This document specifies an XMPP extension for use of the vCard4 XML format in XMPP systems, with the intent of obsoleting the vcard-temp format.
Legal

Copyright

This XMPP Extension Protocol is copyright © 1999 – 2020 by the XMPP Standards Foundation (XSF).

Permissions

Permission is hereby granted, free of charge, to any person obtaining a copy of this specification (the "Specification"), to make use of the Specification without restriction, including without limitation the rights to implement the Specification in a software program, deploy the Specification in a network service, and copy, modify, merge, publish, translate, distribute, sublicense, or sell copies of the Specification, and to permit persons to whom the Specification is furnished to do so, subject to the condition that the foregoing copyright notice and this permission notice shall be included in all copies or substantial portions of the Specification. Unless separate permission is granted, modified works that are redistributed shall not contain misleading information regarding the authors, title, number, or publisher of the Specification, and shall not claim endorsement of the modified works by the authors, any organization or project to which the authors belong, or the XMPP Standards Foundation.

Warranty

## NOTE WELL: This Specification is provided on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. ##

Liability

In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall the XMPP Standards Foundation or any author of this Specification be liable for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising from, out of, or in connection with the Specification or the implementation, deployment, or other use of the Specification (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if the XMPP Standards Foundation or such author has been advised of the possibility of such damages.

Conformance

This XMPP Extension Protocol has been contributed in full conformance with the XSF’s Intellectual Property Rights Policy (a copy of which can be found at <https://xmpp.org/about/xsf/ipr-policy> or obtained by writing to XMPP Standards Foundation, P.O. Box 787, Parker, CO 80134 USA).
11.3.5 TEL .......................................................... 16
11.4 Properties Defined Similarly in vcard-temp, vCard3, and vCard4 .......... 16
11.5 Properties Defined in vCard3 but Removed from vCard4 ............... 17

12 Migration Tools ......................................................... 18
  12.1 Extensible Stylesheet Language Transformation (XSLT) from vcard-temp to vCard4 ........................................... 18
  12.2 Example of vcard-temp Data ....................................... 29
  12.3 Example of vCard4 XML Data .................................... 35

13 Acknowledgements ..................................................... 41
1 Introduction

Since 1999, the Jabber/XMPP community has used an interim, unofficial XML representation of vCard data for personal contacts, called vcard-temp (XEP-0054) \(^1\). Recently, the IETF has upgraded vCard from vCard3 to vCard 4 (RFC 6350 \(^2\)), and at the same time has defined an official XML format for vCard4 (RFC 6351 \(^3\)). This document specifies an XMPP extension for use of the vCard4 XML format in XMPP systems, with the intent of obsoleting the vcard-temp format. Primarily this document defines the encapsulation method itself; secondarily it also defines transport methods and a mapping to the vcard-temp format for migration by clients and servers.

2 Requirements

This specification was designed with the following requirements in mind.

1. Reuse vCard4 as defined in RFC 6350.
2. Reuse the vCard4 XML format as defined in RFC 6351.
3. Ensure that clients and servers can easily migrate from vcard-temp to the new encapsulation format.
4. Notify interested parties of changes to vCard data using standard XMPP extensions, specifically Personal Eventing Protocol (XEP-0163) \(^4\).
5. Support vCards for non-human entities such as XMPP servers and Multi-User Chat (XEP-0045) \(^5\) rooms.

3 Reuse of vCard4

Because there is now an XML namespace for the official vCard format, we can simply re-use that namespace: “urn:ietf:params:xml:ns:vcard-4.0”.
The vCard XML format defined at the IETF specifies that the root element is <vcard/>, where the only defined child element is <vcard/>. For use in XMPP, we specify that the root element shall be <vcard/>, not <vcards/>.

4 Self vCards

This section describes the use of the vCard format for self-publication and retrieval of publicly-accessible information about any entity on an XMPP network, thus fulfilling all the use cases of the old vcard-temp format.

4.1 IQ-Based Publication and Retrieval

As in XEP-0054, the primary method for publishing and retrieving vCards is the XMPP <iq/> stanza. (Although it would have been possible to use Best Practices for Persistent Storage of Public Data via Publish-Subscribe (XEP-0222) for public storage and retrieval, community consensus is that storage via IQ is more backward-compatible with XEP-0054, and that publish-subscribe is more appropriate only for event notifications.)

4.1.1 Retrieval

An XMPP entity retrieves the vCard of another entity (or itself) by sending an IQ-get to the target entity containing a <vcard/> child element (note the lowercase "c"!) qualified by the 'urn:ietf:params:xml:ns:vcard-4.0' namespace.

Listing 1: vCard Retrieval Request

```xml
<iq from='samizzi@cisco.com/foo' id='bx81v356' to='stpeter@jabber.org' type='get'>
  <vcard xmlns='urn:ietf:params:xml:ns:vcard-4.0'/>
</iq>
```

If a vCard exists for the target entity, the responsible entity (e.g., the XMPP server that hosts the account for a bare JID) MUST return the data in an IQ-result:

Listing 2: Server Returns vCard

```xml
<iq from='stpeter@jabber.org' id='bx81v356' to='samizzi@cisco.com/foo' type='result'>
  <vcard xmlns="urn:ietf:params:xml:ns:vcard-4.0">
    <fn><text>Peter Saint-Andre</text></fn>
    <n><surname>Saint-Andre</surname><given>Peter</given><additional/></n>
  </vcard>
</iq>
```

<nickname><text>stpeter</text></nickname>
<nickname><text>psa</text></nickname>
<photo><uri>https://stpeter.im/images/stpeter_oscon.jpg</uri></photo>
<bday><date>1966-08-06</date></bday>
<adr>
  <parameters>
    <type><text>work</text><text>voice</text></type>
    <pref><integer>1</integer></pref>
  </parameters>
  <ext>Suite 600</ext>
  <street>1899 Wynkoop Street</street>
  <locality>Denver</locality>
  <region>CO</region>
  <code>80202</code>
  <country>USA</country>
</adr>
<adr>
  <parameters><type><text>home</text></type></parameters>
  <ext></ext>
  <street></street>
  <locality>Parker</locality>
  <region>CO</region>
  <code>80138</code>
  <country>USA</country>
</adr>
<tel>
  <parameters>
    <type><text>work</text><text>voice</text></type>
    <pref><integer>1</integer></pref>
  </parameters>
  <uri>tel:+1-303-308-3282</uri>
</tel>
<tel>
  <parameters><type><text>work</text><text>fax</text></type></parameters>
  <uri>tel:+1-303-308-3219</uri>
</tel>
<tel>
  <parameters>
    <type><text>cell</text><text>voice</text><text>text</text></type>
  </parameters>
  <uri>tel:+1-720-256-6756</uri>
</tel>
<tel>
  <parameters><type><text>home</text><text>voice</text></type></parameters>
  <uri>tel:+1-303-555-1212</uri>

If no vCard exists, the server MUST return an IQ-result containing an empty `<vcard/>` element.

Listing 3: No vCard (empty element)

```xml
<iq from='stpeter@jabber.org' id='bx81v356' to='samizzi@cisco.com/fo0'></iq>
```
4.1.2 Publication

An XMPP entity publishes or updates its vCard by sending an IQ-set to itself (typically its bare JID), containing a `<vcard/>` child element qualified by the 'urn:ietf:params:xml:ns:vcard-4.0' namespace. The publication request needs to include the entire vCard, not a "diff" against the prior data (if any).

Listing 4: vCard Publication Request

```xml
<iq from='stpeter@jabber.org/squire' id='h3vz319m' to='stpeter@jabber.org' type='set'>
    <vcard xmlns='urn:ietf:params:xml:ns:vcard-4.0'>
        [...]
    </vcard>
</iq>
```

If no error occurs, the responsible entity returns an IQ-result.

Listing 5: Server Acknowledges Publication

```xml
<iq from='stpeter@jabber.org' id='bx81v356' to='stpeter@jabber.org/squire' type='result'/>
```

Note: An entity MAY have authorization to update the vCard of another entity (e.g., a server administrator might have authorization to modify the server’s vCard).

4.2 Event Notifications

Publish-Subscribe (XEP-0060) provides a way to subscribe to events, and Personal Eventing Protocol (XEP-0163) defines a pubsub profile for events associated with instant messaging (IM) accounts. If PEP is supported by an IM server, it can be used to automatically generate event notifications when a user’s vCard is modified.

