

Network Working Group
Internet-Draft
Expires: January 23, 2003

F. Dawson
Nokia
S. Reddy
Oracle
D. Royer
INET-Consulting
E. Plamondon
Oracle
July 25, 2002

**iCalendar DTD Document (xCal)
draft-ietf-calsch-many-xcal-02**

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of [Section 10 of RFC2026](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/1id-abstracts.txt>.

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

This Internet-Draft will expire on January 23, 2003.

Copyright Notice

Copyright (C) The Internet Society (2002). All Rights Reserved.

Abstract

This memo defines a [[XML](#)] Document Type Definition (DTD) that corresponds to the iCalendar, Internet Calendaring and Scheduling Core Object Specification defined by [[RFC 2445](#)]. This DTD provides equivalent functionality to the standard format defined by [[RFC 2445](#)]. Documents structured in accordance with this DTD may also be known as "XML iCalendar" documents or "xCal".

The mailing list for discussion of this memo is "ietf-calendar@imc.org". Send an email to "ietf-calendar-request@imc.org" with the message "SUBSCRIBE" to add your email address to this mailing list. Send an email to "ietf-calendar-request@imc.org" with the message "UNSUBSCRIBE" to remove your email address from this mailing list.

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 [[RFC 2119](#)].

Table of Contents

- [1.](#) Introduction [4](#)
- [2.](#) Using XML For Representating iCalendar [6](#)
- [2.1](#) XML Dependencies [6](#)
- [2.2](#) Document Type Definition [6](#)
- [2.3](#) Working With Standard and XML iCalendar Representations . . [6](#)
- [2.3.1](#) Conversion [6](#)
- [2.3.2](#) Mixed Use of Both Representations [7](#)
- [2.4](#) Using Data Types [7](#)
- [2.5](#) Including Binary Content [8](#)
- [2.6](#) Including Multiple iCalendar Objects [9](#)
- [2.7](#) Mapping Property Parameters to XML [10](#)
- [2.8](#) Mapping Calendar Properties to XML [11](#)
- [2.9](#) Mapping Component Properties to XML [13](#)
- [2.10](#) Parameter Entities [16](#)
- [2.11](#) Namespace [17](#)
- [2.12](#) Emailing the iCalendar XML Representation [17](#)
- [2.13](#) iCalendar XML Representation and File Systems [18](#)
- [3.](#) Example Usage [20](#)
- [3.1](#) A well-formed and valid iCalendar XML document [20](#)
- [3.2](#) Adding non-standard, experimental extensions [20](#)
- [3.3](#) Including binary content in attachments [21](#)
- [3.4](#) iCalendar XML document with multiple iCalendar objects . . . [23](#)
- [3.5](#) Using the iCalendar namespace [24](#)
- [3.6](#) Publish meeting information [25](#)
- [3.7](#) Publish transparent annual event [25](#)
- [3.8](#) Meeting invitation [26](#)
- [3.9](#) Assign a to-do [27](#)
- [3.10](#) Publish a journal entry [28](#)
- [3.11](#) Publish busy time [29](#)
- [3.12](#) Request busy time [29](#)
- [3.13](#) Response to a busy time request [30](#)
- [3.14](#) Published event that references time zone information . . . [30](#)
- [3.15](#) An event with an alarm [31](#)
- [4.](#) iCalendar XML Document Type Definition [34](#)
- [5.](#) Acknowledgments [48](#)

- [6.](#) IANA Considerations [49](#)
- [7.](#) Security Considerations [50](#)
- [8.](#) Bibliography [51](#)
 - Authors' Addresses [51](#)
 - Full Copyright Statement [53](#)

1. Introduction

The Extended Markup Language (XML) as defined in [[XML](#)] is gaining widespread attention as a "web friendly" syntax for representing and exchanging documents and data on the Internet. This interest includes requests for and discussion of possible document type definitions (DTD) and name-space for IETF standard formats such as that defined by [[RFC 2445](#)].

This memo defines how XML can be used to represent iCalendar objects. This memo includes the definition of the XML DTD for a XML document representation of an iCalendar object.

NOTE: The [[RFC 2445](#)] is the definitive reference for the definition of iCalendar semantics. This memo only provides an alternative, XML representation for the standard syntax defined in [[RFC 2445](#)]. This memo does not introduce any semantics not already defined by [[RFC 2445](#)].

An attempt has been made to leverage the standard features of the XML syntax in order to represent the component iCalendar semantics. For example, strong data typing is specified using the XML notation declaration. The element type attributes are used to represent many of the calendar properties that provide a "global attribute" capability in an iCalendar object. Binary content in the ATTACH component property may either be specified through an external entity reference to a non-XML binary content or may be included in the XML document's content information, after first being encoding using the BASE64 scheme of [[RFC 2146](#)]. Parameter entities are used to logically group content particles in the XML DTD in order to facilitate reading and comprehension of the structure specified by the iCalendar XML DTD.

The publication of XML version 1.0 was followed by the publication of a World-wide Web Consortium (W3C) recommendation on "Namespaces in XML". A XML name-space is a collection of names, identified by a URI. In anticipation of the broader use of XML namespaces, this memo includes the definition of the URI to be used to identify the namespace associated with the iCalendar DTD element types in other XML documents. XML applications that conform to this memo and also use namespaces MUST NOT include other non-iCalendar namespaces in an iCalendar XML document.

It is expected that the DTD defined in this memo will not normally be included with iCalendar XML documents that are distributed. Instead, the DTD will be referenced in the document type declaration in the document entity. Such iCalendar XML documents will be well-formed and valid, as defined in [[XML](#)]. In addition, other iCalendar XML

documents will be specified that do not include the XML prolog. Such iCalendar XML documents will be well-formed but not valid.

2. Using XML For Representating iCalendar

XML is a simplified version of the text markup syntax defined by ISO 8879, Standard Generalized Markup Language (SGML). XML was published as a proposed recommendation [[XML](#)] by the World-wide Web Consortium (W3C) on February 10, 1998.

2.1 XML Dependencies

This memo specifies the XML representation for the standard iCalendar format defined by [[RFC 2445](#)]. There are no XML dependencies other than the [[XML](#)] and the [[XMLNS](#)] recommendations.

2.2 Document Type Definition

A XML DTD for iCalendar is defined by the DTD specified in [section 3](#).

The formal public identifier (FPI) for the DTD is:

```
-//IETF//DTD XCAL//iCalendar XML//EN
```

NOTE: The "DTD XCAL" text in the FPI value will be replaced with the text "RFC xxxx", where "xxxx" is the RFC number, when the memo is published as a RFC.

This FPI MUST be used on the DOCTYPE statement within a XML document referencing the DTD defined by this memo.

This FPI SHOULD also be used to identify iCalendar XML documents within operating system registries of file, clipboard and interactive rendering (e.g., memory clipboard or drag/drop) formats.

2.3 Working With Standard and XML iCalendar Representations

This memo defines an alternative, XML representation for the standard iCalendar format defined in [[RFC 2445](#)]. This alternative representation provides the same semantics as that defined in the standard format.

2.3.1 Conversion

The standard format can be converted to and from this XML format without loss of any calendaring information. When the XML representation was defined, every attempt was made to use existing component, property and parameter naming conventions. This greatly facilitates transformations between the two representations.

2.3.2 Mixed Use of Both Representations

As previously indicated, conversion between the standard and XML representations of iCalendar is a straightforward process. In addition, mixed use of both representations is also possible.

With the use of the MIME multipart content-types, compound MIME entities containing a mix of the standard and XML representations can be specified. This capability is useful in applications where both representations might be encountered. In addition, this capability demonstrates the isomeric nature of the two representations. XML applications conforming to this specification MUST be able to properly parse and process a MIME multipart entity containing the MIME type associated with this iCalendar XML document type.

Internet applications conforming to this memo MUST only send the iCalendar XML document in a "multipart/alternative" MIME entity that also contains an equivalent iCalendar object in the standard format defined by [\[RFC2445\]](#). This restriction will guarantee that the iCalendar object can also be processed by Internet applications that only support the standard iCalendar representation.

2.4 Using Data Types

Strong "data typing" is an integral design principle to the iCalendar format. Strong data typing in iCalendar means that the format type for each property value is well known. Within [\[RFC 2445\]](#), the data type is called the "value type". The standard format defined by [\[RFC 2445\]](#) specifies a default value type for each calendar and component property. In addition, many of the property definitions allow for the specification of alternate value types. This XML representation continues this design principle.

Explicit value/data typing in the XML representation is specified with the "value" attribute on each element type. In addition, the XML DTD specifies a default value/data type for each element type. XML documents conforming to this memo need only specify the "value" attribute on element types when the value needs to override the default value/data type. The standard value types defined in [\[RFC2445\]](#) are specified in the XML DTD by individual NOTATION declarations. The formal public identifier for standard value types all have the common string format of:

```
-//IETF//NOTATION XCAL/Value Type/xxx//EN
```

NOTE: The "XCAL" text in the FPI value will be replaced with the text "RFC xxxx", where "xxxx" is the RFC number, when the memo is published as a RFC.

