Event Publishing Extensions to iCalendar

draft-ietf-calext-eventpub-extensions-11

Abstract

This specification updates RFC5545 by introducing a number of new iCalendar properties and components which are of particular use for event publishers and in social networking.

This specification also defines a new STRUCTURED-DATA property for iCalendar RFC5545 to allow for data that is directly pertinent to an event or task to be included with the calendar data.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

The currently existing iCalendar standard [RFC5545] lacks useful methods for referencing additional, external information relating to calendar components. Additionally there is no standard way to provide rich text

descriptions or meta-data associated with the event.

Current practice is to embed this information as links in the description or to add non-standard properties as defined in [RFC5545] section 3.8.8.2.

This document updates [RFC5545] to define a number of properties and a component referencing such external information that can provide additional information about an iCalendar component. The intent is to allow interchange of such information between applications or systems (e.g., between clients, between client and server, and between servers). Formats such as VCARD are likely to be most useful to the receivers of such events as they may be used in other applications - such as address books.

This specification defines a new PARTICIPANT component. Many people or groups may participate in an event. This component provides detailed information. Such participants may act as attendees to the event (or derived events) or may just provide a reference - perhaps for mailing lists.

Additionally this specification defines a new use for the RELATED parameter to allow locations to be defined as the start and/or end location of a component.

The following properties are defined in this specification

STYLED-DESCRIPTION:
> Supports rich-text descriptions, for example HTML. Event publishers typically wish to provide more and better formatted information about the event.

STRUCTURED-LOCATION:
> There may be a number of locations associated with an event. This provides detailed information about the location.

STRUCTURED-RESOURCE:
> Events need resources such as rooms, projectors, conferencing capabilities.

STRUCTURED-DATA:
> The existing properties in iCalendar cover key elements of events and tasks such as start time, end time, location, summary, etc. However, different types of events often have other specific "fields" that it is useful to include in the calendar data. For example, an event representing an airline flight could include the airline, flight number, departure and arrival airport codes, check-in and gate-closing times etc. As another example, a sporting event might contain information about the type of sport, the home and away teams, the league the teams are in, information about nearby parking, etc.

PARTICIPANT-TYPE:
> Used in the PARTICIPANT component to define the type.

CALENDAR-ADDRESS:
> Used in the PARTICIPANT component to provide the calendar address of the participant.

In addition the SOURCE property defined in [RFC7986] is redefined to allow VALUE=TEXT and broaden its usage.

1.1. 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. Components and properties

Previous extensions to the calendaring standards have been largely restricted to the addition of properties or parameters. This is partly because iCalendar libraries had trouble handling components nested deeper than

those defined in [RFC5545]

In a break with this 'tradition' this specification introduces one of these extensions as a component rather than a property. This is a better match for the way XML and JSON handles such structures and allows richer definitions.

It also allows for the addition of extra properties inside the component and resolves some of the problems of trying to add detailed information as a parameter.

3. Typed References

The properties defined here can all reference external meta-data which may be used by applications to provide enhanced value to users. By providing type information as parameters, clients and servers are able to discover interesting references and make use of them, perhaps for indexing or the presenting of additional related information for the user.

The [RFC5545] LOCATION property provides only an unstructured single text value for specifying the location where an event (or task) will occur. This is inadequate for use cases where structured location information (e.g. address, region, country, postal code) is required or preferred, and limits widespread adoption of iCalendar in those settings.

Using STRUCTURED-LOCATION, information about a number of interesting locations can be communicated, for example, address, region, country, postal code as well as other informations such as the parking, restaurants and the venue. Servers and clients can retrieve the objects when storing the event and use them to index by geographic location.

When a calendar client receives a calendar component it can search the set of supplied properties looking for those of particular interest. The TYPE and FMTTYPE parameters, if supplied, can be used to help the selection.

The PARTICIPANT component is designed to handle common use cases in event publication. It is generally important to provide information about the organizers of such events. Sponsors wish to be referenced in a prominent manner. In social calendaring it is often important to identify the active participants in the event, for example a school sports team, and the inactive participants, for example the parents.

The PARTICIPANT component can also be used to provide useful extra data about an attendee. For example a LOCATION property inside the PARTICIPANT gives the actual location of a remote attendee.

3.1. Use Cases

The main motivation for these properties has been event publication but there are opportunities for use elsewhere. The following use cases will describe some possible scenarios.

3.1.1. Piano Concert Performance

In putting together a concert there are many participants: piano tuner, performer, stage hands etc. In addition there are sponsors and various contacts to be provided. There will also be a number of related locations. A number of events can be created, all of which relate to the performance in different ways.

