

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: April 29, 2012

C. Joy
Oracle
C. Daboo
Apple Inc.
M. Douglass
RPI
October 27, 2011

Schema for representing resources for calendaring and scheduling
services
draft-cal-resource-schema-06

Abstract

This specification describes a schema for representing resources for calendaring and scheduling. A resource in the scheduling context is any shared entity that can be scheduled by a calendar user, but does not control its own attendance status.

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 <http://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 April 29, 2012.

Copyright Notice

Copyright (c) 2011 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 (<http://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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	5
2. Conventions Used in This Document	5
3. General Considerations	5
4. Resource Object	5
4.1. LDAP Resource ObjectClass Definition	5
5. Resource Attributes	6
5.1. Common Name	6
5.1.1. LDAP Attribute Definition	6
5.1.2. VCard Property Definition	6
5.2. Kind	6
5.2.1. LDAP Definition	7
5.2.1.1. LDAP Attribute Definition	7
5.2.2. VCard Property Definition	7
5.2.3. Mapping of KIND value between LDAP and VCard representations	8
5.3. Unique ID	8
5.3.1. LDAP Attribute Definition	8
5.3.2. VCard Property Definition	8
5.4. Nick Name	8
5.4.1. LDAP Attribute Definition	9
5.4.2. VCard Property Definition	9
5.5. Description	9
5.5.1. LDAP Attribute Definition	9
5.5.2. VCard Property Definition	9
5.6. Organizational Unit	9
5.6.1. LDAP Attribute Definition	10
5.6.2. VCard Property Definition	10
5.7. Categories	10
5.7.1. LDAP Attribute Definition	10
5.7.2. VCard Property Definition	10
5.8. Group Member	10
5.8.1. LDAP Attribute Definition	11
5.8.2. VCard Property Definition	11
5.9. Admittance Info	11
5.9.1. LDAP ObjectClass Definition	11
5.9.2. Restricted Access	11
5.9.2.1. LDAP Attribute Definition	11
5.9.2.2. VCard Property Definition	12
5.9.3. Admittance Info URL	12
5.9.3.1. LDAP Attribute Definition	12
5.9.3.2. VCard Property Definition	12
5.10. Accessibility	13

5.10.1. LDAP Attribute Definition	13
5.10.2. VCard Property Definition	13
5.11. Capacity	14
5.11.1. LDAP Attribute Definition	14
5.11.2. VCard Property Definition	14
5.12. Inventory Info	14
5.12.1. LDAP ObjectClass Definition	15
5.12.2. Inventory List	15
5.12.2.1. LDAP Attribute Definition	15
5.12.2.2. VCard Property Definition	15
5.12.3. Inventory URL	16
5.12.3.1. LDAP Attribute Definition	16
5.12.3.2. VCard Property Definition	16
5.13. Owner	16
5.13.1. LDAP Attribute Definition	17
5.13.2. VCard Property Definition	17
5.14. Resource Manager	17
5.14.1. LDAP Attribute Definition	17
5.14.2. VCard Property Definition	18
5.15. Calendar URL	18
5.15.1. LDAP Attribute Definition	18
5.15.2. VCard Property Definition	18
5.16. FreeBusy URL	18
5.16.1. LDAP Attribute Definition	19
5.16.2. VCard Property Definition	19
5.17. Scheduling Address	19
5.17.1. LDAP Attribute Definition	19
5.17.2. VCard Property Definition	19
5.18. Time Zone	20
5.18.1. LDAP Attribute Definition	20
5.18.2. VCard Property Definition	20
5.19. Multiple Bookings	20
5.19.1. LDAP Attribute Definition	20
5.19.2. VCard Property Definition	21
5.20. Maximum Instances	21
5.20.1. LDAP Attribute Definition	21
5.20.2. VCard Property Definition	21
5.21. BookingWindow Start	22
5.21.1. LDAP Attribute Definition	22
5.21.2. VCard Property Definition	23
5.22. BookingWindow End	23
5.22.1. LDAP Attribute Definition	24
5.22.2. VCard Property Definition	24
5.23. Scheduling Approval Info	24
5.23.1. LDAP ObjectClass Definition	25
5.23.2. Auto schedule	25
5.23.2.1. LDAP Attribute Definition	25
5.23.2.2. VCard Property Definition	25

5.23.3. Approval Info URL	26
5.23.3.1. LDAP Attribute Definition	26
5.23.3.2. VCard Property Definition	26
5.23.4. Scheduling Admin Contact	27
5.23.4.1. LDAP Attribute Definition	27
5.23.4.2. VCard Property Definition	27
5.24. Cost	27
5.24.1. LDAP ObjectClass Definition	28
5.24.2. Nocost	28
5.24.2.1. LDAP Attribute Definition	28
5.24.2.2. VCard Property Definition	28
5.24.3. Cost URL	29
5.24.3.1. LDAP Attribute Definition	29
5.24.3.2. VCard Property Definition	29
5.25. Related	29
5.25.1. LDAP Attribute Definition	30
5.25.2. VCard Property Definition	30
6. Examples	30
6.1. LDAP Examples	30
6.1.1. Location Resource	30
6.1.2. Role Resources Group	31
6.2. VCard Examples	32
6.2.1. Location Resource	32
6.2.2. Role Resources Group	33
7. Security Considerations	34
8. IANA Considerations	34
8.1. LDAP Objectclass and Attribute Registration	34
8.2. VCard Property and Value Registration	36
9. Recommendations for Calendaring Systems	37
10. Acknowledgments	37
11. Normative References	38

1. Introduction

This specification defines a schema for representing resources to ease the discovery and scheduling of resources between any calendar client and server.

LDAP and vCard mappings of the schema are described in this document. The Object model chosen is the lowest common denominator to adapt for LDAP.

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. General Considerations

Data values must have valid representation for the chosen format with respect to escape characters, line folding, and so on.

4. Resource Object

A resource object definition should contain all information required to find and schedule the right resource. For this, it should contain all, or a set of the attributes described in Section 5. The cn attribute, described in Section 5.1 MUST be present in any resource object. Additional proprietary attributes may be defined as well, but must begin with "X-". Clients encountering attributes they don't know about must ignore them.

Attributes or Properties required to contact the resource are not included in this specification. LDAP attributes defined in [RFC4519] and VCARD properties defined in vCard Format Specification [RFC6350] can be used to include contact information for the resource.

4.1. LDAP Resource ObjectClass Definition

In LDAP, a resource object SHOULD be defined as an objectclass with attributes as defined in Section 5. This objectClass MUST be an auxiliary class. Its Superior class is the calEntry objectClass as defined in Section 2.4.3.1 of [RFC2739].

Definition of the CalendarResource ObjectClass:

```
( 1.3.6.1.1.x.1.1
  NAME 'CalendarResource'
  DESC 'Calendar Resource Object Class'
  SUP calEntry
  AUXILIARY
  MUST (cn)
  MAY (kind $ nickname $ description $ ou $ categories $
        member $ uniquemember $ accessibilityurl $ capacity $
        owner $ resourcemanager $ timezoneid $
        multiplebookings $ maxinstances $
        bookingwindowstart $ bookingwindowend $
        vcarduid $ related) )
```

5. Resource Attributes

5.1. Common Name

Description:

Full name of the resource. This attribute MUST be defined for a resource object.

ValueType:

String value.

Example value:

Room One

5.1.1. LDAP Attribute Definition

cn attribute as defined in Section 2.3 of [RFC4519]. This attribute MUST be present in a CalendarResource object.

5.1.2. VCard Property Definition

FN property as defined in Section 6.2.1 of [RFC6350].

5.2. Kind

Description:

The kind of object represented.

ValueType:

Some of the possible values are "Location", "Individual", "CalendarResource", or "Group".
Location is used for any physical location resource such as room, building, etc.
Individual is used for for a human resource such as driver, technician, etc.

CalendarResource is used for any physical object that can scheduled like projector, printer, etc.

Group is used to specify a group of resources with a specific skill set. For example: drivers, electricians, etc.

Example value:
Location

5.2.1. LDAP Definition

In LDAP, this information can be represented by including the right category objectclass.

Possible objectclasses are:

Person objectclass as defined in Section 3.12 of [RFC4519].

groupOfNames objectclass as defined in Section 3.5 of [RFC4519].

groupOfUniqueNames objectclass as defined in Section 3.6 of [RFC4519].

device objectclass as defined in Section 3.4 of [RFC4519].

room objectclass as defined in Section 3.8 of [RFC4524].

In the absence of an objectclass that accurately describes the type of the object, the KIND attribute defined below MUST be used.

5.2.1.1. LDAP Attribute Definition

Definition of the kind LDAP attribute:

```
( 1.3.6.1.1.x.0.1
  NAME 'Kind'
  DESC 'Kind of Object'
  EQUALITY caseIgnoreMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE )
```

5.2.2. VCard Property Definition

Property KIND that specifies the kind of object represented, as defined in Section 6.1.4 of [RFC6350]. A new value of "calendarresource" will be used to represent any physical object or device.

5.2.3. Mapping of KIND value between LDAP and VCard representations

KIND Value Mapping Table:

LDAP Objectclass	VCard Value
person	individual
groupOfNames or groupOfUniqueNames	group
device	calendarresource
room	location

5.3. Unique ID

Description:

A Unique Identifier.

ValueType:

Single string value.

Example value:

room1-id1

5.3.1. LDAP Attribute Definition

Definition of the vcarduid LDAP attribute:

```
( 1.3.6.1.1.x.0.1
  NAME 'VcardUid'
  DESC 'VCard UniqueID'
  EQUALITY caseExactMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE )
```

5.3.2. VCard Property Definition

UID property as defined in Section 6.7.6 of [RFC6350].

5.4. Nick Name

Description:

A short or popular name for the resource.

ValueType:

String value.

Example value:
TheOne

5.4.1. LDAP Attribute Definition

Definition of the nickname LDAP attribute:

```
( 1.3.6.1.1.x.0.2
  NAME 'NickName'
  DESC 'Nick Name'
  EQUALITY caseIgnoreIA5Match
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 )
```

5.4.2. VCard Property Definition

NICKNAME property as defined in Section 6.2.3 of [RFC6350].

5.5. Description

Description:
Description of the resource.

ValueType:
String value.

Example value:
Room 1 in Building X

5.5.1. LDAP Attribute Definition

description attribute as defined in Section 2.5 of [RFC4519].

5.5.2. VCard Property Definition

NOTE property as defined in Section 6.7.2 of [RFC6350].

