

CALCONNECT DOCUMENT CD 0702

Type: Report
Title: January 2007 CalConnect Interoperability Test Report
Version: 2.0
Date: 2007-04-24
Status: Published
Source: IOPTTEST Technical Committee

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>.

TITLE: CalConnect Interoperability Test Event Report – January 2007
Novell Complex, Provo, UT
Version: 2.0
Date of Publication: April 24, 2007
Document Author: Patricia Egen
Contributing Authors: Tony Becker, Cyrus Daboo, Mike Douglass, Tomas Hnetila , Dave Nutall, Simon Vaillancourt

STATEMENT OF INTELLECTUAL PROPERTY RIGHTS

This document and the information it contains is the work product of The Calendaring and Scheduling Consortium (“Consortium”), and as such, the Consortium claims all rights to any intellectual property contained herein.

STATEMENT OF APPROPRIATE USAGE

Standards Setting Organizations and others who find that this document is of use in their work are hereby granted the right to copy, redistribute, incorporate into their own documents, make derivative works from, and otherwise make further use of the document and the material it contains at no cost and without seeking prior permission from the Consortium, subject to properly attributing the source if unmodified to the Consortium and notifying the Consortium of its use according to the guidelines below:

1. If the document is excerpted or used in its entirety in another document, the text must remain unchanged and a complete citation must be supplied referencing the full title, version, date, and appropriate section/subsection/paragraph identification from the original document.
2. A normative or informative reference to this document may be used in place of excerpting or incorporating the entire original document. Such references should include the full title, version, date and appropriate section/subsection/paragraph identification from the Consortium document being referenced.
3. In either case, the user referencing or excerpting a Consortium document is requested to notify the Consortium of the referencing specification and to provide the Consortium with an appropriate link or other way of reviewing the specification.

DISCLAIMER OF WARRANTY

THIS DOCUMENT AND THE INFORMATION IT CONTAINS IS PROVIDED ON AN “AS IS” BASIS, WITHOUT ANY WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, FROM THE CONSORTIUM, ITS CONTRIBUTORS, AND THE ORGANIZATIONS ITS CONTRIBUTORS REPRESENT OR ARE SPONSORED BY (IF ANY), INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR ANY PARTICULAR PURPOSE, AND NON-INFRINGEMENT.

TABLE OF CONTENTS

Introduction.....	3
General Comments.....	3
What was tested:.....	3
CALDAV Testing Matrix:	3
CALDAV testing	6
Examples of items found were:.....	6
Example of things tested:.....	6
iCalendar Testing:	7
Summary.....	7

CALCONNECT INTEROPERABILITY TEST EVENT
January 2007 – Novell – Provo, UT

Introduction

This document contains notes and results from the January 2007 calendar interoperability event held at the Novell complex in Provo, Utah. The basic purpose of the event was to test CALDAV Free Busy and Scheduling and iCalendar iMIP and iTIP events.

The chart below shows the attendees, their organization and the products they were testing.

<u>Attendees</u>	<u>Organization</u>	<u>Products</u>
Chuck Norris	EVDB	Eventful server
Mikael Rogers	OSAF	Chandler client
Mike Douglass	RPI	Bedework server
Cyrus Daboo	Apple	Apple server and client
Wilfredo Sanchez	Apple	Apple server
Simon Vaillancourt	Oracle	Oracle CALDAV
Tomas Hnetila	Kerio	CALDAV server, icalendar
Stepan Potys	Kerio	CALDAV server, icalendar
Tony Becker	Marware	Project X CALDAV client, iCalendar
Dave Nutall	Novell	Groupwise
Rand Babcock	Novell	Groupwise, CALDAV server

General Comments

What was tested:

The following applications and products were tested:

Four CALDAV servers - RPI, Oracle, Apple, and OSAF
 Four CALDAV clients – OSAF, Kerio, Marware and Apple
 iCalendar interoperability – Kerio, Novell, Oracle, Eventful and Marware

CALDAV Testing Matrix:

The following chart shows the specific items validated during CALDAV testing:

