Network Working Group	M. Douglass
Internet-Draft	Bedework
Updates: 5545 (if approved)	July 30, 2021
Intended status: Standards Track	
Expires: January 31, 2022	

Support for iCalendar Relationships

draft-ietf-calext-ical-relations-07

Abstract

This specification updates RELATED-TO defined in iCalendar (RFC5545) and introduces new iCalendar properties LINK, CONCEPT and REFID to allow better linking and grouping of iCalendar components and related data.

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 January 31, 2022.

Copyright Notice

Table of Contents

1. Introduction

iCalendar entities often need to be related to each other or to associated meta-data. The specifications below support relationships of the following forms:

Structured iCalendar:
: iCalendar entities can be related to each other in some structured way, for example as parent, sibling, before, after.

Grouped iCalendar:
: iCalendar entities can be related to each other as a group. CATEGORIES are often used for this purpose but are problematic for application developers due to their lack of consistency and use as a free-form tag.

Linked:
: Entities can be linked to other entities such as vcards through a URI and associated REL and FMTTYPE parameters.

1.1. Structured iCalendar relationships

The currently existing iCalendar [RFC5545] RELATED-TO property has no support for temporal relationships

as used by standard project management tools.

The RELTYPE parameter is extended to take new values defining temporal relationships, a GAP parameter is defined to provide lead and lag values, and RELATED-TO is extended to allow URI values. These changes allow the RELATED-TO property to define a richer set of relationships useful for project management.

1.2. Grouped iCalendar relationships

This specification defines a new REFID property which allows arbitrary groups of entities to be associated with the same key value.

REFID is used to identify a key allowing the association of components that are related to the same object and retrieval of a component based on this key. Two examples of how this may be used are to identify the tasks associated with a given project without having to communicate the task structure of the project, and to group all tasks associated to a specific package in a package delivery system.

As such, the presence of a REFID property imparts no meaning to the component. It is merely a key to allow retrieval. This is distinct from categorisation which, while allowing grouping also adds meaning to the component to which it is attached.

1.3. Concept relationships

The name CONCEPT is used by the Simple Knowledge Organization System defined in [W3C.CR-skos-reference-20090317]. The term "concept" more accurately defines what we often mean by a category. It's not the text string that is important but the meaning attached to it. For example, the term "football" can mean very different sports.

The introduction of CONCEPT allows a more structured approach to categorization, with the possibility of namespaced and path-like values. Unlike REFID the CONCEPT property imparts some meaning. It is assumed that the value of this property will reference a well defined category.

The current [RFC5545] CATEGORY property is used as a free form 'tagging' field. As such it is difficult to establish formal relationships between components based on their category.

Rather than attempt to add semantics to the CATEGORY property it seems best to continue its usage as an informal tag and establish a new CONCEPT property with more constraints.

1.4. Linked relationships

The currently existing iCalendar standard [RFC5545] lacks a general purpose method for referencing additional, external information relating to calendar components.

This document proposes a method for referencing typed external information that can provide additional information about an iCalendar component. This new LINK property is closely aligned to the LINK header defined in [RFC8288]

The LINK property defines a typed reference or relation to external meta-data or related resources. By providing type and format information as parameters, clients and servers are able to discover interesting references and make use of them, perhaps for indexing or the presentation of interesting links for the user.

It is also often necessary to reference calendar components in other collections. For example, a VEVENT might refer to a VTODO from which it was derived. The PARENT, SIBLING and CHILD relationships defined for the RELATED-TO property only allow for a UID which is inadequate for many purposes. Allowing other value types for those relationships may help but would cause backward compatibility issues. The link property can link components in different collections or even on different servers.

When publishing events it is useful to be able to refer back to the source of that information. The actual

event may have been consumed from a feed or an ics file on a web site. A LINK property can provide a reference to the originator of the event.

Beyond the need to relate elements temporally, project management tools often need to be able to specify the relationships between the various events and tasks which make up a project. The LINK property provides such a mechanism.

