

jmap
Internet-Draft
Intended status: Informational
Expires: December 24, 2020

M. Loffredo
IIT-CNR/Registro.it
R. Stepanek
FastMail
June 22, 2020

**JSContact: Converting from and to vCard
draft-loffredo-jmap-jscontact-vcard-03**

Abstract

This document provides informational guidance for converting the contact card defined by JSContact specification, namely JSCard, from and to vCard.

Status of This Memo

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

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

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on December 24, 2020.

Copyright Notice

Copyright (c) 2020 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include 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.	Motivation	3
1.2.	Scope and Caveats	3
1.3.	Conventions Used in This Document	4
2.	Mapping	4
2.1.	General Properties	4
2.1.1.	BEGIN and END	4
2.1.2.	SOURCE	4
2.1.3.	KIND	5
2.1.4.	XML	6
2.2.	Identification Properties	6
2.2.1.	FN	6
2.2.2.	N and NICKNAME	6
2.2.3.	PHOTO	7
2.2.4.	BDAY, BIRTHPLACE, DEATHDATE, DEATHPLACE, ANNIVERSARY	8
2.2.5.	GENDER	10
2.3.	Delivery Addressing Properties	10
2.3.1.	ADR	10
2.4.	Communications Properties	11
2.4.1.	TEL	12
2.4.2.	EMAIL	12
2.4.3.	IMPP	13
2.4.4.	LANG	14
2.5.	Geographical Properties	15
2.6.	Organizational Properties	16
2.6.1.	TITLE	16
2.6.2.	ROLE	16
2.6.3.	LOGO	17
2.6.4.	ORG	18
2.6.5.	MEMBER	19
2.6.6.	RELATED	20
2.6.7.	CONTACT-URI	22
2.7.	Personal Information Properties	22
2.7.1.	EXPERTISE	22
2.7.2.	HOBBY	23
2.7.3.	INTEREST	24
2.7.4.	ORG-DIRECTORY	25
2.8.	Explanatory Properties	26
2.8.1.	CATEGORIES	26
2.8.2.	NOTE	27
2.8.3.	PRODID	28
2.8.4.	REV	28
2.8.5.	SOUND	29
2.8.6.	UID	30
2.8.7.	CLIENTPIDMAP and PID Parameter	30
2.8.8.	URL	30

2.8.9.	VERSION	31
2.9.	Security Properties	31
2.9.1.	KEY	31
2.10.	Calendar Properties	32
2.10.1.	FBURL	32
2.10.2.	CALADRURI	33
2.10.3.	CALURI	34
2.11.	Extended Properties	35
2.12.	JSCard Required Properties	35
2.13.	JSCard Unmatched Properties	36
3.	IANA Considerations	36
4.	Security Considerations	36
5.	References	36
5.1.	Normative References	36
5.2.	Informative References	37
	Authors' Addresses	37

1. Introduction

1.1. Motivation

The JSContact specification [[draft-ietf-jmap-jscontact](#)] has been defined to represent contact card information as a more efficient alternative to vCard [[RFC6350](#)] and its JSON-based version named jCard [[RFC7095](#)].

While new applications might adopt JSContact as their main format to exchange contact card data, they are likely to interoperate with services and clients that just support vCard/jCard. Similarly, existing contact data providers and consumers already using vCard/jCard might want to represent their data also according to the JSContact specification.

To facilitate this use cases, this document provides informational guidance about how to convert the card defined in JSContact, namely JSCard, from and to vCard.

1.2. Scope and Caveats

JSContact and vCard have a lot of semantics in common, however some differences must be outlined:

- o The JSContact data model defines some contact information that doesn't have a direct mapping with vCard elements. In particular, unlike vCard, JSContact distinguishes between a single contact card, named JSCard, and a group of contact cards, named JSCardGroup.

- o The vCard specification includes some features (like parameters) that have been formally removed from JSCard due to a complete refactoring of vCard content. Anyway, the vCard parameters may appear as JSCard features.
- o Some vCard elements represented individually have been mapped onto members of JSCard objects.
- o The vCard custom elements, identified by the prefix "x-", don't have a direct counterpart among the JSContact specification.

1.3. 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](#)].

2. Mapping

This section contains the mapping between vCard and JSCard. The vCard properties are grouped according to the categories defined by [[RFC6350](#)].

Where it is needed, the JCardGroup is taken into account.

In the following of this document, the vCard features, namely properties and parameters, are written in uppercase while the JSCard features are written in camel case.

2.1. General Properties

2.1.1. BEGIN and END

The BEGIN and END properties don't have a direct match with a JSCard feature.

