vCard Format extension : represent vCard extensions defined by the Open
      Mobile Alliance (OMA) Converged Address Book (CAB) group
                draft-ietf-vcarddav-oma-cab-extensions-03

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, 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 December 20, 2012.

Copyright Notice

Table of Contents

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 some 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. A Brief Introduction to the Converged Address Book

The Converged Address Book (CAB) Enabler provides consistent
mechanisms to manage contact information both in user-facing
applications and in support of network-facing activities. At the
core of this enabler is a network-based contact repository in which a
user can store contact information. That information can be
retrieved by any CAB-enabled device. The network-based repository is
also able to provide specific contact information to other users and
to keep their copies up to date whenever the information is changed.

The CAB Enabler provides synchronization of the contact information
available in the user device(s) with the network-based contact
repository.

The CAB Enabler also manages the distribution of a user's own contact
information. In essence, a user fills out a Personal Contact Card,
which includes all the information a user wishes to store about him
or herself.

Because systems that are supporting the CAB Enabler are likely
supporting multiple users, the CAB Enabler also defines a search
paradigm that permits other users to query those systems to locate
information about the available users.

The CAB Enabler supports many different types of information. It,
therefore, has a data model that is flexible and extensible. It
manages traditional types of contact information (such as name,
address, email, phone number, mobile number) as well as new types of
information (such as websites, blogs, presence subscription
references).

1.2. Terminology Used in This Document

Syntax specifications shown here use the augmented Backus-Naur Form
(ABNF) as described in [RFC5234], and are specified as in the base

vcard specification [RFC6350].

2. vCard Extensions : Properties

The following sections define the CAB Properties.

Note:
Some string-value vCard properties are defined herein for which no
specific list of allowed strings is specified. For those properties,
it is intended that de-facto taxonomies might develop. One vCard
can, for example, specify a hobby of "philately", while another uses
"stamp collecting", and a third has "old postage stamps". Usage, not
specification, may lead to a preference over time for a single term.
In general, these are meant to be understood by humans, rather than
to be used for automated categorization that might require standard
terms and registries.

2.1. Property : EXPERTISE

Namespace:

Property name: EXPERTISE

Purpose: To specify a field of expertise for the object that the
     vCard refers to.

Value type: A single text value.

Cardinality: *

Property parameters: LEVEL (possible values : "beginner", "average",
     "expert"), INDEX

Description: This is intended to be a free-form naming of fields of
     expertise, meant for human consumption, and no specific expertise
     fields are defined. See the note at the beginning of Section 2.

   Format definition:

      EXPERTISE-param = LEVEL-param / INDEX-param / language-param /
             pref-param / altid-param / type-param / any-param

      EXPERTISE-value = text

   Examples:

      EXPERTISE;LEVEL=beginner;INDEX=2:chinese literature

      EXPERTISE;INDEX=1;LEVEL=expert:chemistry

2.2. Property : HOBBY

   Namespace:

   Property name: HOBBY

   Purpose: To specify the hobbies of the object that the vCard refers
      to.

   Value type: A single text value.

   Cardinality: *

   Property parameters: LEVEL (possible values : "high", "medium",
      "low"), INDEX

   Description: This is intended to be a free-form naming of hobbies,
      meant for human consumption, and no specific hobbies are defined.
      See the note at the beginning of Section 2.

      A hobby, as opposed to an interest (see Section 2.3) 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.

    Format definition:

        HOBBY-param = LEVEL-param / INDEX-param / language-param / pref-
                param / altid-param / type-param / any-param

        HOBBY-value = text

    Examples:

        HOBBY;INDEX=1;LEVEL=high:reading

        HOBBY;INDEX=2;LEVEL=high:sewing

2.3. Property : INTEREST

    Namespace:

    Property name: INTEREST

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

    Value type: A single text value

    Cardinality: *

    Property parameters: LEVEL (possible values : "high", "medium",
        "low"), INDEX

    Description: This is intended to be a free-form naming of interests,
        meant for human consumption, and no specific interests are
        defined. See the note at the beginning of Section 2.

        An interest, as opposed to a hobby (see Section 2.2) 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.

Format definition:

```
INTEREST-param =  LEVEL-param / INDEX-param / language-param /
        pref-param / altid-param / type-param / any-param

INTEREST-value =  text
```

Examples:

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

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

2.4. Property : ORG-DIRECTORY

Namespace:

Property name: ORG-DIRECTORY

Purpose: To specify a directory of an organization the vCard's
    entity belongs to.

Value type: A single URI value.

Cardinality: *

Property parameters: PREF, INDEX

Description: This is intended to be a URI that can be used to do an
    organization-directory lookup. Presumably, the entity the vCard
    represents would be found in the directory, though that isn't
    required. This might be used to make it easier to find someone's
    co-workers, management chain, and so on, in a company or

organizational directory.

How the lookup is done depends upon the URI scheme, and no
attempt is made here to specify details of the lookup mechanism.
An HTTP URI might, for example, lead to a web form that's
intended for manual lookup in a browser -- thus, this URI might
or might not be useable for automated lookup or searching.

