeStateName'

The 'homeStateName' attribute type contains the full names of states or provinces.

This attribute may contain a value consistent with the codes specified in [\[ISO3166\]](#) part 2.

```
( 1.3.6.1.4.1.33592.1.3.12 NAME 'homeStateName'  
  SUP name )
```

Example: "California".

[3.13.](#) 'homeStreet'

The 'homeStreet' attribute type contains site information from a postal address (i.e., the street name, place, avenue, and the house number).

```
( 1.3.6.1.4.1.33592.1.3.13 NAME 'homeStreet'  
  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 [\[RFC4517\]](#).

Example: "15 Main St.".

[3.14.](#) 'latLong'

Represents a geographical location using the WGS84 data co-ordinates (used by GPS).

The attribute consists of the latitude in decimal degrees (positive representing north), a semi colon, and then the longitude in decimal degrees (positive representing east). The location may be followed by

an optional additional semicolon and then height in decimal meters above sea level (of the reference geoid). This format is not checked by the directory.

```
( 1.3.6.1.4.1.33592.1.3.14 NAME 'latLong'  
  EQUALITY caseIgnoreMatch  
  SUBSTR caseIgnoreSubstringsMatch  
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 )
```

The DirectoryString (1.3.6.1.4.1.1466.115.121.1.15) syntax and the 'caseIgnoreMatch' and 'caseIgnoreSubstringsMatch' rules are described in [[RFC4517](#)].

Example: "-33.92;151.28".

3.15. 'logo'

Used to store one or more images of a company logo using formats such as the JPEG File Interchange Format [JFIF].

```
( 1.3.6.1.4.1.33592.1.3.15 NAME 'logo'  
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.28 )
```

3.16. 'otherFacsimileTelephoneNumber'

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

Although 'facsimileTelephoneNumber', and other telephone number fields, are already multi-valued, having a separate attribute allows a distinction to be made between the preferred number (usually only one, stored in the main attribute), and any number of additional (not preferred) numbers, stored in the "other" attribute (multiple non-preferred numbers are more common).

Note: Some user applications may not support multiple valued attributes, and so only be able to display at most two numbers in a category (the main one, plus the first "other").

```
( 1.3.6.1.4.1.33592.1.3.16 NAME 'otherFacsimileTelephoneNumber'  
  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 [[RFC4517](#)].

Examples: "+61 3 9896 7801" and "+81 3 347 7418\$fineResolution".

3.17. 'otherHomePhone'

The 'otherHomePhone' attribute specifies additional home telephone numbers (e.g., "+1 775 555 1234") associated with a person.

```
( 1.3.6.1.4.1.33592.1.3.17 NAME 'otherHomePhone'  
  EQUALITY telephoneNumberMatch  
  SUBSTR telephoneNumberSubstringsMatch  
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.50 )
```

The telephoneNumber (1.3.6.1.4.1.1466.115.121.1.50) syntax and the 'telephoneNumberMatch' and 'telephoneNumberSubstringsMatch' rules are described in [[RFC4517](#)].

3.18. 'otherMailbox'

The 'otherMailbox' (rfc822mailbox) attribute type holds Internet mail addresses in Mailbox [[RFC2821](#)] form (e.g., user@example.com).

```
( 1.3.6.1.4.1.33592.1.3.18 NAME 'otherMailbox'  
  EQUALITY caseIgnoreIA5Match  
  SUBSTR caseIgnoreIA5SubstringsMatch  
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.26{256} )
```

The IA5String (1.3.6.1.4.1.1466.115.121.1.26) syntax and the 'caseIgnoreIA5Match' and 'caseIgnoreIA5SubstringsMatch' rules are described in [[RFC4517](#)].

Note: See the 'mail' field from [[RFC2798](#)] for comments on the implementation of this field.

3.19. 'otherMobile'

The 'otherMobile' attribute specifies mobile telephone numbers (e.g., "+1 775 555 6789") associated with a person (or entity).

```
( 1.3.6.1.4.1.33592.1.3.19 NAME 'otherMobile'  
  EQUALITY telephoneNumberMatch  
  SUBSTR telephoneNumberSubstringsMatch  
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.50 )
```

The telephoneNumber (1.3.6.1.4.1.1466.115.121.1.50) syntax and the 'telephoneNumberMatch' and 'telephoneNumberSubstringsMatch' rules are described in [[RFC4517](#)].