1.	Event creation.
1.1.	Create new single-instance meeting titled "Meeting 1.1" with the location "Durham".
1.2.	Create new meeting titled "Meeting 1.2" recurring every Monday from 10:00 AM to 11:00 AM for 4 weeks.
1.3.	Create new single-instance meeting titled "Meeting 1.3" with 2 other attendees.
1.4.	Create new single-instance meeting titled "Meeting 1.4" with an alarm set to trigger 15 minutes prior to the schedule time of the meeting.
2.	Event modification
2.1.	Modify the title of meeting "Meeting 1.1" to "Meeting 1.1bis".
2.2.	Modify the location of the meeting "Meeting 1.1bis" to "Seattle bis".
2.3.	Reschedule meeting "Meeting 1.1bis" to the next day.
2.4.	Add an attendee to "Meeting 1.1bis".
2.5.	Add an alarm to "Meeting 1.1bis".
2.6.	Modify the title of the 1st instance of the recurring meeting created in 1.2.
2.7.	Modify the participation status of the 1st attendee in meeting 1.3 to DECLINED.
2.8.	Cancel the 4th instance of the recurring meeting created in 1.2.

2.9.	One client changes "Meeting 1.1bis" to a different time, second client 'refreshes' its display to see the modification.
3.	Event retrieval
3.1.	calendar-query REPORT
3.1.1.	No filtering (match everything)
3.1.1.1.	Query all components and return all data. (tests <calendar-query> and <filter>)
3.1.1.2.	Query all components and return ETag WebDAV property and all data. (tests <calendar-query>+<DAV:prop> and <filter>)
3.1.1.3.	Query all components and return just entire VEVENT components. (tests <calendar-query> , <filter>+<comp-filter>)
3.1.1.4.	Query all components and return VEVENT components with only DTSTART, DTEND/DURATION, SUMMARY, UID, SEQUENCE, RRULE, RDATE, EXRULE, EXDATE, RECURRENCE-ID. (tests <calendar-query> , <filter>+<comp-filter> , <calendar-data>+<comp>+<prop>)
3.1.2.	time-range filtering
3.1.2.1.	Query all components within a one day time-range and return all data. Make sure that there is a recurring event that starts prior to the chosen time-range but has one non-overridden instance within the time-range. (tests <calendar-query> , <filter>+<time-range>)
3.1.2.2.	Query all components within a one week time-range and return just entire VEVENT components. Make sure that there is a recurring event that starts prior to the chosen time-range but has one overridden instance within the time-range. (tests <calendar-query> , <filter>+<time-range>)
3.1.3.	component based filtering
3.1.3.1.	Query all components that contain an embedded VALARM component. (tests <calendar-query> , <filter>+<comp-filter>)
3.1.3.2.	Query all components that contain an embedded VALARM component whose trigger falls within a specific time-range. (tests <calendar-query> , <filter>+<comp-filter>+<prop-filter>+<time-range>)
3.1.4.	property based filtering
3.1.4.1.	Query all components that contain any ORGANIZER property. (tests <calendar-query> , <filter>+<prop-filter>+<is-defined>)
3.1.4.2.	Query all components that contain an ORGANIZER property with a specific CUA text value case-insensitively. (tests <calendar-query> , <filter>+<prop-filter>+<text-match>+<caseless>)
3.1.4.3.	Query all components that contain an ORGANIZER property with a specific CUA text value case-sensitively. (tests <calendar-query> , <filter>+<prop-filter>+<text-match>+<caseless>)
3.1.5.	parameter based filtering
3.1.5.1.	Query all components that contain a DTSTART property with a TZID parameter. (tests <calendar-query> , <filter>+<prop-filter>+<text-match>+<param-filter>+<is-defined>)
3.1.5.2.	Query all components that contain an ATTENDEE property with PARTSTAT=NEEDS-ACTION parameter. (tests <calendar-query> , <filter>+<prop-filter>+<text-match>+<param-filter>+<text-match>)
3.2.	calendar-multiget REPORT
3.2.1.	Query a specific href and return all data. (tests <calendar-multiget >)
3.2.2.	Query multiple hrefs (some of which do not exist) and return all data. (tests <calendar-multiget >)
3.2.3.	Query a specific href and return ETag WebDAV property and all data. (tests <calendar-multiget >+<DAV:prop >)
3.2.4.	Query multiple hrefs (some of which do not exist) and return ETag WebDAV property and all data. (tests <calendar-multiget >+ <DAV:prop >)
3.2.5.	Query a specific href and return VEVENT components with only DTSTART, DTEND/DURATION, SUMMARY, UID, SEQUENCE, RRULE, RDATE, EXRULE, EXDATE, RECURRENCE-ID. (tests <calendar-query > , <calendar-data>+<comp>+<prop>)