The LINK property SHOULD NOT be treated as just another attachment. The ATTACH property is being extended to handle server-side management and stripping of inline data. Clients may choose to handle attachments differently as they are often an integral part of the message - for example, the agenda. See [RFC8607]

1.5. Caching and offline use

To facilitate offline display the link type may identify important pieces of data which should be downloaded in advance.

In general, the calendar entity should be self explanatory without the need to download referenced meta-data such as a web page.

1.6. Conventions Used in This Document

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.

2. Reference Types

The actual reference value can take three forms specified by the type parameter

URI:
: The default type. This is a URI referring to the target.

UID:
: This allows for linking within a single collection and the value MUST be another component within that collection.

REFERENCE:
: An XPointer. In an XML environment it may be necessary to refer to an external XML artifact. The XPointer is defined in [W3C.WD-xptr-xpointer-20021219] and allows addressing portions of XML documents.

3. Link Relation Types

[RFC8288] defines two forms of relation type: registered and extension. Registered relation types are added to the Link Relations registry as specified in Section 2.1.1 of [RFC8288]. Extension relation types, defined in Section 2.1.2 of [RFC8288], are specified as unique URIs that are not registered in the registry.

The relation types defined in Section 6.1 will be registered with IANA in accordance with the specifications in [RFC8288].

4. New temporal RELTYPE Parameter values

This section defines the usual temporal relationships for use with the RELTYPE parameter redefined in Section 3.2.15 of [RFC5545]: FINISHTOSTART, FINISHTOFINISH, STARTTOFINISH or STARTTOSTART.

The [RFC5545] RELATED-TO property with one or more of these temporal relationships will be present in the predecessor entity and will refer to the successor entity.

The GAP parameter (see Section 6.2) specifies the lead or lag time between the predecessor and the successor. In the description of each temporal relationship below we refer to Task-A, which contains and controls the relationship, and Task-B the target of the relationship.

Figure 1: Finish to start relationship

Figure 2: Finish to finish relationship

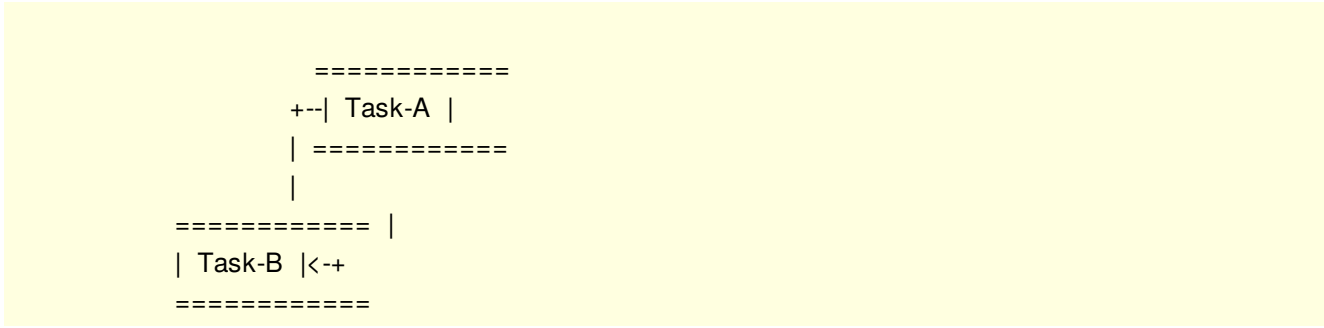

Figure 3: Start to finish relationship

Figure 4: Start to start relationship

RELTYPE=FINISHTOSTART:

Task-B cannot start until Task-A finishes. For example, when sanding is complete, painting can begin.

RELTYPE=FINISHTOFINISH:

Task-B cannot finish before Task-A is finished, that is the end of Task-A defines the end of Task-B. For example, we start the potatoes, then the meat then the peas but they should all be cooked at the same time.

RELTYPE=STARTTOFINISH:

The start of Task-A (which occurs after Task-B) controls the finish of Task-B. For example, ticket sales (Task-B) end when the game starts (Task-A).

RELTYPE=STARTTOSTART:

The start of Task-A triggers the start of Task-B, that is Task-B can start anytime after Task-A starts.