2.1.2. SOURCE

A SOURCE element is represented as a "Resource" object in the "online" array (Figure 1) whose "type" member is set to "uri" and "labels" map contains the entry <"source",true>.

The PREF and MEDIATYPE parameters are mapped onto the "isPreferred" and "mediaType" members respectively.


```
BEGIN:VCARD
VERSION:4.0
...
SOURCE:http://directory.example.com/addressbooks/jdoe/Jean%20Dupont.vcf
...
END:VCARD

{
...
"online":[
...
{
  "type": "uri",
  "labels": { "source": true },
  "value": "http://directory.example.com/addressbooks/jdoe/
Jean%20Dupont.vcf"
},
...
],
...
}
```

Figure 1: SOURCE mapping example

2.1.3. KIND

The KIND element is mapped onto the "kind" member (Figure 2). Allowed values are those, except "group", described in [section 6.1.4 of \[RFC6350\]](#) and extended with the values declared in [\[RFC6473\]](#) and [\[RFC6869\]](#). A group of cards is represented through a JSCardGroup.

```
BEGIN:VCARD
VERSION:4.0
...
KIND:individual
...
END:VCARD

{
...
"kind": "individual",
...
}
```

Figure 2: KIND mapping example

[2.1.4.](#) XML

The XML property doesn't have a direct match with a JSCard feature.

[2.2.](#) Identification Properties

[2.2.1.](#) FN

All the FN elements are globally represented through the "fullName" member. The presence of more than one name is implicitly associated with the full name translations in various languages regardless of the presence of the ALTID parameter. Each translation corresponds to a different entry of the "localizations" map (Figure 3).

[2.2.2.](#) N and NICKNAME

Both the N and NICKNAME elements are converted into equivalent items of the "name" array (Figure 3):

- o the N components are transformed into related "NameComponent" objects as reported in Table 1. Name components SHOULD be ordered such that their values joined by whitespace produce a valid full name of this entity;
- o Each NICKNAME element is mapped onto an object whose "type" member is set to "nickname"

N component	"type" value
Honorific Prefixes	prefix
Given Names	personal
Family Names	surname
Additional Names	additional
Honorific Suffixes	suffix

Table 1: N components mapping


```
BEGIN:VCARD
VERSION:4.0
...
FN:Mr. John Q. Public\, Esq.
N:Public;John;Quinlan;Mr.;Esq.
NICKNAME:Johnny
...
END:VCARD

{
...
"fullName":{
    "value": "Mr. John Q. Public, Esq."
},
"name":[
    { "value":"Mr.", "type": "prefix" },
    { "value":"John", "type": "personal" },
    { "value":"Public", "type": "surname" },
    { "value":"Quinlan", "type": "additional" },
    { "value":"Esq.", "type": "suffix" },
    { "value":"Johnny", "type": "nickname" }
],
...
}
```

Figure 3: FN, N, NICKNAME mapping example

2.2.3. PHOTO

A PHOTO element is represented as a "Resource" object in the "online" array (Figure 4) whose "type" member is set to "uri" and "labels" map contains the entry <"photo",true>.

The PREF and MEDIATYPE parameters are mapped onto the "isPreferred" and "mediaType" members respectively.


```
BEGIN:VCARD
VERSION:4.0
...
PHOTO:http://www.example.com/pub/photos/jqpublic.gif
...
END:VCARD

{
...
"online":[
...
{
  "type": "uri",
  "labels": { "photo": true },
  "value": "http://www.example.com/pub/photos/jqpublic.gif"
},
...
],
...
}
```

Figure 4: PHOTO mapping example

2.2.4. BDAY, BIRTHPLACE, DEATHDATE, DEATHPLACE, ANNIVERSARY

The BDAY and ANNIVERSARY elements and the extensions BIRTHPLACE, DEATHDATE, DEATHPLACE described in [\[RFC6350\]](#) are represented as "Anniversary" objects included in the "anniversaries" array (Figure 5):

- o BDAY and BIRTHPLACE are mapped onto "date" and "place" where "type" is set to "birth"
- o DEATHDATE and DEATHPLACE are mapped onto "date" and "place" where "type" is set to "death"
- o ANNIVERSARY is mapped onto "date" where "type" is set to "other" and "label" is set to a value describing in detail the kind of anniversary (e.g. "marriage date" for the wedding anniversary).

