vObject Minimum Interoperability Profile

Approved Version 1.0 – 02 Oct 2007

Open Mobile Alliance

OMA-TS-vObjectOMAProfile-V1_0-20071002-A

Each Open Mobile Alliance member has agreed to use reasonable endeavors to inform the Open Mobile Alliance in a timely manner of Essential IPR as it becomes aware that the Essential IPR is related to the prepared or published specification. However, the members do not have an obligation to conduct IPR searches. The declared Essential IPR is publicly available to members and non-members of the Open Mobile Alliance and may be found on the "OMA IPR Declarations" list at http://www.openmobilealliance.org/ipr.html. The Open Mobile Alliance has not conducted an independent IPR review of this document and the information contained herein, and makes no representations or warranties regarding third party IPR, including without limitation patents, copyrights or trade secret rights. This document may contain inventions for which you must obtain licenses from third parties before making, using or selling the inventions. Defined terms above are set forth in the schedule to the Open Mobile Alliance Application Form.

NO REPRESENTATIONS OR WARRANTIES (WHETHER EXPRESS OR IMPLIED) ARE MADE BY THE OPEN MOBILE ALLIANCE OR ANY OPEN MOBILE ALLIANCE MEMBER OR ITS AFFILIATES REGARDING ANY OF THE IPR'S REPRESENTED ON THE "OMA IPR DECLARATIONS" LIST, INCLUDING, BUT NOT LIMITED TO THE ACCURACY, COMPLETENESS, VALIDITY OR RELEVANCE OF THE INFORMATION OR WHETHER OR NOT SUCH RIGHTS ARE ESSENTIAL OR NON-ESSENTIAL.

THE OPEN MOBILE ALLIANCE IS NOT LIABLE FOR AND HEREBY DISCLAIMS ANY DIRECT, INDIRECT, PUNITIVE, SPECIAL, INCIDENTAL, CONSEQUENTIAL, OR EXEMPLARY DAMAGES ARISING OUT OF OR IN CONNECTION WITH THE USE OF DOCUMENTS AND THE INFORMATION CONTAINED IN THE DOCUMENTS.

Contents

Tables

1. Scope

This document defines the vObject minimum interoperability profile based on the vCard 2.1, vCalendar 1.0 and vBookmark1.0 specifications. This document also serves as an implementation guideline to ensure consistent and unambiguous interpretation of the relevant base specifications. This specification is aimed at the mobile communications community to foster interoperability across different terminals, platforms, and user agents.

The vObject minimum interoperability profile is to be used in following use cases

- vObject exchange between mobile terminals (peer-to-peer) via local interface e.g. Bluetooth, infrared
- vObject attached to an email or a MMS message
- vObject downloading from a web server

The enablement of applications and advanced function such as data synchronization requires additional metadata and/or mechanism when sending updated vObjects. For this specification, the enablement of such features is out of scope. It is expected that these advance functions will be addressed by a complementary OMA activity focusing on a longer term interoperability solution.

Ambiguities in the vCard2.1, vCalendar1.0 and vBookmark1.0 specifications coupled with the lack of well-known implementation guidelines has led to various commercial implementations that result in interoperability problems. For instance, vCard2.1, vCalendar1.0 and vBookmark1.0 provide several options for content types and file extensions definitions but without defining mandatory support, vCard2.1 allows both ISO 8601basic and extended format to specify date and time, the minimum size of the vObject is not defined and etc.

Therefore the objectives of this specification are three fold.

- to define a set of vObject conformance
- to constrain the implementation choices from original vCard2.1, vCalendar1.0 and vBookmark1.0 specifications
- to clarify the ambiguities of the syntax contained in those specifications.

Interoperability of vObjects covers a range of issues. It is the intention of this specification to address the issues of physical data formatting for these objects as well as specifying a standard presentation for commonly used data properties. These issues have been identified as a subset of the larger set of interoperability; it does not attempt to address all the interoperability issues known to exist with the specifications.

2. References

2.1 Normative References

[IOPPROC] "OMA Interoperability Policy and Process", Version 1.1, Open Mobile Alliance™, OMA-IOP-Process-V1_1, URL:http://www.openmobilealliance.org/

[RFC2119] "Key words for use in RFCs to Indicate Requirement Levels", S. Bradner, March 1997, URL:http://www.ietf.org/rfc/rfc2119.txt

[RFC2234] "Augmented BNF for Syntax Specifications: ABNF". D. Crocker, Ed., P. Overell. November 1997, URL:http://www.ietf.org/rfc/rfc2234.txt

[vCard2.1] "vCard The Electronic Business Card Version 2.1", A versit Consortium Specification, September 18, 1996

URL: http://www.imc.org/pdi/vcard-21.doc

[vCal1.0] "vCalendar The Electronic Calendaring and Scheduling Exchange Format Version 1.0", A versit Consortium Specification, September 18, 1996

URL: http://www.imc.org/pdi/vcal-10.doc

[vBook1.0] "IrDA Infrared Mobile Communications v1.1"

URL: http://www.irda.org/

2.2 Informative References

[SyncML-IOP-Whitepaper] "SyncML Common Object Interoperability White Paper", Open Mobile Alliance™, OMA-DS-2004-0052-Interoperability-White-Paper on OMA DS web portal, URL:http://www.openmobilealliance.org/

[RFC2821] "Simple Mail Transfer Protocol", J. Klensin, AT&T Laboratories, April 2001

http://www.faqs.org/rfcs/rfc2821.html

[RFC2045] "Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies", N. Freed, Innosoft, N. Borenstein, First Virtual, November 1996

[RFC2822] "Internet Message Format", P. Resnick, Editor, QUALCOMM Incorporated, April 2001

http://www.faqs.org/rfcs/rfc2822.html

[RFC1521] "MIME(Multipurpose Internet Mail Extensions) Part One:Mechanisms for Specifying and Describing the Format of Internet Message Bodies", N.Borenstein, Bellcore, N.Freed, Innosoft, September 1993

http://www.faqs.org/rfcs/rfc1521.html

3. Terminology and Conventions

3.1 Conventions

The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described in [RFC2119].