5.6. Organizational Unit

Description:
Organizations the resource belongs to.

ValueType:
String value.

Example value:
EngineeringDepartment

5.6.1. LDAP Attribute Definition

ou attribute as defined in Section 2.20 of [RFC4519].

5.6.2. VCard Property Definition

ORG property as defined in Section 6.6.4 of [RFC6350].

5.7. Categories

Description:

Categories the resource falls under or tags for easy discovery of the resource.

ValueType:

String value. Multi-valued attribute with one attribute per text value in LDAP. One or more text values separated by a COMMA character in VCard property value.

Example value:

Rooms

5.7.1. LDAP Attribute Definition

Definition of the categories LDAP attribute:

```
( 1.3.6.1.1.x.0.3
  NAME 'Categories'
  DESC 'Categories'
  EQUALITY caseIgnoreIA5Match
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 )
```

5.7.2. VCard Property Definition

CATEGORIES property as defined in Section 6.7.1 of [RFC6350].

5.8. Group Member

Description:

List of unique resources in a group of resources object.

ValueType:

URL value.

Example value:

<http://www.example.com/printer1.html>
<http://www.example.com/printer2.html>

5.8.1. LDAP Attribute Definition

member attribute as defined in Section 2.17 or uniquemember attribute as defined in Section 2.40 of [RFC4519].

5.8.2. VCard Property Definition

MEMBER property as defined in Section 6.6.5 of [RFC6350].

5.9. Admittance Info

Description:

Information required to gain access to the resource.

ValueType:

Object value.

5.9.1. LDAP ObjectClass Definition

Definition of the admittanceinfo LDAP objectclass:

```
( 1.3.6.1.1.x.1.2
  NAME 'AdmittanceInfo'
  DESC 'Calendar Resource Admittance Info Class'
  SUP CalendarResource
  AUXILIARY
  MAY (admittanceurl $ restricted) )
```

5.9.2. Restricted Access

Description:

Is access to the resource restricted?

ValueType:

Boolean value.

Example value:

TRUE

5.9.2.1. LDAP Attribute Definition

Definition of the restricted LDAP attribute:

```
( 1.3.6.1.1.x.0.4
  NAME 'Restricted'
  DESC 'Access Restricted'
  EQUALITY booleanMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
```

SINGLE-VALUE)

5.9.2.2. VCard Property Definition

Purpose: To specify if access is restricted or not.

Type value: A single boolean value.

Cardinality: (0,1)

ABNF:

```
RESTRICTEDACCESS-param = ; no parameter allowed
RESTRICTEDACCESS-value = boolean
```

Example:

```
RESTRICTEDACCESS:TRUE
```

5.9.3. Admittance Info URL

Description:

URL pointing to complete information for accessing the resource including getting accessibility rights, special entrances, and so on.

ValueType:

URL value.

Example value:

```
http://www.example.com/room1_admittance.html
```

5.9.3.1. LDAP Attribute Definition

Definition of the admittanceurl LDAP attribute:

```
( 1.3.6.1.1.x.0.5
  NAME 'AdmittanceURL'
  DESC 'Cal Resource Admittance Info URL'
  EQUALITY caseIgnoreIA5Match
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 )
```

5.9.3.2. VCard Property Definition

Purpose: To specify URL pointing to Admission Information.

Type value: URI.

Cardinality: (0,n)

ABNF:

```
ADMISSIONINFO-param = "VALUE=uri" / any-param
ADMISSIONINFO-value = uri
```

Example:

```
ADMISSIONINFO:http://www.example.com/room1_admittance.html
```

5.10. Accessibility

Description:

Special resource accessibility info for the physically disabled.

ValueType:

URL value.

Example value:

```
http://www.example.com/room1_specialaccess.html
```

5.10.1. LDAP Attribute Definition

Definition of the accessibilityurl LDAP attribute:

```
( 1.3.6.1.1.x.0.6
  NAME 'accessibilityURL'
  DESC 'Cal Resource accessibility Info URL'
  EQUALITY caseIgnoreIA5Match
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 )
```

5.10.2. VCard Property Definition

Purpose: To specify URL pointing to Disabled Access Information.

Type value: URI.

Cardinality: (0,n)

ABNF:

```
ACCESSIBILITYINFO-param = "VALUE=uri" / any-param
ACCESSIBILITYINFO-value = uri
```

Example:

```
ACCESSIBILITYINFO:http://www.example.com/room1_specialaccess.html
```

5.11. Capacity

Description:

Capacity of the resource.

ValueType:

Integer.

Example value:

10

5.11.1. LDAP Attribute Definition

Definition of the capacity LDAP attribute:

```
( 1.3.6.1.1.x.0.7
  NAME 'Capacity'
  DESC 'Cal Resource Capacity'
  EQUALITY caseIgnoreIA5Match
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27 )
```

5.11.2. VCard Property Definition

Purpose: To specify Capacity Information.

Type value: integer.

Cardinality: (0,n)

ABNF:

```
CAPACITY-param = "VALUE=integer" / any-param
CAPACITY-value = integer
```

Example:

CAPACITY:10

5.12. Inventory Info

Description:

Information on resources available as part of this resource.

ValueType:

Object value.

5.12.1. LDAP ObjectClass Definition

Definition of the inventoryinfo LDAP attribute:

```
( 1.3.6.1.1.x.1.3
  NAME 'InventoryInfo'
  DESC 'Calendar Resource Inventory Info Class'
  SUP CalendarResource
  AUXILIARY
  MAY (inventorylist $ inventoryurl) )
```

5.12.2. Inventory List

Description:

List of resources available as part of this resource.

ValueType:

String value. Multi-valued attribute with one attribute per text value in LDAP. One or more text values separated by a COMMA character in VCard property value.

Example value:

Printer

5.12.2.1. LDAP Attribute Definition

Definition of the inventorylist LDAP attribute:

```
( 1.3.6.1.1.x.0.8
  NAME 'InventoryList'
  DESC 'Inventory List'
  EQUALITY caseIgnoreIA5Match
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 )
```

5.12.2.2. VCard Property Definition

Purpose: List the resources available as part of this resource.

Type value: One or more text values separated by a COMMA character (ASCII decimal 44).

Cardinality: (0,n)

ABNF:

```
INVENTORYLIST-param = "VALUE=text" / any-param
INVENTORYLIST-value = text
```

Example:
INVENTORYLIST:projector, phone

5.12.3. Inventory URL

Description:
A URL pointing to other resource URLs part of this resource.

ValueType:
URL value.

Example value:
http://www.example.com/room1_inventory.html

5.12.3.1. LDAP Attribute Definition

Definition of the inventoryurl LDAP attribute:

```
( 1.3.6.1.1.x.0.9
  NAME 'InventoryURL'
  DESC 'Cal Resource Inventory Info URL'
  EQUALITY caseIgnoreIA5Match
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 )
```

5.12.3.2. VCard Property Definition

Purpose: To specify URL pointing to Inventory Information.

Type value: URI.

Cardinality: (0,n)

ABNF:
INVENTORYURL-param = "VALUE=uri" / any-param
INVENTORYURL-value = uri

Example:
INVENTORYURL:http://www.example.com/room1_inventory.html

5.13. Owner

Description:
Pointer to the owners of the resource. An owner is anyone who has complete authority over the resource, from naming to overall availability.

ValueType:
URL value.

Example value:
http://www.example.com/room1_ownerinfo.html

5.13.1. LDAP Attribute Definition

owner attribute as defined in Section 2.21 of [RFC4519].

5.13.2. VCard Property Definition

Purpose: To specify URL pointing to Resource Owner. It MAY refer to something other than a vCard object.

Type value: URI.

Cardinality: (0,n)

ABNF:
RESOURCEOWNER-param = "VALUE=uri" / any-param
RESOURCEOWNER-value = uri

Example:
RESOURCEOWNER:http://www.example.com/room1_owner.vcf

5.14. Resource Manager

Description:
Pointer to the managers of the resource. A manager is someone responsible for the day-to-day up keep of the resource.

ValueType:
URL value.

Example value:
http://www.example.com/room1_managerinfo.html

5.14.1. LDAP Attribute Definition

Definition of the resourcemanager LDAP attribute:

```
( 1.3.6.1.1.x.0.10
  NAME 'ResourceManager'
  DESC 'Cal Resource Manager Info'
  EQUALITY distinguishedNameMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.12 )
```

5.14.2. VCard Property Definition

Purpose: To specify URL pointing to Resource Manager.

Type value: URI.

Cardinality: (0,n)

ABNF:

```
RESOURCEMANAGER-param = "VALUE=uri" / any-param  
RESOURCEMANAGER-value = uri
```

Example:

```
RESOURCEMANAGER:http://www.example.com/room1_manager.vcf
```

5.15. Calendar URL

Description:

URL to access calendar data of the resource.

ValueType:

URL value.

Example value:

```
http://www.example.com/calendar/home/Room1/calendar/
```

5.15.1. LDAP Attribute Definition

Calendar access attribute calCAPURI as defined in Section 2.4.4.3 and calOtherCAPURIs as defined in Section 2.4.4.7 of [RFC2739] respectively.

5.15.2. VCard Property Definition

Calendar access property CAPURI as defined in Section 2.3.3 of [RFC2739].

5.16. FreeBusy URL

Description:

URL to read freebusy information of the resource's calendar.

ValueType:

URL value.

Example value:

<http://www.example.com/freebusy/home/Room1/>

5.16.1. LDAP Attribute Definition

Calendar access attribute `calFBURL` as defined in Section 2.4.4.2 and `calOtherFBURLs` as defined in Section 2.4.4.6 of [RFC2739] respectively.

5.16.2. VCard Property Definition

`FBURL` attribute as defined in Section 2.3.1 of [RFC2739] and further explained in Section 6.9.1 of [RFC6350].

5.17. Scheduling Address

Description:

Address used for scheduling the resource by a Calendaring and Scheduling service.

ValueType:

String value.

Example value:

<mailto:room1@example.com>

5.17.1. LDAP Attribute Definition

Scheduling Address attribute `calCalAdrURI` as defined in Section 2.4.4.4 and `calOtherCalAdrURIs` as defined in Section 2.4.4.8 of [RFC2739] respectively. This is the address that would be used by a Scheduling and Calendaring application to schedule the resource. Its value must be a uri string, in most cases a `mailto: uri`. The mail attribute value of the resource should be used for scheduling, in the absence of this attribute.

