

**TITLE:** Calendar Interoperability Testing Report – Public  
September 2007 - Boston, MA – Sponsored by MIT  
**Version:** 2.0  
**Date of Publication:** December 14, 2007  
**Author:** Patricia Egen  
**Contributing Authors:** Tony Becker, Cyrus Daboo, Michael Douglass, Tomas Hnetila, Bill Le,  
Brian Moseley, Clint Talbert

## **INTELLECTUAL PROPERTY RIGHTS, APPROPRIATE USAGE AND DISCLAIMER OF WARRANTY FOR EXTERNAL (PUBLIC) DOCUMENTS**

### **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 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

<b>Introduction.....</b>	<b>3</b>
<b>General Comments.....</b>	<b>3</b>
<b>CalDAV Testing Scenarios .....</b>	<b>3</b>
<b>Vendor A Testing:.....</b>	<b>6</b>
<b>Vendor B Testing .....</b>	<b>6</b>
Vendor 1 testing: .....	6
General Comments.....	9
<b>Vendor C Testing .....</b>	<b>9</b>
Testing Scenario Table: .....	10
General comments/problems: .....	11
<b>Vendor D Testing .....</b>	<b>12</b>
Vendor 1 CalDAV testing:.....	12
Vendor 2 CalDAV testing:.....	12
Vendor 3 CalDAV testing:.....	12
General iCalendar ICS testing: .....	12
Vendor D CalDAV Scenario Testing Results .....	13
General comments. ....	14
<b>Vendor E Testing .....</b>	<b>14</b>
<b>Vendor F Testing.....</b>	<b>14</b>
General comments. ....	14
<b>Vendor G Testing.....</b>	<b>15</b>
<b>Summary.....</b>	<b>15</b>

**CALENDAR INTEROPERABILITY EVENT**  
**September 2007 – MIT – Boston, MA**

**Introduction**

This document contains notes and results from the September 2007 calendar interoperability event, sponsored by MIT, held in Boston, MA. 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.

<b>Attendees</b>	<b>Organization</b>	<b>Products</b>	<b>Version</b>
Cyrus Daboo	Apple	Apple Client and CalDAV server	9A527
Bill Le	IBM	IBM Notes and Domino calendar clients	8.01
Frank Pavelski	IBM		
Tomas Hnetila	Kerio	Kerio CalDAV server and client	
Stepan Potys	Kerio		
Tony Becker	Marware	Marware Calendar, Project X	1.2
Philipp Kewisch	Mozilla	Sunbird and Lightning Calendar clients	0.7
Client Talbert	Mozilla		
Kervin Pierre	Open Connector Groupware		
Simon Vaillancourt	Oracle	Oracle CalDAV Server	
Mikeal Rogers	OSAF	Cosmo CalDAV server and Chandler Client	0.7.0.1
Brian Moseley	OSAF		
Michael Douglass	RPI	bedework CalDAV Server	

**General Comments**

The following are notes from the testing event. Vendors are flagged as A, B, C, etc. to keep the details private, but to reflect the kinds of problems noted during testing.

The following table is the testing script used for CalDAV testing. Comments below will make reference to the numbers corresponding to this table.

**CalDAV Testing Scenarios**

<b>1.</b>	<b>Event creation.</b>
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.
<b>2.</b>	<b>Event modification</b>
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.
<b>3.</b>	<b>Event retrieval</b>
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>)
<b>4.</b>	<b>Event deletion</b>
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.
<b>5.</b>	<b>Access Control</b>
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.
<b>6</b>	<b>Calendar Management</b>
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.
<b>7</b>	<b>Free Busy Reports</b>
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	

7.1.5	Verify two FREEBUSY periods for Friday.
<b>8</b>	<b>Scheduling</b>
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.

### **Vendor A Testing:**

Vendor A testing found some minor issues. Client testing focused on interoperability with other servers.

In some cases we were not able to get past the account setup stage, in others we were able to test access and scheduling features. The account setup problem was debugged and a working copy with a fix was then used to test further.

Overall the interoperability event was very useful to us (as it always is) and we are looking forward to the next one.

### **Vendor B Testing**

The following is a summary of Vendor B's testing.

