

Workgroup: Calendaring Extensions  
Internet-Draft:  
draft-ietf-calext-vcard-jscontact-  
extensions-03

Published: 10 January 2023

Intended Status: Standards Track

Expires: 14 July 2023

Authors: R. Stepanek    M. Loffredo  
          Fastmail        IIT-CNR

## **vCard Format Extension for JSContact**

### **Abstract**

This document defines a set of new properties for vCard and extends the use of existing ones. Their primary purpose is to align the same set of features between the JSContact and vCard formats, but the new definitions also aim to be useful within just the vCard format.

### **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 14 July 2023.

### **Copyright Notice**

Copyright (c) 2023 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 Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

## Table of Contents

- [1. Introduction](#)
  - [1.1. Notational Conventions](#)
  - [1.2. ABNF Notations](#)
- [2. New Properties](#)
  - [2.1. CONTACT-CHANNEL-PREF Property](#)
  - [2.2. CREATED Property](#)
  - [2.3. GRAMMATICAL-GENDER Property](#)
  - [2.4. LOCALE Property](#)
  - [2.5. PRONOUNS Property](#)
  - [2.6. SOCIALPROFILE Property](#)
- [3. New Parameters](#)
  - [3.1. AUTHOR Parameter](#)
  - [3.2. AUTHOR-NAME Parameter](#)
  - [3.3. CREATED Parameter](#)
  - [3.4. DERIVED Parameter](#)
  - [3.5. RANKS Parameter](#)
  - [3.6. PROP-ID Parameter](#)
  - [3.7. SERVICE-TYPE Parameter](#)
- [4. Security Considerations](#)
- [5. IANA Considerations](#)
- [6. References](#)
  - [6.1. Normative References](#)
  - [6.2. Informative References](#)
- [Authors' Addresses](#)

## 1. Introduction

The JSContact [[I-D.ietf-calext-jscontact](#)] format aims to be an alternative to the vCard [[RFC6350](#)] format for representation of contact and address book data. As such, it introduces new semantics that are not covered in the current definition of vCard and its various extensions. Converting contact data between the two formats is defined in [[I-D.ietf-calext-jscontact-vcard](#)] with the goal of not losing any semantics during conversion. In order to do so, this document defines a new set of properties for vCard and extends existing definitions.

### 1.1. Notational Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [[RFC2119](#)] [[RFC8174](#)] when, and only when, they appear in all capitals, as shown here.

## 1.2. ABNF Notations

The ABNF definitions in this document use the notations of [RFC5234]. ABNF rules not defined in this document either are defined in [RFC5234] (such as the ABNF for CRLF, WSP, DQUOTE, VCHAR, ALPHA, and DIGIT) or [RFC6350].

## 2. New Properties

### 2.1. CONTACT-CHANNEL-PREF Property

**Property name:** CONTACT-CHANNEL-PREF

**Purpose:** This property defines which channels the entity associated with this vCard prefers to be contacted with.

**Value type:** A single text value, restricted to an enumerated list of allowed values.

**Cardinality:** \*

**Property parameters:** The PREF parameter defines the preference of this contact channel in relation to other contact channels. The TYPE parameter defines when to use this preference.

**Description:** This defines an order of preferences for contact channels. The contact channels supported in this specification are phone numbers, email, instant messaging and postal delivery. They are identified by the name of the respective vCard property.

**Format definition:** This property is defined by the following notation:

```
contact-channel-pref = "CONTACT-CHANNEL-PREF"
                      *( ";" contact-channel-pref-param )
                      ":" contact-channel-pref-value CRLF

contact-channel-pref-param = type-param / pref-param / any-param

contact-channel-pref-value = "ADR" / "EMAIL" / "IMPP" / "TEL" /
                             x-value
```

**Example(s):**  
 CONTACT-CHANNEL-PREF;PREF=1:EMAIL  
 CONTACT-CHANNEL-PREF;PREF=2:TEL

## 2.2. CREATED Property

**Property name:** CREATED

**Purpose:** This property defines the date and time when the vCard was created

**Value type:** A single timestamp value.

**Cardinality:** \*1

**Property parameters:** VALUE

**Description:** This is the time stamp when the vCard was created. Copying the vCard across systems does not count as a new creation, nor does a new revision. Instead, the time stamp value typically stays unchanged for the existence of the vCard.