Both birth and death places are represented as instances of the "Address" object. The LANGUAGE parameter values of both BIRTHPLACE and DEATHPLACE elements are represented as corresponding entries of the "fullAddress.localizations" map.


```
BEGIN:VCARD
VERSION:4.0
...
BDAY:19531015T231000Z
BIRTHPLACE:Mail Drop: TNE QB\n123 Main Street\nAny Town, CA
91921-1234\nU.S.A.
DEATHDATE:19960415
DEATHPLACE:4445 Courtright Street\nNew England, ND 58647\nU.S.A.
ANNIVERSARY:19860201
...
END:VCARD

{
...
"anniversaries":[
  {
    "type": "birth",
    "date": "19531015T231000Z",
    "place":
      {
        "fullAddress":
          {
            "value": "Mail Drop: TNE QB\n123 Main
Street\nAny Town, CA 91921-1234\nU.S.A."
          }
      }
  },
  {
    "type": "birth",
    "date": "19531015T231000Z",
    "place":
      {
        "fullAddress":
          {
            "value": "4445 Courtright Street\nNew
England, ND 58647\nU.S.A."
          }
      }
  },
  {
    "type": "other",
    "label": "marriage date",
    "date": "19860201"
  }
],
...
}
```


Figure 5: BDAY, BIRTHPLACE, DEATHDATE, DEATHPLACE, ANNIVERSARY
mapping example

[2.2.5.](#) GENDER

TBD

[2.3.](#) Delivery Addressing Properties

[2.3.1.](#) ADR

An ADR element is represented as an "Address" object in the "addresses" array (Figure 6).

The ADR components are transformed into the "Address" members as reported in Table 2.

ADR component	Address member
p.o. box	postOfficeBox
extended address	extension
street address	street
locality	locality
region	region
postal code	postcode
country name	country

Table 2: ADR components mapping

The LABEL parameter is converted into the "fullAddress" member.

The PREF parameter is converted into the "isPreferred" member.

The GEO parameter is converted into the "coordinates" member.

The TZ parameter is converted into by the "timeZone" member.

The TYPE parameter is converted into the "context" member. The "home" value is replaced with the "private" value.

The LANGUAGE parameter values are represented as different entries of the "fullAddress.localizations" map.

The CC parameter defined by [[RFC8605](#)] is converted into the "countryCode" member.


```
BEGIN:VCARD
VERSION:4.0
...
ADR;TYPE=work;CC=US;;;54321 Oak St;Reston;VA;20190;USA
ADR;TYPE=home;CC=US;;;12345 Elm St;Reston;VA;20190;USA
...
END:VCARD

{
...
"addresses": [
  {
    "context": "work",
    "fullAddress": {
      "value": "54321 Oak St\nReston\nVA\n20190\nUSA"
    },
    "street": "54321 Oak St",
    "locality": "Reston",
    "region": "VA",
    "country": "USA",
    "postcode": "20190",
    "countryCode": "US"
  },
  {
    "context": "private",
    "fullAddress": {
      "value": "12345 Elm St\nReston\nVA\n20190\nUSA"
    },
    "street": "12345 Elm St",
    "locality": "Reston",
    "region": "VA",
    "country": "USA",
    "postcode": "20190",
    "countryCode": "US"
  }
]
...
}
```

Figure 6: ADR mapping example

[2.4.](#) Communications Properties

[2.4.1.](#) TEL

A TEL element is represented as a "Resource" object in the "phones" array (Figure 7). The vCard "type-param-tel" values are mapped onto the "type" member values. Those vCard "type-param-tel" values that don't have a counterpart among the "type" member values are represented as entry keys of the "labels" map with the corresponding entry value set to true. The "type-param" values are mapped onto the "context" member values. The "home" value is replaced with the "private" value.

The PREF parameter is mapped onto the "isPreferred" member.

```
BEGIN:VCARD
VERSION:4.0
...
TEL;VALUE=uri;PREF=1;TYPE="voice,home":tel:+1-555-555-5555;ext=5555
TEL;VALUE=uri;TYPE=home:tel:+33-01-23-45-67
...
END:VCARD

{
  ...
  "phones":[
    {
      "context": "private",
      "type": "voice",
      "value": "tel:+1-555-555-5555;ext=5555",
      "isPreferred": true
    },
    {
      "context": "private",
      "value": "tel:+33-01-23-45-67"
    }
  ],
  ...
}
```

Figure 7: TEL mapping example

[2.4.2.](#) EMAIL

An EMAIL element is represented as a "Resource" object in the "emails" array (Figure 8). The vCard "type-param" values are mapped onto the "context" member values. The "home" value is replaced with the "private" value.