#### **Vendor 1 testing:**

Issues found:

- vendor deletes meeting response from various clients, because there isn't a correct SEQUENCE number
- One vendor doesn't recognize meeting requests caused by TIMEZONE definition – see example:  
BEGIN:VTIMEZONE  
TZID:US/Eastern  
\* BEGIN:STANDARD  
\* TZOFFSETFROM:-0400  
\* TZOFFSETTO:-0500  
\* DTSTART:19551030T020000  
\* RRULE:FREQ=YEARLY;UNTIL=20061029T060000Z;BYMONTH=10;BYDAY=-1SU  
\* TZNAME:EST  
\* END:STANDARD  
BEGIN:DAYLIGHT  
TZOFFSETFROM:-0500

TZOFFSETTO:-0400  
 DTSTART:20070311T020000  
 RRULE:FREQ=YEARLY;BYMONTH=3;BYDAY=2SU  
 TZNAME:EDT  
 END:DAYLIGHT  
 BEGIN:STANDARD  
 TZOFFSETFROM:-0400  
 TZOFFSETTO:-0500  
 DTSTART:20071104T020000  
 RRULE:FREQ=YEARLY;BYMONTH=11;BYDAY=1SU  
 TZNAME:EST  
 END:STANDARD  
 END:VTIMEZONE

- Meeting request from one vendor is silently deleted from CalDAV scheduling INBOX by another vendor probably caused by a bug in that vendors application
- iMIP meeting request from one client is not visible in CalDAV INBOX
- One client Error message "Request error - Calendar not found" appears when user creates new calendar
- Calendar is successfully created on the server, but the error message appears
- One client's provides incorrect free/busy (organizer and attendee are swapped)
- Free/Busy - Problem with case sensitive MAILTO - FIXED

Test result:

- 1.1 Ok
- 1.2 Ok
- 1.3 Ok
- 1.4 Ok
- 2.1 Ok
- 2.2 Ok
- 2.3 Ok
- 2.4 Ok
- 2.5 Ok
- 2.6 stored on the server, but some apps aren't able to display this exception
- 2.7 Ok
- 2.8 Ok
- 2.9 Ok
- 3.\* Not tested
- 4.1 Ok
- 4.2 Ok
- 4.3 Ok
- 4.4 Ok
- 4.5 Ok
- 5.\* Not tested
- 6.1 Ok
- 6.2 There is a problem when one application creates calendar on a CalDAV. Calendar is created, but other app reports - "Request error - Calendar not found".
- 6.3 N/A - app doesn't support it
- 6.4 N/A - app doesn't support it
- 6.5 Ok
- 6.6 N/A - app doesn't support it
- 7.\* Not tested
- 8.1 Ok

8.2 Mostly Ok, problem with replies from several apps because incorrect SEQUENCE in reply  
8.3 Ok  
8.4 Ok  
8.5 Mostly Ok, the same problem as 8.2  
8.6 Ok

#### Vendor 2 testing:

##### Issues:

- Why app asks for non-existing URL PROPFIND /calendars/iop-mit.test.nnnnn.com/user2/Calendar/null ?
- One CalDAV server returns HTTP/1.1 401 Unauthorized instead of HTTP/1.1 404 Not found for non-existing URL. - FIXED
- Although one app doesn't support CalDAV Free/Busy it shows Free/Busy dialog. It may be confusing for users.
- Creating event by dragging in calendar and Editing event by double-click don't work correctly.
  - A message "Item Changed on server" appears.
  - New event appears not only as event but also as TODO in one app
    - o REPORT filter doesn't work correctly

##### Test result:

1.1 Ok  
1.2 Meeting is correctly created on the server - other clients displays it correctly, but one client shows only two recurrences instead of 4  
1.3 ORGANIZER is missing in .ics file on the server  
1.4 Ok  
2.\* NOTE: Don't use double click for editing event. It doesn't work correctly  
2.1 Ok  
2.2 Ok  
2.3 Bug in a client GUI. Event is correctly changed on the server, but client doesn't show it until reload.  
2.4 Ok  
2.5 Ok  
2.6 N/A – one client doesn't support such exceptions  
2.7 N/A  
2.8 Ok  
2.9 Ok  
3.\* Not tested  
4.1 Ok  
4.2 Ok  
4.3 Ok  
4.4 Ok  
4.5 Ok  
5.\* Not tested  
6.\* client doesn't support calendar collections.  
7.\* Not tested  
8.\* client doesn't support CalDAV scheduling.