**Format definition:** This property is defined by the following notation:

```
created          = "CREATED" createdparam
                  ":" timestamp CRLF

createdparam     = *(
                    ;
                    ; The following is OPTIONAL,
                    ; but MUST NOT occur more than once.
                    ;
                    (";" "VALUE" "=" "TIMESTAMP") /
                    ;
                    ; The following is OPTIONAL,
                    ; and MAY occur more than once.
                    ;
                    (";" any-param)
                    ;
                    )
```

**Example:**  
CREATED:20220705093412Z  
CREATED;VALUE=TIMESTAMP:20211022T140000-05

## 2.3. GRAMMATICAL-GENDER Property

**Property name:** GRAMMATICAL-GENDER

**Purpose:** This property defines the grammatical gender that shall be used to address the entity represented by this vCard.

**Value type:** A single text value, restricted to an enumerated list of allowed values.

**Cardinality:**

\*

**Property parameters:** LANG

**Description:** This property defines the grammatical gender that the contact prefers to be addressed by or referred at. Multiple occurrences of this property are allowed, but **SHOULD** be distinguished by the LANG parameter.

Many human languages use grammatical genders in salutations and other language constructs. For example, the German language typically distinguishes between the grammatical gender of the recipient in salutations. The allowed values for this property aim to cover grammatical genders for the majority of human languages. This specification registers an [initial list](#) ([Table 3](#)) of allowed values at the IANA vCard Property Values Registry, but this may be extended later.

**Format definition:** This property is defined by the following notation:

```
gram-gender      = "GRAMMATICAL-GENDER" gram-gender-param
                  ":" gram-gender-value CRLF
```

```
gram-gender-param =
    *(
        ;
        ; The following is OPTIONAL,
        ; but MUST NOT occur more than once.
        ;
        (";" language-param) /
        ;
        ; The following is OPTIONAL,
        ; and MAY occur more than once.
        ;
        (";" any-param)
        ;
    )
```

```
gram-gender-value = "animate" /
                    "female" /
                    "inanimate" /
                    "male" /
                    "neuter" /
                    iana-token
                    x-value
```

**Example:** GRAMMATICAL-GENDER:neuter

## 2.4. LOCALE Property

**Property name:** LOCALE

**Purpose:** This property defines the default locale that human-readable text values in this vCard should be assumed written in.

**Value type:** A single Language-Tag value as defined in [Section 4](#) of [\[RFC6350\]](#).

**Cardinality:** \*1

**Property parameters:** The LANG parameter **MUST NOT** be assigned to this property.

**Description:** This property defines the locale in which property values of type TEXT shall be assumed to be written for this vCard. If a vCard property includes the LANG parameter, then the parameter value has higher precedence than the LOCALE property value.

**Format definition:** This property is defined by the following notation:

```
locale          = "LOCALE" any-param ":" Language-Tag CRLF
                  ; Language-Tag is defined in RFC6350, Section 4.
```

**Example:** de-AT

## 2.5. PRONOUNS Property

**Property name:** PRONOUNS

**Purpose:** This property defines the pronouns that shall be used to refer to the entity represented by this vCard.

**Value type:** A single text value.

**Cardinality:** \*

**Property parameters:** LANG, PREF, TYPE

**Description:** This property contains the pronouns that the contact chooses to use for themselves. The value is free-form text. These pronouns shall be used when addressing or referring to the contact. Multiple occurrences of this property **MAY** define pronouns for multiple languages, preferences and contexts.