All sections and appendixes, except “Scope” and “Introduction”, are normative, unless they are explicitly indicated to be informative.

This is an informative document, which is not intended to provide testable requirements to implementations.

3.2 Definitions

vObject composer	An application for user to compose vObject data
vObject parser	An application to parse and register vObject data

3.3 Abbreviations

JPEG	Joint Photographic Expert Group
UTC	Universal Time Coordinated

4. Introduction

The motivation for the vObject OMA minimum interoperability profile is to provide a physical data formatting of vObjects (vCard in [vCard2.1], vCalendar in [vCal1.0] and vBookmark in [vBook1.0]) as well as specifying a standard presentation format for commonly used data properties.

The vObject OMA minimum interoperability profile is a subset of [vCard2.1], [vCal1.0] and [vBook1.0].

5. vObject Implementation Guidelines

This chapter presents implementations guideline for vCard2.1, vCalendar1.0 and vBookmark1.0.

5.1 Recommended Practice with SMTP/MIME

Following table specifies the content types and file extensions that MUST be supported when attaching a vCard2.1, vCalendar1.0 or vBookmark1.0 object to an email, a MMS message, or when downloaded by a browser. Filenames for vObjects MUST use only US-ASCII Character set (e.g., Character sets other than US-ASCII MUST NOT be used in filenames).

vObject	Content-Type	File Extension
vCard2.1	text/x-vCard	.vcf
vCalendar1.0	text/x-vCalendar	.vcs
vBookmark1.0	text/plain	.url (parser MUST support both .url and .vbm)

Table 1: vObject Content Type and File Extension Definition

The conformance requirements on vObject composer and parser are shown below.

The vCard composer MUST support text/x-vCard as content type and .vcf as file extension of a vCard object. The vCard parser MUST be able to identify a vCard object by text/x-vCard content type and .vcf file extension.

The vCalendar composer MUST support text/x-vCalendar as content type and .vcs as file extension of a vCalendar object. The vCalendar parser MUST be able to identify a vCalendar object by text/x-vCalendar content type and .vcs file extension.

The vBookmark composer MUST support text/plain as content type and .url as file extension of a vBookmark object. The vBookmark parser MUST be able to identify a vBookmark object by text/plain content type with both .url and .vbm file extensions.

5.2 Escape

The following characters appearing in the property parameter value MUST be escaped with a backslash character (ASCII 92)

The “backslash””backslash” is an escape for the backslash character.

- Semicolon appearing in a property parameter value
- Forward slash appearing in a property parameter value
- '<' appearing in a property parameter value
- '>' appearing in a property parameter value
- '&' appearing in a property parameter value

5.3 Delimiters

Individual lines within the vCard data stream are delimited by a ([RFC2822]) CRLF line break sequence (US-ASCII decimal 13, followed by US-ASCII decimal 10). Long lines of text can be wrapped into a multiple-line representation using the [RFC2822] “folding” technique. A description of how a folding procedure is performed can be found in [SyncML-IOP-Whitepaper]. A brief description is described below.

- Devices SHOULD fold long lines around the last LWSP character before a non-LWSP character should there be more than one LWSP separating non-LWSP characters.

- Devices SHOULD only perform wrapping on property value fields and not on their names or parameters.

A formatted text line break in a property value must also be represented by ([RFC2822]) a CRLF line break sequence. However, since the CRLF sequence is used to delimit a line, property values with formatted line breaks (i.e., multiple lines) must be encoded using alternate encoding of either Quoted-Printable or Base64, as defined in [RFC1521]. Quoted-Printable lines of text must also be limited to less than 76 characters per line. The 76 characters per line do not include the CRLF ([RFC2822]) line break sequence. WSP MUST NOT be placed in front of the property name and the property value MUST be terminated with a CRLF line break sequence.

5.4 Encoding and Character Set

[vCard2.1] section2.1.5 Encodings "The default encoding for the vCard object is 7-bit. The default encoding may be overridden for an individual property value by using the “ENCODING” property parameter. This parameter value can be “BASE64”, “QUOTED-PRINTABLE”, or “8BIT”. Some transports (e.g., MIME based electronic mail over SMTP) may also provide an encoding property at the transport wrapper level. This transport wrapper level property can be used in these cases for transporting a vCard data stream that has been defined using a default encoding other than 7-bit (e.g., 8-bit). "

[vCard2.1] section 2.1.6 Character set :"The default character set is US-ASCII. The default character set can be overridden for an individual property value by using the “CHARSET” property parameter. "

[vCard2.1] section 2.1.5 character set "Some transports (e.g., MIME based electronic mail over SMTP) may also provide a character set property at the transport wrapper level. This property can be used in these cases for transporting a vCard data stream that has been defined using a default character set other than ASCII (e.g., ISO-2022-JP)."

The following steps are the recommended procedures in [SyncML-IOP-Whitepaper] when using the ENCODING and CHARSET parameters in vCard and vCalendar objects.

1. If the string's character set is US-ASCII then no CHARSET parameter is required; if the string's character is not US-ASCII and the character set does not match the one specified at the transport level, a CHARSET parameter MUST be included specifying which character set is to be used.

2. Examine the string for unsafe Quoted-Printable characters. If there are one or more unsafe Quoted-Printable characters in the string, each of the unsafe characters must be encoded and specified using the QUOTED-PRINTABLE encoding parameter.