#### Vendor 3 testing:

##### Issues:

- MKCALENDAR is missing in OPTIONS



- MKCOL always returns 207 Multistatus. It should return 201 Created if no propertyupdate xml body is present in MKCOL request
- Calendar is read-only when I subscribe it. Is it a bug?

#### Vendor 4 testing:

##### Issues:

- CalDAV scheduling doesn't work at all, because server doesn't support principal search report.

##### Test result:

- 1.1 Ok
- 1.2 Ok
- 1.3 Ok
- 1.4 Ok
- 2.1 Ok
- 2.2 Ok
- 2.3 Ok
- 2.4 Ok
- 2.5 Ok
- 2.6 N/A
- 2.7 Ok
- 2.8 Ok
- 2.9 Ok
- 3.\* Not tested
- 4.1 Ok
- 4.2 Ok
- 4.3 Ok
- 4.4 Ok
- 4.5 Ok
- 5.\* Not tested
- 6.1 Ok
- 6.2 Ok
- 6.3 Created, but server returned incorrect response 207 Multi status
- 6.4 Ok
- 6.5 Ok
- 6.6 Ok
- 7.\* Not tested
- 8.\* Scheduling doesn't work at all.....

#### Vendor 5 testing:

##### Issues:

One server renames calendar when calendar name and display name are different. It causes incompatibility with one client. When the calendar renaming is disabled in the server, publishing calendars from the client works fine. Calendar renaming also causes a problem with another client.

#### General Comments

Calconnect IOP is effective way how to test compatibility.

#### **Vendor C Testing**

This vendor focused on testing iCalendar, iMIP and iTIP objects. They used our standard testing scenario and the results of each event are shown below.

Testing Scenario Table:

Task	Result
A: Non-repeating cases:	
1: User A PUBLISHes an event	Was able to PUBLISH and process an event to/from all vendors
2: User A invites Users B, C, D & E to a meeting: A: ATTACHments: 1: 0	Was able to send all vendors a single without attachments. One vendor accepted, but then vendor's user became chair and sent out 20 rescheduled notice.
2: 1	Was able to send all vendors a single with one attachment (bmp).
3: 1+	Was able to send all vendors a single with two attachment
B: ALTREPs of: 1: DESCRIPTION	Was able to send all vendors a single with AltReps of text in the description field.
2: COMMENT	Was able to send all vendors a single with AltReps of text in the comments dialog.
4: LOCATION	Was able to send all vendors a single with AltReps of text in the location field.
C: Including ALARMS 1: AUDIO only	Was able to send all vendors a single with audio alarm.
2: DISPLAY only	Was able to send all vendors a single with display alarm
3: EMAIL only	Was able to send all vendors a single with email alarm.
F: ATTENDEE property parameters: 1: CUTYPE: A: INDIVIDUAL (Default)	Was able to send to vendors individually
B: GROUP	Was able to send to GROUP of vendors
C: RESOURCE	
D: ROOM	Was able to send all vendors a single Invite with an RnR.
3: ROLE: A: CHAIR	Was able to send a single Invite with Chair role to all vendors.
B: REQ-PARTICIPANT (Default)	Was able to send a single Invite with REQ-PARTICIPANT role to all vendors.
C: OPT-PARTICIPANT	Was able to send a single Invite with OPT-PARTICIPANT role to all vendors.
4: PARTSTAT: A: NEEDS-ACTION (Default)	Was able to send a single Invite with NEEDS-ACTION.
B: ACCEPTED	Was able to send an ACCEPTED out
C: DECLINED	Was able to send a DECLINED out
D: TENTATIVE	Was able to send a TENTATIVE out
5: RSVP A: TRUE	Was able to send with RSVP=TRUE
B: FALSE (Default)	Was able to send with RSVP=FALSE
8: SENT-BY B: 1	Was able to send a single event with SENT-BY.
9: CN B: 1	Was able to send with a CN.