**Format definition:** This property is defined by the following notation:

```

pronouns          = "PRONOUNS" pronouns-param ":" text CRLF

pronouns-param =
    *(
        ;
        ; The following is OPTIONAL,
        ; but MUST NOT occur more than once.
        ;
        (";" language-param / pref-param / type-param) /
        ;
        ; The following is OPTIONAL,
        ; and MAY occur more than once.
        ;
        (";" any-param)
        ;
    )

```

**Example:** PRONOUNS;LANG=en:they/them

## 2.6. SOCIALPROFILE Property

**Property name:** SOCIALPROFILE

**Purpose:** To specify the URI or username for social media profiles associated with the object the vCard represents.

**Value type:** A single URI or TEXT value. The default value type is URI.

**Cardinality:** \*

**Property parameters:** The SERVICE-TYPE parameter **SHOULD** be assigned to this property if the value type is URI. It **MUST** be assigned if the value type is TEXT. In either case, it **MUST NOT** be assigned more than once.

**Description:** Several vCard address book implementations currently use an experimental X-SOCIALPROFILE property to store social media profiles for contacts. This specification provides an IANA-registered property for the same purpose. In addition to the typical use of this property with URI values, it also allows to set usernames for social media services as free-text TEXT values, in which case the service name **MUST** be provided as a parameter.

**Format definition:** This property is defined by the following notation:

```
socialpr      = "SOCIALPROFILE" socialpr-param ":"  
                socialpr-value CRLF
```

```
socialpr-param = "VALUE=uri" / "VALUE=text" /  
                service-type-param / any-param
```

```
socialpr-value = URI / text
```

**Example 1:**  
SOCIALPROFILE;SERVICE-TYPE=Twitter:https://twitter.com/ietf  
SOCIALPROFILE:https://github.com/github  
SOCIALPROFILE;SERVICE-TYPE=SomeSite;VALUE=TEXT:peter94

### 3. New Parameters

#### 3.1. AUTHOR Parameter

**Parameter name:** AUTHOR

**Purpose:** This parameter identifies the author of the associated property value.

**Description:** This parameter **MAY** be set on any property where conveying authorship is desired. It identifies the author as a URI [[RFC3986](#)]. Since every valid URI includes the COLON (U+003A) character, the parameter value **MUST** be quoted. Note that as an alternative or in addition to this parameter, the AUTHOR-NAME parameter allows to name an author as free-text value (see [Section 3.2](#)).

**Format definition:** = "AUTHOR" "=" DQUOTE 1\*QSAFE-CHAR DQUOTE  
; a valid URI from Section 3 of [[RFC3986](#)]

**Example 1:** AUTHOR="mailto:john@example.com":This is some note.

#### 3.2. AUTHOR-NAME Parameter

**Parameter name:** AUTHOR-NAME

**Purpose:** This parameter names the author of the associated property value.

**Description:** This parameter **MAY** be set on any property where conveying authorship is desired. It names the author as a free-text value. The parameter value **MUST NOT** be empty. Implementations **MUST** take care to quote the name part, if otherwise the part would not be a valid param-value (see [Section 3.3](#) of [[RFC6350](#)]). Note that as an alternative or in addition to this parameter, the AUTHOR parameter allows to identify an author by URI (see [Section 3.1](#)).

**Format definition:**

author-name-param = "AUTHOR-NAME" "=" param-value ; not empty

**Example (A):** AUTHOR-NAME=John Doe:This is some note.

NOTE;AUTHOR-NAME="\_:133tHckr:\_":A note by an unusual author name.

### 3.3. CREATED Parameter

**Parameter name:** CREATED

**Purpose:** This parameter defines the date and time when a property was created in a vCard.

**Description:** This parameter **MAY** be set on any property to define the point in time when the property was created. The value **MUST** be a valid **TIMESTAMP** value as defined in [Section 4.3.5](#) of [\[RFC6350\]](#). Generally, updating a property value **SHOULD NOT** change the creation timestamp. It is up to implementations to decide if a property change resembles an update or rather a delete and create.