The PREF parameter is mapped onto the "isPreferred" member.


```
BEGIN:VCARD
VERSION:4.0
...
EMAIL;TYPE=work:jpublic@xyz.example.com
EMAIL;PREF=1:jane_doe@example.com
...
END:VCARD

{
  ...
  "emails":[
    {
      "context": "work",
      "value": "jpublic@xyz.example.com",
    },
    {
      "context": "private",
      "value": "jane_doe@example.com"
      "isPreferred": true
    }
  ],
  ...
}
```

Figure 8: EMAIL mapping example

2.4.3. IMPP

An IMPP element is represented as a "Resource" object in the "online" array (Figure 9) whose "type" member is set to "username" and "labels" map contains the entry <"XMPP",true>.

In case of a contact card related to an account on another online service, the entry key SHOULD be the canonical service name, including capitalisation (e.g. "Twitter", "Facebook", "Skype", "GitHub")

The PREF and MEDIATYPE parameters are mapped onto the "isPreferred" and "mediaType" members respectively.


```
BEGIN:VCARD
VERSION:4.0
...
IMPP;PREF=1:xmpp:alice@example.com
...
END:VCARD

{
...
"online":[
...
{
  "type": "username",
  "labels": { "XMPP": true },
  "value": "alice@example.com"
},
...
],
...
}
```

Figure 9: IMPP mapping example

[2.4.4.](#) **LANG**

A LANG element is represented through the "preferredContactLanguages" map (Figure 10): an entry for each language that may be used for contacting the entity associated with the JSCard. The entry keys correspond to the language tags, the corresponding entry values are arrays of "ContactLanguage" objects.

The TYPE and PREF parameters are mapped onto the "ContactLanguage" members "type" and "preference" respectively.

If both PREF and TYPE parameters are missing, the array of "ContactLanguage" objects MUST be empty.


```
BEGIN:VCARD
VERSION:4.0
...
LANG;TYPE=work;PREF=1:en
LANG;TYPE=work;PREF=2:fr
LANG;TYPE=home:fr
...
END:VCARD

{
...
"preferredContactLanguages" : {
    "en": [
        {
            "type": "work",
            "preference": 1
        }
    ],
    "fr": [
        {
            "type": "work",
            "preference": 2
        },
        {
            "type": "home",
            "preference": 1
        }
    ]
  },
...
}
```

Figure 10: LANG mapping example

2.5. Geographical Properties

The GEO and TZ elements are not directly mapped into equivalent topmost JSCard members because the same information is represented through equivalent "Address" members.

The ALTID parameter is used for associating both GEO and TZ elements with the related address information. When the ALTID parameter is missing, the element should be associated with the first contact address.

[2.6.](#) **Organizational Properties**

[2.6.1.](#) **TITLE**

A TITLE element is mapped onto a "LocalizedString" object included in the "jobTitle" array (Figure 11).

The ALTID parameter is used for for associating the language-dependent alternatives with a given element.

The LANGUAGE parameter values are represented as corresponding entries of the "localizations" map.

```
BEGIN:VCARD
VERSION:4.0
...
TITLE:Research Scientist
...
END:VCARD

{
  ...
  "jobTitle":[
    {
      "value": "Research Scientist"
    }
  ],
  ...
}
```

Figure 11: TITLE mapping example

[2.6.2.](#) **ROLE**

A ROLE element is mapped onto a "LocalizedString" object included in the "role" array (Figure 12).

The ALTID parameter is used for for associating the language-dependent alternatives with a given element.

The LANGUAGE parameter values are represented as corresponding entries of the "localizations" map.


```
BEGIN:VCARD
VERSION:4.0
...
ROLE:Project Leader
...
END:VCARD

{
  ...
  "role":[
    {
      "value": "Project Leader"
    }
  ],
  ...
}
```

Figure 12: ROLE mapping example

2.6.3. LOGO

A LOGO element is represented as a "Resource" object in the "online" array (Figure 13) whose "type" member is set to "uri" and "labels" map contains the entry <"logo",true>.

The PREF and MEDIATYPE parameters are mapped onto the "isPreferred" and "mediaType" members respectively.


```
BEGIN:VCARD
VERSION:4.0
...
LOGO:http://www.example.com/pub/logos/abccorp.jpg
...
END:VCARD

{
...
"online":[
...
{
  "type": "uri",
  "labels": { "logo": true },
  "value": "http://www.example.com/pub/logos/abccorp.jpg"
},
...
],
...
}
```

Figure 13: LOGO mapping example

2.6.4. ORG

An ORG element is mapped onto a "LocalizedString" object included in the "organization" array (Figure 14). The organization name includes the organizational units if any.

The ALTID parameter is used for for associating the language-dependent alternatives with a given element.