3: User B Accepts the invitation:	Sent two vendors a single. Two vendors can accept.
A: but then Declines the invitation	One of the vendor declines. The other vendor mentioned that iTips does not work for him yet, so he cannot accept then decline
4: User C Declines the invitation: A: but then Accepts the invitation:	One vendor accepts. . The other vendor mentioned that iTips does not work for him yet, so he cannot decline then accept
6: User E Delegates to User G: A: User G Accepts the invitation: B: User G Declines the invitation: C: User G requests a Refresh of the invitation: D: User G Counters with a new meeting time: E: User G Delegates to User I:	Couldn't test this scenario since it requires at least two other vendors which support delegation.
7: User A reschedules the meeting: Repeat permutations of 1-6 below as necessary.	
B: Repeating cases: (Repeat A. subcases but expand for instance manipulation including entire set, 1 instance, THISANDPRIOR & THISANDFUTURE ranges  Tests should include the following permutations: RDATEs only RRULEs only RDATEs and RRULEs RDATEs & EXDATEs only RRULEs & EXDATEs only RDATEs & EXRULEs only RRULEs & EXRULEs only RDATEs, EXDATEs & EXRULEs RRULEs, EXDATEs & EXRULEs RDATEs, RRULEs & EXDATEs RDATEs, RRULEs & EXRULEs RDATEs, RRULEs, EXDATEs & EXRULEs )	Was able to send everyone a standard repeating.  Vendors REPLIED with no problem.  Was able to send a resch on time of the whole repeating meeting  Was able to send a reschedule on dates of one instance, thisandprior, thisandfuture instances. One vendor replied with no problem, but we receive error when open the accepted notice  Was able to send a confirm to a single meeting with comments.  Was able to send a cancellation to everyone.  Was able to send a weekly repeating meeting (every other week for 7 weeks). Vendors replied with no problem

#### General comments/problems:

Overall here are problems found at this interoperability event test:

- 1) One vendor accepted our invite, but vendor became chair of the meeting and sent out at least 20 rescheduled notices.
- 2) There is an iCalendar invite from one vendor came in with a complicated RRULE, and we lost one instance when processing it.
- 3) One vendor user accepted our invite with an image, but the image lost when come to our Inbox
- 4) One vendor accepted our multiple reschedule events, and we receive error when opening one of those accepted notice
- 5) One vendor received error when trying to process our task invite. This is a long time, known issue.

## Vendor D Testing

The following are this vendor's testing notes. Scheduling and free/busy testing with CalDAV servers was ignored during this testing.

Of the ICS files, we actually only fail to parse 2 of them. The issues there stem from our parsers requiring a DTSTART when there is a DTEND and not expecting a "timeless" VEVENT" (we handle timeless VTODOs just fine).

We tested the items on the sheet, we also tested VTODO handling on the CalDAV servers.

In general, we didn't find any issues with the CalDAV servers, but we found plenty of issues where we have some mis-steps.

Below are notes from items found during the testing.

### Vendor 1 CalDAV testing:

- Inline editing of title - get a 422 error – another vendor works with this

### Vendor 2 CalDAV testing:

- Giant Attendee list has very poor performance (300 attendees)
- CalDAV Todo oddities 396116
- XPROPs in the alarms when you click on DISMISS and when you click on SNOOZE.

### Vendor 3 CalDAV testing:

- Edit by drag does not send the update to the server if you type in a new title - same issue as another server? Another vendor works with this
- Sending a <url>\null when talking to the CalDAV provider. Why are we doing this? Are we trying to construct a URI and hitting a null javascript object?
- With one server, once we hit this editing wierdness, we continually get the "Submit Change" or "Discard Changes" dialog popping up.
  - If you click "Submit changes" you get nowhere - you stay in the same state.
  - If you click "Discard Changes" the changes are resubitted and the change goes through.
  - Could those buttons be miswired? - not verified that this is not the case, but we are not updating the UI with the new result, and are not really updating the UI if the "submit anyway" fails.
  - The "reload" from the discard seems to refresh the items on the calendar whereas the "submit anyway" doesn't.
- When working with this server, we are creating VTODOs in the UI for each event. We do not export these, and we do not bring them down from the server. Somehow they simply appear in the TODO UI.