[3.20.](#) 'otherPager'

The 'otherPager' attribute specifies pager telephone numbers (e.g., "+1 775 555 5555") for an object.

```
( 1.3.6.1.4.1.33592.1.3.20 NAME 'otherPager'  
  EQUALITY telephoneNumberMatch  
  SUBSTR telephoneNumberSubstringsMatch  
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.50 )
```

The telephoneNumber (1.3.6.1.4.1.1466.115.121.1.50) syntax and the 'telephoneNumberMatch' and 'telephoneNumberSubstringsMatch' rules are described in [[RFC4517](#)].

[3.21.](#) 'otherTelephone'

The 'otherTelephone' attribute type contains telephone numbers that comply with the ITU Recommendation E.123 [[E123](#)]. Each number is one value of this multi-valued attribute.

```
( 1.3.6.1.4.1.33592.1.3.21 NAME 'otherTelephone'  
  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 [[RFC4517](#)].

Example: "+1 234 567 8901".

[3.22.](#) 'sortName'

The 'sortName' attribute type contains name strings for the family names of a person.

```
( 1.3.6.1.4.1.33592.1.3.22 NAME 'sortName'  
  SUP name )
```

Example: "Beethoven", where the sn attribute is "van Beethoven".

[3.23.](#) 'spouseName'

The 'spouseName' attribute type contains name of the spouse of the contact represented by the directory entry.

```
( 1.3.6.1.4.1.33592.1.3.23 NAME 'spouseName'  
  SUP name )
```

Example: "Jane Smith".

3.24. 'tzid'

The usual time zone for the contact represented by the directory entry. Time zones SHOULD be specified using the time zone identifiers from [[UTS35](#)].

```
( 1.3.6.1.4.1.33592.1.3.24 NAME 'tzid'  
  EQUALITY caseIgnoreMatch  
  SUBSTR caseIgnoreSubstringsMatch  
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 )
```

The DirectoryString (1.3.6.1.4.1.1466.115.121.1.15) syntax and the 'caseIgnoreMatch' and 'caseIgnoreSubstringsMatch' rules are described in [[RFC4517](#)].

Examples: "Australia/Sydney", "America/Los_Angeles", "Europe/London".

3.25. 'tzOffset'

The usual time zone offset differential for the person. This does not take into account any daylight savings adjustment.

The attribute contains a plus or minus sign, two digit hour, and then optional two digit minute, similar to the differential portion of Generalized Time syntax. This format is not checked by the directory.

```
( 1.3.6.1.4.1.33592.1.3.25 NAME 'tzOffset'  
  EQUALITY caseIgnoreMatch  
  SUBSTR caseIgnoreSubstringsMatch  
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 )
```

The DirectoryString (1.3.6.1.4.1.1466.115.121.1.15) syntax and the 'caseIgnoreMatch' and 'caseIgnoreSubstringsMatch' rules are described in [[RFC4517](#)].

Example: "+1000".

4. LDAP Object Classes

4.1. 'basicDirectoryCard'

The 'basicDirectoryCard' object class contains basic attributes used in vCard that are supported in the inetOrgPerson object class.

This object class is defined as auxiliary, because it will be used in conjunction with an existing structural object classes.

Any directory which supports inetOrgPerson [[RFC2798](#)] should be able to support this class.

```
( 1.3.6.1.4.1.33592.1.4.1 NAME 'basicDirectoryCard'
  SUP top
  AUXILIARY
  MUST ( cn $
        sn )
  MAY ( audio $ businessCategory $ description $
        displayName $ facsimileTelephoneNumber $
        givenName $ homePhone $ homePostalAddress $
        initials $ l $ labeledURI $ mail $ mobile $
        o $ ou $ pager $ photo $ postalAddress $
        postalCode $ postOfficeBox $ secretary $
        st $ street $ telephoneNumber $ title $
        uid $ userCertificate ) )
```

4.2. 'directoryCard'

The 'directoryCard' provides support for additional vCard attributes that are not in existing LDAP standards, or are in [[RFC4519](#)] and [[RFC4524](#)] but not attributes of inetOrgPerson.