---

4.2.1 Location

The canonical location for notifications regarding a user’s vCard is a pubsub node whose name is "urn:xmpp:vcard4".

4.2.2 Subscribing to vCard Notifications

Let us imagine that Juliet wishes to receive the updates that Romeo publishes to his vCard. She has two options:

1. Implicitly subscribe by advertising support for "urn:xmpp:vcard4+notify" in her Entity Capabilities (XEP-0115) data. Romeo’s PEP service then automatically sends vCard updates to her when it receives presence from her, until and unless she sends presence of type unavailable or stops advertising an interest in vCard updates. This is in accordance with XEP-0060, section 6.1.

2. Explicitly subscribe by sending a formal subscription request to the "urn:xmpp:vcard4" node at Romeo’s JabberID. Romeo’s PEP service might send her all vCard updates even if she is offline at the time (depending on service policies regarding presence integration).

4.2.3 Receiving a vCard Notification

Because Juliet has sent presence to Romeo including Entity Capabilities data that includes the "urn:xmpp:vcard4+notify" feature, Romeo’s XMPP server will send a PEP notification to Juliet. The notification can include an XMPP message body for backward-compatibility with XMPP clients that are not pubsub-capable. This is in accordance with XEP-0060, second 6.1.7.

Listing 6: Receiving a vCard publication/update

```xml
<message from='romeo@montague.lit' to='juliet@capulet.lit' type='headline'>
  <event xmlns='http://jabber.org/protocol/pubsub#event'>
    <items node='urn:xmpp:vcard4'>
      <item id='current'/>
    </items>
  </event>
</message>
```

Note: There is no payload, because this is a pure notification (the receiver needs to retrieve the vCard using an IQ-get as described earlier).

---

5 Contact vCards

In addition to enabling the publication and retrieval of vCards about any entity on an XMPP network, the vCard format can also be used to store information about an entity’s contacts.

5.1 Format

A contact is simply a vCard about someone else (or something else, in the case of automated entities). If the other person or entity is in the user’s roster [RFC 6121], the vCard SHOULD contain the Jabber ID of the person or entity. This enables a user to store information about the contact outside of the roster, thus obviating the need for changes or extensions to the roster namespace itself (as in Annotations (XEP-0145) [11]).

Listing 7: Contact

```xml
<vcard xmlns="urn:ietf:params:xml:ns:vcard-4.0">
  <fn><text>Samantha Mizzi</text></fn>
  <n>
    <surname>Mizzi</surname>
    <given>Samantha</given>
    <additional></additional>
  </n>
  <nickname><text>Sam</text></nickname>
  <nickname><text>samizzi</text></nickname>
  <geo><uri>geo:39.59,-105.01</uri></geo>
  <org>
    <parameters><type><text>work</text></type></parameters>
    <text>Cisco</text>
  </org>
  <note>
    My co-author on XEP-0292. She’s cool!
  </note>
  <impp>
    <parameters><type><text>work</text></type></parameters>
    <uri>xmpp:samizzi@cisco.com</uri>
  </impp>
</vcard>
```


5.2 Storage

Because contact vCards are private information, they are best stored using Best Practices for Persistent Storage of Private Data via Publish-Subscribe (XEP-0223) ¹². The canonical location is a well-known pubsub node "urn:xmpp:contacts". In accordance with XEP-0223, this node MUST have an access type of "whitelist" by default. When a client stores items at this node, it SHOULD NOT include an ItemID, so that the pubsub service can assign those identifiers.

Listing 8: Storing a Contact vCard

```xml
<iq from='stpeter@stpeter.im/squire' type='set' id='h3vs7163'>
  <pubsub xmlns='http://jabber.org/protocol/pubsub'>
    <publish node='urn:xmpp:contacts'>
      <vcard xmlns='urn:ietf:params:xml:ns:vcard-4.0'>
        <fn><text>Samantha Mizzi</text></fn>
        <n>
          <surname>Mizzi</surname>
          <given>Samantha</given>
          <additional></additional>
        </n>
        <nickname><text>Sam</text></nickname>
        <nickname><text>samizzi</text></nickname>
        <geo><uri>geo:39.59,-105.01</uri></geo>
        <org><parameters><type><text>work</text></type></parameters><text>Cisco</text></org>
        <note><text>My co-author on XEP-0292. She's cool!</text></note>
      </vcard>
    </publish>
  </pubsub>
</iq>
```

When a contact’s vCard is stored in a private node, it is pushed out to all of the user’s resources that have included in their entity capabilities (XEP-0115) data a service discovery feature of ”urn:xmpp:contacts+notify” (in the following example those resources are “squire” and “roundabout”).

Listing 9: Publisher resources receive event notification

```xml
<message from='stpeter@stpeter.im' to='stpeter@stpeter.im/squire' type='headline'
    id='ka92g1b5'>
    <event xmlns='http://jabber.org/protocol/pubsub#event'>
        <items node='urn:xmpp:contacts'>
            <item id='9703CC4E-CF7E-4A86-9E61-2C670235F9CB'>
                <vcard xmlns='urn:ietf:params:xml:ns:vcard-4.0'>
                    <fn><text>Samantha Mizzi</text></fn>
                    <n>
                        <given>Samantha</given>
                        <additional></additional>
                    </n>
                    <nickname><text>Sam</text></nickname>
                    <nickname><text>samizzi</text></nickname>
                    <geo><uri>geo:39.59,-105.01</uri></geo>
                    <org>
                        <parameters><type><text>work</text></type></parameters>
                        <text>Cisco</text>
                    </org>
                    <note>
                        My co-author on XEP-0292. She's_cool!
                    </note>
                </vcard>
            </item>
        </items>
    </event>
</message>
```
6 vCards of Automated Entities

Traditionally, vCards have been used on the XMPP network for entities other than human users, e.g. by XMPP servers and chatrooms. When such automated entities use vCards, it is RECOMMENDED to specify a value of "application" for the vCard4 KIND property RFC 6473. 

---

as illustrated in the following example:

Listing 10: vCard for a Thing

```xml
<iq from='jabber.org' id='yhx51c35' to='samizzi@cisco.com/foo' type='result'>
  <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>
    <kind><text>application</text></kind>
  </vcard>
</iq>
```

7 Determining Support

If an XMPP client or server supports the vCard4 namespace, it MUST advertise that fact in its responses to Service Discovery (XEP-0030) information ("disco#info") requests by returning a feature of "urn:ietf:params:xml:ns:vcard-4.0":

Listing 11: A disco#info query

```xml
<iq type='get' from='stpeter@jabber.org/squire' to='samizzi@cisco.com/foo' id='disco1'>
  <query xmlns='http://jabber.org/protocol/disco#info'/>
</iq>
```

Listing 12: A disco#info response

```xml
<iq type='result' from='samizzi@cisco.com/foo' to='stpeter@jabber.org/squire' id='disco1'>
</iq>
```

In order for an application to determine whether an entity supports this protocol, where possible it SHOULD use the dynamic, presence-based profile of service discovery defined in Entity Capabilities (XEP-0115) 15. However, if an application has not received entity capabilities information from an entity, it SHOULD use explicit service discovery instead.

### 8 Security Considerations

The vCard information published to one's XMPP server is world-readable; therefore, users should exercise due caution when determining what information to include (e.g., street addresses, personal telephone numbers, or email addresses).

### 9 IANA Considerations

This document does not require interaction with the Internet Assigned Numbers Authority (IANA) 16.

### 10 XMPP Registrar Considerations

#### 10.1 Well-Known Service Discovery Nodes

The XMPP Registrar 17 shall include 'urn:xmpp:contact' and 'urn:xmpp:vcard4' in its registry of Nodes for Service Discovery and Publish-Subscribe at <https://xmpp.org/registrar/nodes.html>.

---

16The Internet Assigned Numbers Authority (IANA) is the central coordinator for the assignment of unique parameter values for Internet protocols, such as port numbers and URI schemes. For further information, see <http://www.iana.org/>.
17The XMPP Registrar maintains a list of reserved protocol namespaces as well as registries of parameters used in the context of XMPP extension protocols approved by the XMPP Standards Foundation. For further information, see <https://xmpp.org/registrar/>.
11 Mapping from vcard-temp to vCard4

This section provides a more detailed description of mapping vcard-temp properties to vCard4 properties.