The LANGUAGE parameter values are represented as corresponding entries of the "localizations" map.


```
BEGIN:VCARD
VERSION:4.0
...
ORG:ABC\, Inc.;North American Division;Marketing
...
END:VCARD

{
...
"organization":[
  {
    "value": "ABC, Inc.;North American Division;Marketing"
  }
],
...
}
```

Figure 14: ORG mapping example

2.6.5. MEMBER

According to the JSContact specification, a group of contact cards is represented through a JSCardGroup (Figure 15). The contact cards composing the group are included in the "cards" array. Therefore, the MEMBER element doesn't have a direct match with a JSCard feature.


```

BEGIN:VCARD
VERSION:4.0
KIND:group
FN:The Doe family
MEMBER:urn:uuid:03a0e51f-d1aa-4385-8a53-e29025acd8af
MEMBER:urn:uuid:b8767877-b4a1-4c70-9acc-505d3819e519
END:VCARD
BEGIN:VCARD
VERSION:4.0
FN:John Doe
UID:urn:uuid:03a0e51f-d1aa-4385-8a53-e29025acd8af
END:VCARD
BEGIN:VCARD
VERSION:4.0
FN:Jane Doe
UID:urn:uuid:b8767877-b4a1-4c70-9acc-505d3819e519
END:VCARD

{
  "uid": "urn:uuid:ab4310aa-fa43-11e9-8f0b-362b9e155667",
  "name": "The Doe family",
  "cards": [
    {
      "name": {
        "fullName": {
          "value": "John Doe"
        }
      },
      "uid": "urn:uuid:03a0e51f-d1aa-4385-8a53-e29025acd8af"
    },
    {
      "name": {
        "fullName": {
          "value": "Jane Doe"
        }
      },
      "uid": "urn:uuid:b8767877-b4a1-4c70-9acc-505d3819e519f"
    }
  ]
}

```

Figure 15: Group example

2.6.6. RELATED

All the RELATED elements are globally converted into the "relatedTo" map (Figure 16): an entry for each entity the entity described by the

JSCard is associated with. The map keys are the "uid" values of the associated cards.

Each map value is a "Relation" object including only the "relation" member represented as a set of relation types described in [section 6.6.6 of \[RFC6350\]](#).

If the relation type is unspecified, the "relation" is empty.

```

BEGIN:VCARD
VERSION:4.0
...
RELATED;TYPE=friend:urn:uuid:f81d4fae-7dec-11d0-a765-00a0c91e6bf6
RELATED;TYPE=contact:http://example.com/directory/jdoe.vcf
RELATED;VALUE=text:Please contact my assistant Jane Doe for any inquiries.
...
END:VCARD

{
...
"relatedTo":{
    {
        "urn:uuid:f81d4fae-7dec-11d0-a765-00a0c91e6bf6":
        {
            "relation": {
                "friend": true
            }
        }
    },
    {
        "http://example.com/directory/jdoe.vcf":
        {
            "relation": {
                "contact": true
            }
        }
    },
    {
        "Please contact my assistant Jane Doe for any inquiries.":
        {
            "relation": { }
        }
    }
}
...
}

```

Figure 16: RELATED mapping example

2.6.7. CONTACT-URI

A CONTACT-URI element defined by [\[RFC8605\]](#) is represented as a "Resource" object of the "online" array (Figure 17) whose "type" member is set to "uri" and "labels" map contains the entry <"contact-uri",true>.

The PREF parameter is mapped onto the "isPreferred" member.

```
BEGIN:VCARD
VERSION:4.0
...
CONTACT-URI;PREF=1:mailto:contact@example.com
...
END:VCARD

{
  ...
  "online":[
    ...
    {
      "type": "uri",
      "labels": { "contact-uri": true },
      "value": "mailto:contact@example.com",
      "isPreferred": true
    },
    ...
  ],
  ...
}
```

Figure 17: CONTACT-URI mapping example

2.7. Personal Information Properties

2.7.1. EXPERTISE

An EXPERTISE element defined by [\[RFC6715\]](#) is represented as a "PersonalInformation" object in the "personalInfo" array (Figure 18). The "type" member is set to "expertise".

The LEVEL parameter is mapped onto the "level" member with following mapping:

- o "beginner" is converted into "low";
- o "average" is converted into "medium";
- o "expert" is converted into "high".