```
( 1.3.6.1.4.1.33592.1.4.2 NAME 'directoryCard'
  SUP basicDirectoryCard
  AUXILIARY
  MAY ( birthDate $ c $ category $ co $
        generationQualifier $ homeCountry $
        homeCountryName $ homeHouseIdentifier $
        homeLocalityName $ homePostalCode $
        homePostOfficeBox $ homeStateName $
        homeStreet $ houseIdentifier $ latLong $
        logo $ otherFacsimileTelephoneNumber $
        otherHomePhone $ otherMailbox $
        otherMobile $ otherPager $ otherTelephone $
        personalTitle $ sortName $ tzOffset ) )
```

4.3. 'extendedDirectoryCard'

The 'extendedDirectoryCard' provides support for new vCard attributes defined in this document and [\[RFC4770\]](#). It includes some attributes already defined in [\[RFC4519\]](#) and [\[RFC4524\]](#) but not in the original vCard specifications.

```
( 1.3.6.1.4.1.33592.1.4.3 NAME 'extendedDirectoryCard'
  SUP directoryCard
  AUXILIARY
  MAY ( anniversary $ carLicense $ childName $
        drink $ gender $ manager $ messagingURI $
        otherMessagingURI $ spouseName ) )
```

5. vCard Schema Extensions

The following extensions are defined for the vCard [\[VCARD4\]](#) to support common attributes from LDAP specifications and other applications.

vCard type definitions are provided corresponding to the 'carLicense' attribute from [\[RFC2798\]](#), and the 'drink' attribute from [\[RFC4524\]](#).

In addition, one new type (AN) is provided, along with corresponding LDAP attribute defined above ('anniversary').

5.1. AN Type Definition

To: ietf-mime-directory@imc.org

Subject: Registration of text/directory MIME type AN

Type name: AN

Type purpose: To specify the date of marriage, or equivalent, of the person the vCard represents.

Type encoding: 8bit

Type value: The default is a single date value.

Type examples:

ANIV:1998-10-31

5.2. CAR Type Definition

To: ietf-mime-directory@imc.org

Subject: Registration of text/directory MIME type CAR

Type name: CAR

Type purpose: Vehicle license or registration plate associated with the person the vCard represents.

Type encoding: 8bit

Type value: A single text value.

Type special notes: This type is based on the semantics of the [\[RFC2798\]](#) 'carLicense' attribute.

Type examples:

CAR:ABC-123

5.3. DRINK Type Definition

To: ietf-mime-directory@imc.org

Subject: Registration of text/directory MIME type DRINK

Type name: DRINK

Type purpose: Specifies the favorite drinks of an object (or person), for instance, "cola" and "beer".

Type encoding: 8bit

Type value: A single text value.

Type special notes: This type is based on the semantics of the [\[RFC4524\]](#) 'drink' attribute.

Type examples:

DRINK: Cola

[6. Format Conversion](#)

[6.1. Roundtrip support](#)

The conversion rules are intended to support roundtrip conversion from LDAP to vCard and then back into LDAP again.

One purpose of vCard is to allow information to be exchanged between directories. A defined mapping gives assurance that when a "business card" is produced from one directory, it will be accurately represented when imported into another.

In practice this means that the vCard produced by this process will only even be subset of the features possible within vCard.

Several advanced features of vCard, such as being able to include one vCard object within a field of another (such as for the AGENT attribute) are not used.

vCard also has a very large number of possible combinations of attributes for fields such a TEL (telephone number), although in practice only a limited set will usually be used.

Alternative handling for information outside this subset when converting in the reverse direction (from vCard to LDAP) is fully detailed below, but in general the full details of the vCard will not be preserved (for example, the exact same set of properties for a TEL attribute may not be retained).

This means that round tripping in the other direction (from vCard to LDAP and back to vCard) is not guaranteed to work unless the vCard falls within the subset supported by this standard.

6.2. Attribute to Property Mapping

The following table defines a mapping from LDAP attributes to vCard parametrized properties, indicating the specific structured field where necessary.

LDAP Attribute	vCard Property	Structured Field
-----	-----	-----
anniversary	AN	
audio	SOUND	
birthDate	BDAY	
businessCategory	ROLE	
c, co	WORK.ADR	country name
carLicense	CAR	
category	CATEGORIES	
childName	RELATED;TYPE=child	
cn	FN	
description	NOTE	
displayName	NAME	
distinguishedName (*1)	SOURCE;CONTEXT=LDAP	
drink	DRINK	