11.1 Properties Defined in vcard-temp but not in vCard3 or vCard4

11.1.1 DESC

The vcard-temp specification defined a <DESC/> element. This element was not part of the vCard3 schema. Mapping the vcard-temp <DESC/> element to the vCard4 NOTE property is appropriate.

Listing 13: Deprecated DESC element

```xml
<DESC>
    More information about me is located on my personal website: https://stpeter.im/
</DESC>
```

Listing 14: NOTE property

```xml
<note>
    <text>
        More information about me is located on my personal website: https://stpeter.im/
    </text>
</note>
```

11.1.2 JABBERID

The vcard-temp specification defined a <JABBERID/> element:

Listing 15: Deprecated JABBERID element

```xml
<JABBERID>stpeter@jabber.org</JABBERID>
```

Although the JABBERID field was not part of the vCard3 schema and was simply hacked into vcard-temp, RFC 4770\(^\text{18}\) defined an IMPP property for instant messaging and presence addresses, which was ported to vCard4. In the vCard4 XML format, the IMPP property for a JabberID would be as follows.

11.1.3 MIDDLE
The vcard-temp specification defined a `<MIDDLE/>` element as the third allowable element within the `<N/>` ("name") element. This element was not part of the vCard3 schema, although the Dawson drafts did contain an `<other/>` element in the third position of child elements within the `<n/>` element. It is appropriate to map the vcard-temp `<MIDDLE/>` element to the vCard4 "Additional Name" part of the "N" structured property value, which in xCard is the `<additional/>` child of the `<n/>` element.

11.2 Properties Defined Incorrectly in vcard-temp
Several of the properties in vcard-temp are defined differently in vCard3. In fact, the definitions even differ from those provisionally made in the so-called "Dawson drafts" from which vcard-temp was supposedly derived (for reference, the last of these is archived at `<http://www.watersprings.org/pub/id/draft-dawson-vcard-xml-dtd-03.txt>`). The reasons for these discrepancies are unknown. However, care must be taken in correctly mapping these properties from vcard-temp to vCard4.

11.2.1 KEY
The DTD in XEP-0054 provided this definition for the KEY element:

```xml
<!ELEMENT KEY ( TYPE ?, CRED )>
```

However, the DTD in the final Dawson draft provided the following definition:

```xml
<!ELEMENT key ( extref | b64bin )>
```

The relevant RelaxNG definition in vCard4 XML is as follows:

```xml
property-key = element key { 
  ( value-uri | value-text ) 
}
```

The source of the spurious `<TYPE/>` and `<CRED/>` elements is unknown. The vcard-temp `<CRED/>` element is mapped to the vCard4 value-text construction.
11.2.2 SOUND
The DTD in XEP-0054 provided this definition for the SOUND element:

```xml
<!ELEMENT SOUND (PHONETIC | BINVAL | EXTVAL)>
```

However, the DTD in the final Dawson draft provided the following definition:

```xml
<!ELEMENT sound (extref | b64bin)>
```

The source of the spurious vcard-temp `<PHONETIC/>` element is unknown. However, it does not exist in vCard4 and therefore is simply discarded when mapping. The vcard-temp `<BINVAL/>` element is mapped to the vCard4 b64bin construction and the vcard-temp `<EXTVAL/>` element is mapped to the vCard4 extref construction.

11.2.3 VERSION
As explained in XEP-0054, the `<VERSION/>` element from the final Dawson draft was not used in vcard-temp; instead, the vcard-temp protocol used a 'version' attribute (in fact the Dawson drafts were inconsistent, since the DTD defined a `<VERSION/>` element and the body of the specification used a 'version' attribute).

11.3 Properties Defined Differently in vcard-temp, vCard3, and vCard4
The following properties are defined differently in vcard-temp and vCard4. As a result, the mappings are workable but might not preserve all information that could have been contained in vcard-temp data.

11.3.1 ADR
The following address type values allowed in vCard3 were removed from vCard4:

- DOM
- INTL
- PARCEL
- POSTAL
11.3.2 AGENT

In vCard3 and vcard-temp, the AGENT property was allowed to contain the inline vCard of someone who could act as an agent for the primary owner of the referenced vCard. In vCard4, inline vCards are disallowed. Therefore only pointers to external vCard objects are now allowed, by means of a URI.

11.3.3 ORG

The ORGUNIT property was removed from vCard4, with the result that the ORGNAME property becomes the only child of ORG.

11.3.4 SORT-STRING

The SORT-STRING property from vCard3 was renamed to SORT-AS in vCard4.

11.3.5 TEL

The following telephony type values allowed in vCard3 were removed from vCard4:

- BBS
- ISDN
- MODEM
- MSG
- PCS

In addition, in vCard4 the telephone number is represented as a tel: URI, not by means of a NUMBER property.

11.4 Properties Defined Similarly in vcard-temp, vCard3, and vCard4

The following properties are defined similarly in vcard-temp, vCard3, and vCard4. The mappings are fairly straightforward (a future version of this document might provide more detailed narrative descriptions of the mappings).

- BDAY
- CATEGORIES
11 MAPPING FROM VCARD-TEMP TO VCARD4

- EMAIL
- FN
- GEO
- LOGO
- N
- NICKNAME
- NOTE
- PHOTO (mapped to a ’data:’ URI in vCard4, see RFC 2397 \(^{19}\))
- PRODID
- REV
- ROLE
- TITLE
- TZ
- UID
- URL

11.5 Properties Defined in vCard3 but Removed from vCard4

The following properties were defined in vCard3 but were removed from vCard4:

- CLASS
- LABEL
- MAILER

There is no mapping from these properties to vCard4.

12 Migration Tools

This section contains three tools that are intended to help developers in migrating from vcard-temp to vCard4 XML:

1. An Extensible Stylesheet Language Transformation (XSLT) script for automatically translating the vcard-temp XML format into the vCard4 XML format.

2. An example of vcard-temp data that uses most of the elements defined in XEP-0054 that can be mapped to vCard4 properties (note that some of these elements were never used in practice).

3. An example of vCard4 XML data showing the transformation of the vcard-temp example using the XSLT stylesheet.

The tools are purely informational and are not a normative part of this specification.

12.1 Extensible Stylesheet Language Transformation (XSLT) from vcard-temp to vCard4

```xml
<?xml version='1.0' encoding='UTF-8'?>
<!--
Copyright (c) 1999 - 2020 XMPP Standards Foundation

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN
```
ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

-->

<!-- Author: stpeter@jabber.org -->
<!-- Version: 0.0.3 -->
<!-- Last Updated: 2012-09-12 -->

<xsl:stylesheet xmlns='urn:ietf:params:xml:ns:vcard-4.0' xmlns:xsl='http://www.w3.org/1999/XSL/Transform' version='1.0'>
<xsl:output method='xml'/>
<xsl:template match='/'>
<vcard>

<!-- BEGIN VARIABLES -->

<!-- in case we need them, count instances of the vcard-temp elements per the DTD in XEP-0054 -->

<xsl:variable name='FN.count' select='count(/vCard/FN)'/>
<xsl:variable name='N.count' select='count(/vCard/N)'/>
<xsl:variable name='NICKNAME.count' select='count(/vCard/NICKNAME)'/>
<xsl:variable name='PHOTO.count' select='count(/vCard/PHOTO)'/>
<xsl:variable name='BDAY.count' select='count(/vCard/BDAY)'/>
<xsl:variable name='ADR.count' select='count(/vCard/ADR)'/>
<xsl:variable name='LABEL.count' select='count(/vCard/LABEL)'/>
<xsl:variable name='TEL.count' select='count(/vCard/TEL)'/>
<xsl:variable name='EMAIL.count' select='count(/vCard/EMAIL)'/>
<xsl:variable name='JABBERID.count' select='count(/vCard/JABBERID)'/>
<xsl:variable name='MAILER.count' select='count(/vCard/MAILER)'/>
<xsl:variable name='TZ.count' select='count(/vCard/TZ)'/>
<xsl:variable name='GEO.count' select='count(/vCard/GEO)'/>
<xsl:variable name='TITLE.count' select='count(/vCard/TITLE)'/>
<xsl:variable name='ROLE.count' select='count(/vCard/ROLE)'/>
<xsl:variable name='LOGO.count' select='count(/vCard/LOGO)'/>
<xsl:variable name='AGENT.count' select='count(/vCard/AGENT)'/>
<xsl:variable name='ORG.count' select='count(/vCard/ORG)'/>