**Format definition:** = "CREATED" "=" param-value ;  
; a valid **TIMESTAMP** of Section 4.3.5 of [\[RFC6350\]](#)

**Example (S):** CREATED=20221122T151823Z:This is some note.

### 3.4. DERIVED Parameter

**Parameter name:** DERIVED

**Purpose:** This parameter specifies that the value of the associated property is derived from some other property value or values.

**Description:** This property parameter **MAY** be specified on any property when the value is derived from some other property or properties. When present with a value of true, clients **MUST NOT** update the property.

For an example, an implementation may derive the value of the FN property from the name components of the N property. It indicates this by setting the DERIVED parameter on the FN property to true.

**Format definition:** = "DERIVED" "=" ("true" / "false")  
; Default is false

**Example (S):** FN;Quinlan;Mr.;

FN;DERIVED=TRUE:Mr. John Quinlan

### 3.5. RANKS Parameter

**Parameter name:**

RANKS

**Purpose:** This parameter specifies a ranking among the name components of a vCard N property value.

**Description:** The RANKS parameter on a N property assigns a rank among the same-typed name components of a N property value. Some cultures assign ranks to name components, such as a *first* and a *second* surname, or people might prefer to go by their second given name. But the N property value does not allow to infer a culturally or otherwise significant rank from the order in which same-typed name components are stored in it.

The parameter value is structurally equivalent to the multi-valued N property value: ranks of different name component types are separated by semi-colon, ranks among the same name component types are separated by comma. The rank is an integer number larger than or equal to 1 and indicates the first or nth rank. Its location within the RANKS parameter value ranks the name component value at that same position within the N property value. An empty or absent rank indicates that the rank of its related name component value is undefined. If the RANKS parameter is set, its value **MUST NOT** be the empty string.

**Formal definition:** = %x31-39 ; "1" to "9"

```
ranks-param      = ranks-component 4("; " ranks-component)
ranks-component  = rank *(", " rank)
rank              = (NONZERO-DIGIT *DIGIT) / ""
```

**Example:**  
N;RANKS="2,1":Gómez,Rodriguez;Pablo;;;  
; The culturally "first" surname is Rodriguez but it  
; appears at the second position in the N property.

```
N;RANKS="1,1,1":Hamilton,Cartland;Mary,Barbara;;Dame;
; The writer Dame Mary Barbara Hamilton Cartland
; published as "Barbara Cartland"
```

### 3.6. PROP-ID Parameter

**Parameter name:** PROP-ID

**Purpose:** This parameter identifies a property among all its siblings of the same type.

**Description:** This parameter uniquely identifies a property among all of its siblings with the same name within a vCard. A valid PROP-ID value must be of 1 and a maximum of 255 octets in size,

and it **MUST** only contain the ASCII alphanumeric characters (A-Za-z0-9), hyphen (-), and underscore (\_). The identifier only has the purpose to uniquely identify siblings, its value has no other meaning. If an application makes use of PROP-ID it **SHOULD** assign a unique identifier to each sibling property of the same name within their embedding component. The same identifier **MAY** be used for properties of a different name, and it **MAY** also be assigned to a same-named property that is not a sibling.

Resolving duplicate identifier conflicts is specific to the application. Similarly, handling properties where some but not all siblings have a PROP-ID is assigned, is application-specific.

**Format definition:** "PROP-ID" "=" 1\*255(ALPHA / DIGIT / "-" / "\_")

**Example:** PROP-ID=p827:<...remainder of base64-encoded data...>

### 3.7. SERVICE-TYPE Parameter

**Parameter name:** SERVICE-TYPE

**Purpose:** To define the online service name associated with a messaging or social media profile.

**Description:** This parameter **MAY** be specified on a IMPP or SOCIALPROFILE property to name the online service associated with that property value. Its value is case-sensitive, its letter cases **MUST** be preserved.

Several vCard address book implementations currently use an experimental X-SERVICE-TYPE parameter. This specification provides an IANA-registered parameter for the same purpose.

**Format definition:** param = param-value

**Example:** SOCIALPROFILE;SERVICE-TYPE=Twitter:https://twitter.com/ietf

## 4. Security Considerations

This specification extends the vCard Format Specification. The same security considerations as outlined in [Section 9](#) of [\[RFC6350\]](#) apply.