5. Additional New RELTYPE Parameter Values

This section defines the additional relationships below:

RELTYPE=FIRST:

Indicates that the referenced calendar component is the first in a series the referenced calendar component is part of.

RELTYPE=DEPENDS-ON:

Indicates that the current calendar component depends on the referenced calendar component in some manner. For example a task may be blocked waiting on the other, referenced, task.

RELTYPE=REFID:

Establishes a reference from the current component to components with a REFID property which matches the value given in the associated RELATED-TO property.

RELTYPE=CONCEPT:

Establishes a reference from the current component to components with a CONCEPT property which matches the value given in the associated RELATED-TO property.

6. New Property Parameters

6.1. Link Relation

This parameter is defined by the following notation:

```
linkrelparam = "REL" "="
            ("SOURCE"      ; Link to source of this component
            / DQUOTE uri DQUOTE
            / iana-token)   ; Other IANA registered type
```

Parameter name:

LINKREL

Purpose:

To specify the relationship of data referenced by a LINK property.

Format Definition:

Description:

This parameter MUST be specified on all LINK properties, and defines the type of reference. This allows programs consuming this data to automatically scan for references they support. In addition to the values defined here any value defined in [RFC8288] may be used. There is no default relation type.

REL=SOURCE:

identifies the source of the event information.

Registration:

These relation types are registered in [RFC8288]

6.2. Gap

This parameter is defined by the following notation:

```
gapparam      = "GAP" "=" dur-value
```

Parameter name:

GAP

Purpose:

To specify the length of the gap, positive or negative, between two temporaly related components.

Format Definition:

Description:

This parameter MAY be specified on the RELATED-TO property, and defines the duration of time between the predecessor and successor in an interval. When positive it defines the lag time between a task and its logical successor. When negative it defines the lead time.

An example of lag time might be if task A is "paint the room" and task B is "hang the drapes" then task A may be related to task B with RELTYPE=FINISHTOSTART with a gap long enough for the paint to dry.

An example of lead time might be to relate a 1 week task A to the end of task B with RELTYPE=STARTTOFINISH and a negative gap of 1 week so they finish at the same time.

7. New Value Data Types

This specification defines the following new value types to be used with the VALUE property parameter:

UID

VALUE=UID indicates that the associated value is the UID for a component.

REFERENCE

VALUE=REFERENCE indicates that the associated value is an XPointer referencing an associated XML artifact.

8. New Properties

8.1. Concept

This property is defined by the following notation:

```
concept       =  "CONCEPT" conceptparam ":"
                   uri CRLF

conceptparam = *(";" other-param)
```

The following is an example of this property. It points to a server acting as the source for the calendar object.

```
CONCEPT:http://example.com/event-types/arts/music
```

Property name:

CONCEPT

Purpose:

This property defines the formal categories for a calendar component.

Value type:

URI

Property Parameters:

IANA, and non-standard parameters can be specified on this property.

Conformance:

This property can be specified zero or more times in any iCalendar component.

Description:

This property is used to specify formal categories or classifications of the calendar component. The values are useful in searching for a calendar component of a particular type and category.

Within the "VEVENT", "VTODO", or "VJOURNAL" calendar components, more than one formal category can be specified by using multiple CONCEPT properties.

This categorization is distinct from the more informal "tagging" of components provided by the existing CATEGORIES property. It is expected that the value of the CONCEPT property will reference an external resource which provides information about the categorization.

In addition, a structured URI value allows for hierarchical categorization of events.

Possible category resources are the various proprietary systems, for example Library of Congress, or an open source of categorisation data.