The INDEX parameter is represented as the index of the expertise among the declared expertises.

```
BEGIN:VCARD
VERSION:4.0
...
EXPERTISE;LEVEL=beginner;INDEX=2:chinese literature
EXPERTISE;INDEX=1;LEVEL=expert:chemistry
...
END:VCARD

{
...
"personalInfo":[
...
{
"type": "expertise",
"value": "chemistry",
"level": "high"
},
{
"type": "expertise",
"value": "chinese literature",
"level": "low"
}
...
]
...
}
```

Figure 18: EXPERTISE mapping example

2.7.2. HOBBY

An HOBBY element defined by [\[RFC6715\]](#) is represented as a "PersonalInformation" object in the "personalInfo" array (Figure 19). The "type" member is set to "hobby".

The LEVEL parameter is mapped onto the "level" member with a direct mapping.

The INDEX parameter is represented as the index of the hobby among the declared hobbies.


```
BEGIN:VCARD
VERSION:4.0
...
HOBBY;INDEX=1;LEVEL=high:reading
HOBBY;INDEX=2;LEVEL=high:sewing
...
END:VCARD

{
...
"personalInfo":[
    ...
    {
      "type": "hobby",
      "value": "reading",
      "level": "high"
    },
    {
      "type": "hobby",
      "value": "sewing",
      "level": "high"
    }
    ...
  ]
...
}
```

Figure 19: HOBBY mapping example

2.7.3. INTEREST

An INTEREST element defined by [\[RFC6715\]](#) is represented as a "PersonalInformation" object in the "personalInfo" array (Figure 20). The "type" member is set to "interest".

The LEVEL parameter is mapped onto the "level" member with a direct mapping.

The INDEX parameter is represented as the index of the interest among the declared interests.


```
BEGIN:VCARD
VERSION:4.0
...
INTEREST;INDEX=1;LEVEL=medium:r&b music
INTEREST;INDEX=2;LEVEL=high:rock 'n' roll music
...
END:VCARD

{
  ...
  "personalInfo":[
    ...
    {
      "type": "interest",
      "value": "r&b music",
      "level": "medium"
    },
    {
      "type": "interest",
      "value": "rock 'n' roll music",
      "level": "high"
    }
    ...
  ]
  ...
}
```

Figure 20: INTEREST mapping example

2.7.4. ORG-DIRECTORY

An ORG-DIRECTORY element is represented as a "Resource" object in the "online" array (Figure 21) whose "type" member is set to "uri" and "labels" map contains the entry <"org-directory",true>.

The PREF parameter is mapped onto the "isPreferred" member.

The INDEX parameter is represented as the index of the directory among the online resources with the "org-directory" key in the "labels" map.


```
BEGIN:VCARD
VERSION:4.0
...
ORG-DIRECTORY;INDEX=1:http://directory.mycompany.example.com
ORG-DIRECTORY;PREF=1:ldap://ldap.tech.example/
o=Example%20Tech,ou=Engineering
...
END:VCARD

{
...
"online":[
...
{
  "type": "uri",
  "labels": { "org-directory": true },
  "value": "http://directory.mycompany.example.com"
},
{
  "type": "uri",
  "labels": { "org-directory": true },
  "value": "ldap://ldap.tech.example/
o=Example%20Tech,ou=Engineering",
  "isPreferred": true
},
...
],
...
}
```

Figure 21: ORG-DIRECTORY mapping example

[2.8.](#) Explanatory Properties

[2.8.1.](#) CATEGORIES

A CATEGORIES element is converted into an object in the "categories" array (Figure 22).


```
BEGIN:VCARD
VERSION:4.0
...
CATEGORIES:INTERNET,IETF,INDUSTRY,INFORMATION TECHNOLOGY
...
END:VCARD

{
...
"categories":[
    "INTERNET",
    "IETF",
    "INDUSTRY",
    "INFORMATION TECHNOLOGY"
]
...
}
```

Figure 22: CATEGORIES mapping example

2.8.2. NOTE

A NOTE element is mapped onto a "LocalizedString" object included in the "notes" array (Figure 23).

The ALTID parameter is used for associating the language-dependent alternatives with a given element.

The LANGUAGE parameter values are represented as corresponding entries of the "localizations" map.


```
BEGIN:VCARD
VERSION:4.0
...
NOTE:This fax number is operational 0800 to 1715 EST\, Mon-Fri.
...
END:VCARD

{
...
"notes":[
  {
    "value": "This fax number is operational 0800 to 1715 EST, Mon-
Fri."
  }
]
...
}
```

Figure 23: NOTE mapping example

[2.8.3.](#) **PRODID**

The PRODID element is converted into the "prodId" member (Figure 24).

```
BEGIN:VCARD
VERSION:4.0
...
PRODID:-//ONLINE DIRECTORY//NONSGML Version 1//EN
...
END:VCARD

{
...
"prodId": "-//ONLINE DIRECTORY//NONSGML Version 1//EN"
...
}
```

Figure 24: PRODID mapping example

[2.8.4.](#) **REV**

The REV element is transformed into the "updated" member (Figure 25).


```
BEGIN:VCARD
VERSION:4.0
...
REV:19951031T222710Z
...
END:VCARD

{
...
"updated": "19951031T222710Z"
...
}
```

Figure 25: REV mapping example

2.8.5. SOUND

A SOUND element is represented as a "Resource" object in the "online" array (Figure 26) whose "type" member is set to "uri" and "labels" map contains the entry <"sound",true>.