Where "xxx" is replaced with the text specified in the table below.

The following table specifies the XML value/data type that corresponds to each of the standard value types defined in [RFC 2445].

RFC 2445 Value Type	XML Value Type	Notation FPI Text
BINARY	BINARY	Binary
BOOLEAN	BOOLEAN	Boolean
CALADR	CALADR	Calendar User Address
DATE	DATE	Date
DATE-TIME	DATE-TIME	Date-Time
DURATION	DURATION	Duration
FLOAT	FLOAT	Float
INTEGER	INTEGER	Integer
PERIOD	PERIOD	Period of Time
RECUR	RECUR	Recurrence Rule
TEXT	TEXT	Text
TIME	TIME	Time
URI	URI	URI
UTC-OFFSET	UTC-OFFSET	UTC-Offset
Non-standard	X-NAME	X-Name

Other standard XML data types can be specified by including a NOTATION declaration that specifies the formal public identifier associated with the other standard format. In addition, the name of the format specified in the NOTATION declaration is specified in the "value" attribute of any element type that casts to the other standard format.

2.5 Including Binary Content

Binary content can be included in an iCalendar object with the "ATTACH" component property. In the standard iCalendar format this content may either be specified through an external entity reference, using a URI value type, or maybe specified within the iCalendar object, after first BASE64 encoding the content.

The XML representation for iCalendar also supports including binary content in an iCalendar object with the "attach" element type. It also supports either an external reference to the non-XML binary content or inclusion of the binary content after first encoding the binary information using the BASE64 encoding of [RFC 2045].

Any iCalendar properties defined in [[RFC 2445](#)] that can be used to include binary content are defined in the XML representation as an element type with a content model that consists of either the "extref" or the "b64bin" element type. The "extref" element type is used to reference an external entity containing the binary content. An external reference to the binary content is specified by the "uri" attribute on the "extref" element type. For every external reference, an ENTITY declaration and a corresponding NOTATION declaration MUST also be specified in an internal DTD to identify the location and format of the external entity. For example, the following XML snippets would be needed to include a reference to the executable "foo.exe" in the "attach" element type; which corresponds to the "ATTACH" iCalendar component property:

```
<!-- Following specified within the internal DTD. -->

<!NOTATION EXE SYSTEM "Executable Module Format">
<!ENTITY attach-1 SYSTEM "http://host.com/bin/foo.exe" NDATA EXE>

<!-- Following specified within the body of the XML document. -->

<attach><extref uri="attach-1" /></attach>
```

The "b64bin" element type is used to include the binary content within the XML document, after first character encoding the binary information using the BASE64 encoding method of [[RFC 2045](#)]. For example, the following XML snippets would be needed to include the executable "foo.exe" in the "attach" element type; after first BASE64 encoding the binary information:

```
<!-- Following specified within the body of the XML document. -->

<attach><b64bin fmttype="APPLICATION/OCTET-STRING"> MIICajCC
AdOgAwIBAgICBEUwDQEEBQAwdzELMAkGA1UEBhMCMCVVMxLDAqBgNVBAoTI05l
dHNjYXB1IENvbW11bmljYXR5z...and so on...IENvcnBvc==</b64bin>
</attach>
```

[2.6](#) Including Multiple iCalendar Objects

The iCalendar format has the capability for including multiple, individual iCalendar objects in a single data stream. The XML representation can support this also. Individual iCalendar objects are specified by the "vcalendar" element type. One or more "vcalendar" element types are permitted within the parent element type, called "iCalendar". For example:


```

<iCalendar>
  <vcalendar version="2.0">
    <!-- remainder of the first vcalendar object -->
  </vcalendar>

  <vcalendar version="2.0">
    <!-- remainder of the second vcalendar object -->
  </vcalendar>
</iCalendar>

```

2.7 Mapping Property Parameters to XML

The property parameters defined in the standard iCalendar format are represented in the XML representation as an attribute on element types. The following table specifies the attribute name corresponding to each property parameter.

Property Parameter Name	Attribute Name	Attribute Type	Default Value
ALTREP	altrep	ENTITY	IMPLIED
CN	cn	CDATA	Null String
CUTYPE	cutype	NMTOKEN	INDIVIDUAL
DELEGATED-FROM	delegated-from	CDATA	IMPLIED
DELEGATED-TO	delegated-to	CDATA	IMPLIED
DIR	dir	ENTITY	IMPLIED
ENCODING	Not Used	n/a	n/a
FMTTYPE	fmttype	CDATA	REQUIRED
FBTYPE	fbtype	NMTOKEN	BUSY
LANGUAGE	language	CDATA	IMPLIED
MEMBER	member	CDATA	IMPLIED
PARTSTAT	partstat	NMTOKEN	NEEDS-ACTION
RANGE	range	NMTOKEN	THISONLY
RELATED	related	NMTOKEN	START
RELTYPE	reltype	NMTOKEN	PARENT
ROLE	role	NMTOKEN	REQ-PARTICIPANT
RSVP	rsvp	NMTOKEN	FALSE
SENT-BY	sent-by	CDATA	IMPLIED
TZID	tzid	CDATA	IMPLIED
VALUE	value	NOTATION	See elements

The inline "ENCODING" property parameter is not needed in the XML representation. Inline binary information is always included as parsable character data, after first being encoded using the BASE64 encoding of [\[RFC 2045\]](#).

The "RANGE" property parameter defined by [[RFC 2445](#)] does not include the "THISONLY" enumeration. This is the implicit default, if the parameter is not specified on the "RECURRENCE-ID" property. However, the value is needed in the XML representation because all attributes need to explicitly specify a default value in the attribute's declaration in the DTD. This enumeration has been added to the XML representation.

A non-standard, experimental parameter can be added to the XML representation by declaring it in an ATTLIST declaration and assigning it a XML attribute type and corresponding default value.

2.8 Mapping Calendar Properties to XML

Calendar properties defined in the standard iCalendar format provide information about an iCalendar object, as a whole. The calendar properties are represented in the XML representation as an attribute on the "iCalendar" and the "vcalendar" element type. The following table specifies the attribute name permitted on the "iCalendar" element type.

Calendar Property Name	Attribute Name	Attribute Type	Default Value
CALSCALE	calscale	CDATA	IMPLIED
METHOD	method	NMTOKEN	PUBLISH
VERSION	version	CDATA	REQUIRED
PRODID	prodid	CDATA	IMPLIED

In addition to these attributes, the "vcalendar" element type can also have the following attributes:

Attribute Name	Attribute Type	Default Value	Description
xmlns	CDATA	FIXED	Used to specify the default iCalendar XML name space.
xmlns: + <namespace prefix>	CDATA	FIXED	Used to specify the name space.

The semantics of the "xmlns" attribute, and any attribute with "xmlns:" as a prefix, is as specified in [[XMLNS](#)]. It is used to declare a namespace in XML. It can be used to declare the iCalendar

XML namespace in a XML document with a document type other than the iCalendar XML document type. The iCalendar XML document type MUST only use element types from the iCalendar namespace. Non-standard, experimental element types and attributes lists MUST only be specified by declarations in an internal DTD within the iCalendar XML document. To specify the iCalendar namespace, the attribute value for the "xmlns" and any attribute with the prefix "xmlns:" MUST be:

```
'http://www.ietf.org/internet-drafts/draft-ietf-calsch-many-xcal-01.txt'
```

NOTE: This attribute value will be replaced with the URL "http://www.ietf.org/rfc/rfcxxxx.txt", where "xxxx" is the RFC number, when this memo is published as a RFC.

For example:

```
<iCalendar xmlns:iCalv3='http://www.ietf.org/internet-
drafts/draft-ietf-calsch-many-xcal-01.txt'>
<!-- the "iCalendar" prefix is bound to
'http://www.ietf.org/internet-drafts/draft-ietf-calsch-many-xcal-01.txt'
for the "iCalendar" element and contents-->
</iCalendar>
```

The following table specifies the attribute name corresponding to each calendar property. These attributes are only permitted on the "vcalendar" element type.

Calendar Property Name	Attribute Name	Attribute Type	Default Value
CALSCALE	calscale	CDATA	IMPLIED
METHOD	method	NMTOKEN	PUBLISH
VERSION	version	CDATA	REQUIRED
PRODID	prodid	CDATA	IMPLIED

The semantics for these attributes is as specified for the corresponding calendar property in [\[RFC 2445\]](#).

The following table specifies additional attributes that are permitted on the "vcalendar" element type.

Attribute Name	Attribute Type	Default Value	Description
language	CDATA	IMPLIED	Used to specify the default

The "language" attribute permits the default language to be specified for the whole iCalendar object. If the "language" attribute is specified on the XML document, then if the XML representation is converted into the standard format the "LANGUAGE" property parameter MUST be specified on each TEXT valued property to prevent information loss; when translating into the standard format.