There may be an iTip [RFC5546] meeting request for the piano tuner who will arrive before the performance. Other members of staff may also receive meeting requests.

An event can also be created for publication which will have a PARTICIPANT component for the pianist providing a reference to vcard information about the performer. This event would also hold information about parking, local subway stations and the venue itself. In addition, there will be sponsorship information for sponsors of the event and perhaps paid sponsorship properties essentially advertising local establishments.

3.1.2. Itineraries

These additions also provide opportunities for the travel industry. When booking a flight the PARTICIPANT component can be used to provide references to businesses at the airports and to car hire businesses at the destination.

The embedded location information can guide the traveller at the airport or to their final destination. The contact information can provide detailed information about the booking agent, the airlines, car hire companies and the hotel.

3.1.2.1. Reserving facilities

For a meeting, the size of a room and the equipment needed depends to some extent on the number of attendees actually in the room.

A meeting may have 10 attendees non of which are co-located. The current ATTENDEE property dos not allow for the additon of such meta-data. The PARTICIPANT property allows attendees to specify their location.

4. Modifications to Calendar Components

```
eventc    = "BEGIN" ":" "VEVENT" CRLF
          eventprop *alarmc *participantc
          "END" ":" "VEVENT" CRLF


eventprop =/ *(
          ;
          ; The following are OPTIONAL,
          ; and MAY occur more than once.
          ;
          styleddescription / strucloc / strucres / sdataprop
          ;
          )
todoc     = "BEGIN" ":" "VTODO" CRLF
          todoprop *alarmc *participantc
          "END" ":" "VTODO" CRLF


todoprop =/ *(
          ;
          ; The following are OPTIONAL,
          ; and MAY occur more than once.
          ;
          styleddescription / strucloc / strucres / sdataprop
          ;
          )
journalc  = "BEGIN" ":" "VJOURNAL" CRLF
          jourprop *participantc
          "END" ":" "VJOURNAL" CRLF


jourprop =/ *(
```

```
      ;
      ; The following are OPTIONAL,
      ; and MAY occur more than once.
      ;
      styleddescription / sdataprop
      ;
      )
```

The following changes to the syntax defined in iCalendar are made here. New elements are defined in subsequent sections.

5. New Property Parameters

This specification makes use of the LABEL parameter which is defined in [RFC7986]

5.1. Loctype

This parameter is defined by the following notation:

```
loctypeparam   = "LOCTYPE" "=" param-value
```

Parameter name:
        LOCTYPE

Purpose:
        To specify the type of location.

Format Definition:

Description:
        This parameter MAY be specified on STRUCTURED-LOCATION and provides a way to differentiate multiple properties. For example, it allows event producers to provide location information for the venue and the parking.

        Values for this parameter are taken from the values defined in [RFC4589]. New location types SHOULD be registered in the manner laid down in that specification.

5.2. Restype

This parameter is defined by the following notation:

```
restypeparam  = "RESTYPE" "=" restypevalue CRLF

restypevalue  = ("ROOM"
            / "PROJECTOR"
            / "REMOTE-CONFERENCE-AUDIO"
            / "REMOTE-CONFERENCE-VIDEO"
            / x-name         ; Experimental status
            / iana-token)    ; Other IANA-registered
                        ; values
```

Parameter name:
        RESTYPE

Purpose:

To specify the type of resource.

Format Definition:

Description:

>This parameter MAY be specified on STRUCTURED-RESOURCE and provides a way to differentiate multiple properties.

>The registered values are described below. New resource types SHOULD be registered in the manner laid down in this specification.

ROOM:

>A room for the event/meeting.

PROJECTOR:

>Projection equipment.

REMOTE-CONFERENCE-AUDIO:

>Audio remote conferencing facilities.

REMOTE-CONFERENCE-VIDEO:

>Video remote conferencing facilities.

5.3. Order

This parameter is defined by the following notation:

```
orderparam    = "ORDER" "=" integer ;Must be greater than or equal to 1
```

Parameter name:
>ORDER

Purpose:
>To define ordering for the associated property.

Format Definition:

Description:

>The ORDER parameter is OPTIONAL and is used to indicate the relative ordering of the corresponding instance of a property. Its value MUST be an integer greater than or equal to 1 that quantifies the order with 1 being the first in the ordering.

>When the parameter is absent, the default MUST be to interpret the property instance as being at the lowest level of ordering, that is, the property will appear after any other instances of the same property with any value of ORDER.

>When any ORDER parameters have the same value all the associated properties appear as a group within which there is no defined order.

>Note that the value of this parameter is to be interpreted only in relation to values assigned to other corresponding instances of the same property in the same entity. A given value, or the absence of a value, MUST NOT be interpreted on its own.