The PREF and MEDIATYPE parameters are mapped onto the "isPreferred" and "mediaType" members respectively.

```
BEGIN:VCARD
VERSION:4.0
...
SOUND:CID:JOHNQPUBLIC.part8.19960229T080000.xyzMail@example.com
...
END:VCARD

{
...
"online":[
...
  {
    "type": "uri",
    "labels": { "sound": true },
    "value":
"CID:JOHNQPUBLIC.part8.19960229T080000.xyzMail@example.com"
  },
  ...
],
...
}
```

Figure 26: SOUND mapping example

[2.8.6.](#) **UID**

The UID element corresponds to the "uid" member (Figure 27).

```
BEGIN:VCARD
VERSION:4.0
...
UID:urn:uuid:f81d4fae-7dec-11d0-a765-00a0c91e6bf6
...
END:VCARD

{
...
"uid": "urn:uuid:f81d4fae-7dec-11d0-a765-00a0c91e6bf6"
...
}
```

Figure 27: UID mapping example

[2.8.7.](#) **CLIENTPIDMAP and PID Parameter**

TBD

[2.8.8.](#) **URL**

An URL element is represented as a "Resource" object in the "online" array (Figure 28) whose "type" member is set to "uri" and "labels" map contains the entry <"url",true>.

The PREF and MEDIATYPE parameters are mapped onto the "isPreferred" and "mediaType" members respectively.


```
BEGIN:VCARD
VERSION:4.0
...
URL:http://example.org/restaurant.french/~chezchic.html
...
END:VCARD

{
...
"online":[
...
{
  "type": "uri",
  "labels": { "url": true },
  "value": "http://example.org/restaurant.french/~chezchic.html"
},
...
],
...
}
```

Figure 28: URL mapping example

[2.8.9.](#) **VERSION**

The VERSION property doesn't have a direct match with a JSCard feature.

[2.9.](#) **Security Properties**

[2.9.1.](#) **KEY**

A KEY element is represented as a "Resource" object in the "online" array (Figure 29) whose "type" member is set to "uri" and "labels" map contains the entry <"key",true>.

The PREF and MEDIATYPE parameters are mapped onto the "isPreferred" and "mediaType" members respectively.


```
BEGIN:VCARD
VERSION:4.0
...
KEY:http://www.example.com/keys/jdoe.cer
...
END:VCARD

{
...
"online":[
...
{
  "type": "uri",
  "labels": { "key": true },
  "value": "http://www.example.com/keys/jdoe.cer"
},
...
],
...
}
```

Figure 29: KEY mapping example

[2.10.](#) Calendar Properties

[2.10.1.](#) FBURL

A FBURL element is represented as a "Resource" object of the "online" array (Figure 30) whose "type" member is set to "uri" and "labels" map contains the entry <"fburl",true>.

The PREF and MEDIATYPE parameters are mapped onto the "isPreferred" and "mediaType" members respectively.


```
BEGIN:VCARD
VERSION:4.0
...
FBURL;PREF=1:http://www.example.com/busy/janedoe
FBURL;MEDIATYPE=text/calendar:ftp://example.com/busy/project-a.ifb
...
END:VCARD

{
...
"online":[
...
{
  "type": "uri",
  "labels": { "fburl": true },
  "value": "http://www.example.com/busy/janedoe",
  "isPreferred": true
},
{
  "type": "uri",
  "labels": { "fburl": true },
  "value": "ftp://example.com/busy/project-a.ifb",
  "mediaType": "text/calendar"
},
...
],
...
}
```

Figure 30: FBURL mapping example

2.10.2. CALADRURI

A CALADRURI element is represented as a "Resource" object of the "online" array (Figure 31) whose "type" member is set to "uri" and "labels" map contains the entry <"caladruri",true>.