5.17.2. VCard Property Definition

Scheduling Address property `CALADRURI` as defined in Section 2.3.2 [RFC2739] and further explained in Section 6.9.2 of [RFC6350]. This is the address that would be used by a Scheduling and Calendaring application to schedule the resource. Its value must be a uri string, in most cases a `mailto: uri`. The `EMAIL` property value of the resource should be used for scheduling, in the absence of this attribute.

5.18. Time Zone

Description:

TimeZone Identifier for the timezone the resource is in.

ValueType:

String value.

Example value:

America/New_York

5.18.1. LDAP Attribute Definition

Definition of the timezoneid LDAP attribute:

```
( 1.3.6.1.1.x.0.11
  NAME 'TimeZoneID'
  DESC 'Cal Time Zone ID'
  EQUALITY caseIgnoreIA5Match
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 )
```

5.18.2. VCard Property Definition

TimeZone property TZ as defined in Section 6.5.1 of [RFC6350].

5.19. Multiple Bookings

Description:

Number of simultaneous bookings allowed.

ValueType:

Integer value.

Value of 0 indicates no limits.

Example value:

1

5.19.1. LDAP Attribute Definition

Definition of the multiplebookings LDAP attribute:

```
( 1.3.6.1.1.x.0.12
  NAME 'Multiplebookings'
  DESC 'Cal Num Bookings Allowed'
  EQUALITY integerMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE )
```

5.19.2. VCard Property Definition

Purpose: To specify number of simultaneous bookings allowed.

Type value: integer.

Cardinality: (0,1)

ABNF:

```
MULTIBOOK-param = "VALUE=integer" / any-param  
MULTIBOOK-value = integer
```

Example:

```
MULTIBOOK:10
```

5.20. Maximum Instances

Description:

Maximum number of instances of an event, the resource can be scheduled for from NOW.

ValueType:

Integer value.

Value of 0 indicates no limits.

Example value:

```
60
```

5.20.1. LDAP Attribute Definition

Definition of the maxinstances LDAP attribute:

```
( 1.3.6.1.1.x.0.13  
  NAME 'MaxInstances'  
  DESC 'Cal Maximum Instances allowed'  
  EQUALITY integerMatch  
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27  
  SINGLE-VALUE )
```

5.20.2. VCard Property Definition

Purpose: To specify maximum number of instances of an event, the resource can be scheduled for from NOW.

Type value: integer.

Cardinality: (0,1)

ABNF:

```
MAXINSTANCES-param = "VALUE=integer" / any-param
MAXINSTANCES-value = integer
```

Example:

```
MAXINSTANCES:10
```

5.21. BookingWindow Start

Description:

Defines how much time in advance the resource can be booked. The value of this property is used to calculate the earliest date and time when a resource can be reserved for an event starting on a specific date and time.

If this property value is defined, the resource may be booked for an event at a certain time, only if the current time is equal to or after the date and time calculated by subtracting this value from the event's proposed start time. If this property is absent, then the resource may be booked at any time before the end of the booking window.

ValueType: Duration value.

The format is based on the [ISO.8601.2004] duration representation basic format with designators for the duration of time. The format can represent nominal durations (weeks and days) and accurate durations (hours, minutes, and seconds). The syntax is further defined in Appendix A, "Duration" section of [RFC3339].

Example value:

```
P3M
```

5.21.1. LDAP Attribute Definition

Definition of the bookingwindowstart LDAP attribute:

```
( 1.3.6.1.1.x.0.14
  NAME 'BookingWindowStart'
  DESC 'Cal Booking Window Start'
  EQUALITY caseIgnoreIA5Match
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.26
  SINGLE-VALUE )
```

5.21.2. VCard Property Definition

Purpose: To specify how much time in advance the resource can be booked.

Type value: duration.

The format is based on the [ISO.8601.2004] duration representation basic format with designators for the duration of time. The format can represent nominal durations (weeks and days) and accurate durations (hours, minutes, and seconds). The syntax is further defined in Appendix A, "Duration" section of [RFC3339].

Cardinality: (0,1)

ABNF:

```
BOOKINGWINDOWSTART-param = "VALUE=text" / any-param
BOOKINGWINDOWSTART-value = text
```

Example:

```
BOOKINGWINDOWSTART:P3M
```

5.22. BookingWindow End

Description:

Defines how much time in advance the resource booking is closed. The value of this property is used to calculate the latest date and time when a resource can be reserved for an event starting on a specific date and time.

If the current time is equal to or before the value obtained by subtracting BookingWindowEnd from the start date and time of the event, then the resource may be booked. If this property is absent, then the resource may be booked anytime from booking window start to the start of the event.

BookingWindow Start and End together provide the window of time a resource can be booked, relative to the start time of the event.

```
If BookingWindowStart = BwS,
BookingWindowEnd = BwE,
Current Time = CT and
Event Start Time = ST,
a resource can be booked at a certain time only if
CT is equal to or after (ST - BwS)
and CT is equal to or before (ST - BwE)
```

ValueType: Duration value.

The format is based on the [ISO.8601.2004] duration representation basic format with designators for the duration of time. The format can represent nominal durations (weeks and days) and accurate durations (hours, minutes, and seconds). The syntax is further defined in Appendix A, "Duration" section of [RFC3339].

Example value:

P5D

5.22.1. LDAP Attribute Definition

Definition of the bookingwindowend LDAP attribute:

```
( 1.3.6.1.1.x.0.15
  NAME 'BookingWindowEnd'
  DESC 'Cal Booking Window End'
  EQUALITY caseIgnoreIA5Match
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.26
  SINGLE-VALUE )
```

5.22.2. VCard Property Definition

Purpose: To specify how much time in advance the resource booking is closed.

Type value: duration.

The format is based on the [ISO.8601.2004] duration representation basic format with designators for the duration of time. The format can represent nominal durations (weeks and days) and accurate durations (hours, minutes, and seconds). The syntax is further defined in Appendix A, "Duration" section of [RFC3339].

Cardinality: (0,1)

ABNF:

```
BOOKINGWINDOWEND-param = "VALUE=text" / any-param
BOOKINGWINDOWEND-value = text
```

Example:

BOOKINGWINDOWEND:P5D

5.23. Scheduling Approval Info

Description:

Information regarding approval of a scheduling request to the resource.

ValueType:
Object value.

5.23.1. LDAP ObjectClass Definition

Definition of the schedapprovalinfo LDAP objectclass:

```
( 1.3.6.1.1.x.1.4
  NAME 'SchedApprovalInfo'
  DESC 'Calendar Sched Approval Class'
  SUP CalendarResource
  AUXILIARY
  MAY (autoschedule $ approvalinfofourl $ schedadmin) )
```

5.23.2. Auto schedule

Description:
No approval required. Automatically scheduled.

ValueType:
Boolean value.

Example value:
TRUE

5.23.2.1. LDAP Attribute Definition

Definition of the autoschedule LDAP attribute:

```
( 1.3.6.1.1.x.0.16
  NAME 'Autoschedule'
  DESC 'Cal Scheduling no approval required'
  EQUALITY booleanMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
  SINGLE-VALUE )
```

5.23.2.2. VCard Property Definition

Purpose: To specify if invitations should be automatically scheduled.

Type value: Boolean.

Cardinality: (0,1)

ABNF:

```
AUTOSCHEDULE-param = "VALUE=boolean" / any-param
AUTOSCHEDULE-value = "TRUE" / "FALSE"
```

Example:

```
AUTOSCHEDULE:TRUE
```

5.23.3. Approval Info URL

Description:

URL pointing to complete information on scheduling request approval process for the resource.

ValueType:

URL value.

Example value:

```
http://www.example.com/room1_approval.html
```

5.23.3.1. LDAP Attribute Definition

Definition of the approvalinfourl LDAP attribute:

```
( 1.3.6.1.1.x.0.17
  NAME 'ApprovalInfoURL'
  DESC 'Cal Sched Approval Info'
  EQUALITY caseIgnoreIA5Match
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 )
```

5.23.3.2. VCard Property Definition

Purpose: To specify URL pointing to Scheduling Approval Information.

Type value: URI.

Cardinality: (0,n)

ABNF:

```
APPROVALINFO-param = "VALUE=uri" / any-param
APPROVALINFO-value = uri
```

Example:

```
APPROVALINFO:http://www.example.com/room1_approval.html
```

5.23.4. Scheduling Admin Contact

Description:

Contact information for the scheduling approvers, if approval required.

ValueType:

URL value.

Example value:

<http://www.example.com/SchedAdmin1.vcf>

5.23.4.1. LDAP Attribute Definition

Definition of the schedadmin LDAP attribute:

```
( 1.3.6.1.1.x.0.18
  NAME 'SchedAdmin'
  DESC 'Cal Sched Admin Info'
  EQUALITY distinguishedNameMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.12 )
```

5.23.4.2. VCard Property Definition

Purpose: To specify URL pointing to Scheduling Manager.

Type value: URI.

Cardinality: (0,n)

ABNF:

```
SCHEDADMIN-param = "VALUE=uri" / any-param
SCHEDADMIN-value = uri
```

Example:

SCHEDADMIN:<http://www.example.com/SchedAdmin1.vcf>

5.24. Cost

Description:

Scheduling costs for this resource.

ValueType:

Object value.

5.24.1. LDAP ObjectClass Definition

Definition of the cost LDAP objectclass:

```
( 1.3.6.1.1.x.1.5
  NAME 'CalendarResourceCost'
  DESC 'Calendar Resource Cost Object Class'
  SUP CalendarResource
  AUXILIARY
  MAY (nocost $ costurl)
```

5.24.2. Nocost

Description:

No cost for using the resource. Can be used for a resource scheduling query.

ValueType:

Boolean value.

Example value:

TRUE

5.24.2.1. LDAP Attribute Definition

Definition of the nocost LDAP attribute:

```
( 1.3.6.1.1.x.0.19
  NAME 'Nocost'
  DESC 'Free or Priced resource'
  EQUALITY booleanMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
  SINGLE-VALUE )
```

5.24.2.2. VCard Property Definition

Purpose: To specify if resource usage is free.

Type value: A single boolean value.

Cardinality: (0,1)

ABNF:

```
NOCOST-param = ; no parameter allowed
NOCOST-value = boolean
```

Example:
NOCOST:TRUE

5.24.3. Cost URL

Description:
URL pointing to complete pricing information for usage of the resource.

ValueType:
URL value.