2.9 Mapping Component Properties to XML

Component properties in the standard iCalendar format provide calendar information about the calendar component. The component properties defined in the standard iCalendar format are represented in the XML representation as element types. The following tables specify the element types corresponding to each of the properties in the specified property category.

Descriptive Component Properties

Component Property Name	Element Name	Element Content Model
ATTACH	attach	extref or b64bin
	extref	EMPTY
	b64bin	PCDATA
CATEGORIES	categories	Any number of item elements
	item	PCDATA
CLASS	class	PCDATA
COMMENT	comment	PCDATA
DESCRIPTION	description	PCDATA
GEO	geo	lat followed by lon element
	lat	PCDATA
	lon	PCDATA
LOCATION	location	PCDATA
PERCENT	percent	PCDATA
PRIORITY	priority	PCDATA
RESOURCES	resources	Any number of item elements
STATUS	status	PCDATA
SUMMARY	summary	PCDATA

Date and Time Component Properties

Component Property Name	Element Name	Element Content Model
COMPLETED	completed	PCDATA
DTEND	dtend	PCDATA
DUE	due	PCDATA
DTSTART	dtstart	PCDATA
DURATION	duration	PCDATA
FREEBUSY	freebusy	PCDATA
TRANSP	transp	PCDATA

Time Zone Component Properties

Component Property Name	Element Name	Element Content Model
TZID	tzid	PCDATA
TZNAME	tzname	PCDATA
TZOFFSETFROM	tzoffsetfrom	PCDATA
TZOFFSETO	tzoffsetto	PCDATA
TZURL	tzurl	EMPTY

Relationship Component Properties

Component Property Name	Element Name	Element Content Model
ATTENDEE	attendee	PCDATA
CONTACT	contact	PCDATA
ORGANIZER	organizer	PCDATA
RECURRENCE-ID	recurrence-id	PCDATA
RELATED-TO	related-to	PCDATA
URL	url	EMPTY
UID	uid	PCDATA

Recurrence Component Properties

Component Property Name	Element Name	Element Content Model
----------------------------	-----------------	-----------------------

EXDATE	exdate	PCDATA	
EXRULE	exrule	PCDATA	
RDATE	rdate	PCDATA	
RRULE	rrule	PCDATA	
+-----+-----+-----+			

Alarm Component Properties

Component	Element	Element Content Model	
Property Name	Name		
+-----+-----+-----+			
ACTION	action	PCDATA	
REPEAT	repeat	PCDATA	
TRIGGER	trigger	PCDATA	
+-----+-----+-----+			

Change Management Component Properties

Component	Element	Element Content Model	
Property Name	Name		
+-----+-----+-----+			
CREATED	created	PCDATA	
DTSTAMP	dtstamp	PCDATA	
LAST-MODIFIED	last-modified	PCDATA	
SEQUENCE	sequence	PCDATA	
+-----+-----+-----+			

Miscellaneous Component Properties

Component	Element	Element Content Model	
Property Name	Name		
+-----+-----+-----+			
REQUEST-STATUS	request-status	PCDATA	
+-----+-----+-----+			

The following table specifies the element types that represent each of the calendar components.

Component Structuring Properties

Component Property Name	Element Name	Element Content Model
Multiple iCalendar objects	iCalendar	One or more iCal elements
VCALENDAR	vcalendar	calcomp parameter entity
VEVENT	vevent	vevent.opt1 and vevent.optm parameter entity and valarm element
VTODO	vtodo	vtodo.opt1 and vtodo.optm parameter entity and valarm element
VJOURNAL	vjournal	vjournal.opt1 and vjournal.optm parameter entity
VFREEBUSY	vfreebusy	vfreebusy.opt1 and vfreebusy.optm parameter entity
VTIMEZONE	vtimezone	vtimezone.man, vtimezone.opt1, vtimezone.mann parameter entity
STANDARD	standard	standard.man or standard.optm entity
DAYLIGHT	daylight	daylight.man or daylight.optm entity
VALARM	valarm	valarm.audio, valarm.display, valarm.email and valarm.procedure entity

The [\[RFC 2445\]](#) specification specifies that the equivalent component properties to the "comment", "description", "location", "summary" and "contact" element types can contain formatted content, such as is specified by multiple lines of text. In such cases, the formatted text should be specified in as CDATA Section content. The CDATA section specifies arbitrary character data that is not meant to be interpreted. It is not scanned for markup by the XML parser. The CDATA Section in these element types MUST NOT contain markup or other such alternate representation of the property value. The "altrep" attribute is used to reference any such alternate representation for the textual content of these element types.

2.10 Parameter Entities

The external, iCalendar XML DTD specified in [section 3](#) makes use of

parameter entity declarations. This XML feature is used to group declarations within the DTD. This technique has been used in DTD design in order to facilitate the reading and comprehension of the structure specified by the DTD.

2.11 Namespace

[XMLNS] defines "Namespaces in XML" to be a collection of names, identified by a URI, which are used in XML documents as element types and attribute names. The [XML] specification does not include a definition for namespaces, but does set down some guidelines for experimental naming of namespaces.

XML namespaces allow multiple markup vocabulary in a single document. Considering the utility of the iCalendar properties in other applications, it is important for the iCalendar XML DTD to define a namespace for the iCalendar element types.

This memo defines the value that MUST be used in non-iCalendar XML documents that reference element types or attribute lists from the iCalendar namespace.

The following is an example of a well-formed but invalid "xdoc" document type that includes elements and attribute lists from the iCalendar namespace:

```
<?xml version="1.0" encoding="UTF-8"?>
<xdoc>
<xCal:xCal version="3.0" xmlns:xCal="http://www.ietf.org/internet
-drafts/draft-ietf-calsch-many-xcal-01.txt">

<!-- Remainder of the XML document, each element from the -->
<!-- iCalendar namespace with the "xCal:" prefix. -->

</xCal:xCal>
</xdoc>
```

2.12 Emailing the iCalendar XML Representation

It is expected that iCalendar XML documents will need to be sent over SMTP/MIME email. The "text/xml" and "application/xml" content-types have been registered for XML documents. However, use of these content-type definitions present some problems for XML applications such as calendaring and scheduling.

The "text/xml" and "application/xml" content-type definitions do not provide for any header field parameters to identify the type of XML

document contained in the MIME entity. This means that a recipient mail user agent must (MUA) open up each "text/xml" or "application/xml" content in order to determine what object handler is needed to process the information. To a MUA, all XML documents look like just plain "text/xml" or "application/xml" content.

Additionally, it is accepted practice for a MUA to provide iconic feedback to the user for individual content-types that are supported by the MUA. For example, not only would feedback be provided for a calendaring and scheduling content. Some further unique identification would also be provided for each different scheduling message; such as a meeting invitation, response to an invitation, reschedule notice, cancellation notice, etc. In such cases, acceptable performance by the MUA is dependent on the existence of header field information, such as it provided in the definition of the "text/calendar" content-type by [[RFC 2445](#)].

Internet application conforming to this memo MUST identify iCalendar XML documents with the experimental content-type "application/calendar+xml". The content-type header field SHOULD also contain a "component" and "method" parameter to clearly identify a comma-separated list of components and the singular method used in the iCalendar XML document. For example, an iCalendar XML document specifying a REQUEST for a VEVENT and VTODO would be specified with the following content-type header field:

```
content-type:application/  
calendar+xml;method=REQUEST;component=VEVENT,VTODO
```

The content-type can also include the "optinfo" parameter to specify any other optional iCalendar information. The semantics of these content-type parameters is as defined in [[RFC 2445](#)].

Internet applications conforming to this memo MUST only send the iCalendar XML document in a "multipart/alternative" MIME entity that also contains an equivalent iCalendar object in the standard format defined by [[RFC 2445](#)]. This restrict will guarantee that the iCalendar object can also be processed by internet applications that only support the standard iCalendar format.

An XML application supporting the iCalendar XML document type MUST be able to receive and properly process the "application/calendar+xml" document contained within a "multipart" message content-type.

2.13 iCalendar XML Representation and File Systems

The iCalendar XML documents will be stored in file systems. The accepted practice for file extensions for XML documents is the text "XML". However, in order to uniquely identify iCalendar XML

documents for file association with applications that can directly process this document type, it is RECOMMENDED that the file extension be the text "XCS".

3. Example Usage

The following sections provide various examples of iCalendar XML documents.