facsimileTelephoneNumber	TEL;TYPE=fax;PREF=1	
gender	GENDER	
generationQualifier	N	suffixes
givenName	N	given name
homeCountry, homeCountryName	HOME.ADR	country name
homeHouseIdentifier	HOME.ADR	extended
homeLocalityName	HOME.ADR	locality
homePhone	HOME.TEL;PREF=1	
homePostalAddress	HOME.LABEL	
homePostalCode	HOME.ADR	postal code
homePostOfficeBox	HOME.ADR	post office box
homeStateName	HOME.ADR	region
homeStreet	HOME.ADR	street address
houseIdentifier	WORK.ADR	extended address
initials	N	given name
l	WORK.ADR	city
labeledURI	URL	
latLong	GEO	
logo	LOGO	
mail	EMAIL;PREF=1	
manager	RELATED;TYPE=manager	
messagingURI	IMPP;PREF=1	
mobile	TEL;TYPE=cell;PREF=1	
o	ORG	org. name
otherFacsimileTelephoneNumber	TEL;TYPE=fax	
otherHomePhone	HOME.TEL	

otherMailbox	EMAIL	
otherMessagingURI	IMPP	
otherMobile	TEL;TYPE=cell	
otherPager	TEL;TYPE=pager	
otherTelephone	WORK.TEL	
ou	ORG	org. unit names
pager	TEL;TYPE=pager;PREF=1	
personalTitle	N	prefixes
photo	PHOTO	
postalAddress	WORK.LABEL	
postalCode	WORK.ADR	postal code
postOfficeBox	WORK.ADR	post office box
secretary	RELATED;TYPE=assistant	
sn	N	family name
sortName	SORT-STRING	
spouseName	RELATED;TYPE=spouse	
st	WORK.ADR	region
street	WORK.ADR	street address
telephoneNumber	WORK.TEL;PREF=1	
title	TITLE	
tzOffset, tzid	TZ	
uid	NICKNAME	
uniqueIdentifier	UID	
userCertificate	KEY	
whenChanged	REV	

(*1) distinguishedName encoded as an LDAP URI.

The above table includes mappings administrative attributes uniqueIdentifier and whenChanged.

In addition to the above mapping, the SOURCE property should specify CONTEXT=LDAP and provide the LDAP URI with the distinguishedName of the directory entry.

6.3. Converting from LDAP to vCard

Attributes SHOULD be mapped from the LDAP entry to the vCard properties as specified above, with the following rules.

1. An application SHOULD map all fields it holds data for, except an application MAY decide to omit fields (or default to omitting fields) containing more personal or sensitive information (primarily AN, BDAY, RELATED fields of type child and spouse).

2. It is recommended that a user SHOULD be able to configure an application to omit or include fields or use an alternative mapping, and an application MAY fallback to alternative properties if the mappings specified here are not possible.

e.g. If a directory entry does not have a 'photo' attribute, then an application may use (or be configurable to use) an alternative attribute such as 'jpegPhoto'.

3. Where a LDAP attribute has multiple values, all values SHOULD appear in the vCard, for example multiple values of the 'title' attribute will result in multiple "TITLE" properties in the vCard.

4. Where a LDAP attribute has multiple values and the mapping specifies a vCard PREF=1, an application MAY omit the PREF attribute from the second and subsequent values.

e.g. If a directory entry has two 'telephoneNumber' values, the first may be mapped to WORK.TEL;PREF=1 and the second to WORK.TEL.

5. Where a single LDAP value exists for an attribute where the mapping specifies a vCard PREF=1, the application MAY omit the PREF attribute provided that no other value (such as from another directory attribute) maps to the resulting vCard property (without the TYPE=PREF).

e.g. If a directory entry has a single 'telephoneNumber' value, then it may be mapped to WORK.TEL provided there are no 'otherTelephone' values (normally 'otherTelephone' maps to WORK.TEL without the PREF attribute).

6. Where a LDAP attributes has multiple values and maps to part of a structured vCard property (such as the ADR or N properties), then the first value of each attribute SHOULD map to the first property, the second value of each attribute SHOULD map to the second property, etc (leaving structured values blank where necessary).

e.g. If a directory entry has one values for 'sn' and two values for 'givenName', then two N properties will be generated. The first N property has the first value for each attribute, and the second N property has the second 'givenName' value only (other parts of the structured value will be empty).

7. The 'givenName' attribute SHOULD be mapped to the first value of the given name structured field in the N property. The 'initials' attribute SHOULD be mapped to a second value of the given name structured field (separated by a comma).
8. To generate the country name portion of an ADR vCard property, any of the following MAY be used (in order of recommendation, noting that vCard is intended to contain a name, not necessarily a code):