Example value:
<http://www.example.com/cost.html>

5.24.3.1. LDAP Attribute Definition

Definition of the costurl LDAP attribute:

```
( 1.3.6.1.1.x.0.20
  NAME 'CostURL'
  DESC 'Cal Resource Cost Info'
  EQUALITY caseIgnoreIA5Match
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 )
```

5.24.3.2. VCard Property Definition

Purpose: To specify URL pointing Resource Scheduling Cost Information.

Type value: URI.

Cardinality: (0,n)

ABNF:
COSTINFO-param = "VALUE=uri" / any-param
COSTINFO-value = uri

Example:
COSTINFO:<http://www.example.com/cost.html>

5.25. Related

Description:
Specify a relationship with another resource.

ValueType:
URL value.

Example value:
`http://www.example.com/printer1.html`

5.25.1. LDAP Attribute Definition

Definition of the related LDAP attribute:

```
( 1.3.6.1.1.x.0.21
  NAME 'Related'
  DESC 'Related URL'
  EQUALITY uniqueMemberMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.34 )
```

5.25.2. VCard Property Definition

The property RELATED as defined in Section 6.6.6 of [RFC6350].

6. Examples

6.1. LDAP Examples

6.1.1. Location Resource

```
dn: cn=Room One,ou=Engineering,dc=example,dc=com
objectclass: top
objectclass: calendarresource
objectclass: admittanceinfo
objectclass: inventoryinfo
objectclass: schedapprovalinfo
objectclass: calendarresourcecost
objectclass: room
vcarduid: room1-id
cn: Room One
ou: Engineering
nickname: The One
description: Room 1 in Engineering Building X
categories: rooms
categories: engineering_resources
restricted: TRUE
admittanceurl: http://www.example.com/room1_admittance.html
accessibilityurl: http://www.example.com/room1_specialaccess.html
capacity: 100
inventorylist: phone
inventorylist: projector
inventoryurl: http://www.example.com/room1_inventory.html
owner: cn=RoomOwner,ou=Engineering,dc=example,dc=com
resourcemanager: cn=RoomOwner,ou=Engineering,dc=example,dc=com
calcapuri: http://www.example.com/calendar/home/Room1/calendar/
calfburl: http://www.example.com/freebusy/home/Room1/
calcaladruri: mailto:room1@example.com
timezoneid: America/Los_Angeles
multiplebookings: 1
maxinstances: 10
bookingwindowstart: P3M
bookingwindowend: P3D
autoschedule: FALSE
approvalinfourl: http://www.example.com/room1_approval.html
schedadmin: cn=RoomOwner,ou=Engineering,dc=example,dc=com
nocost: FALSE
costurl: http://www.example.com/cost.html
```

6.1.2. Role Resources Group

```
dn: cn=Drivers X,ou=Transportation,dc=example,dc=com
objectclass: top
objectclass: groupOfuniqueNames
objectclass: calendarresource
objectclass: schedapprovalinfo
objectclass: calendarresourcecost
vcarduid: driversX-id
cn: Driver One
ou: Transportation
nickname: The X
description: Drivers in the Transportation department driver pool X
categories: drivers
uniquemember: cn=Driver1,ou=Transportation,dc=example,dc=com
uniquemember: cn=Driver2,ou=Transportation,dc=example,dc=com
uniquemember: cn=Driver3,ou=Transportation,dc=example,dc=com
owner: cn=Transportation_Manager,ou=Transportation,dc=example,dc=com
calfburl: http://www.example.com/freebusy/home/DriversX/
calcaladruri: mailto:driversX@example.com
timezoneid: America/Los_Angeles
multiplebookings: 3
maxinstances: 10
bookingwindowstart: P3M
bookingwindowend: P3D
autoschedule: FALSE
approvalinfourl: http://www.example.com/driversX_approval.html
schedadmin: cn=TransportationManager,ou=Transportation,dc=example,dc=com
nocost: FALSE
costurl: http://www.example.com/driversXcost.html
```

6.2. VCard Examples

6.2.1. Location Resource


```
BEGIN:VCARD
VERSION:4.0
UID:urn:uuid:room1-id
KIND: location
FN: Room One
ORG: Engineering
NICKNAME: The One
NOTE: Room 1 in Engineering Building X
CATEGORIES: rooms, engineering_resources
RESTRICTEDACCESS: TRUE
ADMISSIONINFO: http://www.example.com/room1_admittance.html
ACCESSIBILITYINFO: http://www.example.com/room1_specialaccess.html
CAPACITY: 100
INVENTORYLIST: phone, projector
INVENTORYURL: http://www.example.com/room1_inventory.html
RESOURCEOWNER: http://www.example.com/ResOwner1.vcf
RESOURCEMANAGER: http://www.example.com/ResManager1.vcf
CAPURI: http://www.example.com/calendar/home/Room1/calendar/
FBURL: http://www.example.com/freebusy/home/Room1/
CALADRURI: mailto:room1@example.com
TZ: America/Los_Angeles
MULTIBOOK: 1
MAXINSTANCES: 10
BOOKINGWINDOWSTART:P3M
BOOKINGWINDOWEND: P3D
AUTOSCHEDULE: FALSE
APPROVALINFO: http://www.example.com/room1_approval.html
SCHEDADMIN: http://www.example.com/SchedAdmin1.vcf
NOCOST: FALSE
COSTINFO: http://www.example.com/cost.html
END:VCARD
```

6.2.2. Role Resources Group

```
BEGIN:VCARD
VERSION:4.0
UID:urn:uuid:driverXPool-id
KIND: group
FN: Driver X Pool
ORG: Transportation
NICKNAME: The X Group
NOTE: Drivers in the Transportation department driver pool X
CATEGORIES: drivers
MEMBER:urn:uuid:driver1-id
MEMBER:urn:uuid:driver2-id
MEMBER:urn:uuid:driver3-id
RESOURCEOWNER: http://www.example.com/DriversManager.vcf
FBURL: http://www.example.com/freebusy/home/DriversX/
CALADRURI: mailto:driversX@example.com
TZ: America/Los_Angeles
MULTIBOOK: 3
MAXINSTANCES: 10
BOOKINGWINDOWSTART:P3M
BOOKINGWINDOWEND: P3D
AUTOSCHEDULE: FALSE
APPROVALINFO: http://www.example.com/driversX_approval.html
SCHEDADMIN: http://www.example.com/DriversX_SchedAdmin.vcf
NOCOST: FALSE
COSTINFO: http://www.example.com/driversXcost.html
END:VCARD
```

7. Security Considerations

As this document only defines schema for representing resource information for calendaring and scheduling and does not refer to the actual storage mechanism itself, or the calendaring and scheduling protocol, no special security considerations are required as part of this document.

8. IANA Considerations

8.1. LDAP Objectclass and Attribute Registration

New LDAP objectclasses and attributes defined in this document need to be registered by the Internet Assigned Numbers Authority (IANA) as requested in the following template. Once the assignment is done, this document needs to be updated with the right OID numbers for all the newly defined objectclasses and attributes.

Subject: Request for LDAP Descriptor Registration
 Descriptor (short name): See table below
 Object Identifier: See table below
 Person & email address to contact for further information:
 Ciny Joy <ciny.joy@oracle.com>
 Usage: See table below
 Specification: draft-cal-resource-schema
 Author/Change Controller: IESG

New LDAP ObjectClass and Attributes Table:

Name	Type	Definition	OID
CalendarResource	ObjectClasses	Section 4.1	IANA-ASSIGNED-OID
Kind	Attribute	Section 5.2.1.1	IANA-ASSIGNED-OID
VcardUid	Attribute	Section 5.3.1	IANA-ASSIGNED-OID
NickName	Attribute	Section 5.4.1	IANA-ASSIGNED-OID
Categories	Attribute	Section 5.7.1	IANA-ASSIGNED-OID
AdmittanceInfo	ObjectClasses	Section 5.9.1	IANA-ASSIGNED-OID
Restricted	Attribute	Section 5.9.2.1	IANA-ASSIGNED-OID
AdmittanceURL	Attribute	Section 5.9.3.1	IANA-ASSIGNED-OID
AccessibilityURL	Attribute	Section 5.10.1	IANA-ASSIGNED-OID
Capacity	Attribute	Section 5.11.1	IANA-ASSIGNED-OID
InventoryInfo	ObjectClasses	Section 5.12.1	IANA-ASSIGNED-OID
InventoryList	Attribute	Section 5.12.2.1	IANA-ASSIGNED-OID
InventoryURL	Attribute	Section 5.12.3.1	IANA-ASSIGNED-OID
ResourceManager	Attribute	Section 5.14.1	IANA-ASSIGNED-OID
TimeZoneID	Attribute	Section 5.18.1	IANA-ASSIGNED-OID
MultipleBookings	Attribute	Section 5.19.1	IANA-ASSIGNED-OID
MaxInstances	Attribute	Section 5.20.1	IANA-ASSIGNED-OID

BookingWindowStart	Attribute	Section 5.21.1	IANA-ASSIGNED-0 ID
BookingWindowEnd	Attribute	Section 5.22.1	IANA-ASSIGNED-0 ID
SchedApprovalInfo	ObjectClasses	Section 5.23.1	IANA-ASSIGNED-0 ID
Autoschedule	Attribute	Section 5.23.2	IANA-ASSIGNED-0 ID
ApprovalInfoURL	Attribute	Section 5.23.3	IANA-ASSIGNED-0 ID
SchedAdmin	Attribute	Section 5.23.4	IANA-ASSIGNED-0 ID
CalendarResourceCost	ObjectClasses	Section 5.24.1	IANA-ASSIGNED-0 ID
Nocost	Attribute	Section 5.24.2	IANA-ASSIGNED-0 ID
CostURL	Attribute	Section 5.24.3	IANA-ASSIGNED-0 ID
Related	Attribute	Section 5.25.1	IANA-ASSIGNED-0 ID

8.2. VCard Property and Value Registration

The following new VCard Properties need to be registered by IANA.

New VCard Properties Table:

VCard Property Name	VCard Property Definition
RESTRICTEDACCESS	Section 5.9.2.2
ADMISSIONINFO	Section 5.9.3.2
ACCESSIBILITYINFO	Section 5.10.2
CAPACITY	Section 5.11.2
INVENTORYLIST	Section 5.12.2.2
INVENTORYURL	Section 5.12.3.2
RESOURCEOWNER	Section 5.13.2
RESOURCEMANAGER	Section 5.14.2
MAXINSTANCE	Section 5.20.2
BOOKINGWINDOWSTART	Section 5.21.2
BOOKINGWINDOWEND	Section 5.22.2
AUTOSCHEDULE	Section 5.23.2.2
APPROVALINFO	Section 5.23.3.2
SCHEDADMIN	Section 5.23.4.2
NOCOST	Section 5.24.2.2
COSTINFO	Section 5.24.3.2

The following new VCard Property Values need to be registered by IANA.