6. vCard2.1 Minimum Interoperability Profile and Implementation Guideline

This chapter defines the minimum interoperability profile for vCard2.1 and provides guidelines for implementation

6.1 vCard2.1 Minimum Interoperability Profile

The following table defines the supported properties and corresponding semantics for the vCard object.

For the properties included in the vCard minimum interoperability profile, M shown for the composer indicates that the composer MUST be able to compose the property. VERSION property and N property MUST have instances. Other properties can have null (0x00h) as values. The composer MAY remove any property other than VERSION property and N property if the property does not have a value. M shown for the parser indicates that the parser MUST parse and register the property value.

O shown for the composer indicates that the composer SHOULD be able to compose the property. O shown for the parser indicates that the parser SHOULD be able to parse and register the property value.

The table below shows the vCard2.1 minimum interoperability profile and the composer conformance criteria. Parser conformance criteria is described in the specific property description.

To be able to parser a vCard object properly, the parser MUST have at least 320 bytes of buffer size.

Property	Semantic Description	Composer Conformance
FN	The formatted name string associated with the vCard object.	O
N	A structured representation for the name of the person, place or thing associated with the vCard object	M
PHOTO	An image or photograph of the individual associated with the vCard	O
BDAY	The date of birth of the individual associated with the vCard	O
ADR	A structured representation of the physical delivery address for the vCard object	O
TEL	A canonical number string for a telephone number for telephony communication with the vCard object	M
EMAIL	Address for electronic mail communication with the vCard object	M
NOTE	Supplemental information or a comment that is associated with the vCard object	O
SOUND	A sound annotation for the vCard object	O
URL	A value that represents a URL	O
VERSION	The value of this property must be 2.1 to correspond to this specification.	M
REV	This property specifies the combination of the calendar data and time of day of the last update to the vCard object.	O

Table 2: vCard2.1 Minimum Interoperability Profile

6.2 vCard2.1 Property Implementation Guideline

The format of vCard2.1 object corresponds to the [vCard2.1] specification. This chapter provides implementation guidelines for the vCard2.1 properties and parameters, and provides informative usage examples.

6.2.1 FN Property

Refers to [vCard2.1] specification

6.2.2 N Property

At a minimum, sending of 32 bytes data (excluding the semicolon ";") of N property value MUST be supported.

6.2.3 PHOTO Property

Image data is defined by the TYPE parameter and MUST be encoded in BASE64. The delimiter of the image data MUST follow the guidelines specified in Chapter 5.3. Inclusion of an inline style MUST be supported.

The TYPE parameter is used to specify the graphics format for the PHOTO property value. JPEG MUST be supported as the value for the TYPE parameter.

Example: PHOTO;ENCODING=BASE64;TYPE=JPEG:

R01G0Ddghalienlidghlldgi/

6.2.4 BDAY Property

Time MUST be specified by ISO8601 basic form (extended format MUST NOT be used).
Example Format is "YYYYMMDD"

6.2.5 ADR Property

At a minimum, an input of 100 bytes data (excluding the semicolon ";") MUST be supported for the address field.

6.2.6 TEL Property

The composer of the terminal MUST support at least 3 phone numbers. At a minimum, sending of 23 digits of each TEL data MUST be supported.

The composer MUST support WORK, HOME, VOICE, CELL, FAX property parameters. The order sequence MUST be followed when combining property parameters.

TEL;["PREF;"]["WORK;"/"HOME;"]"VOICE;"/"FAX;"/"MSG;"/"CELL;"/PAGER;"/"BBS;"/"MODEM;"/"CAR;"/"ISDN;"/"VIDEO;"

The Property parameter combinations shown in table 3 MUST be supported on the composer side.

The parser MUST be able to parse and register at least 3 phone numbers.

The parser MUST be able to capture the data type shown in table 3 correctly.. The Parser MUST be able to register TEL data even if it receives a TEL property parameter it does not support. Which TEL type the parser registers the data to is left to implementation.

The parser SHOULD understand the PREF property parameter. The parser MUST register TEL;PREF first (if the parser receives several TEL;PREF the data MUST be registered following the order that TEL;PREF data comes in) then register data in the order that the property parameter comes in.

TYPE value	Semantic Description
TEL;VOICE	personal wired phone number
TEL;CELL	personal mobile phone number
TEL;WORK;VOICE	business wired phone number
TEL;WORK;FAX	Business fax number
TEL;WORK;CELL	Business mobile phone number

Table 3: TEL Type and Semantics

Example: TEL;HOME;VOICE:00000000000

TEL;WORK;VOICE:11111111111

6.2.7 EMAIL Property

The composer of the terminal MUST support at least three email addresses. At a minimum, support of 50 bytes of each EMAIL property value MUST be supported..

Composer MUST support property parameters shown in table 4.

The parser MUST be able to parse and register at least three email addresses.

The parser MUST be able to capture the data type shown in table 5 correctly. The parser MUST be able to register EMAIL data even if it receives EMAIL property parameter it does not support, which EMAIL type the parser registers the data to is left to implementation.

The parser SHOULD understand the PREF property parameter. The parser MUST register EMAIL;PREF first (if the parser receives several EMAIL;PREF the data MUST be registered following the order that the EMAIL;PREF data comes in) then register data in the order that the property parameter comes in.

Example: EMAIL;INTERNET:abc@xxx.xxx.xxx

TYPE value	Semantic Description
EMAIL;INTERNET	Default
EMAIL;CELL	MMS mobile email address
EMAIL;HOME	personal email address
EMAIL;WORK	business email address

Table 4: EMAIL Type and Semantics

6.2.8 NOTE Property

Folding included in NOTE must be encoded according to Chapter5.3.

Example: NOTE;ENCODING=QUOTED-PRINTABLE: this facsimile machine if operational =

0830 to 1715 hours=0D=0A

Monday through Friday. Call +1-213-555-1234 if you have problems=0D=0A=

with access to the machine.