>This parameter MAY be applied to any property that allows multiple instances.

5.4. Schema

This parameter is defined by the following notation:

```
schemaparam    = "SCHEMA" "=" DQUOTE uri DQUOTE
```

```
STRUCTURED-DATA;FMTTYPE=application/ld+json;
 SCHEMA="https://schema.org/FlightReservation";
```

ENCODING=BASE64;VALUE=BINARY:Zm9vYmFy

Parameter Name:
> SCHEMA

Purpose:
> To specify the schema used for the content of a "STRUCTURED-DATA" property value.

Format Definition:

Description:
> This property parameter SHOULD be specified on "STRUCTURED-DATA" properties. When present it provides identifying information about the nature of the content of the corresponding "STRUCTURED-DATA" property value. This can be used to supplement the media type information provided by the "FMTTYPE" parameter on the corresponding property.

Example:

5.5. Derived

This parameter is defined by the following notation:

```
derivedparam    = "DERIVED" "=" ("TRUE" / "FALSE")
; Default is FALSE
```

```
STRUCTURED-DESCRIPTION;FMTTYPE=text/html;
 DERIVED=TRUE:<html>...
```

Parameter Name:
> DERIVED

Purpose:
> To specify that the value of the associated property is derived from some other property value or values.

Format Definition:

Description:
> This property parameter can 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.

> As an example, if a STYLED-DESCRIPTION property is present with FMTTYPE="application/rtf" then there may be an additional STYLED-DESCRIPTION property with FMTTYPE="text/html" and DERIVED=TRUE and a value created from the rtf value.

Example:

6. Redefined Property SOURCE

The SOURCE property defined in [RFC7986] is redefined to allow VALUE=TEXT and broaden its usage to any component.

This property is defined by the following notation:

```
source    = "SOURCE" sourceparam
      (
       (
        ";" "VALUE" "=" "URI"
        ":" uri
       ) /
       (
```

The following is an example referring to a VCARD.

Property name:
> SOURCE

Purpose:
> This property provides a reference to information about a component such as a participant possibly as a vcard or optionally a plain text typed value.

Value type:
> The default value type for this property is URI. The value type can also be set to TEXT to indicate plain text content.

Property Parameters:
> Non-standard or format type parameters can be specified on this property.

Conformance:
> This property MAY be appear in any iCalendar component.

Description:
> This property provides information about the component in which it appears.

> In a resource or participant it may provide a reference to a vcard giving directory information.

> In a VCALENDAR component this property identifies a location where a client can retrieve updated data for the calendar. Clients SHOULD honor any specified "REFRESH-INTERVAL" value when periodically retrieving data. Note that this property differs from the "URL" property in that "URL" is meant to provide an alternative representation of the calendar data rather than the original location of the data.

> In a calendar entity component such as an event the SOURCE property may provide a reference to the original source of the event. This may be used by aggregators to provide a link back.

Format Definition:

Example:

7. New Properties

7.1. Participant Type

This parameter is defined by the following notation:

```
participanttype   = "PARTICIPANT-TYPE" "=" partvalue CRLF

partvalue   = ("ACTIVE"
            / "INACTIVE"
            / "SPONSOR"
            / "CONTACT"
            / "BOOKING-CONTACT"
            / "EMERGENCY-CONTACT"
            / "PUBLICITY-CONTACT"
            / "PLANNER-CONTACT"
            / "PERFORMER"
            / "SPEAKER"
            / x-name         ; Experimental status
            / iana-token)    ; Other IANA-registered
                             ; values
```

The following is an example of this property:

```
PARTICIPANT-TYPE:SPEAKER
```

Property name:
>   PARTICIPANT-TYPE

Purpose:
>   To specify the type of participant.

Value type:
>   The value type for this property is TEXT. The allowable values are defined below.

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

Conformance:
>   This property MUST be specified within a PARTICIPANT component.

Description:
>   This property defines the type of participation in events or tasks. Participants can be individuals or organizations, for example a soccer team, the spectators, or the musicians.

Format Definition:

Example:

The registered values for the PARTICIPANT-TYPE property have the meanings described here:

ACTIVE:
>   A participant taking an active role - for example a team member.

INACTIVE:
>   A participant taking an inactive part - for example an audience member.

SPONSOR:

> A sponsor of the event. The ORDER parameter may be used with this participant type to define the relative order of multiple sponsors.

CONTACT:

> Contact information for the event. The ORDER parameter may be used with this participant type to define the relative order of multiple contacts.