Format definition:

ORG-DIRECTORY-param = pref-param / INDEX-param / language-param
         / pid-param / pref-param / altid-param / type-param /
         any-param

ORG-DIRECTORY-value= uri

Examples:

ORG-DIRECTORY;INDEX=1:http://directory.mycompany.example.com

ORG-DIRECTORY;PREF=1:ldap://ldap.tech.example/
o=Example%20Tech,ou=Engineering

3. vCard extensions : Parameters

The following sections define Parameters used within Properties
definitions.

3.1. Parameter : INDEX

Namespace:

Parameter name: INDEX

Purpose: Used in a multi-valued property to indicate the position of
    this value within the set of values.

Description: When a property is multi-valued, INDEX can be used to
    indicate an ordering or sequence of the values. INDEX values
    MUST be strictly positive. Zero is not allowed.

    Format definition:

        INDEX-param = "INDEX=" INDEX-value

        INDEX-value = integer

    Examples:

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

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

3.2. Parameter : LEVEL

    Namespace:

    Parameter name: LEVEL

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

    Description: Allowable values:

        *   "beginner", "average", "expert" when used with EXPERTISE

        *   "high", "medium", "low" when used with HOBBY or INTEREST

    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 document presents no security considerations beyond those in
   section 9 of the base vcard specification [RFC6350].

5. IANA Considerations

   IANA is requested to add the following entries to the vCard
   Properties registry, defined in [RFC6350] section 10.3.1.

```
   +-------+--------------------------+------------------+
   | Name  |                          |                  |
   | space | Property                 | Reference        |
   +-------+--------------------------+------------------+
   |       | EXPERTISE                | RFCXXXX, sec 2.1 |
   |       | HOBBY                    | RFCXXXX, sec 2.2 |
   |       | INTEREST                 | RFCXXXX, sec 2.3 |
   |       | ORG-URI                  | RFCXXXX, sec 2.4 |
   +-------+--------------------------+------------------+
```

   IANA is requested to add the following entries to the vCard
   Parameters registry, defined in [RFC6350] section 10.3.2.

```
   +-------+--------------------------+------------------+
   | Name  |                          |                  |
   | space | Parameter                | Reference        |
   +-------+--------------------------+------------------+
   |       | INDEX                    | RFCXXXX, sec 3.1 |
   |       | LEVEL                    | RFCXXXX, sec 3.2 |
   +-------+--------------------------+------------------+
```

   IANA is requested to add the following entries to the vCard Values
   registry, defined in [RFC6350] section 10.3.4.

```
   +-----------+-----------+--------------+------------------+
   | Property  | Parameter | Value        | Reference        |
   +-----------+-----------+--------------+------------------+
   | EXPERTISE | LEVEL     | beginner     | RFCXXXX, sec 3.2 |
   | EXPERTISE | LEVEL     | average      | RFCXXXX, sec 3.2 |
   | EXPERTISE | LEVEL     | expert       | RFCXXXX, sec 3.2 |
   | HOBBY     | LEVEL     | high         | RFCXXXX, sec 3.2 |
   | HOBBY     | LEVEL     | medium       | RFCXXXX, sec 3.2 |
   | HOBBY     | LEVEL     | low          | RFCXXXX, sec 3.2 |
   | INTEREST  | LEVEL     | high         | RFCXXXX, sec 3.2 |
   | INTEREST  | LEVEL     | medium       | RFCXXXX, sec 3.2 |
   | INTEREST  | LEVEL     | low          | RFCXXXX, sec 3.2 |
   +-----------+--------------------------+------------------+
```

6. Acknowledgments

Thanks to Simon Perreault, Peter Saint-Andre, Cyrus Daboo, and Chris
Newman for particularly thorough reviews, which led to a much cleaner
submission to the working group.

7. References

7.1. Normative References

[RFC5234]   Crocker, D. and P. Overell, "Augmented BNF for Syntax
            Specifications: ABNF", STD 68, RFC 5234, January 2008.

[RFC6350]   Perreault, S., "vCard Format Specification", RFC 6350,
            August 2011.

7.2. Informative References

[OMA-CAB]   Open Mobile Alliance, "Converged Address Book (CAB)
            Specification", October 2010, <http://
            www.openmobilealliance.org/Technical/release_program/docs/
            CopyrightClick.aspx?pck=CAB&file=V1_0-20101019-C/
            OMA-TS-CAB-V1_0-20101019-C.pdf>.

            Candidate Version 1.0, OMA-TS-CAB-V1_0-20101019-C

[OMA-DS]   Open Mobile Alliance, "Data Synchronization Protocol",
            March 2009, <http://www.openmobilealliance.org/Technical/
            release_program/docs/
            copyrightclick.aspx?pck=DS&file=V1_2_2-20090319-A/
            OMA-TS-DS_Protocol-V1_2_2-20090319-A.pdf>.

            Candidate Version 1.2.2, OMA-TS-DS_Protocol-V1_2_2-
            20090319-A

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.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