The PREF and MEDIATYPE parameters are mapped onto the "isPreferred" and "mediaType" members respectively.


```
BEGIN:VCARD
VERSION:4.0
...
CALADRURI;PREF=1:mailto:janedoe@example.com
CALADRURI:http://example.com/calendar/jdoe
...
END:VCARD

{
  ...
  "online":[
    ...
    {
      "type": "uri",
      "labels": { "caladruri": true },
      "value": "mailto:janedoe@example.com",
      "isPreferred": true
    },
    {
      "type": "uri",
      "labels": { "caladruri": true },
      "value": "http://example.com/calendar/jdoe"
    },
    ...
  ],
  ...
}
```

Figure 31: CALADRURI mapping example

2.10.3. CALURI

A CALURI element is represented as a "Resource" object of the "online" array (Figure 32) whose "type" member is set to "uri" and "labels" map contains the entry <"caluri",true>.

The PREF and MEDIATYPE parameters are mapped onto the "isPreferred" and "mediaType" members respectively.


```
BEGIN:VCARD
VERSION:4.0
...
CALURI;PREF=1:http://cal.example.com/calA
CALURI;MEDIATYPE=text/calendar:ftp://ftp.example.com/calA.ics
...
END:VCARD

{
...
"online":[
...
{
  "type": "uri",
  "labels": { "caluri": true },
  "value": "http://cal.example.com/calA",
  "isPreferred": true
},
{
  "type": "uri",
  "labels": { "caluri": true },
  "value": "ftp://ftp.example.com/calA.ics",
  "mediaType": "text/calendar"
},
...
],
...
}
```

Figure 32: CALURI mapping example

2.11. Extended Properties

If an extended property is a resource, JSCard already allows to represent it by setting the "type" member to "other" and specifying a value for the "labels" map.

Any other property supporting a custom feature MAY be added and its name MUST be prefixed with a specific domain name to avoid conflict, e.g. "example.com/customprop".

2.12. JSCard Required Properties

While converting a vCard into a JSCard, only the topmost "uid" member is required.

2.13. JSCard Unmatched Properties

The "preferredContactMethod" member doesn't match any vCard element.

3. IANA Considerations

This document has no actions for IANA.

4. Security Considerations

This document doesn't report any security consideration.

5. References

5.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC6350] Perreault, S., "vCard Format Specification", [RFC 6350](#), DOI 10.17487/RFC6350, August 2011, <<https://www.rfc-editor.org/info/rfc6350>>.
- [RFC6473] Saint-Andre, P., "vCard KIND:application", [RFC 6473](#), DOI 10.17487/RFC6473, December 2011, <<https://www.rfc-editor.org/info/rfc6473>>.
- [RFC6474] Li, K. and B. Leiba, "vCard Format Extensions: Place of Birth, Place and Date of Death", [RFC 6474](#), DOI 10.17487/RFC6474, December 2011, <<https://www.rfc-editor.org/info/rfc6474>>.
- [RFC6715] Cauchie, D., Leiba, B., and K. Li, "vCard Format Extensions: Representing vCard Extensions Defined by the Open Mobile Alliance (OMA) Converged Address Book (CAB) Group", [RFC 6715](#), DOI 10.17487/RFC6715, August 2012, <<https://www.rfc-editor.org/info/rfc6715>>.
- [RFC6869] Salgueiro, G., Clarke, J., and P. Saint-Andre, "vCard KIND:device", [RFC 6869](#), DOI 10.17487/RFC6869, February 2013, <<https://www.rfc-editor.org/info/rfc6869>>.
- [RFC7095] Kewisch, P., "jCard: The JSON Format for vCard", [RFC 7095](#), DOI 10.17487/RFC7095, January 2014, <<https://www.rfc-editor.org/info/rfc7095>>.

[RFC8605] Hollenbeck, S. and R. Carney, "vCard Format Extensions: ICANN Extensions for the Registration Data Access Protocol (RDAP)", [RFC 8605](https://www.rfc-editor.org/info/rfc8605), DOI 10.17487/RFC8605, May 2019, <<https://www.rfc-editor.org/info/rfc8605>>.

5.2. Informative References

[[draft-ietf-jmap-jscontact](#)]
"JSContact: A JSON representation of contact data",
<<https://datatracker.ietf.org/doc/draft-ietf-jmap-jscontact/>>.

Authors' Addresses

Mario Loffredo
IIT-CNR/Registro.it
Via Moruzzi,1
Pisa 56124
IT

Email: mario.loffredo@iit.cnr.it
URI: <http://www.iit.cnr.it>

Robert Stepanek
FastMail
PO Box 234, Collins St West
Melbourne VIC 8007
AU

Email: rsto@fastmailteam.com
URI: <https://www.fastmail.com>