3.1 A well-formed and valid iCalendar XML document

The following is a simple example of a iCalendar XML document. This document is both a well-formed and valid XML document. The iCalendar object specifies an appointment.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE iCalendar PUBLIC "-//IETF//DTD XCAL/iCalendar XML//EN"
"http://www.ietf.org/internet-drafts/draft-ietf-calsch-many-xcal-01.txt">

<iCalendar>
<vcalendar method="PUBLISH"
  version="2.0"
  prodid="-//HandGen//NONSGML vGen v1.0//EN">
<vevent>
  <uid>19981116T150000@cal10.host.com</uid>
  <dtstamp>19981116T145958Z</dtstamp>
  <summary>Project XYZ Review</summary>
  <location>Conference Room 23A</location>
  <dtstart>19981116T163000Z</dtstart>
  <dtend>19981116T190000Z</dtend>
  <categories>
    <item>Appointment</item>
  </categories>
</vevent>
</vcalendar>
</iCalendar>
```

3.2 Adding non-standard, experimental extensions

The following is an example of a valid iCalendar XML document that also includes a non-standard, experimental extension, as provided for by [RFC 2445](#). The iCalendar object specifies the publication of a to-do with a non-standard experimental property for a customer charge code.

The non-standard experimental property is identified by the "X-" prefix to the element name. All non-standard properties MUST be specified with element types with an "X-" type element name. In addition, a text identifier for the vendor specifying the extension SHOULD be appended to the "X-" text prefix. In this case, the example specifies a "foo" for the name of the vendor specifying the

non- standard property.

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE iCalendar PUBLIC "-//IETF//DTD XCAL/iCalendar XML//EN"
"http://www.ietf.org/internet-drafts/draft-ietf-calsch-many-xcal-01.txt"
[
<!ELEMENT vtodo ((class | completed | created | description | dtstamp
| dtstart | geo | last-modified | location | organizer | percent |
priority |
recurrence-id | sequence | status | summary | uid | url | (due |
duration))* ,
(attach | attendee | categories | comment | contact | exdate | exrule
| request-status | related-to | resources | rdate | rrule | x-foo-cust-
code)* ,
(valarm)*)>

<!ELEMENT x-foo-cust-code (#PCDATA)>
<!ATTLIST x-foo-cust-code value NOTATION (X-NAME) #IMPLIED>
]>

<iCalendar>
<vcalendar method="PUBLISH"
  version="2.0"
  prodid="-//HandGen//NONSGML vGen v1.0//EN">
<vtodo>
  <uid>19981104T130000@cal1.host.com </uid>
  <dtstamp>19981104T125957Z</dtstamp>
  <dtstart>19981105T133000Z</dtstart>
  <due>19981106T133000Z</due>
  <summary>Draft a test plan</summary>
  <x-foo-cust-code>1998-ABC Corp-1234</x-foo-cust-code>
  <priority>1</priority>
</vtodo>
</vcalendar>
</iCalendar>

```

3.3 Including binary content in attachments

The following is an example of a valid iCalendar XML document that also includes an external reference to an attachment. The iCalendar object specifies a meeting invitation with an attachment.


```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE iCalendar PUBLIC "-//IETF//DTD XCAL/iCalendar XML//EN"
"http://www.ietf.org/internet-drafts/draft-ietf-calsch-many-xcal-01.txt"
[
<!ENTITY attach1 SYSTEM "http://host.com/pub/photos/holiday.jpg"
NDATA JPEG>

<!NOTATION JPEG PUBLIC "ISO/IEC 10918:1993//NOTATION Digital
Compression and Coding of Continuous-tone Still Images (JPEG)//EN" >
]>

<iCalendar>
<vcalendar method="REQUEST"
  version="2.0"
  prodid="-//HandGen//NONSGML vGen v1.0//EN">
<vevent>
  <uid>19981211T133000@cal1.host.com</uid>
  <dtstamp>19981211T132928Z</dtstamp>
  <organizer>jim@host.com</organizer>
  <dtstart>19981212T150000Z</dtstart>
  <dtend>19981212T160000Z</dtend>
  <summary>Department Meeting</summary>
  <location>Conference Room 23A</location>
  <attendee role="CHAIR">jim@host.com</attendee>
  <attendee role="REQ-PART"
    rsvp="TRUE">MAILTO:joe@host.com</attendee>
  <attendee role="REQ-PART"
    rsvp="TRUE">MAILTO:steve@host.com</attendee>
  <attach><extref uri="attach1" /></attach>
</vevent>
</vcalendar>
</iCalendar>
```

The following is an example of a well-formed and valid iCalendar XML document that includes an attachment as inline binary content. The iCalendar object specifies a meeting invitation with an attachment.


```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE iCalendar PUBLIC "-//IETF//DTD XCAL/iCalendar XML//EN"
"http://www.ietf.org/internet-drafts/draft-ietf-calsch-many-xcal-01.txt">

<iCalendar>
<vcalendar method="REQUEST"
  version="2.0"
  prodid="-//HandGen//NONSGML vGen v1.0//EN">
<vevent>
  <uid>19981211T133000@cal1.host.com</uid>
  <dtstamp>19981211T132928Z</dtstamp>
  <organizer>MAILTO:jim@host.com</organizer>
  <dtstart>19981212T150000Z</dtstart>
  <dtend>19981212T160000Z</dtend>
  <summary>Department Meeting</summary>
  <location>Conference Room 23A</location>
  <attendee role="CHAIR">MAILTO:jim@host.com</attendee>
  <attendee role="REQ-PART"
    rsvp="TRUE">MAILTO:joe@host.com</attendee>
  <attendee role="REQ-PART"
    rsvp="TRUE">MAILTO:steve@host.com</attendee>
  <attach><b64bin fmttype="IMAGE/JPEG">MIICajCCAdOgAwIBAgI
  CBEUwDQEEBQAwDzELMAkGA1UEBhMCMVVMxLDAqBgNVBAoTI05ldHNjYXB
  lIEjYXRpb25z...and so on...IENvcnBvc==</b64bin></attach>
</vevent>
</vcalendar>
</iCalendar>
```

3.4 iCalendar XML document with multiple iCalendar objects

The following is an example of a well-formed and valid iCalendar XML document that includes more than one iCalendar object.


```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE iCalendar PUBLIC "-//IETF//DTD XCAL/iCalendar XML//EN"
"http://www.ietf.org/internet-drafts/draft-ietf-calsch-many-xcal-01.txt">

<iCalendar annotation="IT Conference Prep">
<vcalendar method="PUBLISH"
  version="2.0"
  prodid="-//HandGen//NONSGML vGen v1.0//EN">
<vtodo>
  <uid>19981009T233000@cal1.host.com</uid>
  <dtstamp>19981009T232928Z</dtstamp>
  <dtstart>19981010T000000Z</dtstart>
  <due>19981010T235959Z</due>
  <summary>Register for conference</summary>
  <priority>2</priority>
</vtodo>
</vcalendar>
<vcalendar version="2.0"
  prodid="-//HandGen//NONSGML vGen v1.0//EN"
  method="PUBLISH">
<vevent>
  <uid>19981009T233010@cal1.host.com</uid>
  <dtstamp>19981009T233000Z</dtstamp>
  <dtstart>19981120T133000Z</dtstart>
  <dtend>19981122T183000Z</dtend>
  <summary>IT Conference</summary>
  <location>Downtown Hotel</location>
</vevent>
</vcalendar>
</iCalendar>

```

3.5 Using the iCalendar namespace

The following is an example of a snippet of a XML document that includes elements from the iCalendar name-space.

```

<x xmlns:xcal="http://www.ietf.org/internet-drafts/
draft-ietf-calsch-many-xcal-01.txt"
xmlns:pdi="http://pdi.org/schema">
<xcal:dtstart>19981123T133000Z</xcal:dtstart>
<xcal:dtend>19981123T203000Z</xcal:dtend>
<pdi:idnum>1234567</pdi:idnum>
<pdi:usage>999.99</pdi:usage>
</x>

```


3.6 Publish meeting information

The following is a snippet of an iCalendar XML document that publishes information about a meeting. The "method" attribute isn't specified since it is the default value.

```
<iCalendar>
  <vcalendar version="2.0"
    prodid="-//hacksw/handcal//NONSGML 1.0//EN">
    <vevent>
      <uid>19970901T130000Z-123401@host.com</uid>
      <dtstamp>19970901T130000Z</dtstamp>
      <dtstart>19970903T163000Z</dtstart>
      <dtend>19970903T190000Z</dtend>
      <summary>Annual Employee Review</summary>
      <class>PRIVATE</class>
      <categories>
        <item>Business</item>
        <item>Human Resources</item>
      </categories>
    </vevent>
  </vcalendar>
</iCalendar>
```

3.7 Publish transparent annual event

The following is a snippet of an iCalendar XML document that publishes information about an annually repeating event that is transparent to busy time searches.


```
<iCalendar>
<vcalendar version="2.0"
  prodid="-//hacksw/handcal//NONSGML 1.0//EN">
<vevent>
  <uid>19990101T125957Z-123403@host.com</uid>
  <dtstamp>19990101T130000Z</dtstamp>
  <dtstart value="DATE">19991102</dtstart>
  <dtend value="DATE">19991102</dtend>
  <summary>Our Blissful Anniversary</summary>
  <class>CONFIDENTIAL</class>
  <transp>TRANSPARENT</transp>
  <categories>
    <item>Anniversary</item>
    <item>Personal</item>
    <item>Special Occasion</item>
  </categories>
  <rrule>FREQ=YEARLY</rrule>
</vevent>
</vcalendar>
</iCalendar>
```