New VCard Property Values Table:

VCard Property Name	Additional VCard Property Value	Value Definition
KIND	calendarresource	Section 5.2.2

9. Recommendations for Calendaring Systems

While this document does not mandate how each of the defined attribute values must be used by calendaring systems, here are some recommendations:

1. BookingWindow Start (Section 5.21), Booking Window End (Section 5.22), and Multiple Bookings (Section 5.19) information should be used in freebusy calculations. A query for a time slot that falls outside the booking window or one that already has the maximum allowed number of simultaneous bookings, must be returned as BUSY_UNAVAILABLE.
2. Calendaring systems that support the auto schedule (Section 5.23.2) attribute, should automatically mark the attendee PARTSTAT for a resource as ACCEPTED, if its auto schedule value is TRUE and the scheduling is successful. If owner approval is required, the PARTSTAT could be automatically marked as TENTATIVE.
3. Information like Capacity (Section 5.11) can be used by calendaring systems to warn end users if the number of attendees exceed the capacity value.

10. Acknowledgments

This specification is a result of discussions that took place within the Calendaring and Scheduling Consortium's Resource Technical Committee. The authors thank the participants of that group, and specifically the following individuals for contributing their ideas and support: Arnaud Quillaud, Adam Lewenberg, Andrew Laurence, Guy Stalnaker, Mimi Mugler, Dave Thewlis, Bernard Desruisseaux, Alain Petit, Andrew Sciberras, and Jason Miller.

11. Normative References

- [ISO.8601.2004] International Organization for Standardization, "Data elements and interchange formats -- Information interchange -- Representation of dates and times", 2004.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC2739] Small, T., Hennessy, D., and F. Dawson, "Calendar Attributes for vCard and LDAP", RFC 2739, January 2000.
- [RFC3339] Klyne, G., Ed. and C. Newman, "Date and Time on the Internet: Timestamps", RFC 3339, July 2002.
- [RFC4519] Sciberras, A., "Lightweight Directory Access Protocol (LDAP): Schema for User Applications", RFC 4519, June 2006.
- [RFC4524] Zeilenga, K., "COSINE LDAP/X.500 Schema", RFC 4524, June 2006.
- [RFC6350] Perreault, S., "vCard Format Specification", RFC 6350, August 2011.

Authors' Addresses

Ciny Joy
Oracle Corporation
4210 Network Circle
Santa Clara, CA 95054
USA

EMail: ciny.joy@oracle.com
URI: <http://www.oracle.com/>

Cyrus Daboo
Apple Inc.
1 Infinite Loop
Cupertino, CA 95014
USA

EMail: cyrus@daboo.name
URI: <http://www.apple.com/>

Michael Douglass
Rensselaer Polytechnic Institute
110 8th Street
Troy, NY 12180
USA

EMail: douglm@rpi.edu
URI: <http://www.rpi.edu/>

vcarddav
Internet-Draft
Intended status: Standards Track
Expires: August 21, 2011

D. Cauchie
France Telecom - Orange
B. Leiba
K. Li
Huawei Technologies
February 17, 2011

vCard Format extension : represent vCard extensions defined by the Open
Mobile Alliance (OMA) Converged Address Book (CAB) group
draft-cauchie-vcarddav-oma-cab-extensions-00

Abstract

This document defines extensions to the vCard data format for representing and exchanging certain contact information. The properties covered here have been defined by the Open Mobile Alliance Converged Address Book group, in order to synchronize, using OMA Data Synchronization, important contact fields that were not already defined in the base vCard 4.0 specification.

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 <http://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 August 21, 2011.

Copyright Notice

Copyright (c) 2011 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 (<http://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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1.	Introduction	3
1.1.	Terminology Used in This Document	3
2.	vCard Extensions : Properties	3
2.1.	Property : CONTACT-STATUS-MAIN	3
2.2.	Property : CONTACT-STATUS-UPDATED	4
2.3.	Property : CONTACT-STATUS-TEMPORARY	5
2.4.	Property : CONTACT-LANGUAGE	5
2.5.	Property : SERVICE	6
2.6.	Property : EXPERTISE	7
2.7.	Property : HOBBY	7
2.8.	Property : INTEREST	8
2.9.	Property : PUBLICNOTE	9
2.10.	Property : ORG-DIRECTORY	10
3.	vCard extensions : Parameters	10
3.1.	Parameter : ACCEPT	10
3.2.	Parameter : ACK	11
3.3.	Parameter : CONTACT-ID-REF	11
3.4.	Parameter : INDEX	12
3.5.	Parameter : LANGUAGE-PROFICIENCY-TYPE	12
3.6.	Parameter : LANGUAGE-FLUENCY-TYPE	13
3.7.	Parameter : LEVEL	13
4.	Security Considerations	14
5.	IANA Considerations	14
6.	Acknowledgments	15
7.	Normative References	15
	Authors' Addresses	15

1. Introduction

Synchronization of an Open Mobile Alliance Converged Address Book (OMA-CAB), using Open Mobile Alliance Data Synchronization (OMA-DS), commonly uses vCard as an exchange format between the DS Server and the DS Client. In order to properly perform synchronization of an OMA-CAB, the CAB specification defines vCard extensions that correspond to some important CAB contact fields not already defined in the vCard base specification. This document re-uses the definitions found in the OMA-CAB specification and describes them as vCard extensions. The following sections define the necessary Properties and Parameters.

1.1. Terminology 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].

Syntax specifications shown here use the augmented Backus-Naur Form (ABNF) as described in [RFC5234], and are specified as in the base vcard specification [I-D.ietf-vcarddav-vcardrev].

2. vCard Extensions : Properties

The following sections define the CAB Properties.

2.1. Property : CONTACT-STATUS-MAIN

Namespace:

Property name: CONTACT-STATUS-MAIN

Purpose: To specify the main properties of the CAB status of the object the vCard represents.

Value type: A single structured value consisting of 3 sub-values separated by the SEMI-COLON character (ASCII decimal 59) :

1. contact-type (possible value: "CAB" if the contact is a CAB user)
2. contact-subscription-status (possible values : "active", "pending", "denied", "invalid filter", "not found", "other_error")
3. contact-source indicating the latest source from which the contact data was obtained or updated (default value "CAB")

Cardinality: 1

Property parameters:

Description:

Format definition:

```
CONTACT-STATUS-MAIN-param = "VALUE=CONTACT-STATUS-MAIN-value"  
CONTACT-STATUS-MAIN-value = text
```

Example:

```
CONTACT-STATUS-MAIN:CAB;active;CAB
```

2.2. Property : CONTACT-STATUS-UPDATED

Namespace:

Property name: CONTACT-STATUS-UPDATED

Purpose: To complete the CAB status of the object the vCard represents.

Value type: A single structured value consisting of a value indicating that the contact has been updated by the CAB server, as a result of automatic updates from incoming subscription request(s) (possible values : "incoming subscription request", "contact subscription", "contact updated", "contactshare"). This field may include a CONTACT-ACK parameter.

Cardinality: *

Property parameters:

Description:

Format definition:

```
CONTACT-STATUS-UPDATED-param = "VALUE=CONTACT-STATUS-UPDATED-  
value"  
CONTACT-STATUS-UPDATED-value = text
```

Example:

```
CONTACT-STATUS-UPDATED;ACK=true:contactshare
```

2.3. Property : CONTACT-STATUS-TEMPORARY

Namespace:

Property name: CONTACT-STATUS-TEMPORARY

Purpose: To complete the CAB status of the object the vCard represents.

Value type: A single structured value consisting of a value indicating that the contact is created by the CAB Server, when the contact is not in the AB of the user, and/or the contact requires interaction from CAB User (possible values : "contact subscription", "contact imported", "incoming subscription request" and "contactshare"). This field shall include a ACCEPT parameter. This field may include a CONTACT-ID-REF parameter.

Cardinality: *1

Property parameters:

Description:

Format definition:

CONTACT-STATUS-TEMPORARY-param = "VALUE=CONTACT-STATUS-
TEMPORARY-value"

CONTACT-STATUS-TEMPORARY-value = text

Example:

CONTACT-STATUS-TEMPORARY;CONTACT-ID-REF=150; ACCEPT=yes:
contactshare

2.4. Property : CONTACT-LANGUAGE

Namespace:

Property name: CONTACT-LANGUAGE

Purpose: To specify the language(s) that may be used for contacting the individual associated with the vCard.

Value type: A single language-tag value.

Cardinality: *

Property parameters:

Description: This property can include the "PREF" parameter to indicate a preferred-language (possible values: from 1 to 100). This property can include "LANGUAGE-PROFICIENCY-TYPE" and/or "LANGUAGE-FLUENCY-TYPE" parameters. This property can include an "INDEX" parameter.

Format definition:

```
CONTACT-LANGUAGE-param = "VALUE=CONTACT-LANGUAGE-value" / pref-  
    param / LANGUAGE-PROFICIENCY-TYPE-param / LANGUAGE-  
    FLUENCY-TYPE-param / INDEX-param  
CONTACT-LANGUAGE-value = language-tag
```

Example:

```
CONTACT-LANGUAGE;INDEX=1;LANGUAGE-PROFICIENCY-  
TYPE=speak;LANGUAGE-FLUENCY-TYPE=fluent:en
```

2.5. Property : SERVICE

Namespace:

Property name: SERVICE

Purpose: To specify the aliases used on different sites by the object that the vCard refers to.

Value type: A single structured value consisting of 3 values separated by the SEMI-COLON character (ASCII decimal 59) :

1. label : indicating a free-text description of the service
2. alias : indicating the alias identifier string used for a service
3. url : indicating the URL pointing to the service resource

Cardinality: *

Property parameters:

Description: This property can include the "INDEX" parameter

Format definition:

```
SERVICE-param = "VALUE=SERVICE-value" / INDEX-param
SERVICE-value = text
```

Example:

```
SERVICE;INDEX=1:facebook;Facili
Tie:http://fr-fr.facebook.com/people/Facili-Tie/100001298828793
```

2.6. Property : EXPERTISE

Namespace:

Property name: EXPERTISE

Purpose: To specify the expertise(s) of the object that the vCard refers to.

Value type: A single string value.

Cardinality: *

Property parameters:

Description: This property can include the LEVEL parameter (possible values : "beginner", "average", "expert"). This property can include the "INDEX" parameter.

