

Calendaring and scheduling — Support for iCalendar Relationships

Committee Draft Standard

Warning for drafts

This document is not a CalConnect Standard. It is distributed for review and comment, and is subject to change without notice and may not be referred to as a Standard. Recipients of this draft are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

Recipients of this draft are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

© 2018 The Calendaring and Scheduling Consortium, Inc.

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from the address below.

The Calendaring and Scheduling Consortium, Inc.

4390 Chaffin Lane
McKinleyville
California 95519
United States of America

copyright@calconnect.org
www.calconnect.org

Contents

Foreword.....	iv
0. Introduction.....	v
Acknowledgements.....	vii
1. Scope.....	1
2. Normative References.....	1
3. Terms and definitions.....	1
4. Reference Types.....	1
5. Link Relation Types.....	1
6. Redefined Relation Type Value.....	1
7. New Property Parameters.....	3
7.1. Rel.....	3
7.2. Gap.....	4
8. New Value Data Types.....	4
9. New Properties.....	5
9.1. Concept.....	5
9.2. Link.....	6
9.3. Refid.....	7
10. Redefined RELATED-TO Property.....	8
10.1.RELATED-TO.....	8
11. Security Considerations.....	9
12. IANA Considerations.....	10
12.1.iCalendar Property Registrations.....	10
12.2.iCalendar Property Parameter Registrations.....	10
12.3.iCalendar Value Data Type Registrations.....	10
12.4.iCalendar RELTYPE Value Registrations.....	10
12.5.New Reference Type Registration.....	11
Bibliography.....	12

Foreword

The Calendaring and Scheduling Consortium (“CalConnect”) is a global non-profit organization with the aim to facilitate interoperability of collaborative technologies and tools through open standards.

CalConnect works closely with international and regional partners, of which the full list is available on our website (<https://www.calconnect.org/about/liaisons-and-relationships>).

The procedures used to develop this document and those intended for its further maintenance are described in the CalConnect Directives.

In particular the different approval criteria needed for the different types of CalConnect documents should be noted. This document was drafted in accordance with the editorial rules of the CalConnect Directives.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. CalConnect shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be provided in the Introduction.

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

This document was prepared by Technical Committee *CALENDAR*.

0. Introduction

0.1.. General

iCalendar entities often need to be related to each other or to associated meta-data. These relationships can take the following forms

Structured iCalendar	iCalendar entities are related to each other in some structured way, for example as parent, sibling, before, after.
Grouped iCalendar	iCalendar entities are related to each other as a group. CATEGORIES are often used for this purpose but are problematic for application developers.
Linked	Entities are linked to each other through typed references.

0.2.. Structured iCalendar relationships

The currently existing iCalendar [RFC 5545](#) 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.

0.3.. 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. This may be, for example, to identify the tasks associated with a given project without having to communicate the task structure of the project, or, for example, in a package delivery system all tasks associated to a specific package.

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.

0.4.. Concept relationships

The name CONCEPT is used by the Simple Knowledge Organization System defined in [W3C CR skos-reference-20090317](#). This more accurately defines what we mean by a category. It's not the words but the meaning.

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 [RFC 5545](#) 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 current property it seems best to continue its usage as an informal tag and establish a new property with more constraints.

0.5.. Linked relationships

The currently existing iCalendar standard [RFC 5545](#) 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 [RFC 5988](#)

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 often necessary to relate calendar components. The current RELATED-TO property only allows for a UID which is inadequate for many purposes. Allowing other value types for that property may help but might raise a number of 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 [I-D.daboo-caldav-attachments](#)

0.6.. 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.

0.7.. 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 [RFC 2119](#).

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.

Calendaring and scheduling — Support for iCalendar Relationships

1. Scope

This specification updates RELATED-TO defined in [RFC 5545](#) and introduces new iCalendar properties LINK, CONCEPT and REFID to allow better linking and grouping of iCalendar components and related data.

2. Normative References

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

RFC 5545, *Internet Calendaring and Scheduling Core Object Specification (iCalendar)*

RFC 5988, *Web Linking*

RFC 6638, *Scheduling Extensions to CalDAV*

W3C CR skos-reference-20090317,

W3C WD xptr-xpointer-20021219,

I-D.daboo-caldav-attachments, *CalDAV Managed Attachments*

3. Terms and definitions

No terms and definitions are listed in this document.

