

**CALCONNECT DOCUMENT CD1012**

**Type:** Report  
**Title:** An Introduction to Internet Calendaring  
**Version:** 1.0  
**Date:** 2010-10-14  
**Status:** Published  
**Source:** CalConnect

This document incorporates by reference the CalConnect Intellectual Property Rights, Appropriate Usage, Trademarks and Disclaimer of Warranty for External (Public) Documents as located at

<http://www.calconnect.org/documents/disclaimerpublic.pdf>.

# An Introduction to Internet Calendaring

## ICALNDAR

### History

The iCalendar specification was first produced by the IETF in 1998 as RFC 2445 [1]. Since then it has become the dominant standard for calendar data interchange on the internet and between devices (desktop computers, mobile phones etc). The specification was revised in 2009 as RFC 5545 [4].

Alongside iCalendar is the iTIP specification (RFC 2446 [2] and revised as RFC 5546[5]) that defines how iCalendar is used to carry out scheduling operations (for example, how an organizer can invite attendees to a meeting and receive their replies). This forms the basis for email-based scheduling using iMIP (the specification that describes how to use iTIP with email - RFC 2447 [3]).

iCalendar itself is a text-based data format. However, an XML format is also available, providing a one-to-one mapping to the text format (draft [7]).

iCalendar data files typically have a .ics file name extension. Most desktop calendar clients can import or export iCalendar data, or directly access such data over the Internet using a variety of protocols.

### Data model

The iCalendar data format has a well defined data model. "iCalendar objects" encompass a set of "iCalendar components" each of which contains a set of "iCalendar properties" and possibly other sub-components. An iCalendar property consists of a name, a set of optional parameters (specified as "key-value" pairs) and a value.

iCalendar components include:

- "VEVENT" which represents an event
- "VTODO" which represents a task or to-do

- "VJOURNAL" which represents a journal entry
- "VFREEBUSY" which represents periods of free or busy time information
- "VTIMEZONE" which represents a timezone definition (timezone offset and daylight saving rules)

"VALARM" is currently the only defined sub-component and is used to set alarms or reminders on events or tasks.

Properties include:

- "DTSTART" which represents a start time for a component
- "DTEND" which represents an end time for a component
- "SUMMARY" which represents a title or summary for a component
- "RRULE" which can specify rules for repeating events or tasks (for example, every day, every week on Tuesdays, etc)
- "ORGANIZER" which represents the calendar user who is organizing an event or assigning a task
- "ATTENDEE" which represents calendar users attending an event or assigned a task

In addition to this data model and the pre-defined properties, the specification defines how all those are used together to define the semantics of calendar objects and scheduling. The semantics are basically a set of rules stating how all the components and properties are used together to ensure that all iCalendar products can work together to achieve good interoperability. For example, a rule requires that all events must have one and only one "DTSTART" property. The most important part of the iCalendar specification is the semantics of the calendaring model that it represents. The use of text or XML to encode those is secondary.

## Scheduling

The iTIP specification defines how iCalendar objects are exchanged in order to accomplish the key task needed to schedule events or tasks. An example of a simple workflow is as follows:

- To schedule an event, an organizer creates the iCalendar object representing the event and adds calendar users as attendees.

- The organizer then sends an iTIP "REQUEST" message to all the attendees.
- Upon receipt of the scheduling message, each attendee can decide whether they want to attend the meeting or not.
- Each attendee can then respond back to the organizer using an iTIP "REPLY" message indicating their own attendance status.

iTIP supports other types of scheduling messages, for example, to cancel meetings, add new instances to a repeating meeting, etc.

## Extensibility

iCalendar was designed to be extensible, allowing for new components, properties and parameters to be defined as needed. A registry exists to maintain the list of standard extensions with references to their definitions to ensure anyone can use them and work well with others.

# CALENDAR DATA ACCESS AND EXCHANGE PROTOCOLS

## Internet Calendar Downloads

Calendar data can be downloaded from a web server via a web browser and then imported directly into an iCalendar aware client. Alternatively, calendar data could be retrieved in some other fashion from the Internet (e.g., sent via email as an attachment) and then imported into a calendar client.

This solution works well for calendar data that is not likely to change over time (for example the list of national holidays for the next year).

## Internet Calendar Subscriptions

An Internet calendar subscription is simply an iCalendar data file made available on a web server. However, rather than using a web browser to download the calendar data, calendar clients can instead use the URL to the calendar data on the web server to download the calendar data themselves. Additionally, the clients can check the web server on a regular

basis for updates to the calendar data, and then update their own cached copy of it. This allows calendar data that changes over time to be kept synchronized.