3.2.6.	Query multiple hrefs (some of which do not exist) and return VEVENT components with only DTSTART, DTEND/DURATION, SUMMARY, UID, SEQUENCE, RRULE, RDATE, EXRULE, EXDATE, RECURRENCE-ID. (tests <calendar-query >, <calendar-data>+<comp>+<prop>)
4.	Event deletion
4.1.	Delete a single non-recurring meeting.
4.2.	Delete a single recurring meeting with no overridden instances.
4.3.	Delete a single recurring meeting with overridden instances.
4.4.	Delete a non-overridden instance of a recurring meeting.
4.5.	Delete an overridden instance of a recurring meeting.
5.	Access Control
5.1.	View access control details on current user's main calendar.
5.2.	Change access control details on current user's main calendar to add another user with read-only access. Verify that other user can view the calendar but not change it.
5.3.	Change access control details on current user's main calendar to add another user with read-write access. Verify that other user can view the calendar and change it. Verify that changes done by one user are seen by the other.
5.4.	Remove another user's access to the current user's main calendar and verify they can no longer access the calendar.
6	Calendar Management
6.1	Browse the list of calendars on the server, including the current user's personal calendars.
6.2	Create a new calendar in the current user's personal calendar space.
6.3	Create a regular collection in the current user's personal calendar space.
6.4	Create a new calendar inside the collection created in 6.3.
6.5	Delete the calendar created in 6.2.
6.6	Delete the collection created in 6.3.
7	Free Busy Reports
Setup	Create a new calendar and populate it with the following for one week: Event on Monday, 9 am - 11 am, recurs every day for five times Event on Monday, 12 pm - 1 pm, status tentative Event on Monday, 2 pm - 3 pm, status cancelled Event on Tuesday, 11 am - 12 pm Event on Tuesday, 2 pm - 4 pm, recurs every day for four times Event on Tuesday, 3 pm - 5 pm Event on Wednesday, 11 am - 12 pm, status tentative Event on Wednesday, 3 pm - 5 pm, status tentative Event on Thursday, 11 am - 12 pm, status cancelled Event on Thursday, 3 pm - 5 pm, status cancelled
7.1	Run a free-busy report for the entire week.
7.1.1	Verify two FREEBUSY periods for Monday, the second is BUSY-TENTATIVE.
7.1.2	Verify two FREEBUSY periods for Tuesday.
7.1.3	Verify four FREEBUSY periods for Wednesday, second and fourth are BUSY-TENTATIVE and one hour long.
7.1.4	Verify two FREEBUSY periods for Thursday.
7.1.5	Verify two FREEBUSY periods for Friday.
8	Scheduling
Setup	Three user accounts user1 (role Organizer), user2 (role Attendee), user3 (role Attendee) provisioned with suitable principal properties for calendar home, inbox, outbox and user addresses.
8.1	Organizer (user1) sends non-recurring message invite for Monday at 9am (1 hour) to each attendee. Verify that each attendee Inbox receives a copy of the invite.
8.2	Attendee (user2) accepts invite and sends back reply. Verify that reply is placed in Organizer Inbox.