 - a. The 'co' (or 'homeCountryName') attribute.
 - b. The value of the 'c' (or 'homeCountry') attribute, converted from a two character code into a country name via lookup.
 - c. The 'c' (or 'homeCountry') attribute (a two character code).
9. To generate the TZ property, any of the following MAY be used (in order of recommendation):
 - a. The 'tzid' attribute (as a text value).
 - b. The 'tzid' attribute timezone, converted to an offset via lookup. The date used for the conversion SHOULD be the time specified in the REV property.
 - c. The 'tzOffset' attribute.
10. The 'manager' and 'secretary' attributes of a directory entry are mapped to RELATED properties by finding the entry they refer to (the attributes contain distinguished names) and taking the 'cn' attribute of the referenced entry.

6.4. Converting from vCard to LDAP

As the mapping from LDAP only utilizes a subset of the vCard format, additional rules are required to consistently import vCard formats that lie outside that subset.

1. An application SHOULD map all fields contained in the vCard.
2. It is recommended that a user SHOULD be able to configure an application to omit or include fields or use an alternative mapping.

3. Where a mapping is defined for a property with PREF=1 and the vCard does not contain a corresponding property, but does contain has a TEL property of that type without PREF=1, then the first property of that type SHOULD be treated as if it contains PREF=1 (and map accordingly).

e.g. If a vCard contains a property WORK.TEL, but no property WORK.TEL;PREF=1, then the value is mapped to the LDAP attribute 'telephoneNumber'.

4. Where a vCard has multiples of the same property with PREF=1, then the application MAY choose to treat the second and subsequent attributes as if they do not have PREF=1 (and map accordingly).

e.g. If a vCard contains multiple properties for the value EMAIL;PREF=1, then the first value must map to LDAP attribute 'mail', but the other values may be mapped to either 'mail' or 'otherMailbox' (i.e. treated as if they are just EMAIL).

5. Any TEL property types other than FAX, CELL, PAGER should be ignored when mapping. If the resulting TEL property has no TYPE and is not in the HOME group:
 - a. If the vCard has no WORK.TEL (either with or without PREF=1), then it SHOULD be mapped to the LDAP attribute 'telephoneNumber'.
 - b. Otherwise, it SHOULD be mapped to the LDAP attribute 'otherTelephone'.

e.g. If a vCard contains TEL;TYPE=cell,voice;PREF=1 then it is treated as if it is TEL;TYPE=cell, and if it contains TEL;TYPE=bbs,modem,car then it is treated as WORK.TEL (or WORK.TEL;PREF=1).

These mappings are necessary because LDAP supports a much smaller range of telephone number types and combinations than vCard.

6. Any ADR or LABEL property TYPE should be ignored when mapping.
7. Any ADR or LABEL property without a group should be treated as in the WORK group, i.e. WORK.ADR or WORK.LABEL.

8. The country name field of a structured ADR property SHOULD be used to set both the 'c' and 'co' attributes (or 'homeCountry' and 'homeCountryName' attributes) if possible. Note that applications should expect that this field may contain either a country name or a two character code.
9. The TZ property SHOULD be used to set both the 'tzid' and 'tzOffset' attributes if possible. If TZ only contains a numeric offset then it should be considered as the offset as at the time specified in the REV property.
10. Any RELATED property of type manager or assistant SHOULD try to be resolved as the 'cn' of another directory entry if possible. Otherwise, an application MAY either:
 - a. Create a new directory entry with the 'cn', using the last word of the value as the 'sn'. In this case the new entry should be tracked in case it can be replaced by another vCard being imported.
 - b. Store the property name and 'cn' value as an additional 'description' attribute of the entry, e.g. "assistant: Bob Smith".

7. Calendar LDAP Schema Update

[RFC2739] defines calendar attributes for vCard and LDAP. The OID values used are from the ISO USA Microsoft tree (1.2.840.113556), however Microsoft is not listed as one of the contributors to the RFC.

In fact, the the OID used for calCAPURI (1.2.840.113556.1.4.480) conflicts with an existing OID used within Microsoft Active Directory (for an attribute named defaultGroup).

Clarification is also needed as to the attributes of the calEntry object class.

The LDAP definitions in [[RFC2739](#)] should be updated to the following values.

7.1. Calendar Attributes

7.1.1. 'calCalAdrURI'

