CALCONNECT.ORG
The CALENDARING & SCHEDULING CONSORTIUM

**Title:** Mobile Recurrence Interoperability Recommendations

**Version:** 1.0

**Date:** 29 July 2008

**Contributors:** TC-Mobile Technical Committee

**Editors:** Chris Dudding, Symbian Ltd
Mark Paterson, Oracle Corporation

**Status:** Published

This document describes interoperability problems observed in the implementation of recurring events in mobile devices and provides recommendations for how these issues can be avoided.

# INTELLECTUAL PROPERTY RIGHTS, APPROPRIATE USAGE, TRADEMARKS 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.

## TRADEMARKS

All terms used in this document that are known to be trademarks or service marks have been appropriately capitalized; however the Consortium cannot guarantee that its information is accurate in all cases. The use of any trademark or service mark in this document should not be taken as affecting its validity.

## 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.

# Executive Summary

The ability to share calendar information among different applications and across network boundaries has become an important business need, as a growing number of organizations look for ways to leverage their investments in collaborative applications.

The Mobile Technical Committee (TC-MOBILE) of the Calendaring & Scheduling Consortium published the results of a mobile calendaring questionnaire in July 2006. Of concern were answers related to calendar synchronization. Synchronization was one of the main things users did, but it was also singled out as one of the main things that did not work well yet. This can be attributed, in a large part, to issues related to data object interoperability.

One of the main issues is that iCalendar has not been widely adopted within certain application spaces. Although adopted by all major time management solution vendors, there has been reluctance within the mobile industry to migrate from vCalendar (iCalendar's predecessor) based solutions and to fully embrace iCalendar. To help persuade mobile vendors the Mobile Technical Committee (TC-MOBILE) of the Calendaring & Scheduling Consortium recently published a white paper titled: The Benefits of iCalendar for the Mobile Industry [MOBILEICAL].

Even with iCalendar based solutions however significant issues often exist with recurring events. The Recurrence Technical Committee (TC-RECURR) of the Calendaring & Scheduling Consortium published a full set of problems and recommendations: iCalendar RECURRENCE PROBLEMS AND RECOMMENDATIONS [RECURISSUES].

The Mobile Calendar Interoperability Test Suite [TESTSUITE] describes a test suite to assess a mobile device's capability to synchronize calendar data with a calendar store. The repeating events section of this test suite often reveals the interoperability issues regarding these types of events.

This white paper further explores issues related to recurring events that are specific to the mobile space and recommends possible solutions for both client and server vendors.

## Problems with the implementation of Recurrences on Mobile Devices

Although mobile calendar synchronization solutions have matured over the past years, providing for the most part reliable mobile-side representations of user's calendar information, severe problems can exist with synchronization of repeating events. Often the calendar on the device does not accurately reflect irregular or changing instances of repeating events on the server. This basic mismatch can then turn into corruption of the server data if these irregularities are sent back to the server for example in the case where an OMA Data Synchronization slow sync might be triggered.

There are several causes of mismatched device and server recurring events that are directly related to the way in which devices support [iCal] (in many cases [vCal]).

- Some devices do not support either sending or receiving RDATE or EXDATE although they support RRULE.
- Some devices support one or the other.
- Some devices don't support any of the three.

For implementers that insist on continuing to use [vCal] rather then [iCal] there is the issue that [vCal] does not support the notion of making any changes to an instance of a recurring event other than rescheduling the start time and date; yet many events also change their LOCATION. The use of the [iCal] RECURRENCE-ID property would easily solve this issue but implementers need to be willing to move to [iCal].

Even if an implementation supports RRULE the actual set of rules that the implementation can handle is normally far less then what is defined in [iCal].

One large underlying problem surrounds the misinterpretation of the OPTIONAL nature of some properties in [vCal] and [iCal]. This has lead to the belief that supporting the full set of properties is not required, resulting in poor interoperability between products that support different sub-sets of properties.

In order to deal with these interoperability problems there needs to be a proper understanding between client and server implementers on what should be supported. The next section defines three levels of support. The two sections following it provide recommendations on how both device implementations and server implementations should deal with each level.

## Recurrence Support Levels for Mobile Devices

Level 0: Device provides no support for repeating events

The easiest level of support for a mobile device to provide is not to support repeating events at all. Although this restricts the usability of the mobile calendar, clients that do not support repeating events are straightforward for server implementations to react to accordingly.

Level 1: Device can handle recurrence rules (i.e. supports RRULE)

If a mobile device allows a user to create a repeating event then it should support the following recurrence patterns:

1. Daily, Weekly, Monthly by date, Yearly
2. Plus: Monthly by day
3. Plus: Weekdays
4. Plus: Repeating every n weeks

These repeating events should be sent to the server using RRULE accordingly and the client should be able to accept these same RRULEs coming from the server.

Level 2: Device can handle recurrence rules with exceptions and extra dates (i.e. supports RRULE, RDATE, EXDATE and RECURRENCE-ID)

If a mobile device supports this level of support then users should be able to create repeating events as defined as part of Level 1 and in addition be able add exceptions and extra dates.