8.3	Organizer (user1) updates invite with user2 accept state and resends invite. Verify that each attendee Inbox receives a copy of the new invite.
8.4	Attendee (user3) accepts updated invite and sends back reply. Verify that reply is placed in Organizer Inbox.
8.5	Organizer (user1) updates invite with user3 accept state and resends invite. Verify that each attendee Inbox receives a copy of the new invite.
8.6	Organizer (user1) cancels the invite. Verify that each attendee Inbox receives the cancellation.

iCalendar testing:

A iTIP test matrix and iCalendar test streams were validated against various products.

The following are generic notes that describe some of the results of the interop testing.

CALDAV testing

Several servers and clients were able to test much of the CALDAV matrix. On the server front, some minor issues were found during testing, but for the most part the servers are holding up well. Again, due to bugs found during testing, not much of the Free Busy or Scheduling was able to be tested.

On the client side, a number of issues with CALDAV interoperability with other servers were found with problems occurring on all sides. Some server problems were fixed and re-tested as working.

Examples of items found were:

- Calendar-query report not matching any event occurrences in some cases
- COPY to the same location should not be allowed
- HTTP Error 409 Conflict returned when overwrite HTTP header is false, should be 401 or 403
- VFREEBUSY component should always contain GMT times
- Publishing only a project calendar using CALDAV.
- Not deleting events from CALDAV server during next publishing.
- Needing to publish more information about tasks in event Descriptions
- Not properly handling all-day meeting invitation generated by another vendor's product.
- Using incorrect Content-Class:urn:content-classes:calendarmessage instead of Content-Class:urn:content-classes:task when sending tasks.
- client expecting an etag on collections
- One vendor adds an Organizer to the events they created.
- bugs were associated with setting of calendar properties.
- problems with user principals
- products sending many simultaneous requests.

Example of things tested:

- Ability to connect and publish VEVENT's and VTODO's to a CALDAV server
- Adding a configuration pane to the Application to support changing server connections.
- Summaries and descriptions to each task.
- Subprojects as all day events.
- TODO support and MKCalendar support
- absorb and completely ignore a vvenue component.

Cyrus Daboo of Apple created a test tool which was run against several servers at the event. A brief report on results was posted and made available to other vendors. This showed that there is still much work to do to have servers with full compliance to all details of the CALDAV spec, but progress is being made. A number of issues were reported back to vendors. Note – the report created by the CALDAV tool mentioned above can be found later in this document.

All vendors felt that the Interop Testing event was an effective way to test compatibility. Several vendors mentioned the need for more test cases for exceptions, particularly with recurring events with multiple exceptions.

iCalendar Testing:

Examples of things found during iCalendar testing

- Task interoperability issues that stem from a simple IMIP component tag missing.
- Not handling TENTATIVE status and broken Cancel.
- All day appointment expectations where there is no time (unlike Microsoft).
- And of course, the odds and ends little bugs that are always found.
- A lot is working today with most of the attendees.

Summary

As usual there were several bugs found during testing. Quite a bit more of CALDAV was able to be tested this event and several iCalendar iMIP and iTIP objects were passed among the vendors for testing. The CALDAV testing matrix is the same one used in the September 2006 testing. At that time, not everyone was supporting scheduling. We continue to test as much as we can on the new scheduling sections.

As suggested by several participants, we will be looking at some virtual interop testing between onsite events. The virtual interops are not meant to take the place of on-site testing. Too much value is deriving from the one-on-one, in person interactions. However, continued testing between events will help find discrepancies that can be resolved prior to the next onsite event.

Respectfully submitted, Pat Egen. Interoperability Event Manager

Note: The CALDAV Tester Tool Report follows:

CALDAV Tester Results – tool created by Cyrus Daboo of Apple.