BOOKING-CONTACT:

> Contact information for reservations or payment

EMERGENCY-CONTACT:

> Contact in case of emergency

PUBLICITY-CONTACT:

> Contact for publicity

PLANNER-CONTACT:

> Contact for the event planner or organizer

PERFORMER:

> A performer - for example the soloist or the accompanist. The ORDER parameter may be used with this participant type to define the relative order of multiple performers. For example, ORDER=1 could define the principal performer or soloist.

SPEAKER:

> Speaker at an event

7.2. Calendar Address

This parameter is defined by the following notation:

```
calendaraddress   = "CALENDAR-ADDRESS" "=" cal-address
```

Property name:

> CALENDAR-ADDRESS

Purpose:

> To specify the calendar address for a participant.

Value type:

> CAL-ADDRESS

Property Parameters:

> IANA or non-standard property parameters can be specified on this property.

Conformance:

> This property MAY be specified within a PARTICIPANT component.

Description:

> This property provides a calendar user address for the participant. If there is an ATTENDEE property with the same value then the participant is schedulable.

Format Definition:

7.3. Styled-Description

This property is defined by the following notation:

```
styleddescription = "STYLED-DESCRIPTION" styleddescparam ":"
         (
          (
```

```
                      ";" "VALUE" "=" "URI"
                      ":" uri
                   )  /
                   (
                     ";" "VALUE" "=" "TEXT"
                     ":" text
                   )
                 )
                 CRLF

    styleddescparam   = *(
                    ;
                    ; The following are OPTIONAL,
                    ; but MUST NOT occur more than once.
                    ;
                    (";" altrepparam) / (";" languageparam) /
                    (";" fmttypeparam) / (";" derivedparam) /
                    ;
                    ; the following is OPTIONAL
                    ; and MAY occur more than once
                    ;
                    (";" other-param)
                    )
```

The following is an example of this property. It points to an html description.

```
    STYLED-DESCRIPTION;VALUE=URI:http://example.org/desc001.html
```

Property name:
> STYLED-DESCRIPTION

Purpose:
> This property provides for one or more rich-text descriptions to replace that provided by the DESCRIPTION property.

Value type:
> There is no default value type for this property. The value type can be set to URI or TEXT. Other text-based value types can be used when defined in the future. Clients MUST ignore any properties with value types they do not understand.

Property Parameters:
> IANA, non-standard, id, alternate text representation, format type, derived and language property parameters can be specified on this property.

Conformance:
> The property can be specified multiple times in the "VEVENT", "VTODO", "VJOURNAL", "PARTICIPANT", or "VALARM" calendar components.

> If it does appear more than once there MUST be exactly one instance of the property with no DERIVED parameter or DERIVED=FALSE. All others MUST have DERIVED=TRUE.

> Additionally, if there is one or more STYLED-DESCRIPTION property then the DESCRIPTION property should be either absent or have the parameter DERIVED=TRUE.

Description:
> This property is used in the "VEVENT" and "VTODO" to capture lengthy textual descriptions

associated with the activity. This property is used in the "VJOURNAL" calendar component to capture one or more textual journal entries. This property is used in the "VALARM" calendar component to capture the display text for a DISPLAY category of alarm, and to capture the body text for an EMAIL category of alarm. In the PARTICIPANT component it provides a detailed description of the participant.

VALUE=TEXT is used to provide rich-text inline as the property value.

VALUE=URI is used to provide a link to rich-text content which is expected to be displayed inline as part of the event.

In either case the DESCRIPTION property should be absent or contain a plain text rendering of the styled text.

Applications MAY attempt to guess the media type of the resource via inspection of its content if and only if the media type of the resource is not given by the "FMTTYPE" parameter. If the media type remains unknown, calendar applications SHOULD treat it as type "text/html".

Multiple STYLED-DESCRIPTION properties may be used to provide different formats or different language variants. However all but one MUST have DERIVED=TRUE.

Format Definition:

Example:

7.4. Structured-Location

This property is defined by the following notation:

```
strucloc      = "STRUCTURED-LOCATION" struclocparam
              (
               (
                 ";" "VALUE" "=" "URI"
                 ":" uri
               ) /
               (
                 ";" "VALUE" "=" "TEXT"
                 ":" text
               )
              )
              CRLF

struclocparam  = *(
              ;
              ; the following are OPTIONAL
              ; but MUST NOT occur more than once
              ;
              (";" fmttypeparam) /
              (";" labelparam) /
              (";" languageparam) /
              (";" trigrelparam) /
              (";" loctypeparam) /
              ;
              ; the following is OPTIONAL
              ; and MAY occur more than once
              ;
              (";" other-param)
              )
```

The following is an example of this property. It points to a venue.

```
         STRUCTURED-LOCATION;LABEL="The venue";
          VALUE=URI:
          http://dir.example.com/venues/big-hall.vcf
```

Property name:
    STRUCTURED-LOCATION

Purpose:
    This property provides a typed reference to external information about the location of an event or optionally a plain text typed value.