### General iCalendar ICS testing:

- Bug fixed - application doesn't handle timezone definitions without a TZNAME specified in it
- If something has a DTSTART and DURATION, our application puts a DTEND on it before sending it up to the CalDAV/ICS server. This is wrong!
- Not able to change attendee PARTSTAT through the event dialog for a specific attendee
- Import of an ICS into a storage calendar (like home) is not refreshing the month view
- Cannot import items that are indented in ICS files (entire file has 3 spaces to the left, for instance, and indented lines have another extra space).
- 4.2.2.ics fails - no start/end on vevent

# Vendor D CalDAV Scenario Testing Results

Item #	CalDAV Servers					Comments
	Srv1	Srv2	Srv3	Srv4	Srv5	
1.1.	P	P	P	P	P	Creating calendar with one server, the URL must be properly cased. If it is incorrectly cased, we fail to create the calendar claiming that the resource is DAV but not calDAV
	P	P		P	P	
1.2.						
1.3.	P	P	P	P	P	
1.4.	P	P	P	P	P	Issues with editing inline on two servers, but editing with opened dialog is ok. One server update failed due to problematic DURATION handling on client side.
	P & F	P	P	P	P & F	
2.1.						
2.2.	P	P	P	P	F due to 2.1	
2.3.	P	P	P	P	P	(not accessible through UI) – the attendee would have to respond with a Decline response.
2.4.	P	P	P	P	P	
2.5.	P	P	P	P	P	
2.6.	P	P	P	P	P	
	N	N	N	N	N	
2.7.						
2.8.	P	P	P	P	P	
2.9.	P	P	P	P	P	
3.1.						
3.1.1.	P	P	P	P	P	
3.1.1.1.						
3.1.1.2.	P	P	P	P	P	
3.1.1.3.	P	P	P	P	P	
3.1.1.4.						
3.1.2.						
3.1.2.1.	P	P	P	P	P	
3.1.2.2.	P	P	P	P	P	
3.1.3.						
3.1.3.1.						
3.1.3.2.						
3.1.4.						
3.1.4.1.						
3.1.4.2.						
3.1.4.3.						
3.1.5.						
3.1.5.1.						
3.1.5.2.						
3.2.						

3.2.1.	P	P	P	P	P
3.2.2.	P	P	P	P	P
3.2.3.	P	P	P	P	P
3.2.4.					
3.2.5.					
3.2.6.					
4.1.	P	P	P	P	P
4.2.	P	P	P	P	P
4.3.	P	P	P	P	P
4.4.	P	P	P	P	P
4.5.	P	P	P	P	P

#### General comments.

Thanks for a very useful and well-run session.

#### **Vendor E Testing**

We did limited testing with one client, only a half hour or so, but everything looked good.

Another client was tested heavily against one server all week. No issues were ever reported to me. It's possible that the these folks will have client errors to report.

One client was not really tested. There's a bug in the client that keeps it from accepting absolute urls in DAV responses. I spent most of Monday working on our application but by the time that was available to test, other vendors had no time to test.

#### **Vendor F Testing**

For the first time, server to server functionality was tested between 3 vendors and many issues were fixed in time for a demo at the end of the interop.

Floating event issues were discovered when doing tests with one vendor.

Overall it was a very interesting interop mostly due to the new server to server functionality. There should put extra effort put into inviting more CalDAV client implementors in the upcoming interop events.

#### **Vendor G Testing**

Several clients and servers were tested. Some do not support floating time or do not fall back to TZ on calendar collection.

Another server doesn't support "setting" properties on new collections, no inbox?

One server created, but can't propfind (they move collection just after creation) - was fixed and then the server worked.

One server had no issued and worked right off.

#### General comments.

New scheduling was untested and new Free/Busy was untested.

## **Vendor H Testing**

This vendor focused on free/busy and scheduling CalDAV testing. These are their notes.

Monday - mostly fixing bugs we encountered between two vendors' servers. By the end of the day that seemed to work OK.

Tuesday - spent more time working with another vendor server.

Wednesday we discovered a few more problems.

The end result was we had working scheduling between one client and one server and server to server between three CalDAV server. .

Some of the problems we ran into were mostly in the form of data returned.

One plugin is unable to process some forms of valid freebusy.

There was very little time for testing against any other clients.

## **Summary**

We continue to have good results testing CalDAV clients and servers. General comments again are that it is always good to have interops in person.

We need to get more vendors in to test iCalendar, iMIP and iTIP objects, particularly with respect to changes to the specification that came out of the CALSIFY working group at the IETF.

The Free/Busy testing and demos are starting to gain headway. We hope to do more intensive testing at future interoperability events as more clients and servers support the specifications.

Respectfully submitted, Pat Egen. Interoperability Event Manager