3.8 Meeting invitation

The following is a snippet of an iCalendar XML document that specifies an invitation for a meeting. The meeting occurs on the first Monday of each year for five years.


```
<iCalendar>
<vcalendar method="REQUEST"
  version="2.0"
  prodid="-//hacksw/handcal//NONSGML 1.0//EN">
<vevent>
  <uid>19981220T130000Z-123403@host.com</uid>
  <dtstamp>19981220T130050Z</dtstamp>
  <organizer>MAILTO:corp@host.com</organizer>
  <dtstart>19990104T140000Z</dtstart>
  <dtend>19990104T220000Z</dtend>
  <summary>Annual Stockholders Meeting</summary>
  <location>One Corporate Drive, Wilmington, DL</location>
  <attendee role="CHAIR">MAILTO:mrbig@host.com</attendee>
  <attendee cutype="GROUP"
    rsvp="TRUE">MAILTO:stockholders@host.com</attendee>
  <categories>
    <item>Business</item>
    <item>Meeting</item>
    <item>Special Occasion</item>
  </categories>
  <rrule>FREQ=YEARLY;COUNT=5;BYDAY=1M0</rrule>
</vevent>
</vcalendar>
</iCalendar>
```

3.9 Assign a to-do

The following is a snippet of an iCalendar XML document that assigns a to-do.


```
<iCalendar>
<vcalendar method="REQUEST"
  version="2.0"
  prodid="-//hacksw/handcal//NONSGML 1.0//EN">
<vtodo>
  <uid>19990104T133402@ical1.host.com</uid>
  <dtstamp>19990104T133410Z</dtstamp>
  <dtstart value="DATE">19990104</dtstart>
  <due value="DATE">19990129</due>
  <organizer>MAILTO:dboss@host.com</organizer>
  <summary>Periodic Self Review</summary>
  <description>Complete your self review.
  Contact me if you questions.</description>
  <priority>1</priority>
  <class>CONFIDENTIAL</class>
  <attendee>MAILTO:dilbert@host.com</attendee>
</vtodo>
</vcalendar>
</iCalendar>
```

3.10 Publish a journal entry

The following is a snippet of an iCalendar XML document that publishes a journal entry.

```
<iCalendar>
<vcalendar method="PUBLISH"
  version="2.0"
  prodid="-//hacksw/handcal//NONSGML 1.0//EN">
<vjournal>
  <uid>19990104T170003@ical1.host.com</uid>
  <dtstamp>19990104T170001Z</dtstamp>
  <dtstart value="DATE">19990104</dtstart>
  <organizer>MAILTO:corprel@host.com</organizer>
  <class>PUBLIC</class>
  <description>Year end report for Worldwide Calendar Company
  The complete report can be found at the Corporate website.
  http://www.host.com/annualreport</description>
  <categories>
    <item>Annual Report</item>
    <item>Business</item>
  </categories>
</vjournal>
</vcalendar>
</iCalendar>
```


3.11 Publish busy time

The following is an iCalendar XML document that publishes busy time information. The default value for the "method" attribute is "PUBLISH" and does not need to be specified in this example.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE iCalendar SYSTEM "xcal.dtd" [

<!ENTITY jsmith.ifb SYSTEM
"http://www.host.com/calendar/busytime/jsmith.ifb" NDATA BINARY>
]>

<iCalendar>
<vcalendar version="2.0"
  prodid="-//hacksw/handcal//NONSGML 1.0//EN">
<vfreebusy>
  <uid>19980313T133000@ical1.host.com</uid>
  <dtstamp>19990104T133010Z</dtstamp>
  <organizer>MAILTO:jsmith@host.com</organizer>
  <dtstart>19980313T141711Z</dtstart>
  <dtend>19980410T141711Z</dtend>
  <url uri="jsmith.ifb" />
  <freebusy>19980314T233000Z/19980315T003000Z</freebusy>
  <freebusy>19980316T153000Z/19980316T163000Z</freebusy>
  <freebusy>19980318T030000Z/19980318T040000Z</freebusy>
</vfreebusy>
</vcalendar>
</iCalendar>
```

3.12 Request busy time

The following is a snippet of an iCalendar XML document that requests a calendar user's busy time information.


```

<iCalendar>
<vcalendar method="REQUEST"
  version="2.0"
  prodid="-//hacksw/handcal//NONSGML 1.0//EN">
<vfreebusy>
  <uid>19970901T083000@ical1.host.com</uid>
  <dtstamp>19970901T083000Z</dtstamp>
  <organizer>MAILTO:jane_doe@host1.com</organizer>
  <dtstart>19971015T050000Z</dtstart>
  <dtend>19971016T050000Z</dtend>
  <attendee>MAILTO:john_public@host2.com</attendee>
</vfreebusy>
</vcalendar>
</iCalendar>

```

3.13 Response to a busy time request

The following is an iCalendar XML document that responds to request for busy time information.

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE iCalendar SYSTEM "xcal.dtd" [
<!ENTITY jpublic-01.ifb SYSTEM "http://host2.com/pub/busy/jpublic-
01.ifb" NDATA BINARY>
]>

<iCalendar>
<vcalendar method="REPLY"
  version="2.0"
  prodid="-//hacksw/handcal//NONSGML 1.0//EN">
<vfreebusy>
  <uid>19970901T083000@ical1.host.com</uid>
  <dtstamp>19970901T100000Z</dtstamp>
  <organizer>MAILTO:jane_doe@host1.com</organizer>
  <url uri="jpublic-01.ifb" />
  <attendee>MAILTO:john_public@host2.com</attendee>
  <freebusy value="PERIOD">19971015T050000Z/PT8H30M,
  19971015T160000Z/PT5H30M,19971015T223000Z/PT6H30M</freebusy>
</vfreebusy>
</vcalendar>
</iCalendar>

```

3.14 Published event that references time zone information

The following is a snippet of an iCalendar XML document that publishes calendar information about an event that includes date/time

values that reference a time zone definition.

```

<iCalendar>
<vcalendar version="2.0"
  prodid="-//hacksw/handcal//NONSGML 1.0//EN">
<vtimezone>
  <tzid>US-Eastern</tzid>
  <standard>
    <dtstart>19981025T020000</dtstart>
    <tzoffsetfrom>-0400</tzoffsetfrom>
    <tzoffsetto>0500</tzoffsetto>
    <rdate>19981025T020000</rdate>
    <tzname>EST</tzname>
  </standard>
  <daylight>
    <dtstart>19990404T020000</dtstart>
    <tzoffsetfrom>-0500</tzoffsetfrom>
    <tzoffsetto>-0400</tzoffsetto>
    <rdate>19990404T020000</rdate>
    <tzname>EDT</tzname>
  </daylight>
</vtimezone>
<vevent>
  <dtstamp>19980309T231000Z</dtstamp>
  <uid>guid-1.host1.com</uid>
  <dtstart tzid="US-Eastern">19980312T083000</dtstart>
  <dtend tzid="US-Eastern">19980312T093000</dtend>
  <organizer>MAILTO:mrbig@host.com</organizer>
  <description>Project XYZ Review Meeting</description>
  <class>PUBLIC</class>
  <summary>XYZ Project Review</summary>
  <location>1CP Conference Room 4350</location>
  <categories>
    <item>Meeting</item>
  </categories>
  <attendee rsvp="TRUE"
    role="REQ-PART"
    cutype="GROUP">MAILTO:employee-@host.com
  </attendee>
</vevent>
</vcalendar>
</iCalendar>

```

3.15 An event with an alarm

The following is an iCalendar XML with associated alarms. The event specifies alarm definitions for a "display", "audio", "email" and

"procedure" type of alarms. The "method" attribute isn't specified since it is the default value.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE iCalendar PUBLIC "-//IETF//DTD XCAL/iCalendar XML//EN"
"http://www.ietf.org/internet-drafts/draft-ietf-calsch-many-xcal-01.txt"
[
<!ENTITY boom SYSTEM "ftp://host.com/sounds/cannon.wav" NDATA wave>
<!NOTATION wave SYSTEM "WAVE Audio Format">
<!ENTITY doit SYSTEM "http://procs.host.com/litesout.exe" NDATA
binary>
<!NOTATION binary SYSTEM "Foo Bar Executable Format">
]>
<iCalendar>
<vcalendar version="2.0"
  prodid="-//hacksw/handcal//NONSGML 1.0//EN">
<vevent>
  <uid>19990104T130000@host.com</uid>
  <dtstamp>19990104T130100Z</dtstamp>
  <dtstart>19990704T230000Z</dtstart>
  <dtend>19970705T040000Z</dtend>
  <summary>Firework Celebration</summary>
  <categories>
    <item>Holiday</item>
    <item>Special Occasion</item>
  </categories>
  <valarm>
    <action>DISPLAY</action>
    <description>Firework Celebration Tonight at
    6 PM !!!</description>
    <trigger>19990704T224500Z</trigger>
    <repeat>2</repeat>
    <duration>PT15M</duration>
  </valarm>
  <valarm>
    <action>AUDIO</action>
    <trigger>19990704T224500Z</trigger>
    <repeat>2</repeat>
    <duration>PT15M</duration>
    <attach><extref uri="boom"/></attach>
  </valarm>
  <valarm>
    <action>EMAIL</action>
    <description>Firework Celebration Tonight
    at 6 PM on Channel 6!!!</description>
    <summary>*** Firework Celebration On TV ***</summary>
    <trigger>19990704T224500Z</trigger>
    <attendee>MAILTO:PIN1234@pager.host.com</attendee>
```



```
</valarm>
<valarm>
  <action>PROCEDURE</action>
  <attach><extref uri="doit"/></attach>
  <trigger>19990704T230000Z</trigger>
</valarm>
</vevent>
</vcalendar>
</iCalendar>
```

4. iCalendar XML Document Type Definition

The following DTD conforms to XML version 1.0, as specified by [[XML](#)].