</vcard>
</xsl:template>
</xsl:stylesheet>
<xsl:variable name='CATEGORIES.count' select='count(/vCard/CATEGORIES)'/>
<xsl:variable name='NOTE.count' select='count(/vCard/NOTE)'/>
<xsl:variable name='PRODID.count' select='count(/vCard/PRODID)'/>
<xsl:variable name='REV.count' select='count(/vCard/REV)'/>
<xsl:variable name='SORT-STRING.count' select='count(/vCard/SORT-STRING)'/>
<xsl:variable name='SOUND.count' select='count(/vCard/SOUND)'/>
<xsl:variable name='UID.count' select='count(/vCard/UID)'/>
<xsl:variable name='URL.count' select='count(/vCard/URL)'/>
<xsl:variable name='CLASS.count' select='count(/vCard/CLASS)'/>
<xsl:variable name='KEY.count' select='count(/vCard/KEY)'/>
<xsl:variable name='DESC.count' select='count(/vCard/DESC)'/>

<! -- END VARIABLES -->

<! -- BEGIN VCARD4 OUTPUT -->

<! -- FN is required -->
<br/>&lt;fn&gt;&lt;text&gt;&lt;xsl:value-of select='/vCard/FN'/&gt;&lt;/text&gt;&lt;/fn&gt;

<! -- N is required -->
<br/>&lt;xsl:apply-templates select='/vCard/N'/&gt;

<! -- NICKNAME can be included one or more times -->
<br/>&lt;xsl:for-each select='/vCard/NICKNAME'/&gt;
  &lt;nickname&gt;&lt;text&gt;&lt;xsl:value-of select='.'/&gt;&lt;/text&gt;&lt;/nickname&gt;
&lt;/xsl:for-each&gt;

<! -- PHOTO can be included one or more times -->
<br/>&lt;xsl:for-each select='/vCard/PHOTO'/&gt;
  &lt;xsl:variable name='PHOTO.ext' select='count(EXTVAL)'/&gt;
  &lt;xsl:variable name='PHOTO.type' select='TYPE'/&gt;
  &lt;xsl:choose&gt;
    &lt;xsl:when test='$PHOTO.ext=1'/&gt;
      &lt;photo&gt;&lt;uri&gt;&lt;xsl:value-of select='EXTVAL'/&gt;&lt;/uri&gt;&lt;/photo&gt;
    &lt;/xsl:when&gt;
    &lt;xsl:otherwise&gt;
      &lt;photo&gt;&lt;uri&gt;
        &lt;xsl:text data:'&lt;xsl:value-of select='TYPE'&gt;&lt;/xsl:text&gt;&lt;/uri&gt;&lt;/photo&gt;
      &lt;/xsl:otherwise&gt;
    &lt;/xsl:choose&gt;
  &lt;/xsl:for-each&gt;

<! -- content is either a pointer to a URL or inline binary, which is mapped to a data: URI in vCard4 -->
BINVAL' />
</uri>
</photo>
</xsl:otherwise>
</xsl:choose>
</xsl:for-each>
<!-- BDAY can be included once -->
<xsl:if test='BDAY.count=1'>
  <bday><date><xsl:value-of select='vCard/BDAY'/></date></bday>
</xsl:if>

<!-- ADR can be included one or more times -->
<xsl:apply-templates select='/vCard/ADR'/>

<!-- NOTE: vcard-temp allowed a LABEL element, but it was removed from vCard4 -->
<!-- one or more TEL elements can be included -->
<xsl:apply-templates select='/vCard/TEL'/>

<!-- one or more EMAIL elements can be included -->
<xsl:apply-templates select='/vCard/EMAIL'/>

<!-- JABBERID can be included one or more times -->
<!-- we map this to the vCard4 IMPP property -->
<xsl:for-each select='/vCard/JABBERID'>
</uri></impp>
</xsl:for-each>

<!-- NOTE: vcard-temp allowed a MAILER element, but it was removed from vCard4 -->
<!-- one or more TZ elements can be included -->
<xsl:for-each select='/vCard/TZ'>
  <tz><text><xsl:value-of select='.'/></text></tz>
</xsl:for-each>

<!-- one or more GEO elements can be included -->
<xsl:for-each select='/vCard/GEO'>
</xsl:for-each>

<!-- one or more TITLE elements can be included -->
<xsl:for-each select='/vCard/TITLE'>
<title><xsl:value-of select='.'/></title>
</xsl:for-each>

<!-- one or more ROLE elements can be included -->
<xsl:for-each select='/vCard/ROLE'>
<role><xsl:value-of select='.'/></role>
</xsl:for-each>

<!-- one or more LOGO elements can be included -->
<!-- content is either a pointer to a URL or inline binary, which is mapped to a data: URI in vCard4 -->
<xsl:for-each select='/vCard/LOGO'>
<xsl:variable name='LOGO.ext' select='count(EXTVAL)'/>
<xsl:variable name='LOGO.type' select='TYPE'/>
<xsl:choose>
<xsl:when test='$LOGO.ext=1'>
<logo><uri><xsl:value-of select='EXTVAL'/></uri></logo>
</xsl:when>
<xsl:otherwise>
<logo>
</logo>
</xsl:otherwise>
</xsl:choose>
</xsl:for-each>

<!-- one or more AGENT elements can be included -->
<!-- however, inline vcards are not supported in vCard4 -->
<!-- the relevant vCard4 property is RELATED -->
<!-- only EXTVAL is supported now via URIs -->
<!-- NOTE: this element was probably unused in vcard-temp -->
<xsl:for-each select='/vCard/AGENT'>
<xsl:variable name='AGENT.ext' select='count(EXTVAL)'/>
<xsl:if test='$AGENT.ext=1'>
<agent><uri><xsl:value-of select='EXTVAL'/></uri></agent>
</xsl:if>
</xsl:for-each>
12  MIGRATION TOOLS

</xsl:if>
</xsl:for-each>

<!-- one or more ORG elements can be included -->
<!-- NOTE: ORGUNIT is no longer used in vcard4 -->

<xsl:for-each select="/vCard/ORG">
<org>
  <xsl:variable name="ORGNAME.count" select="count(ORGNAME)"/>
  <xsl:if test="$ORGNAME.count=1">
    <text><xsl:value-of select="'ORGNAME'"/></text>
  </xsl:if>
</org>
</xsl:for-each>

<!-- one or more CATEGORIES elements can be included -->
<!-- NOTE: this element was probably unused in vcard-temp -->

<xsl:for-each select="/vCard/CATEGORIES'">
<categories><text><xsl:value-of select="'.'"></text></categories>
</xsl:for-each>

<!-- one or more NOTE elements can be included -->
<!-- NOTE: this element was probably unused in vcard-temp -->

<xsl:for-each select="/vCard/NOTE'">
<note><text><xsl:value-of select="'.'"></text></note>
</xsl:for-each>

<!-- PRODID can be included exactly once -->
<!-- NOTE: this element was probably unused in vcard-temp -->

<xsl:if test="$PRODID.count=1'">
<prodid><text><xsl:value-of select="'.'"></text></prodid>
</xsl:if>

<!-- REV can be included exactly once -->
<!-- NOTE: this element was probably unused in vcard-temp -->

<xsl:if test="$REV.count=1'">
<rev><timestamp><xsl:value-of select="'.'"></timestamp></rev>
</xsl:if>

<!-- one or more SORT-STRING elements can be included -->
<!-- this element maps to SORT-AS in vCard4 -->

<xsl:for-each select="/vCard/SORT-STRING'">
<sort-as><xsl:value-of select="'.'"></sort-as>
</xsl:for-each>
<xsl:for-each>

<!-- one or more SOUND elements can be included -->

<!-- NOTE: for some reason, vcard-temp allowed a <PHONETIC/> child element, but that was not documented in the original Dawson I-Ds and is not supported in vCard4 -->