The following chart shows the results of the CALDAVTester tool run by Cyrus at the Interop event. The products tested and their results are shown in the following chart:

CALDAVTester Test Script	Vendor1	Vendor2	Vendor3	Vendor4
acl	Not supported by server	Does not support per-event ACLs.	VTODO put fails;	Not supported by server.
acldisabled	Not supported by server	Not supported by server.	Not supported by server.	Not supported by server.
aclreports	Not supported by server	Not supported by server.	Gave back 400 response in some cases where a 403 or multi-status should have been returned; supported-report-set does not list all ACL reports;	Not supported by server.
attachments	Failed all - server reports error with last line of ATTACHMENT property	Passed all.	ATTACHMENT property not returned after being PUT	
availability	Not supported by server	Not supported by server.	Not supported by server.	Not supported by server.
CALDAVIOP	Passed all	Passed all.		Passed all.
calendaruserproxy	Not supported by server	Not supported by server.	Not supported by server.	Not supported by server.
copymove	COPY/MOVE not supported by server	Failed: allowed copy of event back to same calendar; return 409 instead of 412 for Overwrite:F	MKCALENDAR fails during start up	500 errors for nearly all.
delete	Passed all.	Passed all.	VTODO PUT failed, VEVENT OK.	Failed during startup: Cannot put VTODO.
depthreports		Whole bunch of multiget problems; no results coming back for calendar-query; fb property value periods not in iCalendar format -0500 offset in period values	Failed during startup: could not create calendar within new regular collection.	Failed during startup: cannot put VFREEBUSY (looks like same error as VTODO).
depthreportsacl	Not supported by server	Not supported by server.	Not supported by server.	Not supported by server.
dropbox	Not supported by server	Not supported by server.	Not supported by server.	Not supported by server.
encodedURIs	MKCALENDAR with calendar%202 fails with 400	Location header was present in a PUT response.	Creates a calendar instead of a collection.	COPYs fail.
errors		Won't allow resource in calendar home.	Failed during startup: initial PUT fails.	No DAV: error's returned.
floating	MKCALENDAR failed in startup	All fail.	Failed during startup: cannot create sub-calendar.	Failure that causes calendar to no longer open or be deleted.
get	Duplicate DTSTAMP in VEVENT response; could not write resource in calendar home	Failed but that was due to normal server restrictions.	500 for store of .ics in calendar home; does directory listing rather than returning whole calendar	Failures - but OK due to some event rewriting and directory listing.

mkcalendar	MKCALENDAR without body fails; with body does not generate proper DAV:error reponse	Bad request properties were returned as 403 response, but should have been 403 prop-status codes in a 207.	400 when no body; no DAV:error for precondition failures	No DAV:error's returned.
notifications	Not supported by server	Not supported by server.	Not supported by server.	Not supported by server.
propfind	DAV:getcontentlength not returned on a calendar collection; did not reject invalid XML; unknown XML element generated a 500	Passed all.	Accepted invalid XML.	Invalid XML accepted.
proppatch	Does not recognise non-standard DAV elements.	Unknown DAV: properties were ignored.	Got a <#test> XML element in output, plus other failures.	500 errors; xml:lang on property is lost.
proxyauthz	Not supported by server	Not supported by server.	Not supported by server.	Not supported by server.
put	VTODO's failed - duplicate DTSTAMP	Some failures - probably due to server re-writing data content.	VTODOs fail; unbounded daily events hang the server	500 errors for VTODOs.
quota	Not supported by server	Not supported by server.	Not supported by server.	Not supported by server.
recurrenceput	java.lang.NullPointerException on some recurring vtodos; vevents were OK	Passed all.	VTODOs all fail; VEVENTs OK.	VTODOs all fail; VEVENTs OK.
reports	Failed during startup whilst trying to put an alarm	Various failures as per depthreports.		Failed during startup: cannot put VFREEBUSY (looks like same error as VTODO).
schedulepost		Not supported by server.	Not supported by server.	Not supported by server.
schedulepostacl	Not supported by server	Not supported by server.	Not supported by server.	Not supported by server.
schedulepostauto	Not supported by server	Not supported by server.	Not supported by server.	Not supported by server.
schedulepostnormal		Not supported by server.	Not supported by server.	Not supported by server.
scheduleprops	Passed - but actually wrong: returned hrefs had "null" for user name	Not supported by server.	Not supported by server.	Not supported by server.