```
<?xml version="1.0" encoding="UTF-8"?>

<!-- ***** -->
<!-- Entity declarations -->
<!-- ***** -->

<!ENTITY % attr.altrep "
altrep ENTITY #IMPLIED
">

<!ENTITY % attr.cn "
cn CDATA ''
">

<!ENTITY % attr.cutype "
cutype NMTOKEN 'INDIVIDUAL'
">
<!-- Valid name tokens are "INDIVIDUAL", "GROUP", "RESOURCE" -->
<!-- "ROOM", "UNKNOWN", a non-standard "X-" name or another -->
<!-- IANA registered name. -->

<!ENTITY % attr.delegated-from "
delegated-from CDATA #IMPLIED
">
<!-- delegated-from value is a calendar user address -->

<!ENTITY % attr.delegated-to "
delegated-to CDATA #IMPLIED
">
<!-- delegated-to value is one or more calendar user addresses -->

<!ENTITY % attr.dir "
dir ENTITY #IMPLIED
">
<!-- dir value is a URI to a directory entry -->

<!ENTITY % attr.fmttype "
fmttype CDATA #REQUIRED
">
<!-- fmttype value is any IANA registered content type -->

<!ENTITY % attr.fbtype "
fbtype NMTOKEN 'BUSY'
">
```



```
<!-- Valid token values are "FREE", "BUSY", "BUSY-UNAVAILABLE", -->
<!-- "BUSY-TENTATIVE", a non-standard "X-" name or another -->
<!-- IANA registered name. -->
```

```
<!ENTITY % attr.language "
language CDATA #IMPLIED
">
```

```
<!-- language value is a valid RFC 1766 language string -->
```

```
<!ENTITY % attr.member "
member CDATA #IMPLIED
">
```

```
<!-- member value is one or more calendar user addresses -->
```

```
<!ENTITY % attr.partstat "
partstat NMTOKEN 'NEEDS-ACTION'
">
```

```
<!-- Valid token value for VEVENT: "NEEDS-ACTION", "ACCEPTED", -->
<!-- "DECLINED", "TENTATIVE", "DELEGATED", a non-standard "X- -->
<!-- name or another IANA registered name. -->
```

```
<!-- Valid token value for VTODO: "NEEDS-ACTION", "ACCEPTED", -->
<!-- "DECLINED", "TENTATIVE", "DELEGATED", "COMPLETED", -->
<!-- "IN-PROGRESS, a non-standard "X- name or another IANA -->
<!-- registered name. -->
<!-- Valid token value for VJOURNAL: "NEEDS-ACTION", "ACCEPTED", -->
<!-- "DECLINED", a non-standard "X- name or another IANA -->
<!-- registered name. -->
```

```
<!ENTITY % attr.range "
range NMTOKEN 'THISONLY'
">
```

```
<!-- Valid token values are "THISONLY" or "THISANDPRIOR" or -->
<!-- "THISANDFUTURE" -->
```

```
<!ENTITY % attr.related "
related NMTOKEN 'START'
">
```

```
<!-- Valid token values are "START" or "END" -->
```

```
<!ENTITY % attr.reltype "
reltype NMTOKEN 'PARENT'
">
```

```
<!-- Valid token values are "PARENT", "CHILD", "SIBLING", -->
<!-- a non-standard "X-" name or any IANA registered name. -->
```

```
<!ENTITY % attr.role "
role NMTOKEN 'REQ-PARTICIPANT'
```



```
">
<!-- Valid token values are "CHAIR", "REQ-PARTICIPANT", -->
<!-- "OPT-PARTICIPANT", "NON-PARTICIPANT", a non-standard "X-" -->
<!-- name or any IANA registered name. -->

<!ENTITY % attr.rsvp "
rsvp NMTOKEN 'FALSE'
">
<!-- Valid token values are "TRUE" or "FALSE", -->

<!ENTITY % attr.sent-by "
sent-by CDATA #IMPLIED
">
<!-- sent-by value is a calendar user address -->

<!ENTITY % attr.tzid "
tzid CDATA #IMPLIED
">
<!-- tzid value is a time zone identifier -->

<!ENTITY % cal.comp "
vevent | vtodo | vjournal | vfreebusy | vtimezone
">

<!ENTITY % vevent.opt1 "
class | created | description | dtstamp | dtstart | geo |
last-modified | location | organizer | priority | recurrence-id |
sequence | status | summary | transp | uid | url |
(dtend | duration)
">
<!-- These properties may only appear once in a VEVENT -->
<!ENTITY % vevent.optm "
attach | attendee | categories | comment | contact |
exdate | exrule | rdate | related-to | resources | request-status |
rrule
">
<!-- These properties may appear one or more times in a VEVENT -->

<!ENTITY % vtodo.opt1 "
class | completed | created | description | dtstamp | dtstart |
geo | last-modified | location | organizer | percent | priority |
recurrence-id | sequence | status | summary | uid | url |
(due | duration)
">
<!-- These properties may only appear once in a VTOD0 -->

<!ENTITY % vtodo.optm "
attach | attendee | categories | comment | contact |
```



```
exdate | exrule | request-status | related-to | resources |
rdate | rrule
">
<!-- These properties may appear one or more times in a VTODO -->

<!ENTITY % vjournal.opt1 "
class | created | description | dtstart | dtstamp | last-modified |
organizer | recurrence-id | sequence | status | summary | uid | url
">
<!-- These properties may only appear once in a VJOURNAL -->

<!ENTITY % vjournal.optm "
attach | attendee | categories | comment | contact |
exdate | exrule | related-to | rdate | rrule | request-status
">
<!-- These properties may appear one or more times in a VJOURNAL -->

<!ENTITY % vfreebusy.opt1 "
contact | dtstamp | dtstart | dtend | duration |
organizer | uid | url
">
<!-- These properties may only appear once in a VFREEBUSY -->

<!ENTITY % vfreebusy.optm "
attendee | comment | freebusy | request-status
">
<!-- These properties may appear one or more times in a -->
<!-- VFREEBUSY -->

<!ENTITY % vtimezone.man "
tzid
">
<!-- These properties must appear in a VTIMEZONE -->

<!ENTITY % vtimezone.opt1 "
last-modified | tzurl
">
<!-- These properties may only appear once in a VTIMEZONE -->

<!ENTITY % vtimezone.mann "
(standard | daylight), (standard | daylight)*
">
<!-- These properties must appear in a VTIMEZONE and may -->
<!-- appear multiple times -->

<!ENTITY % standard.man "
dtstart | tzoffsetto | tzoffsetfrom
">
```



```
<!-- These properties must appear in a STANDARD, but only once -->

<!ENTITY % standard.optm "
comment | rdate | rrule | tzname
">
<!-- These properties may appear one or more times in a STANDARD -->

<!ENTITY % daylight.man "
dtstart | tzoffsetto | tzoffsetfrom
">
<!-- These properties must appear in a DAYLIGHT, but only once -->

<!ENTITY % daylight.optm "
comment | rdate | rrule | tzname
">
<!-- These properties may appear one or more times in a DAYLIGHT -->

<!ENTITY % audio.man "
action, trigger
">
<!-- These properties must appear in an audio VALARM. -->

<!ENTITY % audio.optx "
duration | repeat
">
<!-- These properties may appear once in an audio VALARM. If one -->
<!-- appears, then both must appear. -->

<!ENTITY % audio.opt1 "
attach
">
<!-- These properties may appear once in an audio VALARM. -->

<!ENTITY % valarm.audio "
(%audio.man;), (%audio.optx;)*, (%audio.opt1;)
">

<!ENTITY % display.man "
action, description, trigger
">
<!-- These properties must appear in a display VALARM. -->
<!ENTITY % display.optx "
duration | repeat
">
<!-- These properties may appear once in a display VALARM. If -->
<!-- one appears, then both must appear. -->