## CalDAV

CalDAV is a calendar access protocol and is defined in RFC 4791 [6]. The protocol is based on WebDAV which is an extension to HTTP that provides enhanced capabilities for document management on web servers.

CalDAV is used in a variety of different environments, ranging from very large internet service providers, to large and small corporations or institutions, and to small businesses and individuals.

CalDAV clients include desktop applications, mobile devices and browser-based solutions. It can also be used by "applets", for example, a web page panel that displays a user's upcoming events.

One of the key aspects of CalDAV is its data model. Simply put, it defines a "calendar home" for each calendar user, within which any number of "calendars" can be created. Each "calendar" can contain any number of iCalendar objects representing individual events, tasks or journal entries. This data model ensures that clients and servers can interoperate well.

In addition to providing simple operations to read, write and delete calendar data, CalDAV provides a querying mechanism to allow clients to fetch calendar data matching specific criteria. This is commonly used by clients to do "time-range" queries, i.e., find the set of events that occur within a given start/end time period.

CalDAV also supports access control allowing for features such as delegated calendars and calendar sharing.

CalDAV also specifies how scheduling operations can be done using the protocol, rather than, for example, sending scheduling messages via email. Whilst it uses the semantics of the iTIP protocol, it simplifies the process by allowing simple calendar data write operations to trigger the sending of scheduling messages, and it has the server automatically process the receipt of scheduling messages. Scheduling can be done with other users on the CalDAV server or with calendar users on other systems (via some form of "gateway").

## ActiveSync/SyncML

ActiveSync [8] and SyncML [9] are technologies that allow multiple devices to synchronize data with a server, with calendar data being one of the classes of data supported. These have

typically been used for mobile devices with a broad spectrum of capabilities, but most often in "smart" phones.

## CalWS

CalWS [10] is a web services calendar access application programming interface developed by The Calendaring and Scheduling Consortium and the OASIS organization, to be used as part of the OASIS WS-Calendar standard. It provides a programming interface to access and manipulate calendar data stored on a server that is more suited to a web services programming environment. It follows a similar data model to CalDAV and has been designed to co-exist with a CalDAV service offering the same data.

## iSchedule

iSchedule [11] is a protocol to allow scheduling between users on different calendaring systems and across different internet domains. It transports iTIP scheduling messages using HTTP between servers. Servers use DNS and various security mechanisms to determine the authenticity of messages received.

It has been specifically designed to be independent of any calendar system in use at the endpoints, so that it is compatible with many different systems. This allows organizations with different calendar systems to exchange scheduling messages with each other, and also allows a single organization with multiple calendar systems (for example due to mergers, or different departmental requirements) to exchange scheduling messages between users of each system.

# REFERENCES

[1] <https://datatracker.ietf.org/doc/rfc2445/> : 'Internet Calendaring and Scheduling Core Object Specification'

[2] <https://datatracker.ietf.org/doc/rfc2446/> : 'iCalendar Transport-Independent Interoperability Protocol'

[3] <https://datatracker.ietf.org/doc/rfc2447/> : 'iCalendar Message-Based Interoperability Protocol'

[4] <https://datatracker.ietf.org/doc/rfc5545/> : 'Internet Calendaring and Scheduling Core Object Specification'

[5] <https://datatracker.ietf.org/doc/rfc5546/> : 'iCalendar Transport-Independent Interoperability Protocol'

[6] <https://datatracker.ietf.org/doc/rfc4791/> : 'Calendaring Extensions to WebDAV'

[7] <https://datatracker.ietf.org/doc/draft-daboo-et-al-icalendar-in-xml/> : 'xCal: The XML format for iCalendar'

[8] <http://msdn.microsoft.com/en-us/library/aa913903.aspx> : ActiveSync

[9] <http://www.openmobilealliance.org/tech/affiliates/syncml/syncmlindex.html> : SyncML

[10] <http://www.calconnect.org/pubdocs/CDIO11%20CalWS-Rest%20Restful%20Web%20Services%20Protocol%20for%20Calendaring.pdf> : 'Restful Web Services Protocol for Calendaring'

[11] <http://tools.ietf.org/id/draft-desruisseaux-ischedule-01.txt> : 'Internet Calendar Scheduling Protocol'