Format definition:

```
EXPERTISE-param = "VALUE=EXPERTISE-value" / LEVEL-param / INDEX-
param
EXPERTISE-value = text
```

Examples:

```
EXPERTISE;LEVEL=beginner;INDEX=2:chinese literature
EXPERTISE;INDEX=1;LEVEL=expert:chemistry
```

2.7. Property : HOBBY

Namespace:

Property name: HOBBY

Purpose: To specify the hobbies of the object that the vCard refers to. A hobby, as opposed to an interest (see Section 2.8) is an activity that one actively engages in for entertainment, intellectual stimulation, creative expression, or the like.

- * "Art" might be a hobby if one actively sculpts or paints.
- * "Tennis" might be a hobby if one enjoys playing, rather than just watching matches.

Value type: A single string value.

Cardinality: *

Property parameters:

Description: This property can include the LEVEL parameter (possible values : "high", "medium", "low"). This property can include the INDEX parameter.

Format definition:

HOBBY-param = "VALUE=HOBBY-value" / LEVEL-param / INDEX-param
HOBBY-value = text

Examples:

HOBBY;INDEX=1;LEVEL=high:reading

HOBBY;INDEX=2;LEVEL=high:sewing

2.8. Property : INTEREST

Namespace:

Property name: INTEREST

Purpose: To specify the interest(s) of the object that the vCard refers to. An interest, as opposed to a hobby (see Section 2.7) is an activity or topic that one finds interesting, but doesn't necessarily actively engage in.

- * "Art" might be an interest if one likes looking at art, but doesn't create art.
- * "Tennis" might be an interest if one enjoys watching matches, but doesn't play.

Value type: A single string value

Cardinality: *

Property parameters:

Description: This property can include the LEVEL parameter (possible values : "high", "medium", "low"). This property can include the INDEX parameter.

Format definition:

INTEREST-param = "VALUE=INTEREST-value" / LEVEL-param / INDEX-param

INTEREST-value = text

Examples:

INTEREST;INDEX=1;LEVEL=medium:r&b music

INTEREST;INDEX=2;LEVEL=high:rock'n roll music

2.9. Property : PUBLICNOTE

Namespace:

Property name: PUBLICNOTE

Purpose: To specify additional information associated with the object the vCard refers to.

Value type: A single string value

Cardinality: *

Property parameters:

Description:

Format definition:

PUBLICNOTE-param = "VALUE=PUBLICNOTE-value" /language-param

PUBLICNOTE-value = text

Example:

PUBLICNOTE;LANGUAGE=en:Out of my office today

2.10. Property : ORG-DIRECTORY

Namespace:

Property name: ORG-DIRECTORY

Purpose: To specify the organization-directory of the object the vCard represents.

Value type: A single URI value.

Cardinality: *

Property parameters:

Description: This property can include the PREF and INDEX parameters.

Format definition:

```
ORG-DIRECTORY-param = "VALUE=ORG-DIRECTORY-value" / pref-param /  
                      INDEX-param  
ORG-DIRECTORY-value= uri
```

Examples:

```
ORG-DIRECTORY;INDEX=1:http://mycompany.example1.com
```

```
ORG-DIRECTORY;PREF=1;INDEX=2:http://mycompany.example2.com
```

3. vCard extensions : Parameters

The following sections define Parameters used within Properties definitions.

3.1. Parameter : ACCEPT

Namespace:

Parameter name: ACCEPT

Purpose: Used in CONTACT-STATUS-TEMPORARY to indicate, if the user has accepted the temporary contact or not.

Description:

Format definition:

```
ACCEPT-param = "ACCEPT=" ACCEPT-value
ACCEPT-value = "yes" / "no"
```

Example:

```
CONTACT-STATUS-TEMPORARY;CONTACT-ID-REF=150; ACCEPT=yes:
contactshare
```

3.2. Parameter : ACK

Namespace:

Parameter name: ACK

Purpose: Used in CONTACT-STATUS-UPDATED to indicate whether the updated contact has been acknowledged or read by the CAB Client or not.

Description:

Format definition:

```
ACK-param = "ACK=" ACK-value
ACK-value = "true" / "false"
```

Example:

```
CONTACT-STATUS-UPDATED;ACK=true:contactshare
```

3.3. Parameter : CONTACT-ID-REF

Namespace:

Parameter name: CONTACT-ID-REF

Purpose: Used in CONTACT-STATUS-TEMPORARY to indicate, when temporary element is used, a reference to the Contact Entry to which the contact activity-status is associated with.

Description:

Format definition:

```
CONTACT-ID-REF-param = "CONTACT-ID-REF=" CONTACT-ID-REF-value
CONTACT-ID-REF-value = integer
```

Example:

```
CONTACT-STATUS-TEMPORARY;CONTACT-ID-REF=150; ACCEPT=yes:
contactshare
```

3.4. Parameter : INDEX

Namespace:

Parameter name: INDEX

Purpose: Used to indicate the range of each value when a parameter can take several values. possible values : token.

Description:

Format definition:

```
INDEX-param = "INDEX=" INDEX-value
INDEX-value = integer
```

Examples:

```
ORG-DIRECTORY;INDEX=1:http://mycompany.example1.com
```

```
ORG-DIRECTORY;PREF=1;INDEX=2:http://mycompany.example2.com
```

3.5. Parameter : LANGUAGE-PROFICIENCY-TYPE

Namespace:

Parameter name: LANGUAGE-PROFICIENCY-TYPE

Purpose: Used to indicate which degree of proficiency the object the vCard represents attained in the corresponding language. possible values : "read only", "speak", "read/write".

Description:

Format definition:

```
LANGUAGE-PROFICIENCY-TYPE-param = "LANGUAGE-PROFICIENCY-TYPE="
                                LANGUAGE-PROFICIENCY-TYPE-value
LANGUAGE-PROFICIENCY-TYPE-value = "read only" / "speak" / "read/
                                write"
```

Example:

```
CONTACT-LANGUAGE;LANGUAGE-PROFICIENCY-TYPE=speak:en
```

3.6. Parameter : LANGUAGE-FLUENCY-TYPE

Namespace:

Parameter name: LANGUAGE-FLUENCY-TYPE

Purpose: Used to indicate which degree of fluency the object the vCard represents attained in the corresponding language.

Possible values : "beginner", "average", "fluent".

Description:

Format definition:

```
LANGUAGE-FLUENCY-TYPE-param = "LANGUAGE-FLUENCY-TYPE=" LANGUAGE-
                                FLUENCY-TYPE-value
LANGUAGE-FLUENCY-TYPE-value = "beginner" / "average" / "fluent"
```

Example:

```
CONTACT-LANGUAGE;LANGUAGE-FLUENCY-TYPE=fluent:en
```

3.7. Parameter : LEVEL

Namespace:

Parameter name: LEVEL

Purpose: Used to indicate a level of expertise, hobby or interest attained by the object the vCard represents.

Possible values:

- * "beginner", "average", "expert" when used with EXPERTISE
- * "high", "medium", "low" when used with HOBBY or INTEREST

Description:

Format definition:

```
LEVEL-param = "LEVEL=" LEVEL-value
LEVEL-value = "beginner" / "average" / "expert" / "high" /
              "medium" / "low"
```

Examples:

```
EXPERTISE;LEVEL=beginner:chinese literature
```

```
HOBBY;LEVEL=high:reading
```

```
INTEREST;LEVEL=medium:r&b music
```

4. Security Considerations

This presents no security considerations beyond those in section 9 of the base vcard specification [I-D.ietf-vcarddav-vcardrev].

5. IANA Considerations

IANA is requested to add the following entries to the vCard Properties registry, defined in [I-D.ietf-vcarddav-vcardrev] section 10.3.1.

Name space	Property	Status	Reference
	CONTACT-STATUS-MAIN	Current	RFCXXXX, sec 2.1
	CONTACT-STATUS-UPDATED	Current	RFCXXXX, sec 2.2
	CONTACT-STATUS-TEMPORARY	Current	RFCXXXX, sec 2.3
	CONTACT-LANGUAGE	Current	RFCXXXX, sec 2.4
	SERVICE	Current	RFCXXXX, sec 2.5
	EXPERTISE	Current	RFCXXXX, sec 2.6
	HOBBY	Current	RFCXXXX, sec 2.7
	INTEREST	Current	RFCXXXX, sec 2.8
	PUBLICNOTE	Current	RFCXXXX, sec 2.9
	ORG-DIRECTORY	Current	RFCXXXX, sec 2.10

IANA is requested to add the following entries to the vCard Parameters registry, defined in [I-D.ietf-vcarddav-vcardrev] section 10.3.2.

Name space	Parameter	Status	Reference
	ACCEPT	Current	RFCXXXX, sec 3.1
	ACK	Current	RFCXXXX, sec 3.1
	CONTACT-ID-REF	Current	RFCXXXX, sec 3.1
	INDEX	Current	RFCXXXX, sec 3.2
	LANGUAGE-PROFICIENCY-TYPE	Current	RFCXXXX, sec 3.3
	LANGUAGE-FLUENCY-TYPE	Current	RFCXXXX, sec 3.4
	LEVEL	Current	RFCXXXX, sec 3.5

6. Acknowledgments

TBD

7. Normative References

- [I-D.ietf-vcarddav-vcardrev]
Perreault, S. and P. Resnick, "vCard Format Specification", draft-ietf-vcarddav-vcardrev-15 (work in progress), December 2010.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC5234] Crocker, D. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, January 2008.

Authors' Addresses

Dany Cauchie
France Telecom - Orange
2 Avenue Pierre Marzin
Lannion 22307
France

Phone: +33 2 96 05 31 16
Email: dany.cauchie@orange-ftgroup.com

Barry Leiba
Huawei Technologies

Phone: +1 646 827 0648
Email: barryleiba@computer.org
URI: <http://internetmessagingtechnology.org/>

Kepeng Li
Huawei Technologies

Phone: +86 755 28974289
Email: likepeng@huawei.com

vcarddav
Internet-Draft
Intended status: Standards Track
Expires: August 17, 2011

R. George
B. Leiba
K. Li
Huawei Technologies
A. Melnikov
Isode Limited
February 13, 2011

vCard Format Extension : To Represent the Social Network Information of
an Individual
draft-george-vcarddav-vcard-extension-03

Abstract

This document defines an extension to the vCard data format for representing and exchanging a variety of social network information.

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 <http://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 August 17, 2011.