<!ENTITY % valarm.display "
```



```
(&#x000D;), (&#x000A;)*
">

<!ENTITY % email.man "
action, description, summary, trigger
">
<!-- These properties must appear in an email VALARM. -->

<!ENTITY % email.optx "
duration | repeat
">
<!-- These properties may appear once in an email VALARM. If one -->
<!-- appears, then both must appear. -->

<!ENTITY % email.optm "
attach
">
<!-- These properties may appear one or more times in an email -->
<!-- VALARM. -->

<!ENTITY % email.mann "
attendee
">
<!-- These properties must appear in an email VALARM. The may -->
<!-- appear more than once. -->

<!ENTITY % valarm.email "
(&#x000D;), (&#x000A;)*, (&#x000D;)*,
(&#x000D;)*
">

<!ENTITY % procedure.man "
action, attach, trigger
">
<!-- These properties must appear in an audio VALARM. -->

<!ENTITY % procedure.optx "
duration | repeat
">
<!-- These properties may appear once in an procedure VALARM. -->
<!-- If one appears, then both must appear. -->

<!ENTITY % procedure.opt1 "
description
">
<!-- These properties may appear once in a procedure VALARM -->
<!ENTITY % valarm.procedure "
(&#x000D;), (&#x000A;)*, (&#x000D;)?
```


">

<!-- ***** -->

<!-- iCalendar value type notation declarations -->

<!-- ***** -->

<!-- NOTE: The "XCAL" text in the following NOTATION values will be replaced with the text "RFC xxxx", where "xxxx" is the RFC number, when this memo is published as a RFC. -->

<!NOTATION BINARY PUBLIC "-//IETF//NOTATION XCAL/Value Type/Binary//EN">

<!NOTATION BOOLEAN PUBLIC "-//IETF//NOTATION XCAL/Value Type/Boolean//EN">

<!NOTATION CALADR PUBLIC "-//IETF//NOTATION XCAL/Value Type/Calendar User Address//EN">

<!NOTATION DATE PUBLIC "-//IETF//NOTATION XCAL/Value Type/Date//EN">

<!NOTATION DATE-TIME PUBLIC "-//IETF//NOTATION XCAL/Value Type/Date-Time//EN">

<!NOTATION DURATION PUBLIC "-//IETF//NOTATION XCAL/Value Type/Duration//EN">

<!NOTATION FLOAT PUBLIC "-//IETF//NOTATION XCAL/Value Type/Float//EN">

<!NOTATION INTEGER PUBLIC "-//IETF//NOTATION XCAL/Value Type/Integer//EN">

<!NOTATION PERIOD PUBLIC "-//IETF//NOTATION XCAL/Value Type/Period of Time//EN">

<!NOTATION RECUR PUBLIC "-//IETF//NOTATION XCAL/Value Type/Recurrence Rule//EN">

<!NOTATION TEXT PUBLIC "-//IETF//NOTATION XCAL/Value Type/Text//EN">

<!NOTATION TIME PUBLIC "-//IETF//NOTATION XCAL/Value Type/Time//EN">

<!NOTATION URI PUBLIC "-//IETF//NOTATION XCAL/Value Type/URI//EN">

<!NOTATION UTC-OFFSET PUBLIC "-//IETF//NOTATION XCAL/Value Type/UTC-Offset//EN">

<!NOTATION X-NAME PUBLIC "-//IETF//NOTATION XCAL/Value Type/X-Name//EN">

<!-- ***** -->

<!-- iCalendar property element/attribute declarations -->


```

<!-- ***** -->

<!ELEMENT br EMPTY>
<!-- Signifies a new line in the TEXT value content information -->

<!-- Description component properties element type declarations -->

<!ELEMENT attach (extref | b64bin)>
<!-- extref holds a reference to an external entity that -->
<!-- has the attachment. b64bin holds the inline BASE64 encoded -->
<!-- binary data for the attachment as defined in RFC 2045. -->

<!ELEMENT extref EMPTY>
<!ATTLIST extref
uri ENTITY #REQUIRED>

<!ELEMENT b64bin (#PCDATA)>
<!ATTLIST b64bin
%attr.fmttype;
value NOTATION (BINARY) #IMPLIED>

<!ELEMENT categories (item)*>

<!ELEMENT item (#PCDATA)>
<!ATTLIST item
%attr.language;
value NOTATION (TEXT) #IMPLIED>

<!ELEMENT class (#PCDATA)>
<!ATTLIST class
%attr.language;
value NOTATION (TEXT) #IMPLIED>

<!ELEMENT comment (#PCDATA)*>
<!ATTLIST comment
%attr.language;
%attr.altrep;
value NOTATION (TEXT) #IMPLIED>

<!ELEMENT description (#PCDATA)*>
<!ATTLIST description
%attr.language;
%attr.altrep;
value NOTATION (TEXT) #IMPLIED>

<!ELEMENT geo (lat, lon)>

<!ELEMENT lat (#PCDATA)>

```



```
<!ATTLIST lat value NOTATION (FLOAT) #IMPLIED>
<!-- A decimal degree float number to 6 decimal places -->

<!ELEMENT lon (#PCDATA)>
<!ATTLIST lon value NOTATION (FLOAT) #IMPLIED>
<!-- A decimal degree float number to 6 decimal places -->

<!ELEMENT location (#PCDATA)>
<!ATTLIST location
%attr.language;
%attr.altrep;
value NOTATION (TEXT) #IMPLIED>

<!ELEMENT percent (#PCDATA)>
<!ATTLIST percent
value NOTATION (INTEGER) #IMPLIED>

<!ELEMENT priority (#PCDATA)>
<!ATTLIST priority
value NOTATION (INTEGER) #IMPLIED>

<!ELEMENT resources (#PCDATA)>
<!ATTLIST resources
%attr.language;
%attr.altrep;
value NOTATION (TEXT) #IMPLIED>

<!ELEMENT status (#PCDATA)>
<!ATTLIST status
%attr.language;
%attr.altrep;
value NOTATION (TEXT) #IMPLIED>
<!-- Text value must match the valid values for the particular -->
<!-- calendar component. -->

<!ELEMENT summary (#PCDATA)>
<!ATTLIST summary
%attr.language;
%attr.altrep;
value NOTATION (TEXT) #IMPLIED >

<!-- Data and time component property element type declarations -->

<!ELEMENT dtstart (#PCDATA)>
<!ATTLIST dtstart
%attr.tzid;
value NOTATION (DATE-TIME | DATE) "DATE-TIME">
```



```
<!ELEMENT dtend (#PCDATA)>
<!ATTLIST dtend
%attr.tzid;
value NOTATION (DATE-TIME | DATE) "DATE-TIME">
```

```
<!ELEMENT due (#PCDATA)>
<!ATTLIST due
%attr.tzid;
value NOTATION (DATE-TIME | DATE) "DATE-TIME">
```

```
<!ELEMENT completed (#PCDATA)>
<!ATTLIST completed
value NOTATION (DATE-TIME) #IMPLIED>
```

```
<!ELEMENT duration (#PCDATA)>
<!ATTLIST duration
value NOTATION (DURATION) #IMPLIED>
```

```
<!ELEMENT freebusy (#PCDATA)>
<!ATTLIST freebusy
%attr.fbtype;
value NOTATION (PERIOD) #IMPLIED>
```

```
<!ELEMENT transp (#PCDATA)>
<!ATTLIST transp
value NOTATION (TEXT) #IMPLIED>
<!-- Text value must be one of the valid enumerations. -->
```

```
<!-- Time zone component property element type declarations -->
```

```
<!ELEMENT tzid (#PCDATA)>
<!ATTLIST tzid
value NOTATION (TEXT) #IMPLIED>
```

```
<!ELEMENT tzname (#PCDATA)>
<!ATTLIST tzname
%attr.language;
value NOTATION (TEXT) #IMPLIED>
```

```
<!ELEMENT tzoffsetfrom (#PCDATA)>
<!ATTLIST tzoffsetfrom
value NOTATION (UTC-OFFSET) #IMPLIED>
```

```
<!ELEMENT tzoffsetto (#PCDATA)>
<!ATTLIST tzoffsetto
value NOTATION (UTC-OFFSET) #IMPLIED>
```

```
<!ELEMENT tzurl EMPTY>
```



```
<!ATTLIST tzurl
uri ENTITY #REQUIRED>
```

```
<!-- Relationship component property element type declarations -->
```

```
<!ELEMENT attendee (#PCDATA)>
<!ATTLIST attendee
%attr.language;
%attr.cn;
%attr.role;
%attr.partstat;
%attr.rsvp;
%attr.cutype;
%attr.member;
%attr.delegated-to;
%attr.delegated-from;
%attr.sent-by;
%attr.dir;
value NOTATION (CALADR) #IMPLIED>
```

```
<!ELEMENT contact (#PCDATA)*>
<!ATTLIST contact
%attr.language;
%attr.altrep;
value NOTATION (TEXT) #IMPLIED>
```