<xsl:for-each select='/vCard/SOUND'>
<xsl:variable name='SOUND.ext' select='count(EXTVAL)'/>
<xsl:variable name='SOUND.bin' select='count(BINVAL)'/>
<xsl:choose>
  <xsl:when test='$SOUND.ext=1'>
    <sound><uri><xsl:value-of select='EXTVAL'/></uri></sound>
  </xsl:when>
  <xsl:when test='$SOUND.ext=1'>
    <sound>
      <uri><xsl:text data:audio/basic;base64 ,</xsl:text><xsl:value-of select='BINVAL'/></uri>
    </sound>
  </xsl:when>
  <xsl:otherwise>
  </xsl:otherwise>
</xsl:choose>
</xsl:for-each>

<!-- UID can be included exactly once -->

<xsl:if test='$UID.count=1'>
  <uid><uri><xsl:value-of select='/vCard/UID'/></uri></uid>
</xsl:if>

<!-- URL can be included one or more times -->

<xsl:for-each select='/vCard/URL'>
  <url><uri><xsl:value-of select='.'/></uri></url>
</xsl:for-each>

<!-- NOTE: vcard-temp allowed a CLASS element, but it was removed from vCard4 -->

<!-- KEY can be included one or more times -->

<xsl:for-each select='/vCard/KEY'>
  <key><text><xsl:value-of select='CRED'/></text></key>
</xsl:for-each>

<!-- DESC can be included one or more times -->

<!-- this existed in vcard-temp but not vCard3 -->
<!-- mapped to the NOTE element -->
<xsl:for-each select='/vCard/DESC'>
  <note><text><xsl:value-of select='.'/></text></note>
</xsl:for-each>
</vcard>
</xsl:template>

<xsl:template match='N'>
  <n>
    <xsl:variable name='FAMILY.count' select='count(FAMILY)'/>
    <xsl:variable name='GIVEN.count' select='count(GIVEN)'/>
    <xsl:variable name='MIDDLE.count' select='count(MIDDLE)'/>
    <xsl:variable name='PREFIX.count' select='count(PREFIX)'/>
    <xsl:variable name='SUFFIX.count' select='count(SUFFIX)'/>
    <xsl:if test='$FAMILY.count=1'>
      <surname><xsl:value-of select='FAMILY'/></surname>
    </xsl:if>
    <xsl:if test='$GIVEN.count=1'>
      <given><xsl:value-of select='GIVEN'/></given>
    </xsl:if>
    <xsl:if test='$MIDDLE.count=1'>
      <additional><xsl:value-of select='MIDDLE'/></additional>
    </xsl:if>
    <xsl:if test='$PREFIX.count=1'>
      <prefix><xsl:value-of select='PREFIX'/></prefix>
    </xsl:if>
    <xsl:if test='$SUFFIX.count=1'>
      <suffix><xsl:value-of select='SUFFIX'/></suffix>
    </xsl:if>
  </n>
</xsl:template>

<xsl:template match='ADR'>
  <adr>
    <xsl:variable name='HOME.count' select='count(HOME)'/>
    <xsl:variable name='WORK.count' select='count(WORK)'/>
    <xsl:if test='$HOME.count>0'>
      <home><xsl:value-of select='HOME'/></home>
    </xsl:if>
    <xsl:if test='$WORK.count>0'>
      <work><xsl:value-of select='WORK'/></work>
    </xsl:if>
    <xsl:if test='$HOME.count=0 and $WORK.count=0'>
      <work><xsl:value-of select='HOME'/></work>
    </xsl:if>
    </adr>
</xsl:template>
<xsl:variable name='PCODE.count' select='count(PCODE)'/>

<!-- NOTE: yes, vcard-temp has CTRY, not COUNTRY -->
<xsl:variable name='CTRY.count' select='count(CTRY)'/>

<!-- first we count the number of vCard TYPE parameters -->
<xsl:variable name='TYPE.count' select='$HOME.count+$WORK.count'/>

<!-- now we output all the parameters -->
<xsl:if test='$TYPE.count &gt; 0'>
  <parameters>
    <type>
      <xsl:if test='$HOME.count=1'>
        <text>home</text>
      </xsl:if>
      <xsl:if test='$WORK.count=1'>
        <text>work</text>
      </xsl:if>
    </type>
    <xsl:if test='$PREF.count=1'>
      <pref><integer>1</integer></pref>
    </xsl:if>
  </parameters>
</xsl:if>
<xsl:if test='$POBOX.count=1'>
  <pobox><xsl:value-of select='POBOX'/></pobox>
</xsl:if>
<xsl:if test='$EXTADD.count=1'>
  <ext><xsl:value-of select='EXTADD'/></ext>
</xsl:if>
<xsl:if test='$STREET.count=1'>
  <street><xsl:value-of select='STREET'/></street>
</xsl:if>
<xsl:if test='$LOCALITY.count=1'>
  <locality><xsl:value-of select='LOCALITY'/></locality>
</xsl:if>
<xsl:if test='$REGION.count=1'>
  <region><xsl:value-of select='REGION'/></region>
</xsl:if>
<xsl:if test='$PCODE.count=1'>
  <code><xsl:value-of select='PCODE'/></code>
</xsl:if>
<xsl:if test='$CTRY.count=1'>
  <country><xsl:value-of select='CTRY'/></country>
</xsl:if>
<adr>
</adr>
</xsl:template>

<xsl:template match='TEL'>
<tel>
  <xsl:variable name='HOME.count' select='count(HOME)'/>
  <xsl:variable name='WORK.count' select='count(WORK)'/>
  <xsl:variable name='TEXT.count' select='count(TEXT)'/>
  <xsl:variable name='VOICE.count' select='count(VOICE)'/>
  <xsl:variable name='FAX.count' select='count(FAX)'/>
  <xsl:variable name='CELL.count' select='count(CELL)'/>
  <xsl:variable name='VIDEO.count' select='count(VIDEO)'/>
  <xsl:variable name='PAGER.count' select='count(PAGER)'/>
  <xsl:variable name='TEXTPHONE.count' select='count(TEXTPHONE)'/>
</tel>
</xsl:template>

<!-- NOTE: vcard-temp allowed telephony types of MSG, BBS, MODEM, ISDN, and PCS but they were removed from vCard4 -->

<xsl:variable name='PREF.count' select='count(PREF)'/>
<xsl:variable name='NUMBER.count' select='count(NUMBER)'/>
<!-- first we count the number of vCard TYPE parameters -->
<xsl:variable name='TYPE.count' select='HOME.count + WORK.count + TEXT.count + VOICE.count + FAX.count + CELL.count + VIDEO.count + PAGER.count + TEXTPHONE.count'/>

<!-- now we output all the parameters -->

<xsl:if test='TYPE.count &gt; 0'>
  <parameters>
    <type>
      <xsl:if test='HOME.count=1'>
        <text>home</text>
      </xsl:if>
      <xsl:if test='WORK.count=1'>
        <text>work</text>
      </xsl:if>
      <xsl:if test='TEXT.count=1'>
        <text>text</text>
      </xsl:if>
      <xsl:if test='VOICE.count=1'>
        <text>voice</text>
      </xsl:if>
    </type>
  </parameters>
</xsl:if>
<xsl:if test='$FAX.count=1'>
  <text>fax</text>
</xsl:if>
<xsl:if test='$CELL.count=1'>
  <text>cell</text>
</xsl:if>
<xsl:if test='$VIDEO.count=1'>
  <text>video</text>
</xsl:if>
<xsl:if test='$PAGER.count=1'>
  <text>pager</text>
</xsl:if>
<xsl:if test='$TEXTPHONE.count=1'>
  <text>texphone</text>
</xsl:if>
</type>
<xsl:if test='$PREF.count=1'>
  <pref><integer>1</integer></pref>
</xsl:if>
</parameters>
<xsl:if test='$NUMBER.count=1'>
</xsl:if>
</tel>
</xsl:template>
<xsl:template match='EMAIL'>
  <email>
    <xsl:variable name='HOME.count' select='count(HOME)'/>
    <xsl:variable name='WORK.count' select='count(WORK)'/>

    <!-- NOTE: vcard-temp allowed email types of INTERNET and X400, but they were never in vCard3 -->
    <xsl:variable name='PREF.count' select='count(PREF)' />
    <xsl:variable name='USERID.count' select='count(USERID)' />

    <!-- first we count the number of vCard TYPE parameters -->
    <xsl:variable name='TYPE.count' select='HOME.count+$WORK.count' />