```
( 1.3.6.1.4.1.33592.1.3.26 NAME 'calCalAdrURI'  
  EQUALITY caseIgnoreMatch  
  SUBSTRING caseIgnoreMatch  
  SYNTAX 'IA5String'  
  USAGE userApplications )
```

7.1.2. 'calCalURI'

```
( 1.3.6.1.4.1.33592.1.3.27 NAME 'calCalURI'  
  EQUALITY caseIgnoreMatch  
  SUBSTRING caseIgnoreMatch  
  SYNTAX 'IA5String'  
  USAGE userApplications )
```

7.1.3. 'calCAPURI'

```
( 1.3.6.1.4.1.33592.1.3.28 NAME 'calCAPURI'  
  EQUALITY caseIgnoreMatch  
  SUBSTRING caseIgnoreMatch  
  SYNTAX 'IA5String'  
  USAGE userApplications )
```

7.1.4. 'calFBURL'

```
( 1.3.6.1.4.1.33592.1.3.29 NAME 'calFBURL'  
  EQUALITY caseIgnoreMatch  
  SUBSTRING caseIgnoreMatch  
  SYNTAX 'IA5String'  
  USAGE userApplications )
```

7.1.5. 'calOtherCalAdrURIs'

```
( 1.3.6.1.4.1.33592.1.3.30 NAME 'calOtherCalAdrURIs'  
  EQUALITY caseIgnoreMatch  
  SUBSTRING caseIgnoreMatch  
  SYNTAX 'IA5String'  
  MULTI-VALUE  
  USAGE userApplications )
```

7.1.6. 'calOtherCalURIs'

```
( 1.3.6.1.4.1.33592.1.3.31 NAME 'calOtherCalURIs'  
  EQUALITY caseIgnoreMatch
```



```
SUBSTRING caseIgnoreMatch
SYNTAX 'IA5String'
MULTI-VALUE
USAGE userApplications )
```

7.1.7. 'calOtherCAPURIs'

```
( 1.3.6.1.4.1.33592.1.3.32 NAME 'calOtherCAPURIs'
  EQUALITY caseIgnoreMatch
  SUBSTRING caseIgnoreMatch
  SYNTAX 'IA5String'
  MULTI-VALUE
  USAGE userApplications )
```

7.1.8. 'calOtherFBURLs'

```
( 1.3.6.1.4.1.33592.1.3.33 NAME 'calOtherFBURLs'
  EQUALITY caseIgnoreMatch
  SUBSTRING caseIgnoreMatch
  SYNTAX 'IA5String'
  MULTI-VALUE
  USAGE userApplications )
```

7.2. Calendar Object Classes

7.2.1. 'calEntry'