6.2.9 SOUND Property

For the usage of X-IRMC-N/ORG to add sound annotation, the composer MUST compose SOUND property according to IRMC version 1.1 Errata 2000 07 18.

If X-IRMC-N/ORG is not present, the parser SHALL parse and register the value according to [vCard2.1] specification.

Errata 2000 07 18 constrains the value of the SOUND property to a string, usage example in Errata 2000 07 18 is shown below

N:last;first;;;

SOUND;X-IRMC-N:last_pronunciation;first_pronunciation;;;

ORG:IrDA

SOUND;X-IRMC-ORG:errderr

The CHARSET and ENCODING parameters are used in the same way as with the N field

6.2.10 URL Property

Refer to [vCard2.1] specification.

6.2.11 VERSION Property

The composer MUST support the VERSION property and the value for this property MUST be 2.1 to correspond to this specification. The Value MUST NOT be null(0x00h) or other values. The composer MUST compose the VERSION property directly after BEGIN:VCARD.

The parser MUST be able to handle VERSION property regardless of the order in which it is embedded in a vCard object.

Example: VERSION:2.1

6.2.12 REV Property

Refer to [vCard2.1] specification.

Last revision time MUST be specified by ISO8601 basic form (extended format MUST NOT be used).

Example: REV:20040511T005000Z

UTC MUST be used.

7. vCalendar1.0 Minimum Interoperability Profile and Implementation Guideline

This chapter defines the vCalendar1.0 minimum interoperability profile and implementation guidelines. The vCalendar object includes the vEvent object and the vTodo object.

For the properties included in the vCalendard minimum interoperability profile, M shown for the composer indicates that the composer MUST be able to compose the property. VERSION property MUST have instances. Other properties can have null (0x00h) as values. M shown for the parser indicates that the parser MUST parse and register the property value.

O shown for the composer indicates that the composer SHOULD be able to compose the property O shown for the parser indicates that the parser SHOULD be able to parse and register the property value.

The table below shows the vCalendar1.0 minimum interoperability profile and composer conformance criteria. Parser conformance criteria is described in the specific property description.

To be able to parse a vCalendar object properly, the parser MUST have at least 370 bytes of buffer size

7.1 vCalendar1.0 Minimum Interoperability Profile

Table5 shows supported properties of the vEvent object. Table6 shows supported properties of the vTodo object.

Property	Semantic Description	Composer Conformance
VERSION	The value of this property must be 1.0 to correspond to this specification	M
AALARM	An audio reminder for the vCalendar object	O
CATEGORIES	Categories (e.g. ,BUSINESS, HOLIDAY)	M
CLASS	Access classification (e.g. PUBLIC, PRIVATE)	O
DESCRIPTION	A more complete description of the vCalendar object	M
DALARM	A display reminder for the vCalendar object	O
DTEND	The date and time that the event will end	M
LOCATION	The intended location for a vCalendar object	O
RRULE	A rule or repeating pattern for a recurring vCalendar object	M
DTSTART	The date and time that the event will start	M
SUMMARY	A short summary or subject of the vCalendar object	O

Table 5: vEvent Minimum Interoperability Profile

Property	Semantic Description	Composer Conformance
VERSION	The value of this property must be 1.0 to correspond to this specification	M
AALARM	An audio reminder for the vCalendar entity	O
CATEGORIES	Categories e.g. BUSINESS, HOLIDAY	M
CLASS	Access classification	O
COMPLETED	The date and time that the todo event was actually completed	M
DESCRIPTION	A more complete description of the vCalendar entity	M

DALARM	A display reminder for the vCalendar entity	O
DUE	The date and time that the todo is due to be completed	M
PRIORITY	Priority for the vCalendar object	O
STATUS	Status associated with the vCalendar entity	O
SUMMARY	A short summary or subject of the vCalendar entity	O

Table 6: vTodo Minimum Interoperability Profile

7.2 vCalendar1.0 Implementation Guideline

This chapter defines implementation guidelines for vCalendar 1.0.

7.2.1 Specifying Times

7.2.1.1 UTC and Local time

Unless specified otherwise, all times MUST be specified as UTC. A UTC time MUST use the trailing Z form of ISO8601 basic.

Example: 20040511T005000Z

Where specified, local times MAY be used. Local times are specified by not including a time zone in the ISO8601 basic form (extended format MUST NOT be used).

Example: 20040511T005000

A vCalendar parser MAY accept local times where possible. A parser MUST treat all times specified without a time zone as local time. Parsers which do not understand time zones MAY interpret all non-UTC times as local time. A parser SHOULD treat a local time as the time in the parser's current time zone.

7.2.1.2 All Day Events

[vCal1.0] specification does not define how to describe an all day event, a solution can be found in [SyncML-IOP-Whitepaper] and is described below.

Devices SHOULD represent an all day event as an event in local time in the ISO8601 format, starting at 00:00:00 and ending at 24:00:00. The date field for both the start and end times SHOULD be identical.

Example of an all day event:

```
 BEGIN:VEVENT

 DTSTART:20021224T000000

 DTEND:20021224T240000

 END:VEVENT
```

In order to avoid confusion with an actual 24-hour event which begins at midnight, those events SHOULD indicate a begin time of 00:00:00 and end time of 00:00:00 the following day. The time 24:00:00 SHOULD not be used except to represent an all day event.

Example of a 24-hour event beginning at midnight, UTC:

```
 BEGIN:VEVENT

DTSTART:20021224T000000Z
```

DTEND:20021225T000000Z

END:VEVENT

7.2.2 VERSION Property

The composer MUST support VERSON property. The value of this property MUST be 1.0 to correspond to this specification. The value of VERSION MUST NOT be NULL(0x00h) or other values. The composer MUST compose the VERSION property directly after BEGIN:VCALENDAR.