Value type:
    There is no default value type for this property. The value type can be set to URI or TEXT.

Property Parameters:
    IANA, non-standard, label, loctype, related or format type parameters can be specified on this property.

Conformance:
    This property MAY be specified zero or more times in any iCalendar component.

Description:
    When used in a component the value of this property provides information about the event venue or of related services such as parking, dining, stations etc..

    When a LABEL parameter is supplied the language of the label must match that of the content and of the LANGUAGE parameter if present.

Use of the related parameter:
    This allows a location to define the start and/or end timezone of the associated component. If a location is specified with a RELATED parameter then the affected DTSTART or DTEND properties MUST be specified as floating DATE-TIME value.

    If the RELATED parameter is present with a value of START, then the "DTSTART" property MUST be present in the associated "VEVENT" or "VTODO" calendar component.

    For an event, if the RELATED parameter is present with a value of END, then the "DTEND" property or the "DTSTART" and "DURATION " properties MUST be present in the associated "VEVENT" calendar component.

    For a to-do with a RELATED value of END, then either the "DUE" property or the "DTSTART" and "DURATION " properties MUST be present in the associated "VTODO" calendar component.

    If there is a location specified with RELATED=START and no location is specified with RELATED=END then the event is assumed to start and end in the same timezone.

Format Definition:

Example:

7.5. Structured-Resource

This property is defined by the following notation:

```
strucres      = "STRUCTURED-RESOURCE" strucresparam /
          (
           (
```

```
                ";" "VALUE" "=" "URI"
                ":" uri
            ) /
            (
                ";" "VALUE" "=" "TEXT"
                ":" text
            )
        )
          CRLF


    strucresparam  = *(
                ;
                ; the following are OPTIONAL
                ; but MUST NOT occur more than once
                ;
                (";" fmttypeparam) /
                (";" labelparam) /
                (";" languageparam) /
                (";" restypeparam) /
                ;
                ; the following is OPTIONAL
                ; and MAY occur more than once
                ;
                (";" other-param)
                )
```

The following is an example of this property. It refers to a projector.

```
        STRUCTURED-RESOURCE;value=uri;restype="projector":
         http://dir.example.com/projectors/3d.vcf
```

Property name:
   STRUCTURED-RESOURCE

Purpose:
   This property provides a typed reference to external information about a resource or optionally a plain
   text typed value. Typically a resource is anything that might be required or used by a calendar entity
   and possibly has a directory entry.

Value type:
   There is no default value type for this property. The value type can be set to URI or TEXT.

Property Parameters:
   IANA, non-standard, label, restype or format type parameters can be specified on this property.

Conformance:
   This property MAY be specified zero or more times in any iCalendar component.

Description:
   When used in a component the value of this property provides information about resources used for
   the event.

   Such resources may be a room or a projector. This RESTYPE value registry provides a place in
   which resource types may be registered for use by scheduling sevices.

   When a LABEL parameter is supplied the language of the label must match that of the content and of

the LANGUAGE parameter if present.

Format Definition:

Example:

7.6. Structured-Data

This property is defined by the following notation:

```
sdataprop   = "STRUCTURED-DATA" sdataparam
              (":" text) /
              (
                ";" "ENCODING" "=" "BASE64"
                ";" "VALUE" "=" "BINARY"
                ":" binary
              ) /
              (
                ";" "VALUE" "=" "URI"
                ":" uri
              )
              CRLF

sdataparam  = *(
              ;
              ; The following is OPTIONAL for a URI value,
              ; RECOMMENDED for a TEXT or BINARY value,
              ; and MUST NOT occur more than once.
              ;
              (";" fmttypeparam) /
              (";" schemaparam) /
              ;
              ; The following is OPTIONAL,
              ; and MAY occur more than once.
              ;
              (";" other-param)
              ;
            )
```

```
STRUCTURED-DATA;FMTTYPE=application/ld+json;
 SCHEMA="https://schema.org/SportsEvent";
 VALUE=TEXT:{\n
   "@context": "http://schema.org"\,\n
   "@type": "SportsEvent"\,\n
   "homeTeam": "Pittsburgh Pirates"\,\n
   "awayTeam": "San Francisco Giants"\n
 }\n
```

Property Name:

STRUCTURED-DATA

Purpose:

This property specifies ancillary data associated with the calendar component.

Value Type:

TEXT, BINARY or URI

Property Parameters:

> IANA, non-standard, inline encoding, and value data type property parameters can be specified on this property. The format type and schema parameters can be specified on this property and are RECOMMENDED for text or inline binary encoded content information.