Format Definition:

Example:

8.2. Link

This property is defined by the following notation:

```
link          = "LINK" linkparam ":"
                  ( text /  ; for VALUE=REFERENCE
                    uri /   ; for VALUE=URI
                    text ) ; for VALUE=TEXT
                 CRLF

linkparam      = ; the elements herein may appear in any order,
                ; and the order is not significant.

                (";" "VALUE" "=" ("UID" /
                                  "URI" /
                                  "TEXT"))
                1*(";" linkrelparam)
                (";" fmttypeparam)
                (";" labelparam)
                *(";" other-param)
```

The following is an example of this property which provides a reference to the source for the calendar object.

```
LINK;LINKREL=SOURCE;LABEL=Venue;VALUE=URI:http://example.com/events
```

The following is an example of this property which provides a reference to an entity from which this one was derived. The link relation is a vendor defined value

```
LINK;LINKREL="https://example.com/linkrel/derivedFrom";VALUE=URI:
 http://example.com/tasks/01234567-abcd1234.ics
```

Property name:
: LINK

Purpose:
: This property provides a reference to external information about a component.

Value type:
: URI, TEXT or REFERENCE

Property Parameters:
: The VALUE parameter is required. Non-standard, reference type or format type parameters can also be specified on this property. The LABEL parameter is defined in [RFC7986]

Conformance:
: This property MAY be specified in any iCalendar component.

Description:
: When used in a component the value of this property points to additional information related to the component. For example, it may reference the originating web server.

Format Definition:

Example:

Example:

8.3. Refid

This property is defined by the following notation:

```
refid      = "REFID" refidparam ":" text CRLF


refidparam      = *(";" other-param)
```

The following is an example of this property.

```
REFID:itinerary-2014-11-17
```

Property name:
: REFID

Purpose:
: This property value acts as a key for associated iCalendar entities.

Value type:
: TEXT

Property Parameters:
: Non-standard parameters can be specified on this property.

Conformance:

This property MAY be specified multiple times in any iCalendar component.

Description:

The value of this property is free-form text that creates an identifier for associated components. All components that use the same REFID value are associated through that value and can be located or retrieved as a group. For example, all of the events in a travel itinerary would have the same REFID value, so as to be grouped together.

Format Definition:

Example:

9. Redefined RELATED-TO Property

9.1. RELATED-TO

This property is defined by the following notation:

```
related    = "RELATED-TO" relparam ":"
                    ( uid /  ; for VALUE=UID
                      uri /  ; for VALUE=URI
                      text ) ; for VALUE=TEXT or default
             CRLF

relparam   = ; the elements herein may appear in any order,
             ; and the order is not significant.
             [";" "VALUE" "=" ("UID" /
                               "URI" /
                               "TEXT")]
             [";" reltypeparam]
             [";" gapparam]
             *(";" other-param)
```

The following are examples of this property.

```
RELATED-TO:jsmith.part7.19960817T083000.xyzMail@example.com

RELATED-TO:19960401-080045-4000F192713-0052@example.com

RELATED-TO;VALUE=URI;RELTYPE=STARTTOFINISH:
 http://example.com/caldav/user/jb/cal/
 19960401-080045-4000F192713.ics
```

Property name:

RELATED-TO

Purpose:

This property is used to represent a relationship or reference between one calendar component and another. The definition here extends the definition in Section 3.8.4.5 of [RFC5545] by allowing URI or UID values and a GAP parameter.

Value type:

URI, UID or TEXT

Property Parameters:

Relationship type, IANA and non-standard property parameters can be specified on this property.

Conformance:

This property MAY be specified in any iCalendar component.

Description:

By default or when VALUE=UID is specified, the property value consists of the persistent, globally unique identifier of another calendar component. This value would be represented in a calendar component by the "UID" property.

By default, the property value points to another calendar component that has a PARENT relationship to the referencing object. The "RELTYPE" property parameter is used to either explicitly state the default PARENT relationship type to the referenced calendar component or to override the default PARENT relationship type and specify either a CHILD or SIBLING relationship or a temporal relationship.

The PARENT relationship indicates that the calendar component is a subordinate of the referenced calendar component. The CHILD relationship indicates that the calendar component is a superior of the referenced calendar component. The SIBLING relationship indicates that the calendar component is a peer of the referenced calendar component.

To preserve backwards compatibility the value type MUST be UID when the PARENT, SIBLING or CHILD relationships are specified.

The FINISHTOSTART, FINISHTOFINISH, STARTTOFINISH or STARTTOSTART relationships define temporal relationships as specified in the reltype parameter definition.