4. 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 is assumed to 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.

5. Link Relation Types

[RFC 5988](#) defines two forms of relation types, registered and extension. Registered relation types are added to a registry defined by [RFC 5988](#) while extension relation types are specified as unique unregistered URIs, (at least unregistered in the [RFC 5988](#) registry).

The relation types defined here will be registered with IANA in accordance with the specifications in [RFC 5988](#).

6. Redefined Relation Type Value

Relationship parameter type values are defined in [RFC 5545, section 3.2.15](#). This specification redefines that type to include the new temporal relationship values FINISHTOSTART, FINISHTOFINISH, STARTTOFINISH and

STARTTOSTART. It also adds the DEPENDS-ON value to provide a link to a component upon which the current component depends.

Format Definition

This property parameter is defined by the following notation:

```
reltypeparam = "RELTYPE" "="
               ("PARENT"      ; Parent
relationship - Default
               / "CHILD"      ; Child
relationship
               / "SIBLING"    ; Sibling
relationship
               / "DEPENDS-ON" ; refers to
previous component
               / "REFID"      ; Relationship
based on REFID
               / "CONCEPT"
                           ; Relationship based
on CONCEPT
               / "FINISHTOSTART" ; Temporal
relationship
               / "FINISHTOFINISH" ; Temporal
relationship
               / "STARTTOFINISH" ; Temporal
relationship
               / "STARTTOSTART" ; Temporal
relationship
               / iana-token    ; Some other
IANA-registered
                           ; iCalendar
relationship type
               / x-name)      ; A non-
standard, experimental
                           ; relationship
type
```

Figure 1

Description

This parameter can be specified on a property that references another related calendar component. The parameter may specify the hierarchical relationship type of the calendar component referenced by the property when the value is PARENT, CHILD or SIBLING. If this parameter is not specified on an allowable property, the default relationship type is PARENT. Applications MUST treat x-name and iana-token values they don't recognize the same way as they would the PARENT value.

This parameter defines the temporal relationship when the value is one of the project management standard relationships FINISHTOSTART, FINISHTOFINISH, STARTTOFINISH or STARTTOSTART. This property will be present in the predecessor entity and will refer to the successor entity. The GAP parameter 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.

RELTYPE=PARENT

See [RFC 5545, section 3.2.15](#).

RELTYPE=CHILD	See RFC 5545, section 3.2.15 .
RELTYPE=SIBLING	See RFC 5545, section 3.2.15 .
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.
RELTYPE=FINISHTOSTART	Task-B cannot start until Task-A finishes. For example, when sanding is complete, painting can begin.

===== | Task-A | --+ ===== || V ===== | Task-B | =====

Figure 1 — Finish to start relationship

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.
------------------------	--

===== | Task-A | --+ ===== || ===== || Task-B | <-+ =====

Figure 2 — Finish to finish relationship

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).
-----------------------	---

===== +-| Task-A | | ===== | ===== || Task-B | <-+ =====

Figure 3 — Start to finish relationship

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

===== +-| Task-A | | ===== | ===== +->| Task-B | =====

Figure 4 — Start to start relationship

7. New Property Parameters

7.1. Rel

Parameter name	REL
Purpose	To specify the relationship of data referenced by a LINK property.

Format Definition

This parameter is defined by the following notation:

```
relparam      = "REL" "="
                  ("SOURCE"          ; Link to source
                  / DQUOTE uri DQUOTE
                  / x-name           ; Experimental
                  / iana-token)      ; Other IANA
                  registered type
```

Figure 2

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 [RFC 5988](#) 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 [RFC 5988](#)

7.2. Gap

Parameter name

GAP

Purpose

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

Format Definition

This parameter is defined by the following notation:

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

Figure 3

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.

8. 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.