```
(1.3.6.1.4.1.33592.1.4.4 NAME 'calEntry'
  SUP top
  AUXILIARY
  MAY ( calCalAdrURI $ calCalURI $ calCAPURI $
        calFBURL $ calOtherCalAdrURIs $ calOtherCalURIs $
        calOtherCAPURIs $ calOtherFBURLs ) )
```

8. Security Considerations

Transporting directory information via vCard has several security considerations outlined in [[VCARD4](#)].

These include the transport of cryptographic keys, security classification policies for vCards, spoofing of vCards, and revision date considerations.

As detailed in the mapping section, an application SHOULD allow a user to configure what information is included in a vCard, particularly for personal information such as birth date and family members.

An application MAY implement a mapping or other way of designating a CLASS property for a vCard, or a process for handling the CLASS property of imported vCards. Like [\[VCARD4\]](#), this document does not detail any specific policy for handling CLASS attributes.

9. IANA Considerations

This memo defines IANA registered extensions to the attributes defined by LDAP [\[RFC4512\]](#) and vCard [\[RFC2426\]](#).

IANA registration proposals for vCard, detailed in [section 5](#), are to be emailed to the registration agent for the "text/directory" MIME content-type, <MAILTO: ietf-mime-directory@imc.org> using the format defined in [\[VCARD\]](#).

The Internet Assigned Numbers Authority (IANA) has updated the LDAP descriptors registry [\[RFC4520\]](#) as indicated in the following template:

Subject: Request for LDAP Descriptor Registration Update
Descriptor (short name): see comment
Object Identifier: see comments
Person & email address to contact for further information:
 Stephen Gryphon <sgryphon@computer.org>
Usage: see comments
Specification: [draft-gryphon-ldap-schema-vcard-01](#)
Author/Change Controller: IESG

Comments:

The following descriptors have been updated to refer to this RFC.

NAME	Type	OID
-----	----	-----
anniversary	A	1.3.6.1.4.1.33592.1.3.1
basicDirectoryCard	O	1.3.6.1.4.1.33592.1.4.1
birthDate	A	1.3.6.1.4.1.33592.1.3.2
calCalAdrURI	A	1.3.6.1.4.1.33592.1.3.26
calCalURI	A	1.3.6.1.4.1.33592.1.3.27
calCAPURI	A	1.3.6.1.4.1.33592.1.3.28
calEntry	O	1.3.6.1.4.1.33592.1.4.4
calFBURL	A	1.3.6.1.4.1.33592.1.3.29
calOtherCalAdrURIs	A	1.3.6.1.4.1.33592.1.3.30
calOtherCalURIs	A	1.3.6.1.4.1.33592.1.3.31
calOtherCAPURIs	A	1.3.6.1.4.1.33592.1.3.32
calOtherFBURLs	A	1.3.6.1.4.1.33592.1.3.33
category	A	1.3.6.1.4.1.33592.1.3.3
childName	A	1.3.6.1.4.1.33592.1.3.4
directoryCard	O	1.3.6.1.4.1.33592.1.4.2
extendedDirectoryCard	O	1.3.6.1.4.1.33592.1.4.3
gender	A	1.3.6.1.4.1.33592.1.3.5
homeCountry	A	1.3.6.1.4.1.33592.1.3.6
homeCountryName	A	1.3.6.1.4.1.33592.1.3.7
homeHouseIdentifier	A	1.3.6.1.4.1.33592.1.3.8
homeLocalityName	A	1.3.6.1.4.1.33592.1.3.9
homePostalCode	A	1.3.6.1.4.1.33592.1.3.10
homePostOfficeBox	A	1.3.6.1.4.1.33592.1.3.11
homeStateName	A	1.3.6.1.4.1.33592.1.3.12
homeStreet	A	1.3.6.1.4.1.33592.1.3.13
latLong	A	1.3.6.1.4.1.33592.1.3.14
logo	A	1.3.6.1.4.1.33592.1.3.15
otherFacsimileTelephoneNumber	A	1.3.6.1.4.1.33592.1.3.16
otherHomePhone	A	1.3.6.1.4.1.33592.1.3.17
otherMailbox	A	1.3.6.1.4.1.33592.1.3.18
otherMobile	A	1.3.6.1.4.1.33592.1.3.19
otherPager	A	1.3.6.1.4.1.33592.1.3.20
otherTelephone	A	1.3.6.1.4.1.33592.1.3.21
sortName	A	1.3.6.1.4.1.33592.1.3.22
spouseName	A	1.3.6.1.4.1.33592.1.3.23
tzid	A	1.3.6.1.4.1.33592.1.3.24
tzOffset	A	1.3.6.1.4.1.33592.1.3.25

where Type A is Attribute, Type O is ObjectClass, and * indicates that the registration is historic in nature.

10. References

10.1. Normative References

- [E123] Notation for national and international telephone numbers, ITU-T Recommendation E.123, 1988
- [ISO3166] ISO 3166, "Codes for the representation of names of countries".
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC2798] Smith, M., "Definition of the inetOrgPerson LDAP Object Class", [RFC 2798](#), April 2000.
- [RFC4512] Zeilenga, K., "Lightweight Directory Access Protocol (LDAP): Directory Information Models", [RFC 4512](#), June 2006.