Changes to a calendar component referenced by this property can have an implicit impact on the related calendar component. For example, if a group event changes its start or end date or time, then the related, dependent events will need to have their start and end dates changed in a corresponding way. Similarly, if a PARENT calendar component is cancelled or deleted, then there is an implied impact to the related CHILD calendar components. This property is intended only to provide information on the relationship of calendar components. It is up to the target calendar system to maintain any property implications of this relationship.

Format Definition:

Example:

10. Security Considerations

Applications using the LINK property need to be aware of the risks entailed in using the URIs provided as values. See [RFC3986] for a discussion of the security considerations relating to URIs.

The CONCEPT and redefined RELATED-TO property have the same issues in that values may be URIs.

11. IANA Considerations

11.1. iCalendar Property Registrations

The following iCalendar property names have been added to the iCalendar Properties Registry defined in Section 8.3.2 of [RFC5545]

Property	Status	Reference
CONCEPT	Current	Section 8.1
LINK	Current	Section 8.2
REFID	Current	Section 8.3

11.2. iCalendar Property Parameter Registrations

The following iCalendar property parameter names have been added to the iCalendar Parameters Registry defined in Section 8.3.3 of [RFC5545]

Parameter	Status	Reference
GAP	Current	Section 6.2
REL	Current	Section 6.1

11.3. iCalendar Value Data Type Registrations

The following iCalendar property parameter names have been added to the iCalendar Value Data Types Registry defined in Section 8.3.4 of [RFC5545]

Value Data Type	Status	Reference
REFERENCE	Current	Section 7
UID	Current	Section 7

11.4. iCalendar RELTYPE Value Registrations

The following iCalendar "RELTYPE" values have been added to the iCalendar Relationship Types Registry defined in Section 8.3.8 of [RFC5545]

Relationship Type	Status	Reference
CONCEPT	Current	Section 5
DEPENDS-ON	Current	Section 5
FINISHTOFINISH	Current	Section 4
FINISHTOSTART	Current	Section 4
FIRST	Current	Section 5
REFID	Current	Section 5
STARTTOFINISH	Current	Section 4
STARTTOSTART	Current	Section 4

11.5. New Reference Type Registration

The following link relation values have been added to the Reference Types Registry defined in Section 6.2.2 of [RFC8288]

Name	Status	Reference
SOURCE	Current	Section 6.1

12. Acknowledgements

The author would like to thank the members of the Calendaring and Scheduling Consortium technical committees and the following individuals for contributing their ideas, support and comments:

Adrian Apthorp, Cyrus Daboo, Marten Gajda, Ken Murchison

The author would also like to thank CalConnect, the Calendaring and Scheduling Consortium for advice with this specification.

13. 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.

[RFC5545] Desruisseaux, B., "Internet Calendaring and Scheduling Core Object Specification (iCalendar)", RFC 5545, DOI 10.17487/RFC5545, September 2009.

[RFC7986] Daboo, C., "New Properties for iCalendar", RFC 7986, DOI 10.17487/RFC7986, October 2016.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017.

[RFC8288] Nottingham, M., "Web Linking", RFC 8288, DOI 10.17487/RFC8288, October 2017.

[RFC8607] Daboo, C., Quillaud, A. and K. Murchison, "Calendaring Extensions to WebDAV (CalDAV): Managed Attachments", RFC 8607, DOI 10.17487/RFC8607, June 2019.

[W3C.CR-skos-reference-20090317] Bechhofer, S. and A. Miles, "SKOS Simple Knowledge Organization System Reference", World Wide Web Consortium CR CR-skos-reference-20090317, March 2009.

[W3C.WD-xptr-xpointer-20021219] DeRose, S., Daniel, R. and E. Maler, "XPointer xpointer() Scheme", World Wide Web Consortium WD WD-xptr-xpointer-20021219, December 2002.

Author's Address

Michael Douglass
Bedework
226 3rd Street
Troy, NY 12180
USA
EMail: mdouglass@bedework.com
URI: http://bedework.com