The parser MUST be able to handle VERSION property regardless of the order in which it is embedded in a vCalendar object

Example: VERSION:1.0

7.2.3 AALARM Property

Alarm time MUST be specified by ISO8601 basic form (extended format MUST NOT be used).

Example: AALARM:20040511T005000Z

UTC MUST be used.

7.2.4 CATEGORIES Property

PERSONAL, HOLIDAY, TRAVEL, BUSINESS, MEETING MUST be supported as property values. Other values CAN be used.

Example: CATEGORIES:MEETING

7.2.5 CLASS Property

.Refer to [vCal1.0] specification.

Example: CLASS:PRIVATE

7.2.6 DESCRIPTION Property

This property provides a more complete description of the vCalendar entity than that provided by the SUMMARY property. Line break appears in this property following the rule described in Chapter 5.3. At a minimum, the composer MUST support the sending of 200 bytes of data for the DESCRIPTION property.

Example: DESCRIPTION:deadline of vObject specification

7.2.7 DALARM Property

Time of alarm MUST be specified in accordance to ISO 8601 basic form format (extended format MUST NOT be used)

Example: DALARM:20040511T005000Z;MEETING

UTC MUST be used.

7.2.8 DTEND Property

Local time MUST be used to specify all day events. UTC MUST be used for other than all day events.

7.2.9 LOCATION Property

Rsefer to [vCal1.0] specification.

Example: LOCATION:Conference Room

7.2.10 RRULE Property

The repeating pattern for a recurring vCalendar entity MUST be based on the Basic Recurrence Rule Grammar of XAPIA's CSA. Daily, weekly, monthly and yearly recurrence rules MUST be supported. The rule MUST be created to be applied to UTC time

Example: RRULE:W1#5 repeating 5 times a week

7.2.11 DTSTART Property

Local time MUST be used to specify all day events. UTC MUST be used for other than all day events.

7.2.12 SUMMARY Property

At a minimum, the composer MUST support the sending of 40 bytes of data for the SUMMARY property.

Example: SUMMARY: shopping

7.2.13 COMPLETED Property

Time MUST be specified in accordance to ISO 8601 basic form format (extended format MUST NOT be used) Example: COMPLETED:20040512T000000Z

UTC MUST be used.

7.2.14 DUE Property

Time MUST be specified in accordance to ISO 8601 basic form format (extended format MUST NOT be used)

Example: DUE: 20040512T000000Z

UTC MUST be used.

7.2.15 PRIORITY Property

When the vCalendar composer supports the setting of a calendar entries priority the composer SHALL conform to [vCal1.0] in the following way:

- provide the priority property whenever the priority is explicitly set, and
- provide a undefined/default priority by not inserting the PRIORITY property or by explicitly inserting a the priority property with value "0" (PRIORITY:0) and
- provide the priority property with value "1" (PRIORITY:1) for the highest priority of vCalendar notification when the composer supports different priorities, and
- provide the priority property with value "9" (PRIORITY:9) for the lowest priority of vCalendar notification when the composer supports different priorities, and

Use property values between "2" (PRIORITY:2) to "8" (PRIORITY:8) respectively for decreasing ordinal priority below the highest priority (PRIORITY:1) but of higher priority than the lowest priority (PRIORITY:9) when additional granularity of priority is supported .

The undefined/default value MAY be considered a Normal priority. The Normal priority MAY also be provided by explicitly inserting a priority property with value "5" (PRIORITY:5)

The vCalendar composer SHALL be able to compose value "0", value "1" and value "9" and MAY be able to compose value "2" to value "8" according to the semantics described above.

The vCalendar parser SHALL conform to [vCal1.0] in the following way when PRIORITY is supported:

- support at least the undefined/default value or its specific value of “0” (PRIORITY:0), and
- treat vCalendar objects with the priority property set to value “1” (PRIORITY:1) as the highest priority of vCalendar notification, and
- treat vCalendar objects with the priority property set to value “9” (PRIORITY:9) as the lowest priority of vCalendar notification
- treat vCalendar objects where the values are “2” (PRIORITY:2) or higher, up to “9” (PRIORITY:9), as successively lower priorities than the highest priority.

The vCalendar parser SHALL support at least the undefined/default (PRIORITY:0), a high (PRIORITY:1) and low (PRIORITY:9) priority and MAY support additional priorities when PRIORITY is supported.

The undefined/default value whether specified (PRIORITY:0), or specified (PRIORITY:5) or not present MAY be considered the normal priority by the vCalendar parser when supported. Such a normal priority should be considered approximately a middle priority

Example: PRIORITY:2

7.2.16 STATUS Property

Refer to [vCal1.0] specification.

Example: STATUS:NEEDS ACTION

8. vBookmark1.0 Minimum Interoperability Profile and Implementation Guideline

This chapter defines the vBookmark minimum interoperability profile and implementation guidelines.

For the properties included in the vBookmark minimum interoperability profile, M shown for the composer indicates that the composer MUST be able to compose the property. VERSION property MUST have instances. Other properties can have null as values. M shown for the parser indicates that the parser MUST parse and register the property value.

O shown for the composer indicates that the composer SHOULD be able to compose the property. O shown for the parser indicates that the parser SHOULD be able to parse and register the property value.

The table 7 shows the vBookmark1.0 minimum interoperability profile and composer conformance criteria. Parser conformance criteria is described in the specific property description.

To be able to parse a vBookmark object properly, the parser Should have at least 1087 bytes of buffer size.

8.1 vBookmark1.0 Minimum Interoperability Profile

This chapter defines the vBookmark minimum interoperability profile.