Conformance:

> This property can be specified multiple times in an iCalendar object. Typically it would be used in "VEVENT", "VTODO", or "VJOURNAL" calendar components.

Description:

> This property is used to specify ancillary data in some structured format either directly (inline) as a "TEXT" or "BINARY" value, or as a link via a "URI" value.

> Rather than define new iCalendar properties for the variety of event types that might occur, it would be better to leverage existing schemas for such data. For example, schemas available at https://schema.org include different event types. By using standard schemas, interoperability can be improved between calendar clients and non-calendaring systems that wish to generate or process the data.

> This property allows the direct inclusion of ancillary data whose schema is defined elsewhere. This property also includes parameters to clearly identify the type of the schema being used so that clients can quickly and easily spot what is relevant within the calendar data and present that to users or process it within the calendaring system.

> iCalendar does support an "ATTACH" property which can be used to include documents or links to documents within the calendar data. However, that property does not allow data to be included as a "TEXT" value (a feature that "STRUCTURED-DATA" does allow), plus attachments are often treated as "opaque" data to be processed by some other system rather than the calendar client. Thus the existing "ATTACH" property is not sufficient to cover the specific needs of inclusion of schema data. Extending the "ATTACH" property to support a new value type would likely cause interoperability problems. Thus a new property to support inclusion of schema data is warranted.

Format Definition:

Example:

> The following is an example of this property:

8. New Components

8.1. Participant

This property is defined by the following notation:

```
participantc  = "BEGIN" ":" "PARTICIPANT" CRLF
         partprop *alarmc
         "END" ":" "PARTICIPANT" CRLF

partprop     = *(
         ;
         ; The following are REQUIRED,
         ; but MUST NOT occur more than once.
         ;
         dtstamp  / participanttype /
         ;
         ; The following are OPTIONAL,
         ; but MUST NOT occur more than once.
         ;
```

```
        created / description / last-mod / priority / seq /
        source / status / calendaraddress / summary / url /
        ;
        ; The following are OPTIONAL,
        ; and MAY occur more than once.
        ;
        attach / categories / comment /
        contact / location / rstatus / related /
        resources / strucloc / strucres / styleddescription /
        x-prop / iana-prop
        ;
        )
```

The following is an example of this component. It contains a SOURCE property which points to a VCARD providing information about the event participant.

```
        BEGIN:PARTICIPANT
        PARTICIPANT-TYPE:PRINCIPAL_PERFORMER
        SOURCE:http://dir.example.com/vcard/aviolinist.vcf
        END:PARTICIPANT
```

The following is an example for the primary contact.

```
        BEGIN: PARTICIPANT
        SOURCE;FMTTYPE=text/vcard;
         http://dir.example.com/vcard/contacts/contact1.vcf
        PARTICIPANT-TYPE:PRIMARY-CONTACT
        DESCRIPTION:A contact:
        END:PARTICIPANT
```

Component name:
    PARTICIPANT

Purpose:
    This component provides information about a participant in an event or optionally a plain text typed value.

Conformance:
    This component MAY be appear in any iCalendar component.

Description:
    This component provides information about an participant in an event, task or poll. A participant may be an attendee in a scheduling sense and the ATTENDEE property may be specified in addition. Participants in events can be individuals or organizations, for example a soccer team, the spectators, or the musicians.

    The SOURCE property if present may refer to an external definition of the participant - such as a vcard.

    The CALENDAR-ADDRESS property if present will provide a cal-address. If an ATTENDEE property has the same value the participant is considered schedulable. The PARTICIPANT component can be used to contain additional meta-data related to the attendee.

Format Definition:

Note:

> When the PRIORITY is supplied it defines the ordering of PARTICIPANT components with the same value for the TYPE parameter.

Example:

Example:

8.2. Schedulable Participant

A PARTICIPANT component may represent someone or something that needs to be scheduled as defined for ATTENDEE in [RFC5545] and [RFC5546]. The PARTICIPANT component may also represent someone or something that is NOT to receive scheduling messages.

A PARTICIPANT component is defined to be schedulable if

- It contains a CALENDAR-ADDRESS property
- That property value is the same as the value for an ATTENDEE property.

If both of these conditions apply then the participant defined by the value of the URL property will take part in scheduling operations as defined in [RFC5546].

An appropriate use for the PARTICIPANT component in scheduling would be to store SEQUENCE and DTSTAMP properties associated with replies from each ATTENDEE. A LOCATION property within the PARTICIPANT component might allow better selection of meeting times when participants are in different timezones.

9. Extended examples

The following are some examples of the use of the properties defined in this specification. They include additional properties defined in [RFC7986] which includes IMAGE.