These repeating events should be sent to the server using an appropriate combination of RRULE, RDATE, EXDATE and RECURRENCE-ID properties and the client should be able to accept these same combinations coming from the server.

## Recommendations for Synchronization Clients (Mobile Devices)

**Recommendation #1**: The starting point for solving interoperability issues is for all implementations to be based on [iCal].

**Recommendation #2**: The level of recurrence support supported by your implementation should comply with one of the 3 levels defined in this paper

**Recommendation #3**: The client's calendar store should have Level 2 recurrence support to achieve the best interoperability.

Level 0: OMA DS based solutions should not indicate support for any [iCal] related recurrence properties within their device information object.

Level 1: OMA DS based solutions should indicate support for the [iCal] RRULE property within their device information object and by doing so are indicating they support all of the recurrence patterns defined as required for this level of support.

Level 2: OMA DS based solutions should indicate support for the [iCal] RRULE, RDATE, EXDATE and RECURRENCE-ID properties within their device information object and by doing so are indicating they support all of the recurrence patterns defined as required for this level of support as well as support for modified and delete exceptions as well as support for extra dates.

By adhering to these recommendations server implementations can then reliably react to these 3 levels of support as defined in the following section.

It is strongly encouraged that support for recurrence level 2 is provided in the client's calendar store. This will ensure that the complex recurrences can received and displayed, even if the mobile user interface restricts the creation and modification of events to level 1 recurrences.

## Recommendations for Synchronization Servers

**Recommendation #1**: The starting point for solving interoperability issues is for all implementations to be based on [iCal].

**Recommendation #2**: Your implementation must be able to react to the level of recurrence support reported by the client implementation connecting to your server as follows:

Level 0: All repeating events should be expanded and single instance events should be sent to the mobile device. This should not in any way affect the fact that server side the event is considered recurring. The mobile limitation should not degrade the level of support provided by the Calendar store.

A full set of test cases designed to assess a server's ability to deal with a mobile device claiming level 0 support can be found in [TESTSUITE].

Level 1: The server should be able to support receiving the [iCal] RRULE property from device implementations. Repeating events created on the server, which adheres to the patterns defined as required for level 1 support, should be sent to the device as an RRULE. Any repeating event that is created with a pattern more advanced then those defined for level 1 support however should be expanded and single instance events should be sent to the mobile device (as is defined for level 0 support). Finally events that get edited on the server which were sent to the device using an RRULE which now contain any form of exception or extra date should result in a delete being sent to the device to remove the repeating event and the newly edited event on the server should be expanded and single instance events should be sent to the mobile device (as is defined for level 0 support).

A full set of test cases designed to assess a server's ability to deal with a mobile device claiming level 1 support can be found in [TESTSUITE].

Level 2: Support should be as defined for Level 1 except that if a user edits a repeating event on the device the server must be able to accept the appropriate combination of RRULE, RDATE, EXDATE and RECURRENCE-ID that will result and if such an edit is done server side it should send the device the appropriate combination (rather than a delete and expanding the edited meeting as defined for level 1).

The server should not expand repeating events to single instance (non-repeating) events. The RDATE property can be used represent advanced recurrence patterns beyond what is defined in level 1 support. This will ensure the instances are shown as a single recurrence set on the device.

A full set of test cases designed to assess a server's ability to deal with a mobile device claiming level 2 recurrence support can be found in [TESTSUITE].

## Conclusion

Significant interoperability issues often exist with recurring events. This is damaging user confidence in mobile calendar synchronization solutions. The support levels defined in this paper and the recommendations for both client and server implementations, if adhered to, should go a long way to helping address these interoperability issues.

By reacting to the levels of support per these recommendations server implementations can ensure that the user always sees an accurate representation of repeating events on their mobile device. The fact that an event is part of a recurrence may not make it to the mobile device in some cases but users will always have an accurate representation of their day.

## References

**[MOBILEICAL]** "The Benefits of iCalendar for the Mobile Industry", CalConnect$^{SM}$, http://www.calconnect.org/publications/iCalendarforthemobileindustryv1.0.pdf

**[RECURISSUES]** "iCalendar RECURRENCE PROBLEMS AND RECOMMENDATIONS", CalConnect$^{SM}$, http://www.calconnect.org/publications/icalendarrecurrenceproblemsandrecommendationsv1.0.pdf

**[TESTSUITE]** "The Mobile Calendar Interoperability Test Suite", CalConnect$^{SM}$, http://www.calconnect.org/publications/mobilecalendartestsuitev1.0.pdf

**[iCal]** "RFC2445 – Internet Calendaring and Scheduling Core Object Specification", IETF, http://www.ietf.org/rfcs/rfc2445.txt

**[vCal]** "The Electronic Calendaring and Scheduling Exchange Format", Internet Mail Consortium, http://www.imc.org/pdi/vcal-10.txt