Table 7 shows supported vBookmark properties.

Property	Semantic Description	Composer Conformance
VERSION	The value of this property must be 1.0 to correspond to this specification	M
URL	A value that represents a URL	M
TITLE	Title of the bookmark	M
X-IRMC-URL	Internet shortcut	O

Table 7: vBookmark1.0 Minimum Interoperability Profile

8.2 vBookmark1.0 Implementation Guideline

This chapter defines the vBookmark implementation guidelines.

8.2.1 VERSION Property

The composer MUST support the VERSION property. The value of this property must be 1.0 to correspond to this specification, null(0x00h) or other values MUST NOT be used. The composer MUST compose the VERSION property directly after BEGIN:VBKM.

The parser MUST be able to handle VERSION property regardless of the order in which it is embedded in a vBookmark object.

Example: VERSION:1.0

8.2.2 URL Property

The composer SHOULD compose 1024 bytes URL, the composer may restrict the composition length to 256 bytes. The parser SHALL be able to parse and register 1024 bytes URL.

Example: URL: http://www.openmobilealliance.org

8.2.3 TITLE Property

The composer MUST support the sending of 24 bytes of data for the TITLE property.

Example: TITLE:open mobile alliances

8.2.4 X-IRMC-URL

Refer to [vBook1.0] specification

The value for this property MUST be encoded using QUOTED-PRINTABLE

Example:

BEGIN:ENV

X-IRMC-URL;ENCODING=QUOTED-PRINTABLE:=

[InternetShortcut]=

URL=http://www.openmobilealliance.org

END:ENV

Appendix A. Change History (Informative)

A.1 Approved Version 1.0 History

Reference	Date	Description
OMA-TS-vObjectOMAProfile-V1_0	02 Oct 2007	Status changed to Approved by TP TP Ref# OMA-TP-2007-0347-INP_vObject_V1_0_ERP_for_Final_Approval.doc

Appendix B. DAYLIGHT property

The DAYLIGHT property is an optional item in the [vCal1.0] specification that describes the daylight savings time rule for the originating calendar system. If this property is not specified for a recurring appointment, it is possible that a calendar system in a different time zone could store the appointment incorrectly. For example, assume that someone in New York (Eastern Time Zone) creates a recurring meeting that occurs once a week at 10:00 AM local time and shares it with her colleague in Bangalore. Everything will be fine until the US enters daylight savings time in April, at which point the time of the meeting will change from 10:00 AM EST (15:00 GMT) to 10:00 AM EDT (14:00 GMT). If the calendar system was not aware of this time change, it would be in error, since Bangalore does not use daylight savings time. Simply saving the meeting in GMT will not solve the problem, since the "local" time of the meeting remains constant (10:00 AM) while the "absolute" time of the meeting changes (15:00 to 14:00 GMT). However, if the vCalendar object contained the DAYLIGHT property, the calendar system would know that it had to adjust the time of the meeting when the originator entered daylight savings time, and the calendar in Bangalore would be correct.

Note that the DAYLIGHT property may not work for recurring appointments that last longer than one year, because the exact DST transition date may change from year to year.

Appendix C. Recurring Events

For many enterprises, recurring events with exceptions and rescheduling of single occurrences of recurring events happens frequently. With [vCal1.0], exceptions to regularly occurring repeating events are represented by EXDATE and RDATE. EXDATE is used to represent the dates to be removed or excluded from the series, and RDATE can be used to represent the rescheduled date.

So, for example, if a meeting were to occur every Monday, it could be represented by RRULE. If for one week, the meeting were cancelled, the exception for that week could be represented by EXDATE. If, additionally, the weekly meeting would be held on Wednesday instead of Monday, the new date for that week could be represented using RDATE.

Here is an example of a portion of a vEvent which illustrates the use of these properties:

BEGIN:VEVENT DTSTART:20040927T170000Z DTEND:20040927T180000Z RRULE:D7 20050627T170000Z RDATE:20041020T180000Z EXDATE:20041004T170000Z;20041018T170000Z

This weekly (or Daily, occurring every 7 days) event starts on 9/27/2004 and ends on 6/27/2005, but 10/04 and 10/18 are excluded. For the week of 10/04 there is no meeting. For the week of 10/18, the Monday meeting is replaced by a meeting on 10/20 Wednesday one hour later.

[vCal1.0] allows the user to represent changes in date and time for a recurring series with EXDATE and RDATE, but it is not until vCal 2.0 that we have the vocabulary to represent other changes to the event (e.g. Location) with the use of RECURRENCE-ID.

Appendix D. Static Conformance Requirements (Normative)

The notation used in this appendix is specified in [IOPPROC].

The SCRs are categorised in terms of composer and parser rather than client and server. The rational for this is that a client needs to support the parser function or composer function or both. Likewise a server needs to support the parser function or composer function or both.

D.1 VObject General Support

Item	Feature / Application	Reference	Status	Requirement
vObjProfile -Gen-C-001	vObject support		M	vObjProfile-Gen-C-002 OR vObjProfile-Gen-C-003 OR vObjProfile-Gen-C-004
vObjProfile -Gen-C-002	vObject vCard 2.1 support		O	vObjProfile-vCardC-C-000 OR vObjProfile-vCardP-C-000
vObjProfile -Gen-C-003	vObject vCalendar 1.0 support		O	vObjProfile-vCalC-C-000 OR vObjProfile-vCalP-C-000
vObjProfile -Gen-C-004	vObject vBookmark 1.0 support		O	vObjProfile-vBookC-C-000 OR vObjProfile-vBookP-C-000