9.1. Example 1

The following is an example of a VEVENT describing a concert. It includes location information for the venue itself as well as references to parking and restaurants.

```
BEGIN:VEVENT
CREATED:20170216T145739Z
DESCRIPTION: Piano Sonata No 3\n
 Piano Sonata No 30
DTSTAMP:20171116T145739Z
DTSTART;TZID=America/New_York:20170315T150000Z
DTEND;TZID=America/New_York:20170315T163000Z
LAST-MODIFIED:20170216T145739Z
SUMMARY:Beethoven Piano Sonatas
UID:123456
STRUCTURED-LOCATION;LABEL="The venue";VALUE=URI:
 http://dir.example.com/venues/big-hall.vcf
STRUCTURED-LOCATION;LABEL="Parking for the venue";VALUE=URI:
 http://dir.example.com/venues/parking.vcf
IMAGE;VALUE=URI;DISPLAY=BADGE;FMTTYPE=image/png:h
 ttp://example.com/images/concert.png
BEGIN:PARTICIPANT
PARTICIPANT-TYPE:SPONSOR
SOURCE:http://example.com/sponsor.vcf
END:PARTICIPANT
```

```
BEGIN:PARTICIPANT
PARTICIPANT-TYPE:PERFORMER:
SOURCE:http://www.example.com/people/johndoe.vcf
END:PARTICIPANT
END:VEVENT
```

9.2. Example 2

The following is an example of a VEVENT describing a meeting. One of the attendees is a remote participant.

```
BEGIN:VEVENT
CREATED:20170216T145739Z
DTSTAMP:20101116T145739Z
DTSTART;TZID=America/New_York:20170315T150000Z
DTEND;TZID=America/New_York:20170315T163000Z
LAST-MODIFIED:20170216T145739Z
SUMMARY:Conference plaaning
UID:123456
ORGANIZER:mailto:a@example.com
ATTENDEE;PARTSTAT=ACCEPTED;CN=A:mailto:a@example.com
ATTENDEE;RSVP=TRUE;CN=B:mailto:b@example.com
BEGIN:PARTICIPANT
PARTICIPANT-TYPE:ACTIVE:
SOURCE:http://www.example.com/people/b.vcf
LOCATION:At home
END:PARTICIPANT
END:VEVENT
```

10. Security Considerations

Applications using these properties 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.

Security considerations relating to the "ATTACH" property, as described in [RFC5545], are applicable to the "STRUCTURED-DATA" property.

11. Privacy Considerations

Properties with a "URI" value type can expose their users to privacy leaks as any network access of the URI data can be tracked. Clients SHOULD NOT automatically download data referenced by the URI without explicit instruction from users. This specification does not introduce any additional privacy concerns beyond those described in [RFC5545].

12. IANA Considerations

This section defines updates to the tables defined in [RFC5545] and new tables.

12.1. Additional iCalendar Registrations

12.1.1. Properties

This document defines the following new iCalendar properties to be added to the registry defined in Section 8.2.3 of [RFC5545]:

Property	Status	Reference
CALENDAR-ADDRESS	Current	RFCXXXX, Section 7.2
PARTICIPANT-TYPE	Current	RFCXXXX, Section 7.1
SOURCE	Current	RFCXXXX, Section 6
STRUCTURED-DATA	Current	RFCXXXX, Section 7.6
STYLED-DESCRIPTION	Current	RFCXXXX, Section 7.3
STRUCTURED-LOCATION	Current	RFCXXXX, Section 7.4
STRUCTURED-RESOURCE	Current	RFCXXXX, Section 7.5

12.1.2. Parameters

This document defines the following new iCalendar property parameters to be added to the registry defined in Section 8.2.4 of [RFC5545]:

Property Parameter	Status	Reference
LOCTYPE	Current	RFCXXXX, Section 5.1
ORDER	Current	RFCXXXX, Section 5.3
RESTYPE	Current	RFCXXXX, Section 5.2
SCHEMA	Current	RFCXXXX, Section 5.4

12.1.3. Components

This document defines the following new iCalendar components to be added to the registry defined in Section 8.3.1 of [RFC5545]:

Component	Status	Reference
PARTICIPANT	Current	RFCXXXX, Section 8.1

12.2. New Registration Tables

This section defines new registration tables for PARTICIPANT-TYPE and RESTYPE values. These tables may be updated using the same approaches laid down in Section 8.2.1 of [RFC5545]

12.2.1. Participant Types

The following table has been used to initialize the participant types registry.