## 5. IANA Considerations

IANA is requested to add the following entries to the "vCard Properties" registry, defined in Section 10.3.1. of [\[RFC6350\]](#).

Namespace	Property	Reference
	CONTACT-CHANNEL-PREF	This document, <a href="#">Section 2.1</a>
	CREATED	This document, <a href="#">Section 2.2</a>
	GRAMMATICAL-GENDER	This document, <a href="#">Section 2.3</a>
	LOCALE	This document, <a href="#">Section 2.4</a>
	PRONOUNS	This document, <a href="#">Section 2.5</a>
	SOCIALPROFILE	This document, <a href="#">Section 2.6</a>

Table 1: New VCARD Properties

IANA is requested to add the following entries to the "vCard Parameters" registry, defined in Section 10.3.2. of [[RFC6350](#)].

Namespace	Parameter	Reference
	AUTHOR	This document, <a href="#">Section 3.1</a>
	AUTHOR-NAME	This document, <a href="#">Section 3.2</a>
	CREATED	This document, <a href="#">Section 3.3</a>
	DERIVED	This document, <a href="#">Section 3.4</a>
	PROP-ID	This document, <a href="#">Section 3.6</a>
	RANKS	This document, <a href="#">Section 3.5</a>
	SERVICE-TYPE	This document, <a href="#">Section 3.7</a>

Table 2: New VCARD Parameters

IANA is requested to add the following entries to the "vCard Property Values" registry, defined in Section 10.3.4. of [[RFC6350](#)].

Property	Value	Reference
GRAMMATICAL-GENDER	animate	This document, <a href="#">Section 2.3</a>
GRAMMATICAL-GENDER	female	This document, <a href="#">Section 2.3</a>
GRAMMATICAL-GENDER	inanimate	This document, <a href="#">Section 2.3</a>
GRAMMATICAL-GENDER	male	This document, <a href="#">Section 2.3</a>
GRAMMATICAL-GENDER	neuter	This document, <a href="#">Section 2.3</a>

Table 3: New VCARD Property Values

## 6. References

### 6.1. Normative References

- [RFC2119] Bradner, S. and RFC Publisher, "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>>.
- [RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, DOI 10.17487/RFC3986, January 2005, <<https://www.rfc-editor.org/info/rfc3986>>.

**[RFC5234]**

Crocker, D., Ed. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, DOI 10.17487/RFC5234, January 2008, <<https://www.rfc-editor.org/info/rfc5234>>.

**[RFC6350]**

Perreault, S., "vCard Format Specification", RFC 6350, DOI 10.17487/RFC6350, August 2011, <<https://www.rfc-editor.org/info/rfc6350>>.

**[RFC8174]**

Leiba, B. and RFC Publisher, "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.

## 6.2. Informative References

**[I-D.ietf-calext-jscontact]** Stepanek, R. and M. Loffredo, "JSContact: A JSON representation of contact data", Work in Progress, Internet-Draft, draft-ietf-calext-jscontact-07, 10 January 2023, <<https://datatracker.ietf.org/doc/html/draft-ietf-calext-jscontact-07>>.

**[I-D.ietf-calext-jscontact-vcard]** Loffredo, M. and R. Stepanek, "JSContact: Converting from and to vCard", Work in Progress, Internet-Draft, draft-ietf-calext-jscontact-vcard-06, 10 January 2023, <<https://datatracker.ietf.org/doc/html/draft-ietf-calext-jscontact-vcard-06>>.

## Authors' Addresses

Robert Stepanek  
Fastmail  
PO Box 234, Collins St West  
Melbourne VIC 8007  
Australia

Email: [rsto@fastmailteam.com](mailto:rsto@fastmailteam.com)

Mario Loffredo  
IIT-CNR  
Via Moruzzi,1  
56124 Pisa  
Italy

Email: [mario.loffredo@iit.cnr.it](mailto:mario.loffredo@iit.cnr.it)