D.2 vCard2.1 minimum interoperability profile composer SCRs

Item	Function	Reference	Status	Requirement
vObjProfile-vCardC-C-000	vCard 2.1 Composer Minimum support	6.1	O	vObjProfile-vCardC-C-002 AND vObjProfile-vCardC-C-006 AND vObjProfile-vCardC-C-007 AND vObjProfile-vCardC-C-008 AND vObjProfile-vCardC-C-009 AND vObjProfile-vCardC-C-010 AND vObjProfile-vCardC-C-011 AND vObjProfile-vCardC-C-012 AND vObjProfile-vCardC-C-013 AND vObjProfile-vCardC-C-014 AND vObjProfile-vCardC-C-015 AND vObjProfile-vCardC-C-016 AND vObjProfile-vCardC-C-020 AND vObjProfile-vCardC-C-022 AND vObjProfile-vCardC-C-023
vObjProfile-vCardC-C-001	FN	6.2.1	O	
vObjProfile-vCardC-C-002	N	6.2.2	O	
vObjProfile-vCardC-C-003	PHOTO	6.2.3	O	
vObjProfile-vCardC-C-004	BDAY	6.2.4	O	
vObjProfile-vCardC-C-005	ADR	6.2.5	O	
vObjProfile-vCardC-C-006	TEL	6.2.6	O	

vObjProfile-vCardC-C-007	TEL;VOICE	6.2.6	O	
vObjProfile-vCardC-C-008	TEL;CELL	6.2.6	O	
vObjProfile-vCardC-C-009	TEL;WORK;VOICE	6.2.6	O	
vObjProfile-vCardC-C-010	TEL;WORK;FAX	6.2.6	O	
vObjProfile-vCardC-C-011	TEL;WORK;CELL	6.2.6	O	
vObjProfile-vCardC-C-012	EMAIL	6.2.7	O	
vObjProfile-vCardC-C-013	EMAIL;INTERNET	6.2.7	O	
vObjProfile-vCardC-C-014	EMAIL;CELL	6.2.7	O	
vObjProfile-vCardC-C-015	EMAIL;HOME	6.2.7	O	
vObjProfile-vCardC-C-016	EMAIL;WORK	6.2.7	O	
vObjProfile-vCardC-C-017	NOTE	6.2.8	O	
vObjProfile-vCardC-C-018	SOUND	6.2.9	O	
vObjProfile-vCardC-C-019	URL	6.2.10	O	
vObjProfile-vCardC-C-020	VERSION	6.2.11	O	
vObjProfile-vCardC-C-021	REV	6.2.12	O	
vObjProfile-vCardC-C-022	text/x-vCard content type	5.1	O	
vObjProfile-vCardC-C-023	.vcf file extension	5.1	O	
vObjProfile-vCardC-C-024	TEL;PREF;CELL	6.2.6	O	
vObjProfile-vCardC-C-024	EMAIL;PREF;CELL	6.2.7	O	

D.3 vCard minimum interoperability profile parser SCR

Item	Function	Reference	Status	Requirement
vObjProfile-vCardP-C-000	vCard 2.1 Parser Minimum support	6.1	O	vObjProfile-vCardP-C-002 AND vObjProfile-vCardP-C-006 AND vObjProfile-vCardP-C-007 AND vObjProfile-vCardP-C-008 AND vObjProfile-vCardP-C-009 AND vObjProfile-vCardP-C-010 AND vObjProfile-vCardP-C-011 AND vObjProfile-vCardP-C-012 AND vObjProfile-vCardP-C-013 AND vObjProfile-vCardP-C-014 AND vObjProfile-vCardP-C-015 AND vObjProfile-vCardP-C-016 AND vObjProfile-vCardP-C-020 AND vObjProfile-vCardP-C-022 AND vObjProfile-vCardP-C-023 AND vObjProfile-vCardP-C-024 AND vObjProfile-vCardP-C-025

vObjProfile-vCardP-C-001	FN	6.2.1	O	
vObjProfile-vCardP-C-002	N	6.2.2	O	
vObjProfile-vCardP-C-003	PHOTO	6.2.3	O	
vObjProfile-vCardP-C-004	BDAY	6.2.4	O	
vObjProfile-vCardP-C-005	ADR	6.2.5	O	
vObjProfile-vCardP-C-006	TEL	6.2.6	O	
vObjProfile-vCardP-C-007	TEL;VOICE	6.2.6	O	
vObjProfile-vCardP-C-008	TEL;CELL	6.2.6	O	
vObjProfile-vCardP-C-009	TEL;WORK;VOICE	6.2.6	O	
vObjProfile-vCardP-C-010	TEL;WORK;FAX	6.2.6	O	
vObjProfile-vCardP-C-011	TEL;WORK;CELL	6.2.6	O	
vObjProfile-vCardP-C-012	EMAIL	6.2.7	O	
vObjProfile-vCardP-C-013	EMAIL;INTERNET	6.2.7	O	
vObjProfile-vCardP-C-014	EMAIL;CELL	6.2.7	O	
vObjProfile-vCardP-C-015	EMAIL;HOME	6.2.7	O	
vObjProfile-vCardP-C-016	EMAIL;WORK	6.2.7	O	
vObjProfile-vCardP-C-017	NOTE	6.2.8	O	
vObjProfile-vCardP-C-018	SOUND	6.2.9	O	
vObjProfile-vCardP-C-019	URL	6.2.10	O	
vObjProfile-vCardP-C-020	VERSION	6.2.11	O	
vObjProfile-vCardP-C-021	REV	6.2.12	O	
vObjProfile-vCardP-C-022	text/x-vCard content type	5.1	O	
vObjProfile-vCardP-C-023	.vcf file extension	5.1	O	
vObjProfile-vCardP-C-024	TEL;PREF;CELL	6.2.6	O	
vObjProfile-vCardP-C-025	EMAIL;PREF;CELL	6.2.7	O	