Participant Type	Status	Reference
ACTIVE	Current	RFCXXXX, Section 7.1
INACTIVE	Current	RFCXXXX, Section 7.1
SPONSOR	Current	RFCXXXX, Section 7.1

Participant Type	Status	Reference
CONTACT	Current	RFCXXXX, Section 7.1
BOOKING-CONTACT	Current	RFCXXXX, Section 7.1
EMERGENCY-CONTACT	Current	RFCXXXX, Section 7.1
PUBLICITY-CONTACT	Current	RFCXXXX, Section 7.1
PLANNER-CONTACT	Current	RFCXXXX, Section 7.1
PERFORMER	Current	RFCXXXX, Section 7.1
SPEAKER	Current	RFCXXXX, Section 7.1

12.2.2. Resource Types

The following table has been used to initialize the resource types registry.

Resource Type	Status	Reference
PROJECTOR	Current	RFCXXXX, Section 5.2
ROOM	Current	RFCXXXX, Section 5.2
REMOTE-CONFERENCE-AUDIO	Current	RFCXXXX, Section 5.2
REMOTE-CONFERENCE-VIDEO	Current	RFCXXXX, Section 5.2

13. Acknowledgements

The author would like to thank Chuck Norris of eventful.com for his work which led to the development of this RFC.

The author would also like to thank the members of CalConnect, The Calendaring and Scheduling Consortium, the Event Publication technical committee and the following individuals for contributing their ideas and support:

Cyrus Daboo, John Haug, Dan Mendell, Ken Murchison, Scott Otis,

14. References

14.1. 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.

[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.

[RFC4589] Schulzrinne, H. and H. Tschofenig, "Location Types Registry", RFC 4589, DOI 10.17487/RFC4589, July 2006.

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

[RFC5546] Daboo, C., "iCalendar Transport-Independent Interoperability Protocol (iTIP)", RFC 5546, DOI 10.17487/RFC5546, December 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.

14.2. Informative References

[iana-property-registry] "IANA iCalendar Element Registries"

Appendix A. Open issues

None at the moment

Appendix B. Change log

calext-v09 2018-08-30 MD

- Sorted out inconsistencies in refs to 5546

calext-v08 2018-07-06 MD

- Add some text for equal ORDER values
- Switched scheduleaddress to calendaraddress in participant abnf. Also added more properties
- Fixed PARTICIPANT abnf

calext-v04 2017-10-11 MD

- Change SCHEDULE-ADDRESS to CALENDAR-ADDRESS
- Explicitly broaden scope of SOURCE
- Add initial registry for RESTYPE and move new tables into separate section.
- Fix PARTTYPE/PARTICPANT-TYPE inconsistency

calext-v03 2017-10-09 MD

- Mostly typographical and other minor changes

calext-v02 2017-04-20 MD

- Add SCHEDULE-ADDRESS property
- PARTICIPANT becomes a component rather than a property. Turn many of the former parameters into properties.
- Use existing ATTENDEE property for scheduling.

calext-v01 2017-02-18 MD

- Change ASSOCIATE back to PARTICIPANT
- PARTICIPANT becomes a component rather than a property. Turn many of the former parameters into properties.

calext-v00 2016-08-?? MD

- Name changed - taken up by calext working group

v06 2016-06-26 MD

- Fix up abnf
- change ref to ietf from daboo
- take out label spec - use Cyrus spec

v05 2016-06-14 MD

- Remove GROUP and HASH. they can be dealt with elsewhere if desired
- Change ORDER to integer >= 1.
- Incorporate Structured-Data into this specification.

v04 2014-02-01 MD

- Added updates attribute.
- Minor typos.
- Resubmitted mostly to refresh the draft.

v03 2013-03-06 MD

- Replace PARTICIPANT with ASSOCIATE plus related changes.
- Added section showing modifications to components.
- Replace ID with GROUP and modify HASH.
- Replace TITLE param with LABEL.
- Fixed STYLED-DESCRIPTION in various ways, correct example.

v02 2012-11-02 MD

- Collapse sections with description of properties and the use cases into a section with sub-sections.
- New section to describe relating properties.
- Remove idref and upgrade hash to have the reference
- No default value types on properties..

v01 2012-10-18 MD Many changes.

- SPONSOR and STRUCTURED-CONTACT are now in PARTICIPANT
- Add a STRUCTURED-RESOURCE property
- STYLED-DESCRIPTION to handle rich text
- Much more...

2011-01-07

- Remove MEDIA - it's going in the Cyrus RFC
- Rename EXTENDED-... to STRUCTURED-...
- Add TYPE parameter to SPONSOR

v00 2007-10-19 MD Initial version

Author's Address

Michael Douglass
Spherical Cow Group
226 3rd Street
Troy, NY 12180
USA
EMail: mdouglass@sphericalcowgroup.com
URI: http://sphericalcowgroup.com