Copyright Notice

Copyright (c) 2011 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 (<http://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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as

described in the Simplified BSD License.

Table of Contents

1.	Introduction	3
1.1.	Terminology Used in This Document	3
2.	Social Network Properties	3
2.1.	Property: OPENID	3
2.2.	Property: SOCIALPROFILE	4
2.3.	Property: ALBUM	5
2.4.	Property: DEPICTION	5
2.5.	Property: SOCIALCODE	6
2.6.	Property: INTEREST	7
2.7.	Property: XX	8
3.	Security Considerations	8
4.	IANA Considerations	9
5.	References	9
5.1.	Normative References	9
5.2.	Informative References	9
	Authors' Addresses	10

1. Introduction

As social networking has become common, it has become clear that users would like to include information in their vCards [I-D.ietf-vcarddav-vcardrev] about their social networks. Well organized social network information allows the vCard owner to share his profile information and to import or subscribe to profile information of others on joining a new network.

1.1. Terminology 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].

Syntax specifications shown here use the augmented Backus-Naur Form (ABNF) as described in [RFC5234], and are specified as in the base vcard specification [I-D.ietf-vcarddav-vcardrev].

2. Social Network Properties

These properties are related to sharing social-network information. The basis for these properties came from the "Friend of a Friend" (FOAF) specification, <http://xmlns.com/foaf/spec/>, and we should consider other aspects of that specification.

[[anchor1: *** Do we want to align directly with FoaF items? Are there more FoaF items we want to include, even if we're not aligning completely? *** Barry]]

2.1. Property: OPENID

[[anchor2: *** Maybe this should be something like "authentication;type=openid:" instead? That would allow for other authentication types. *** Barry]]

Namespace:

Property name: OPENID

Purpose: OpenID is an open, decentralized user identification standard, allowing users to log onto many services with the same digital identity. Inclusion of an OpenID URI in a vCard lets others add the vCard owner's ID to their authorization lists.

Value type: A single URI value.

Cardinality: *

Property parameters: (none)

Description:

Format definition:

```
OPENID-param = pid-param / pref-param /  
               any-param  
OPENID-value = uri
```

Example:

```
OPENID:http://www.alice.openid.example.org
```

2.2. Property: SOCIALPROFILE

Namespace:

Property name: SOCIALPROFILE

Purpose: Designates the vCard owner's profile page on a particular social network.

Value type: A single URI value.

Cardinality: *

Property parameters: TYPE

Description: This property SHOULD include the parameter "TYPE" to specify the name of the social network that it refers to. Usually, that will also be discernible from the URI, which is why it's optional. But it can be helpful to have it specified explicitly.

Format definition:

```
SOCIALPROFILE-param = pid-param / pref-param /  
                      any-param  
SOCIALPROFILE-value = uri
```

Examples:

```
SOCIALPROFILE;type=linkedin:http://www.linkedin.com/in/barryleiba
```

```
SOCIALPROFILE;type=facebook:http://www.facebook.com/barackobama
```

2.3. Property: ALBUM

Namespace:

Property name: ALBUM

Purpose: Designates an online album, such as a photo album or video album.

Value type: A single URI value.

Cardinality: *

Property parameters: TYPE

Description: This property SHOULD include the parameter "TYPE" to specify the type of album that it refers to. Usually, that will also be discernible from the URI, which is why it's optional. But it can be helpful to have it specified explicitly.

Format definition:

```
ALBUM-param = pid-param / pref-param /  
              any-param  
ALBUM-value = uri
```

Example:

```
ALBUM;type=photo:http://picasaweb.google.com/barryleiba  
ALBUM;type=video:http://www.youtube.com/user/barryleiba
```

2.4. Property: DEPICTION

```
[[anchor3: *** What's the difference between this and "photo", from  
the base spec? This is in FoaF, but do we really need it here? Any  
comments from folks who are well versed in FoaF? I get the sense  
that this is meant to be different, but I'm not sure how. ***  
(Barry)]]
```

Namespace:

Property name: DEPICTION

Purpose: A depiction of something.

Value type: A single value. The default is binary value. It can also be reset to uri value.

Cardinality: *

Property parameters: VALUE

Description: A common use of depiction is to indicate the contents of a digital image, for example the people or objects represented in an online photo gallery.

The basic notion of 'depiction' could also be extended to deal with multimedia content (video clips, audio),

Format definition:

```
DEPICTION-param = pid-param / pref-param /  
                  any-param  
DEPICTION-value = text
```

Example:

```
DEPICTION;value=uri:http://www.example.com/pub/photos/jqpublic.gif
```

2.5. Property: SOCIALCODE

Namespace:

Property name: SOCIALCODE

Purpose: Description of the vCard owner, in the form of a "social code", such as the "geek code" (see http://en.wikipedia.org/wiki/Geek_code). Social codes are popularly used to exchange a large amount of social information in a compact way, and provide a somewhat frivolous and willfully obscure "fun" mechanism for characterizing technical expertise, interests, and habits.

Value type: A single text value.

Cardinality: *

Property parameters: TYPE

Description: This property MUST include the parameter "TYPE" to specify the type of social network code being used. There are no predefined values for "TYPE", here -- the types will be understood (or not) by the vCard users.

If the code contains characters that have to be quoted, such as COLON, SEMICOLON, or COMMA, the value MUST be enclosed in quotes.

Format definition:

```
SOCIALCODE-param = pid-param / pref-param /  
                    any-param  
SOCIALCODE-value = text
```

Example:

```
SOCIALCODE;type=geek:"s: a--"
```

[Which means "I'm average size, and my age is 20-24."]

2.6. Property: INTEREST

Namespace:

Property name: INTEREST

Purpose: Lists the vCard owner's interests (social, recreational, technical, etc.). This allows users to identify others with common interests.

Value type: A string value consisting of one or more text values separated by a COMMA character (ASCII decimal 44).

Cardinality: *

Property parameters: TYPE, LANGUAGE

Description: This property MAY include the parameter "TYPE" to group interests in categories. TYPE might be used to separate "business" interests from "social" interests, for example. There are no predefined values for "TYPE", here -- the types will be understood (or not) by the vCard users, and it's likely that an ad hoc taxonomy will develop, as has happened with social tagging.

Format definition:

```
INTEREST-param = pid-param / pref-param /  
                  any-param  
INTEREST-value = text
```


Example:

```
INTEREST;type=business:Internet standards,consulting,job offers
INTEREST;type=social:friends and family,new friends
INTEREST;type=hobby:model trains,reading Sci Fi,travel
INTEREST;type=music:classical,jazz,folk,opera
```

2.7. Property: XX

[[anchor4: Template for adding another property, because we expect to add more properties here. Remove this section before publishing.]]
(This will also hold some references for the time being: [RFC2425] [RFC2426] [RFC2739] [RFC4770])

Namespace:

Property name:

Purpose:

Value type: A single text value.

Cardinality: *

Property parameters: VALUE, LANGUAGE

Description:

Format definition:

```
XX-param = pid-param / pref-param /
           any-param
XX-value = text
```

Example:

```
xx:zz
```

3. Security Considerations

This presents no security considerations beyond those in section 9 of the base vcard specification [I-D.ietf-vcarddav-vcardrev].

[[anchor5: *** I'm quite sure there's more to say here, and that there are some real security (and privacy) considerations, so this is just a placeholder. We need to think about this seriously before we're done. *** (Barry)]]

4. IANA Considerations

The IANA is requested to add the following entries to the vCard Properties registry, defined in [I-D.ietf-vcarddav-vcardrev] section 10.3.1.

Namespace	Property	Status	Reference
	OPENID	Current	RFCXXXX, section 2.1
	SOCIALPROFILE	Current	RFCXXXX, section 2.2
	ALBUM	Current	RFCXXXX, section 2.3
	DEPICTION	Current	RFCXXXX, section 2.4
	SOCIALCODE	Current	RFCXXXX, section 2.5
	INTEREST	Current	RFCXXXX, section 2.6

5. References

5.1. Normative References

- [I-D.ietf-vcarddav-vcardrev]
Perreault, S. and P. Resnick, "vCard Format Specification", draft-ietf-vcarddav-vcardrev-15 (work in progress), December 2010.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC5234] Crocker, D. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, January 2008.

5.2. Informative References

- [RFC2425] Howes, T., Smith, M., and F. Dawson, "A MIME Content-Type for Directory Information", RFC 2425, September 1998.
- [RFC2426] Dawson, F. and T. Howes, "vCard MIME Directory Profile", RFC 2426, September 1998.
- [RFC2739] Small, T., Hennessey, D., and F. Dawson, "Calendar Attributes for vCard and LDAP", RFC 2739, January 2000.
- [RFC4770] Jennings, C. and J. Reschke, Ed., "vCard Extensions for Instant Messaging (IM)", RFC 4770, January 2007.

Authors' Addresses

Robins George
Huawei Technologies
Bangalore, Karnataka 560071
India

Phone: +91-080-41117676
Email: robinsgv@gmail.com

Barry Leiba
Huawei Technologies

Phone: +1 646 827 0648
Email: barryleiba@computer.org
URI: <http://internetmessagingtechnology.org/>

Kepeng Li
Huawei Technologies
Huawei Base, Bantian, Longgang District
Shenzhen, Guangdong 518129
P. R. China

Phone: +86-755-28974289
Email: likepeng@huawei.com

Alexey Melnikov
Isode Limited
5 Castle Business Village
36 Station Road, Hampton Middlesex TW12 2BX
UK

Email: Alexey.Melnikov@isode.com
URI: <http://www.melnikov.ca/>

vcarddav
Internet-Draft
Intended status: Standards Track
Expires: January 6, 2012

K. Li
B. Leiba
Huawei Technologies
July 5, 2011

vCard Format Extensions : place of birth, place and date of death
draft-li-vcarddav-vcard-id-property-extensions-03

Abstract

The base vCard 4.0 specification defines a large number of properties, including date of birth. This specification adds three new properties to vCard 4.0, for place of birth, place of death, and date of death.

Note

Discussion and suggestions for improvement are requested, and should be sent to vcarddav@ietf.org.

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 <http://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 January 6, 2012.

Copyright Notice

Copyright (c) 2011 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 (<http://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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
1.1. Terminology Used in This Document	3
2. Identification Property Extensions	3
2.1. Property: BIRTHPLACE	3
2.2. Property: DEATHPLACE	4
2.3. Property: DEATHDATE	4
3. Security Considerations	5
4. IANA Considerations	5
5. Acknowledgements	6
6. Normative References	6
Authors' Addresses	6

1. Introduction

The base vCard 4.0 specification [I-D.ietf-vcarddav-vcardrev] defines a large number of properties, including date of birth. This specification adds three new properties to vCard 4.0, for place of birth, place of death, and date of death.