- [RFC4517] Legg, S., Ed., "Lightweight Directory Access Protocol (LDAP): Syntaxes and Matching Rules", [RFC 4517](#), June 2006.
- [RFC4519] Sciberras, A. (Editor), "Lightweight Directory Access Protocol (LDAP): Schema for User Applications", [RFC 4519](#), June 2006.
- [RFC4524] Zeilenga, E., Ed., "COSINE LDAP/X.500 Schema", [RFC 4524](#), June 2006.
- [SEX] ISO 5218:2004, "Codes for the representation of human sexes".
- [UTS35] Davis, M., "Unicode Locale Data Markup Language (LDML)", Unicode Technical Standard #35, version 1.7, May 2007.
- [VCARD4] [draft-ietf-vcarddav-vcardrev-07](#)

10.2. Informative References

- [WGS84] National Geospatial-Intelligence Agency, "NIMA Technical Report TR8350.2 Department of Defense World Geodetic System 1984, Its Definition and Relationships With Local Geodetic Systems", Third Edition.

11. Acknowledgments

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).

Thanks to Andrew Sciberras, Anton Bobrov, Cullen Jennings, Dan Mosedale, David Bienvenu, Denis Hennessy, Frank Dawson, Julian Reschke, Kurt Zeilenga, Mark Smith, Matthew Barnes, Pat Egen, Rich Megginson, Steven Legg, Tim Howes, and Tony Small for reviewing this document.

This document was prepared using 2-Word-v2.0.template.dot.

Appendix A.**Schema Summary****A.1. Attribute source specifications**

This appendix provides a summary of all the attribute types used in the 'basicDirectoryCard', 'directoryCard' and 'extendedDirectoryCard' object classes defined in this document along with their source.

Attributes without a source listed are specified in this document.

A.1.1. 'basicDirectoryCard' Summary

Attribute	Source
-----	-----
audio	RFC2798
businessCategory	RFC2798
cn	RFC4519
description	RFC4519
displayName	RFC2798
facsimileTelephoneNumber	RFC4519
givenName	RFC2798
homePhone	RFC2798
homePostalAddress	RFC2798
initials	RFC2798
l	RFC4519
labeledURI	RFC2798
mail	RFC2798
mobile	RFC2798
o	RFC2798
ou	RFC4519
pager	RFC2798
photo	RFC2798
postalAddress	RFC4519
postalCode	RFC4519
postOfficeBox	RFC4519
secretary	RFC2798
sn	RFC4519
st	RFC4519
street	RFC4519
telephoneNumber	RFC4519
title	RFC4519
uid	RFC2798
userCertificate	RFC2798

A.1.2. 'directoryCard' Summary

In addition to the attributes specified for 'basicDirectoryCard'.

Attribute	Source
-----	-----
co	RFC4524
generationQualifier	RFC4519
homeCountry	
homeCountryName	
homeHouseIdentifier	
homeLocalityName	
homePostalCode	
homePostOfficeBox	
homeStateName	
homeStreet	
houseIdentifier	RFC4519
latLong	
logo	
otherFacsimileTelephoneNumber	
otherHomePhone	
otherMailbox	
otherMobile	
otherPager	
otherTelephone	
personalTitle	RFC4524
sortName	
tzid	
tzOffset	

A.1.3. 'extendedDirectoryCard' Summary

In addition to the attributes specified for 'directoryCard'.

Attribute	Source
-----	-----
drink	RFC4524
gender	
manager	RFC2798
messagingURI	RFC4770
otherMessagingURI	RFC4770
spouseName	

Appendix B. Changes (to be removed before publication)

B.1. Changes from [draft-gryphon-ldap-schema-vcard-00](#)

- o Update from vCard 3.0 to vCard 4.0 draft, including:
- o Mapping for 'initials' attribute changed from additional names to a second value for given name in the N property.
- o Remove CHILD, MGR, SEX, SPOUSE additional vCard properties.
- o Update mapping for AGENT, CHILD, MGR, SPOUSE to use new RELATED property.
- o Remove mention of inline vCards
- o Update mapping for gender (from SEX to GENDER).
- o Change TYPE=WORK, HOME to WORK., HOME.
- o TODO: Check updates for new vCard drafts, e.g. mapping should refer to PREF ordering (not just PREF=1).
- o TODO: Considerations for PID values, URIs in TEL, RELATED.
- o TODO: Submit additional properties (AN, CAR, DRINK) back into VCARD4 (not this document), i.e. remove part 5. Also submit recommendations on GENDER coding (use ISO5218) and TYPE=spouse for the RELATED property.
- o TODO: Calendar now party of vCard 4.0, so move to part of main document (e.g. after [section 4](#)).

Authors' Addresses

Stephen Gryphon
Gryphon Technology Pty Ltd
P.O. Box 1615
Macquarie Centre NSW 2113
Australia

Phone: +61 4 1454 2562
Email: sgryphon@computer.org