D.4 vCalendar minimum interoperability profile composer SCR

Item	Function	Reference	Status	Requirement
vObjProfile-vCalC-C-000	vCalendar 1.0 composer mandatory support	7.1	O	vObjProfile-vCalC-C-001 AND vObjProfile-vCalC-C-003 AND vObjProfile-vCalC-C-005 AND vObjProfile-vCalC-C-007 AND vObjProfile-vCalC-C-009 AND vObjProfile-vCalC-C-010 AND vObjProfile-vCalC-C-012 AND vObjProfile-vCalC-C-013 AND vObjProfile-vCalC-C-016 AND vObjProfile-vCalC-C-017
vObjProfile-vCalC-C-001	VERSION	7.2.1	O	

vObjProfile-vCalC-C-002	AALARM	7.2.2	O	
vObjProfile-vCalC-C-003	CATEGORIES	7.2.3	O	
vObjProfile-vCalC-C-004	CLASS	7.2.4	O	
vObjProfile-vCalC-C-005	DESCRIPTION	7.2.5	O	
vObjProfile-vCalC-C-006	DALARM	7.2.6	O	
vObjProfile-vCalC-C-007	DTEND	7.2.7	O	
vObjProfile-vCalC-C-008	LOCATION	7.2.8	O	
vObjProfile-vCalC-C-009	RRULE	7.2.9	O	
vObjProfile-vCalC-C-010	DTSTART	7.2.10	O	
vObjProfile-vCalC-C-011	SUMMARY	7.2.11	O	
vObjProfile-vCalC-C-012	COMPLETED	7.2.12	O	
vObjProfile-vCalC-C-013	DUE	7.2.13	O	
vObjProfile-vCalC-C-014	PRIORITY	7.2.14	O	
vObjProfile-vCalC-C-015	STATUS	7.2.15	O	
vObjProfile-vCalC-C-016	text/x-vCalendar content type	5.1	O	
vObjProfile-vCalC-C-017	.vcs file extension	5.1	O	

D.5 vCalendar minimum interoperability profile parser SCR

Item	Function	Reference	Status	Requirement
vObjProfile-vCalP-C-000	vCalendar 1.0 Parser mandatory support	7.1	O	vObjProfile-vCalP-C-001 AND vObjProfile-vCalP-C-003 AND vObjProfile-vCalP-C-005 AND vObjProfile-vCalP-C-007 AND vObjProfile-vCalP-C-009 AND vObjProfile-vCalP-C-010 AND vObjProfile-vCalP-C-012 AND vObjProfile-vCalP-C-013 AND vObjProfile-vCalP-C-016 AND vObjProfile-vCalP-C-017
vObjProfile-vCalP-C-001	VERSION	7.2.1	O	
vObjProfile-vCalP-C-002	AALARM	7.2.2	O	
vObjProfile-vCalP-C-003	CATEGORIES	7.2.3	O	
vObjProfile-vCalP-C-004	CLASS	7.2.4	O	
vObjProfile-vCalP-C-005	DESCRIPTION	7.2.5	O	
vObjProfile-vCalP-C-006	DALARM	7.2.6	O	
vObjProfile-vCalP-C-007	DTEND	7.2.7	O	
vObjProfile-vCalP-C-008	LOCATION	7.2.8	O	
vObjProfile-vCalP-C-009	RRULE	7.2.9	O	
vObjProfile-vCalP-C-010	DTSTART	7.2.10	O	

vObjProfile-vCalP-C-011	SUMMARY	7.2.11	O	
vObjProfile-vCalP-C-012	COMPLETED	7.2.12	O	
vObjProfile-vCalP-C-013	DUE	7.2.13	O	
vObjProfile-vCalP-C-014	PRIORITY	7.2.14	O	
vObjProfile-vCalP-C-015	STATUS	7.2.15	O	
vObjProfile-vCalP-C-016	text/x-vCalendar content type	5.1	O	
vObjProfile-vCalP-C-017	.vcs file extension	5.1	O	

D.6 vBookmark minimum interoperability profile composer SCR

Item	Function	Reference	Status	Requirement
vObjProfile-vBookC-C-000	vBookmark 1.0 Composer mandatory support	8.1	O	vObjProfile-vBookC-C-001 AND vObjProfile-vBookC-C-002 AND vObjProfile-vBookC-C-003 AND vObjProfile-vBookC-C-005 AND vObjProfile-vBookC-C-006
vObjProfile-vBookC-C-001	VERSION	8.2.1	O	
vObjProfile-vBookC-C-002	URL	8.2.2	O	
vObjProfile-vBookC-C-003	TITLE	8.2.3	O	
vObjProfile-vBookC-C-004	X-IRMC-URL	8.2.4	O	
vObjProfile-vBookC-C-005	text/plain	5.1	O	
vObjProfile-vBookC-C-006	.url	5.1	O	

D.7 vBookmark minimum interoperability profile parser SCR

Item	Function	Reference	Status	Requirement
vObjProfile-vBookP-C-000	vBookmark 1.0 Parser mandatory support	8.1	O	vObjProfile-vBookP-C-001 AND vObjProfile-vBookP-C-002 AND vObjProfile-vBookP-C-003 AND vObjProfile-vBookP-C-005 AND vObjProfile-vBookP-C-006 AND vObjProfile-vBookP-C-007
vObjProfile-vBookP-C-001	VERSION	8.2.1	O	
vObjProfile-vBookP-C-002	URL	8.2.2	O	
vObjProfile-vBookP-C-003	TITLE	8.2.3	O	
vObjProfile-vBookP-C-004	X-IRMC-URL	8.2.4	O	
vObjProfile-vBookP-C-005	text/plain	5.1	O	
vObjProfile-vBookP-C-006	.url	5.1	O	
vObjProfile-vBookP-C-007	.vbm	5.1	O	