1.1. Terminology 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].

Syntax specifications shown here use the augmented Backus-Naur Form (ABNF) as described in [RFC5234], and are specified as in the base vcard specification [I-D.ietf-vcarddav-vcardrev].

2. Identification Property Extensions

2.1. Property: BIRTHPLACE

Namespace:

Property name: BIRTHPLACE

Purpose: To specify the place of birth of the object the vCard represents.

Value type: A single text value (default) or a single URI value.

Cardinality: *1

Property parameters: VALUE, LANGUAGE

Description:

Format definition:

```
BIRTHPLACE-param = "VALUE=" / ("text" / "uri")
BIRTHPLACE-value = text / uri
; Value and parameter MUST match.
```

```
BIRTHPLACE-param =/ altid-param / language-param / any-param
```

Examples:

```
BIRTHPLACE:Babies'R'Us Hospital
BIRTHPLACE;VALUE=uri:http://example.com/hospitals/babiesrus.vcf
BIRTHPLACE;VALUE=uri:geo:46.769307,-71.283079
```

2.2. Property: DEATHPLACE

Namespace:

Property name: DEATHPLACE

Purpose: To specify the place of death of the object the vCard represents.

Value type: A single text value (default) or a single URI value.

Cardinality: *1

Property parameters: VALUE, LANGUAGE

Description:

Format definition:

```
DEATHPLACE-param = "VALUE=" / ("text" / "uri")
DEATHPLACE-value = text / uri
; Value and parameter MUST match.
```

```
DEATHPLACE-param =/ altid-param / language-param / any-param
```

Examples:

```
DEATHPLACE:Aboard the Titanic\, near Newfoundland
DEATHPLACE;VALUE=uri:http://example.com/ships/titanic.vcf
DEATHPLACE;VALUE=uri:41.731944,-49.945833
```

2.3. Property: DEATHDATE

Namespace:

Property name: DEATHDATE

Purpose: To specify the date of death of the object the vCard represents.

Value type: The default is a single date-and-or-time value. It can also be reset to a single text value.

Cardinality: *1

Property parameters: VALUE, CALSCALE, LANGUAGE

CALSCALE can only be present when the value is a
date-and-or-time value and actually contains a date or date-time.
LANGUAGE can only be present when the value is text.

Description:

Format definition:

DEATHDATE-param = DEATHDATE-param-date / DEATHDATE-param-text
DEATHDATE-value = date-and-or-time / text
; Value and parameter MUST match.

DEATHDATE-param-date = "VALUE=date-and-or-time"

DEATHDATE-param-text = "VALUE=text" / language-param

DEATHDATE-param =/ altid-param / calscale-param / any-param
; calscale-param can only be present when DEATHDATE-value is
; date-and-or-time and actually contains a date or date-time.

Examples:

DEATHDATE:19960415

DEATHDATE:--0415

DEATHDATE;19531015T231000Z

DEATHDATE;VALUE=text:circa 1800

3. Security Considerations

This presents no security considerations beyond those in section 9 of the base vcard specification [I-D.ietf-vcarddav-vcardrev].

4. IANA Considerations

The IANA is requested to add the following entries to the vCard Properties registry, defined in [I-D.ietf-vcarddav-vcardrev] section 10.3.1.

Namespace	Property	Status	Reference
	BIRTHPLACE	Current	RFCXXXX, section 2.1
	DEATHPLACE	Current	RFCXXXX, section 2.2
	DEATHDATE	Current	RFCXXXX, section 2.3

5. Acknowledgements

The author of this draft would like thank the authors of draft-ietf-vcarddav-vcardrev-13, because much of the text is copied from there.

6. Normative References

- [I-D.ietf-vcarddav-vcardrev]
Perreault, S., "vCard Format Specification",
draft-ietf-vcarddav-vcardrev-22 (work in progress),
May 2011.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC5234] Crocker, D. and P. Overell, "Augmented BNF for Syntax
Specifications: ABNF", STD 68, RFC 5234, January 2008.

Authors' Addresses

Kepeng Li
Huawei Technologies
Huawei Base, Bantian, Longgang District
Shenzhen, Guangdong 518129
P. R. China

Phone: +86-755-28974289
Email: likepeng@huawei.com

Barry Leiba
Huawei Technologies

Phone: +1 646 827 0648
Email: barryleiba@computer.org
URI: <http://internetmessagingtechnology.org/>

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: January 11, 2012

P. Saint-Andre
Cisco
July 10, 2011

vCard KIND:application
draft-saintandre-vcarddav-thing-02

Abstract

This document defines a value of "application" for the vCard KIND property so that vCards can be used to represent software applications.

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 <http://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 January 11, 2012.

Copyright Notice

Copyright (c) 2011 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 (<http://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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. Meaning	3
3. Example	4
4. IANA Considerations	5
5. Security Considerations	5
6. Acknowledgements	5
7. References	6
7.1. Normative References	6
7.2. Informative References	6
Author's Address	6

1. Introduction

Version 4 of the vCard specification [VCARD] defines a new "KIND" property to specify the type of entity that a vCard represents. During its work on the base vCard4 specification, the VCARDDAV Working Group defined values of "individual", "organization", "group", and "location" for the KIND property. The working group considered but then removed a value of "thing" to represent any type of physical entity, machine, software application, etc., with the expectation that such a value might be defined in a vCard extension. This document does not define a generic "thing" value, but instead defines a more narrow "application" value so that vCards can be used to represent software applications.

2. Meaning

When the KIND property has a value of "application", the vCard represents a software application such as a server, an online service (e.g., a chatroom), or an automated softwarebot. More formally, an "application" is functionally equivalent to the 'applicationProcess' object class used in the Lightweight Directory Access Protocol [RFC4519] and derived from the Open Systems Interconnection model [X.521] and [X.200]. As one example of the "application" KIND, vCards are currently used in the Extensible Messaging and Presence Protocol [RFC6120] to represent instant messaging servers that are deployed on the network.

The properties included in an application's vCard apply to one of the following:

- o The application itself (e.g., the FN property might represent the friendly name of an application service, the URL property might represent a website that contains further information about the service, and the ADR, GEO, and TZ properties might represent the physical address, geographical location, and timezone of the machine where the service is hosted).
- o An organization or person that makes the application available on the network (e.g., the LOGO property might represent the corporate logo of a service provider).
- o A person or role that maintains the application (e.g., the TEL, EMAIL, and IMPP properties might represent ways to contact a server administrator).

Because software applications do not have work places and personal lives, it makes no sense to include the "work" and "home" values of

the TYPE parameter in an application's vCard (see Section 5.6 of [VCARD]).

The following base properties make sense for vCards that represent software applications (this list is not exhaustive, and other properties might be applicable as well):

- o ADR
- o EMAIL
- o FN
- o GEO
- o IMPP
- o KEY
- o KIND
- o LANG
- o LOGO
- o NOTE
- o ORG
- o PHOTO
- o REV
- o SOURCE
- o TEL
- o TZ
- o URL

Although it might be desirable to define a more fine-grained taxonomy of applications (e.g., a KIND of "application" with a subtype of "server" or "IM server"), such a taxonomy is out of scope for this document.

3. Example

The following example of an XMPP server is borrowed from [XEP-0292] and uses the XML representation of vCard described in [VCARDXML].

```
<vcard xmlns="urn:ietf:params:xml:ns:vcard-4.0">
  <fn><text>jabber.org IM service</text></fn>
  <url><uri>http://www.jabber.org/</uri></url>
  <lang>
    <parameters><pref>1</pref></parameters>
    <language-tag>en</language-tag>
  </lang>
  <email><text>xmpp@jabber.org</text></email>
  <impp><uri>xmpp:jabber.org</uri></impp>
  <logo><uri>http://www.jabber.org/images/logo.png</uri></logo>
  <geo><uri>geo:42.25,-91.05</uri></geo>
  <tz><text>America/Chicago</text></tz>
  <source><uri>xmpp:jabber.org?vcard</uri></source>
  <rev><timestamp>19990104T122100Z</timestamp></rev>
  <kind><text>application</text></kind>
</vcard>
```

4. IANA Considerations

The IANA is requested to add "application" to the registry of property values for vCard4. In conformance with Section 10.2.6 of [VCARD], the registration is as follows, where the reference is to RFCXXXX.

Value: application

Purpose: The entity represented by the vCard is a software application (e.g., a server, an online service such as a chatroom, or an automated software bot).

Conformance: This value can be used with the "KIND" property.

Example: See Section 3 of RFCXXXX.

[[NOTE TO RFC EDITOR: Please change XXXX to the number assigned to this specification, and remove this paragraph on publication.]]

5. Security Considerations

Use of vCards to represent physical objects and software applications is not envisioned to introduce security considerations beyond those specified for vCards in [VCARD].

6. Acknowledgements

Thanks to Cyrus Daboo, Barry Leiba, Kepeng Li, and Simon Perreault for their feedback.

7. References

7.1. Normative References

- [VCARD] Perreault, S., "vCard Format Specification", draft-ietf-vcarddav-vcardrev-22 (work in progress), May 2011.

7.2. Informative References

- [RFC4519] Sciberras, A., "Lightweight Directory Access Protocol (LDAP): Schema for User Applications", RFC 4519, June 2006.
- [RFC6120] Saint-Andre, P., "Extensible Messaging and Presence Protocol (XMPP): Core", RFC 6120, March 2011.
- [VCARDXML] Perreault, S., "vCard XML Representation", draft-ietf-vcarddav-vcardxml-11 (work in progress), May 2011.
- [X.200] International Telecommunications Union, "Information Technology - Open Systems Interconnection - Basic Reference Model: The Basic Model", ITU-T Recommendation X.521, ISO Standard 9594-7, February 2001.
- [X.521] International Telecommunications Union, "Information Technology - Open Systems Interconnection - The Directory: Selected Object Classes", ITU-T Recommendation X.200, ISO Standard 7498-1, July 1994.
- [XEP-0292] Saint-Andre, P. and S. Mizzi, "vCard4 over XMPP", XSF XEP 0292, July 2011.

Author's Address

Peter Saint-Andre
Cisco
1899 Wyknoop Street, Suite 600
Denver, CO 80202
USA

Phone: +1-303-308-3282
Email: psaintan@cisco.com