    <!-- now we output all the parameters -->
    <xsl:if test='$TYPE.count&ge;0'>
      <parameters>
        <type>
12.2 Example of vcard-temp Data

```
<vCard>
  <FN>Peter Saint-Andre</FN>
  <N>
    <FAMILY>Saint-Andre</FAMILY>
    <GIVEN>Peter</GIVEN>
    <MIDDLE/>
  </N>
  <NICKNAME>stpeter</NICKNAME>
  <NICKNAME>psa</NICKNAME>
  <PHOTO><EXTVAL>http://stpeter.im/images/stpeter_oscon.jpg</EXTVAL></PHOTO>
  <PHOTO><EXTVAL>http://stpeter_im/images/stpeter_hell.jpg</EXTVAL></PHOTO>
  <BDAY>1966-08-06</BDAY>
  <ADR>
    <WORK/>
    <PREF/>
    <EXTADD>Suite 600</EXTADD>
    <STREET>1899 Wynkoop Street</STREET>
    <LOCALITY>Denver</LOCALITY>
    <REGION>CO</REGION>
    <PCODE>80202</PCODE>
    <CTRY>USA</CTRY>
  </ADR>
  <ADR>
    <HOME/>
```
12 MIGRATION TOOLS
12 MIGRATION TOOLS

af125JIB6qf+4xK0/8
AWFhNHMF8xMNYevFEaVq1lpsiWWozNLZnIK4ckmP0Uk9vf2quLdbPKmq
k0g3AxZyOWPlIoG5jZp2lYYWPyo+RCigEdwR7/nXxolJ+5
TMC0MYcj1loDdCtdTjZcBh2KnOAe
vFePCyszF7KtPbSG3lAkK6icfn2oW3t71BnTIwcYB7ntjPz+
1lP14juxcxtTPHCHaRKuj0Bk+hX+
NYObho1sDdG4h0+31d/MJKAo0d45PFp6j6us70t0jywYCMzux3zE+
KWX7Lc21r5YjMq1Y9ylLmg4y
Pfk0PqazRaxpi1lxquAw6ZArOyNysWmdl1hY3owk9VPrX1EwGyJCyHqaTPmg0+
DJzINRse9A
julIMn7vbHuxrDjuiy0z1NTxaz8jsnj9v8AzvsiA+
vyvTSw8b6TTk7xgwNRL3uq1zqcmM44z0pP+
+1tT1rWwJjnJAcA87YPsDyy9168FE2jX0iXY3Pkk9Gz3oLT/FC+e9vM6smyMVv+
mR2hFnpXnHNG
8rlIzkw5Ygnht01kySEKhcd661T00sl906C2qpcIVGDXSHxRMsPh+51I/anEafE9/
koVafALLnTkp
ToFWpViu9zw2QHL/tSdfyb+dGHyklmdQbQFeuNP0G2uIQ50fKjJ/m4yR+
BpmY7a7t75TGq8cyh1B
HUHtf5fxJIX0FdoCwvGgD19k148H3u+0e11bLRANPicED5mrKPLwKDKTHNl/
0p7ZpjM8xsCd0R9P
7fT3+6mhuI1FJUTK2OMqQb+Nabc+vEw7sja+X16+8VN+
JpLs8R33C20MgRgC2Mb87Rk18FHdHUR
Eqs826ijF3jGjEcNj8ql7QasY41UJuwgDHA+
JR5qGofZFEc7V0U7tbUUyfJGrsrsqtjGA
nBOpFXsCzczvNZP6FzvxvFAiycu1SmqAcE3zzXTWJPr2y0uNuy8MGXbcAOQMC/
KtfiG6x6HBCqX
8sbQPUsCvfihmZ99G8uEYFsqkhedowF+
f1VeHc6bMWwG3nVQYD1rFemqQ0E3TBjgsUpBjnFLv
ClmtlaaiizrgewFYdomci/NZHCsAJAcEgpsyuumi/Wi/
Gy6jShH6i6juRcxGv12tG3TPrTP8AVOnmuJNO
1RIw05FJwz04xrA7mSU7hjFry6ehEWr/swwhlcE8cKv7N0s3x/H+x0
kikksoJu2z7041fzpRHnP5
YzxedGCVMuZbH9NUFpFv0p3qMemelBajMzXMa5q9r1r/S0cUj2zyDwx04+
NKNbnNzePt10xAg+Pj
/wa0RdxQhavT1vypNLMMmzH3JAt9a2wpQ2SeRk1xRaeLNSwi2sCHOUU4IRU/
0i19juyjN0B1h
h6huP808Uc6qs8sG4aG3zAKT/
kvIGTE8RHOUCpKoSJDDqkhsknJ10F1SZqzhkZL1Fe53ExuQ8
55/I1nb/AL/AHG4/8U0g0Lm21l1jIJZfKdPAQCCuxs1L+
YoYnbJ0EmvQJejfe77Gj8bC1YLCc
IOBk5xxkVXw6L4FsmDRMMMSM18B5zvP0Pbri+mCCKJKrYAYxGyrePhv+T0C/
u06rpzk6p9ufbcuW
KsZbLkke9aC3tgq+
DffT3B7UykJySRC5e576AmMtW2nG0eDl44R0pFGtntrTjFBD7l1IQNgxgD/vNK
rtnB0M0XDHfBFPMUdGtI4Md0uyd40FdxlERo0FJdH0Y81nCn2A6tChtrRHYUmJ0p3JpFtMVQ
RqSSedTuFf2qppMWT2ee50SKS4jcmR4Z6RhgnsP5GtyF7cXsK7sAm3sOKQrc+
bcTM37G2nk9M
12 MIGRATION TOOLS

Y4V5wxyOzu04x5aRzuTj6UHvUtoLo3eEFxMwPaF69ce+
ksQyVhQnIrt0dks9DSJW3GWvutM7e4h
1jXzhQp5B8Aw7i1FrL5qOMflZAw+
FNN7e3jV17DrXsQ0qha2aANv0DUokEMq5sIXIDH1bhIHu
qQvKsrSJJhPevACdPZvgsasb2b8GxYVFD0+YcZJlZ/G/
2aYuJylG3C46qT3rGeVAvOTKQjXDKW10T80y
F93f51To5ihiQZAWJ0B/20nqRgWiaQnLsNicDja6mqaa1DuG2/wAqqjeWaqFp0zRHKNR/
eCtei/x
ArKyvVZmEah+9Wvp/gz8Kysp10cIZ/3bUv16r/dWV13Z0x4maF/6wvz/ADN0tB/
eXF8A8jV1ZUCz
aPYxq38EFnUbcfxxyFzWUMQch0LTP4SH+2nX/sn4V1ZQx/VEz7EF19/50
bpf8SPgPzryuA0P43
0vx/+owq7D4D8qlsyczv7jro/9k=
</BINVAL>
</LOGO>
<ORG>
<ORGNAME>XMPP Standards Foundation</ORGNAME>
</ORGUNIT>
</ORG>
<URL>https://stpetert.im/</URL>
<URL>http://www.saint-andre.com/</URL>
<KEY>
<CRED>
-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: GnuPG/MacGPG2 v2.0.18 (Darwin)
mQINBFETdzesBEAC0FOv1N3JzIIXn6cKD475KS9CHDPeYpegcOIPnL5eY1DChEh/
/IwS1757CePtiybNoV9Fs14PNKUknzXQxa6LiVEdAR/LU1hGJKjg+gslp8bqEILh/
g13ecH66HwL59arbQK4777K8miIPBFC6E3A4Lq17+euw6UcLhKgoYkMOjdi/
WrMgKnVpch5yDKpm/z2o0s8Rgp1PuTeCrXXYnjHXLVFNZxy04Uzs7P5u5KV/
fxS57uQisr8pXtyLd6SpT06ShgKBvB15u20rvqXhsJojiQtOFXWznAja55FUO0Rq9/
CkIG5cMOUAT8TNf6v0ktxsawWL1ELDVPQy1m7mtzo+VREG+0xU6AJMo/GHb1W1U/
U7Mf9yCiULMslp/HLrFjouaLQV85wulQ2jquPe3tkK8h15ucx1Acp01QvQa1DQFbe/
ulQLTJTF8YHpdHPYH7/71117ZBKGa0y7u7F7w7J03gUazwdz9fjW7OIk7ATw0