```
<!ELEMENT organizer (#PCDATA)>
<!ATTLIST organizer
%attr.language;
%attr.cn;
%attr.sent-by;
%attr.dir;
value NOTATION (CALADR) #IMPLIED>
```

```
<!ELEMENT recurrence-id (#PCDATA)>
<!ATTLIST recurrence-id
%attr.tzid;
%attr.range;
value NOTATION (DATE-TIME | DATE) "DATE-TIME">
```

```
<!ELEMENT related-to (#PCDATA)>
<!ATTLIST related-to
%attr.reltype;
value NOTATION (TEXT) #IMPLIED>
```

```
<!ELEMENT url EMPTY>
<!ATTLIST url
uri ENTITY #REQUIRED>
```



```
<!ELEMENT uid (#PCDATA)>
<!ATTLIST uid
value NOTATION (TEXT) #IMPLIED>

<!-- Recurrence component property element type declarations -->

<!ELEMENT exdate (#PCDATA)>
<!ATTLIST exdate
%attr.tzid;
value NOTATION (DATE-TIME | DATE) "DATE-TIME">

<!ELEMENT exrule (#PCDATA)>
<!ATTLIST exrule
value NOTATION (RECUR) #IMPLIED>

<!ELEMENT rdate (#PCDATA)>
<!ATTLIST rdate
%attr.tzid;
value NOTATION (DATE-TIME | DATE) "DATE-TIME">

<!ELEMENT rrule (#PCDATA)>
<!ATTLIST rrule
value NOTATION (RECUR) #IMPLIED>

<!-- Alarm component property element type declarations -->

<!ELEMENT action (#PCDATA)>
<!ATTLIST action
value NOTATION (TEXT) #IMPLIED>
<!-- Text value must be a valid enumeration -->

<!ELEMENT repeat (#PCDATA)>
<!ATTLIST repeat
value NOTATION (INTEGER) #IMPLIED>

<!ELEMENT trigger (#PCDATA)>
<!ATTLIST trigger
%attr.related-to;
value NOTATION (DURATION | DATE-TIME) "DURATION">

<!-- Change management component property element type -->
<!-- declarations -->

<!ELEMENT created (#PCDATA)>
<!ATTLIST created
value NOTATION (DATE-TIME) #IMPLIED>

<!ELEMENT dtstamp (#PCDATA)>
```



```
<!ATTLIST dtstamp
value NOTATION (DATE-TIME) #IMPLIED>

<!ELEMENT last-modified (#PCDATA)>
<!ATTLIST last-modified
value NOTATION (DATE-TIME) #IMPLIED>

<!ELEMENT sequence (#PCDATA)>
<!ATTLIST sequence
value NOTATION (INTEGER) #IMPLIED>

<!-- Miscellaneous component property element type declarations -->

<!ELEMENT request-status (#PCDATA)>
<!ATTLIST request-status
%attr.language;
value NOTATION (TEXT) #IMPLIED>

<!-- iCalendar object element type declarations -->

<!ELEMENT iCalendar (vcalendar+)>

<!ELEMENT vcalendar (%cal.comp;)*>
<!ATTLIST vcalendar
%attr.language;
xmlns CDATA #FIXED 'http://www.ietf.org/internet-drafts/draft-
ietf-calsch-many-xcal-01.txt'
calscale CDATA "GREGORIAN"
method CDATA "PUBLISH"
version CDATA #REQUIRED
prodid CDATA #IMPLIED>
<!-- version - Must be "2.0" if document conforms to this spec. -->
<!-- calscale - Calendar scale. Default is GREGORIAN. -->
<!-- method - C&S method. Default is iTIP PUBLISH. -->
<!-- prodid - ISO 9070 FPI for product that generated iCalendar. -->

<!-- "vevent" element type declaration -->
<!ELEMENT vevent ((%vevent.opt1;)*, (%vevent.optm;)*, valarm*)>

<!-- "vtodo" element type declaration -->
<!ELEMENT vtodo ((%vtodo.opt1;)*, (%vtodo.optm;)*, valarm*)>

<!-- "vjournal" element type declaration -->
<!ELEMENT vjournal ((%vjournal.opt1;)*, (%vjournal.optm;)*)>

<!-- "vfreebusy" element type declaration -->
<!ELEMENT vfreebusy ((%vfreebusy.opt1;)*, (%vfreebusy.optm;)*)>
```



```
<!-- "vtimezone" element type declaration -->
<!ELEMENT vtimezone (%vtimezone.man;; (%vtimezone.opt1;)*,
(%vtimezone.mann;)*)>

<!ELEMENT standard (((%standard.man;)*), (%standard.optm;)*)>

<!ELEMENT daylight (((%daylight.man;)*), (%daylight.optm;)*)>

<!ELEMENT valarm ((%valarm.audio;) | (%valarm.display;) |
(%valarm.email;) | (%valarm.procedure;))>
```

5. Acknowledgments

The following have participated in the drafting and discussion of this memo:

Greg FitzPatrick, Charles Goldfarb, Paul Hoffman, Lisa Lippert, Thomas Rowe.

6. IANA Considerations

This document defines a XML Formal Public Identifier (FPI), based on a format defined in [ISO 9070], that identifies a XML document type corresponding to this memo. Publication of this memo constitutes registration of this identifier.

In addition, this memo defines the XML FPIs corresponding to each of the value types specified in [[RFC 2445](#)].

7. Security Considerations

CDATA Sections - - A XML iCalendar document may contain CDATA sections to represent content for specific element types. The CDATA section specifies arbitrary character data that is not meant to be interpreted. It is not scanned by the XML parser for markup. While this memo restricts that any CDATA section MUST NOT contain markup or other such alternate representation for the property value, in general, CDATA section from a non-conformant implementation can contain content such as HTML markup. HTML text can be used to invoke programs. Implementors should be aware that this may leave an implementation open to malicious attack that might occur as a result of executing the markup in the CDATA section.

PROCEDURAL ALARMS - - A XML iCalendar document can be created that contains a "VEVENT" and "VTODO" calendar component with "VALARM" calendar components. The "VALARM" calendar component can be of type PROCEDURE and can have an attachment containing some sort of executable program. Implementations that incorporate these types of alarms are subject to any virus or malicious attack that might occur as a result of executing the attachment.

ATTACHMENTS - - A XML iCalendar document can include references to Uniform Resource Locators that can be programmed resources. Implementers and users of this memo should be aware of the network security implications of accepting and parsing such information.

In addition, the security considerations observed by implementations of electronic mail systems should be followed for this memo.

8. Bibliography

[FPI] F. Dawson, "iCalendar Formal Public Identifier", Internet Draft, <http://www.internic.net/internet-drafts/draft-calsch-icalfpi-00.txt>, September 1998.

[ISO9070] "Information Technology_SGML Support Facilities_Registration Procedures for Public Text Owner Identifiers", ISO/IEC 9070, Second Edition, International Organization for Standardization, April 1991.

[RFC 2045] N. Freed, N. Borenstein, "Multipurpose Internet Mail Extensions (MIME) - Part One: Format of Internet Message Bodies", [RFC 2045](http://www.ietf.org/rfc/rfc2045), November 1996.

[RFC 2119] S. Bradner, "Key words for use in RFCs to Indicate Requirement Levels", [RFC 2119](http://www.ietf.org/rfc/rfc2119.txt), <http://www.ietf.org/rfc/rfc2119.txt>, March 1997.

[RFC 2445] F. Dawson and D. Stenerson, "Internet Calendaring and Scheduling Core Object Specification (iCalendar)", [RFC 2445](http://www.ietf.org/rfc/rfc2445.txt), <http://www.ietf.org/rfc/rfc2445.txt>, November 1998.

[XML] "Extensible Markup Language (XML)", Worldwide Web Consortium, <http://www.w3.org/TR/1998/REC-xml-19980210>, February 1998.

[XML] "Extensible Markup Language (XML)", Worldwide Web Consortium, <http://www.w3.org/TR/1998/REC-xml-19980210>, February 1998.

Authors' Addresses

Frank Dawson
Nokia Corporation

Phone: +1 (972) 894 4083
EMail: frank.dawson@nokia.com

Surendra K. Reddy
Oracle
M/S 6op3
500 Oracle Parkway
Redwoodshores, CA 94065
US

Phone: +1 (650) 506 5441
Fax: +1 (650) 654 6205
EMail: skreddy@us.oracle.com

Doug Royer
INET-Consulting LLC
1795 W. Broadway #266
Idaho Falls, ID 83402
US

Phone: +1 (208) 520 4044
Fax: +1 (208) 552 1179
EMail: doug@royer.com

Eric R. Plamondon
Oracle
2000 Peel Street, 4th Floor
Montreal, QC H3A 2W5
Canada

Phone: +1 (514) 733 8500
Fax: +1 (514) 733 8878
EMail: ericp@steltor.com

Full Copyright Statement

Copyright (C) The Internet Society (2002). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.