9. New Properties

9.1. Concept

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	<p>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.</p> <p>Within the “VEVENT”, “VTODO”, or “VJOURNAL” calendar components, more than one formal category can be specified by using multiple CONCEPT properties.</p> <p>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.</p> <p>In addition, a structured URI value allows for hierarchical categorization of events.</p> <p>Possible category resources are the various proprietary systems, for example Library of Congress, or an open source derived from something like the dmoz.org data.</p>

Format Definition

This property is defined by the following notation:

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

conceptparam = *(
    ;
    ; The following is OPTIONAL,
    ; and MAY occur more than once.
    ;
    (";" other-param)
    ;
```

)

Figure 4**EXAMPLE**

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

Figure 5**9.2. Link**

Property name	LINK
Purpose	This property provides a reference to external information about a component.
Value type	URI, TEXT or REFERENCE
Property Parameters	Non-standard, reference type or format type parameters can be specified on this property.
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	This property is defined by the following notation:

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

```
linkparam = *(
              ; the following is MANDATORY
              ; and MAY occur more than once

              (";" relparam) /

              ; the following are MANDATORY
```

```
once                                     ; but MUST NOT occur more than  
  
(";" fmttypeparam) /  
(";" labelparam) /  
; labelparam is defined in ...  
  
; the following is OPTIONAL  
; and MAY occur more than once  
  
(";" xparam)  
  
)
```

Figure 6

EXAMPLE

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

LINK;REL=SOURCE;LABEL=The Egg:http://example.com/events

Figure 7

9.3. Refid

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 a text identifier that allows associated components to be located or retrieved as a whole. For example all of the events in a travel itinerary would have the same REFID value.

Format Definition

This property is defined by the following notation:

```
refid      = "REFID" refidparam ":" text CRLF  
  
refidparam = *(  
  
    ; the following is OPTIONAL  
    ; and MAY occur more than once  
  
    (";" xparam)  
  
    )
```

Figure 8

EXAMPLE

The following is an example of this property.

REFID:itinerary-2014-11-17

Figure 9

10. Redefined RELATED-TO Property

10.1. RELATED-TO

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 RFC 5545, section 3.8.4.5 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	<p>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.</p> <p>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.</p> <p>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.</p> <p>The FINISHTOSTART, FINISHTOFINISH, STARTTOFINISH or STARTTOSTART relationships define temporal relationships as specified in the reltype parameter definition.</p> <p>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</p>

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

This property is defined by the following notation:

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

relparam     = *(
              ;
              ; The following are OPTIONAL,
              ; but MUST NOT occur more than
once.
              ;
              (";" reltypeparam) /
              (";" gapparam) /
              ;
              ; The following is OPTIONAL,
              ; and MAY occur more than once.
              ;
              (";" other-param)
              ;
              )
```

Figure 10

EXAMPLE

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

Figure 11

11. 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.

12. IANA Considerations

12.1. iCalendar Property Registrations

The following iCalendar property names have been added to the iCalendar Properties Registry defined in [RFC 5545, section 8.3.2](#)

Table 1

Property	Status	Reference
CONCEPT	Current	Clause 9.1
LINK	Current	Clause 9.2
REFID	Current	Clause 9.3

12.2. iCalendar Property Parameter Registrations

The following iCalendar property parameter names have been added to the iCalendar Parameters Registry defined in [RFC 5545, section 8.3.3](#)

Table 2

Parameter	Status	Reference
REL	Current	Clause 7.1
GAP	Current	Clause 7.2

12.3. iCalendar Value Data Type Registrations

The following iCalendar property parameter names have been added to the iCalendar Value Data Types Registry defined in [RFC 5545, section 8.3.4](#)

Table 3

Value Data Type	Status	Reference
UID	Current	Clause 8
REFERENCE	Current	Clause 8

12.4. iCalendar RELTYPE Value Registrations

The following iCalendar “RELTYPE” values have been added to the iCalendar Relationship Types Registry defined in [RFC 5545, section 8.3.8](#)

Table 4

Relationship Type	Status	Reference
DEPENDS-ON	Current	Clause 6
REFID	Current	Clause 6
CONCEPT	Current	Clause 6

Relationship Type	Status	Reference
FINISHTOSTART	Current	Clause 6
FINISHTOFINISH	Current	Clause 6
STARTTOFINISH	Current	Clause 6
STARTTOSTART	Current	Clause 6

12.5. New Reference Type Registration

The following link relation values have been added to the Reference Types Registry defined in [RFC 5988, section 6.2.2](#)

Table 5

Name	Status	Reference
SOURCE	Current	Clause 7.1

Bibliography

- [1] RFC 2119, *Key words for use in RFCs to Indicate Requirement Levels*