33
11lzmWDn+2MgzbHOGUMX5hsa8eBdieR2QoLdn27Fip7kBMTJ2+GISrfnJTNQ
OQvmj08DXxadxHu2c4QgzmZbkg35n12y9xXn9NczqzrGlr6vZl3LgX6Sbhi5l7
GgWY6CapebPiDb0v7457n9FV0SdR1gQ505qyi3A30P5aKrIp2TNAkuIwA4bQAB
tCZQZXRlicBTYWlucB1bmrRy3ZACb3r4Xm1rnc9BdHB1dGryMlmitP03cCQQAq1A
IwUCURMP0wIwbAqCQ0gHwA1agKcKwQWAbA4bAheAAo0EpGpEXra2p
6bgQAKpxu8cMDOLC4+EgBHi9NWXYvVyboEfvhuYaLkKprhrHzW1Ow+BpryISR
t9qzX1elCVAojaEv6k6d8M5z2kFf1j3y3j17B1W1+yBr7FfXyyZBbXXx49e1n6
9cl9mbr6vFgaEa6t6nPITz9nOuAb9s0mnrG036EwetHAveQvldQ0j17h6c+Cv71Z
Qbgfj9rfZKmhfQ435nCQaKL91a4v4fqI7fupKnx6uQhx3jNlGvnVo3wiJ+f/jF0Ck
51rTrRmiC13kc/bxg32pr7jHEFpXALMhbmuZ2u2ca+TEZeC396mgyXAUCwcdCFwry
beIEb6ptz5i1L3kAvaQ0h/GgCnncmGn0Ma0BatnwlTdz/vklojy7jQbPr0P1UXf
v5491x7FpIrrH0owXp6Wut86fcqH6hMhPzvRtusj2ornKVvY+yG0LsMCMCTxcXRJRG
7a0lYV72t/pjzfGwSaaxolx1DIZ6B+76jrIhiUHiuW0/gv/4n+DN6B1IC2Qj6xxjxj
462Famy72kpaM0ufTbxw0ujHz/K/KP2Fay/41pX7pvV0wRCu4IkLsLJk
PS7eDaaMvuUxtEnF9d/9LqGwU118Bbse98PLMI8sxKcxic3UXMvd911910Hyqa+1bP
3ZedmHuihckgmJrZugEEgckwP/dneyr6M1FMyfaeQINFBEdsBeAABD0sEHPjUmkhRujH97ek87dn5P/Vh/Y/L/hptcgCK40TL/C/kYdkd3HyteMEF061Pmms
/Rq8K37f3U3VODy9SBP6bktgksKSYUtIpkVka09Q9QNPqyUwNf0F+iAjaVmuuda
EVJy7bHF3l0RDwLY51vLeCxtcv+Vv/1+i772d2WpDp+zLj830C4ybKJSjxv8V
CnQf7DnVL1E16NcNHTsZNDZVM1gYPnxwSg8TEPCG7H5L1wGxc3h1eQjnWnpCpA
mQmEPnDwCwcF5uwo2JNECodsWZ1NDbgfawF3ZI1bo+GClujTu941jVg25j1ATWm
9wafcFz0aUP11RgdIBdUbwUwFQLqd40RLZtXnxGmHlymVrc2cx5gxup+1j80
qh1Kgk8xbimWheF99H2kIBhngVMs177NWn16N51j3QuWbhnuGzF0skGNY
slqNQRbd6qfhaKCK7k7b0t8TyTcYSLg102zhHTbgx2UwLthujjqbd1IQ/GRmlr0hI
WkLXRsn1PnWnEYeFpnTsas2cn9jptTDah7PB81Fui9ntf/XDvfpgsAB5
G3CDV7Q2NgbA16gQHLLaMxvZSP635G83mda0TKXQHXDyLjTTKn+WFWU7t4muULt+
0DSvU8jJXHY5MyUWTNG9PWPAHxFncq/n/1QVbwARAQaqB1fQBFBBgAgABJQR
Ew87AhsM0A0E0eOgpJerxa2pqfpg/ApN+TRu2BibGaw1dr3aznSsh48IPuxDh3
udZvrBgbUtZ/A+mei3ZEm/cnmmBw2G1AuqjNlTzcq18TP+EQ9U7fJH7fVsoh
UyvrMfy3t1MA9jgj9ZM0uloevijijmFeNa0ROGdb/ubOT7Qjzi1klno3l080H
wxBHEXEOlOIs1usZajKvIk5bQwKZ6PHRjLw1Wk4o1i6VgLhGgj79zzL3uhML26
63m7imShv2hklTwryR58c2bNoEotzekYerieA1p7yHurusV+q1t32PQ0bMbBA7
py9d11bw+w3LIAUGZE+Z5hB0WOFXytw+w/U/X0BjGJylw3M2W0uuExe+exzZwn
akygKjmL22z4njFjKjNC1VZ3FmxP0gWdA01nc4jfxDh0RnrdsMrBFDFItMptoz
gfDkIrpic9xKZ/25P6nx27oTOHwkAc0QhxSH+huVBBK8RkAgF+zAbD2e4/L+kan
SrycIXW/wCZwBq61aWsz2Qhhkujozvk+hk4DR+CFjzAfnjxywYER3nuXVQaKWTCw
dNcT19Rv98IsNnrw9Y41JEaG6CJNPmID5+EASycqa0uToRSuKr88sQQLWLPyTnzec
a6Gx7a+fntWwK2HuYoSDHJrY1IW/w1M1k0B4m4tweMAo0zcQH6AQV/GzVQKSGq
rLmuVIV/
=11gw
-----END PGP PUBLIC KEY BLOCK-----
</vCard>
12.3 Example of vCard4 XML Data

```xml
<vcard xmlns="urn:ietf:params:xml:ns:vcard-4.0">
  <fn>
    <text>Peter Saint-Andre</text>
  </fn>
  <n>
    <surname>Saint-Andre</surname>
    <given>Peter</given>
    <additional></additional>
  </n>
  <nickname>
    <text>stpeter</text>
  </nickname>
  <nickname>
    <text>psa</text>
  </nickname>
  <photo>
    <uri>http://stpeter.im/images/stpeter_oscon.jpg</uri>
  </photo>
  <photo>
    <uri>http://stpeter.im/images/stpeter_hell.jpg</uri>
  </photo>
  <bday>
    <date>1966-08-06</date>
  </bday>
  <adr>
    <type><text>work</text></type>
    <pref><integer>1</integer></pref>
    <ext>Suite 600</ext>
    <street>1899 Wynkoop Street</street>
    <locality>Denver</locality>
    <region>CO</region>
    <code>80202</code>
    <country>USA</country>
  </adr>
  <adr>
    <type><text>home</text></type>
    <ext>
      <street>Parker</street>
      <locality>Parker</locality>
      <region>CO</region>
      <code>80138</code>
      <country>USA</country>
    </ext>
  </adr>
  <tel>
    <parameters>
    </parameters>
</vcard>
```
<type><text>work</text><text>voice</text></type>
<pref><integer>1</integer></pref>

</parameters>
</uri>tel:+1-303-308-3282</uri>
</tel>

tel
<parameters>
  <type><text>work</text><text>fax</text></type>
</parameters>
</uri>tel:+1-303-308-3219</uri>
</tel>

tel
<parameters>
  <type><text>cell</text><text>voice</text><text>text</text></type>
</parameters>
</uri>tel:+1-720-256-6756</uri>
</tel>

tel
<parameters>
  <type><text>home</text><text>voice</text></type>
</parameters>
</uri>tel:+1-303-555-1212</uri>
</tel>

<email>
  <text>stpeter@jabber.org</text>
</email>

<email>
  <parameters>
    <type><text>work</text></type>
  </parameters>
  <text>psaintan@cisco.com</text>
</email>

<impp>
  <uri>xmpp:stpeter@jabber.org</uri>
</impp>

text>America/Denver</text>
</tz>
<geo>
  <uri>geo:39.59,-105.01</uri>
</geo>
<title>
  <text>Executive Director</text>
</title>
<role>
  <text>Patron Saint</text>
</role>
<logo>
12 MIGRATION TOOLS

MrPwXrEnomp3EynYApYoMkYGOFrfaHuqekansd1HOMTOeOSeLaxjzm1763ayalcBp1W22K&3noCR1
gT3x2xW21v4bosodIRgKHGBgBPTpjPwzKjmhrYrBFNghPSjLYPNK745oGfNrtDnGCCc6+g
grvhhDZsY
rS3LBeN/9YsvbPTi+KK6GwGBucg0T5HljqSPbXMz4/
uFma1brrzk4qisFS3UicMR6g9qL12f53d
FasL00MAV5ktMDIwamLxYFagmWTc+g6OHF+
k0w85UzIfToJhprbsqTKYwyn17V1zMJDp2fzKQw+
VDJqfrqspMw0w8wclfWtYV3kKLz1SaeLakZ5acT6jhZ Iz0Vhj4HtXqOFLuTjaXGTitJxtZSuGjY

EgjHuomQ7kh1HfspVh5h8Gfs+
e50K6juvdCQoAMj0FKgzwse08DXIuxITBYAHYelKdRw85h
uPImyg31T0z61WadpVteeCdOu40Kk8KZFVWd53DHPUDDUdioieSyrSyziJ4jGsaipvBp2pG21Cwz
iLSzRz0zyC16H48kUvH2zRjlTzdOngh+23
gSOgOFzDPmVkgEv4Pr299btN8qw3es8kenw01xjgvs
BaMPKZ0ex+HRQxInWjDQo10m0kLwwkCurt0kk/wDFe3zo/
waP6KqxyW8h05wWIhFXZSj5boeXNO
2UK2bmayaK75xaAPsPC48p3qsvynJEFSVpuIq/
TzsYsCkkD4uNLtL61Czp7XzwR8xWku7LeK
emckk0a8VZgmBYfcg7qaqvB+
1szymjH7031N0x4b3rUmYty2i6F2JApxo8ZWfftC4x4xTSboIE
KPRz7xp89veMsSkomKQRZn9Tkgtz17H1zXW9XysYrv4qKcZQZrH1FjK8LXayfsLob3mJuo
x1qJwUttk3pDzwX8REEkATA96GHH1vq41Gznv8ASpoLMSceuZQ1ihwbbIPB7Vvh0j7PEPtOeY3rj

AHzw9KuA3c3EvMy7aXy3LejFHXGeFnnSOTsbhy+Jmm61jLdhS9e/02
oqTHDdNg3KF6X9D6HptPoJayk
aF125JJBef+4qXK0/8
AWFhNHmF8MyevFEEaVq11ps1WWWozNZniK4ckmP08ukvf2quLdBPkmq
k0igAxz0wP1IoGj5zP2iYWyPyo+RcIgEdwr7/nXosJ5+
TMOfy6cjl01dC10tjZCbh2KnOae
vFExCyS6ZKQToPB3G31ak6Cfn20oWt3tJbNtiwcYB7ntjcz+
1P4i1jucxtPThcHARkJ068k+h+x+
NYObh015b0DNG0/3id/MJKA0d45pFj6u6s7t0j0yYCMzxud3Ze+
KXW7C21r5yMqIY9yLgMg4
Pfk0PqazRaxiP1xuQwAO2AroNysWBMqdlIbhYH30wK9VPx1EvwXJCYHqaTpMG0+
DjzINrSe9A
julMn7vbHuRdjUj0Yz1ZtXa8s9m9v8AzvsiA+
yvuNT5Sw8b7Tkx70vWRL3cUIqZc5qM44z0qP
+1TLt9WWJnJAcA88YPSdD9168F2E2jX0iXy3Pkk9Gz030LT/FC+9v6M6syzMVv+
mRznFnpNhNG
8rilWzk5Ynht01kEyEkCdc6e1T00s19o62qcICVDXtSXHBMsp+51i/anEafE9/
oKvafLLFntk
ToFWpiu9zSwQ2HL/Tsfydb+DGyKlMdQbQFeuNP0G2uIQ0fKjJ/m4yR+
Bpym7a7yt5Qg8cHy1B
HUHts5FxJX0fdoCxxvGqD19k1k48H3u+0e1b1LARNCpcED5mrKPLbkDTHNL/
o7pZJbM8xsC0d9R9P
12 MIGRATION TOOLS

7fT3+6mhuIFJUTK20MqQB+Nabc+Vew7sja+XI6+8VN+
JpLS8R3sC20MrgC2M87Rkf18fKhdHUR
Eq8s26jJf3Jj1cNj8q1K7QasY41JuwgDHA+
Jr5qGoFZEC7QwccU8057tbjUY1YjGrsrstJgA
nBOPkFxScZc6vNZP6FzxvFAlyCu15Qa5eC3zzTXJWJpr2y0uNUy8MGXbcAOQMC/
KtfiGAXr6hBCqX
8sbQPUScVfhmHz9G8uEYFsqqhkeAdowf+
flVeHc6bbMWWG3nVQYDIFreMqQ0eK3TbJgsUqBjnFLv
ClmtlaaizrgewFYdomci/NZHCsAJAcEGpsiuWi/
G6jsMHh61juRcxovGV12tG3TPrTP8AV0nMUJNO
1Riw05FJzw04rxA7mU7chhJDy6heWr/swwHc8EcKv7N0os3X/h+0
kikkSo230z7041fzprRHpN5
YzxedcGVMzuh9NUNFpfvOp3qzqMemelBajMzxMa59r1rm/S0Cujz2YdWX04+
NKnbnNzePToXAg+Pj
/wa0RdxQhATV1ypNLMMmzHj3Ata92wQp2SeRk1rxAeLn5ni2sCHOUUn4IRU/
o119juyJ0BLh
h6hupP8088uc0s8ga4Gz3akT/
kViGTE8RH0UCQxko5DjqkksknJ10f157zqzhKL1Fe53EuxQ8
S5/Iinb/AL/AG14/8U0g01Lmz11JiJFzkDPAXQCCuxs1L+
YoYnb0jHqeQFjeR7gjtbY1KLCv
I0B5xkXVXw6L4fsDMRMMSs18B5zvP0Pfri+mCCKJKrCYAXgYrePhv+TOc/
or6pkrk6pufbCuw
Ks2bLkF9aC1tgq+
DgfTB37uyklySrceE5z76AmAwW2nGOearhjR4Rp6VtnrTfjFB9IqIGxGd/vNk
tbntBM0XDDHfFbPMUgdTLaMd0y40fdXeRhe0fjDhoY8inCn2A6TrChtrH7uMBj0p3JrFtMVQ
RqSSecedUff2qpmWTeeeoSKS4jcMrZlI6zRhgsnsP5T7cXko7sAm3sOKrqc+
bcTm3Ghh2nk9M
8AUHBLExeXEH2Hq7f1v907Lw+IlgFesiqR7hz/
hgK5DSj3GW3xu3uyocUrn8wrvBbwXtzRDB
082RUJ9Mmmtnvo11rumW/aaNGZBG15InbaxK8ZHzr2r52Zk1rrVjbeSvx5E5uEB3jbgZ5q9xa/
wgU0P
HE4k+x1H/ZoZAMD8balk15ljb1I/MVTEncPww8v7omqjj/4c
WphWw5xyDSJ6Cdyvd9Vnk4c1SLiwq
4tywsgC44GkmCf2z9gGz3qxsJozYQFm5K/NI00XYmXV3LbujBqFOKG86VwSu3Cdc0E1/
Pautvs
wHGCc0aJhC51s6h+pu165mCudxdGYXtn00GFCxzSGjxCRgd+rGjkmik02T9n10M1V/
a0mWOgjj
/wcmhe60cgAXOx/bjIY1veSBybnU9QBS2aNI5s7n9c1mdMDSyFZPU9qmbT+
Mhiradp1+0Dc
O+
tBeQ6SckJykd5JNqYt70Myaq2d3y3jry4Uwhx0qyRYkN6j30MmMrEqiuMw2b7RHwCRLAviaQyf
Y4V5wxOPU04x5ERZuTj6UhdvUtoL3eEFxMowP6F9ce+
koQVhvSwNtDdkS9DSJWGW3GVutM7e4h
JxjHDQPL5ZBhW7lifL5yQ0mFzLazw+
fNN5e3jV17DrXs0qoha2anV0uDu0kEMqxS1xIDHIbIHHu
qQvrK5sJRHPEVcaA4PzVgasbs28gx1vFDO+yCzoIzG/
g2aYuYjLG3C46qT3rGeFvAOtKQXDNkw10T0y
13 Acknowledgements

Thanks to Dave Cridland, Todd Herman, Joe Hildebrand, Waqas Hussain, and Matt Miller for their feedback.