WS-Calendar Version 1.0

Committee Specification Draft ~~03~~04 / Public Review Draft ~~02~~03

~~25 March~~17 June 2011

Specification URIs:

This ~~Version~~version:

~~http://docs.oasis-open.org/ws-calendar/ws-calendar-spec/v1.0/csprd02/ws-calendar-spec-v1.0-csprd02.pdf (Authoritative)~~
~~http://docs.oasis-open.org/ws-calendar/ws-calendar-spec/v1.0/csprd02/ws-calendar-spec-v1.0-csprd02.html~~
~~http://docs.oasis-open.org/ws-calendar/ws-calendar-spec/v1.0/csprd02/ws-calendar-spec-v1.0-csprd02.doc~~
http://docs.oasis-open.org/ws-calendar/ws-calendar-spec/v1.0/csprd03/ws-calendar-spec-v1.0-csprd03.pdf (Authoritative)
http://docs.oasis-open.org/ws-calendar/ws-calendar-spec/v1.0/csprd03/ws-calendar-spec-v1.0-csprd03.html
http://docs.oasis-open.org/ws-calendar/ws-calendar-spec/v1.0/csprd03/ws-calendar-spec-v1.0-csprd03.doc

Previous ~~Version~~version:

~~http://docs.oasis-open.org/ws-calendar/ws-calendar-spec/v1.0/csd02/ws-calendar-spec-v1.0-csd02.pdf (Authoritative)~~
~~http://docs.oasis-open.org/ws-calendar/ws-calendar-spec/v1.0/csd02/ws-calendar-spec-v1.0-csd02.html~~
~~http://docs.oasis-open.org/ws-calendar/ws-calendar-spec/v1.0/csd02/ws-calendar-spec-v1.0-csd02.doc~~
http://docs.oasis-open.org/ws-calendar/ws-calendar-spec/v1.0/csd03/ws-calendar-spec-v1.0-csd03.pdf (Authoritative)
http://docs.oasis-open.org/ws-calendar/ws-calendar-spec/v1.0/csd03/ws-calendar-spec-v1.0-csd03.html
http://docs.oasis-open.org/ws-calendar/ws-calendar-spec/v1.0/csd03/ws-calendar-spec-v1.0-csd03.doc

Latest ~~Version~~version:

~~http://docs.oasis-open.org/ws-calendar/ws-calendar/v1.0/ws-calendar-1.0-spec.pdf (Authoritative)~~
~~http://docs.oasis-open.org/ws-calendar/ws-calendar/v1.0/ws-calendar-1.0-spec.html~~
~~http://docs.oasis-open.org/ws-calendar/ws-calendar/v1.0/ws-calendar-1.0-spec.doc~~
http://docs.oasis-open.org/ws-calendar/ws-calendar/v1.0/ws-calendar-1.0-spec.pdf (Authoritative)
http://docs.oasis-open.org/ws-calendar/ws-calendar/v1.0/ws-calendar-1.0-spec.html
http://docs.oasis-open.org/ws-calendar/ws-calendar/v1.0/ws-calendar-1.0-spec.doc

Technical Committee:

~~OASIS Web Services Calendar (WS-Calendar) TC~~

OASIS Web Services Calendar (WS-Calendar) TC

Chair~~(s):~~:

~~Toby Considine~~

~~Editor(s):~~

~~Toby Considine~~
~~Mike Douglass~~
Toby Considine

Editors:

Toby Considine
Mike Douglass

Related work:

~~XML Schemas for WS-Calendar Version 1.0~~

This specification is related to:

- ~~·~~IETF RFC5545, ICalendar
- ~~·~~IETF RFC5546, ICalendar Transport
- ~~·~~IETF RFC2447, ICalendar Message Based Interoperability
- ~~·~~~~IETF XCAL~~IETF XCAL specification in progress
- ~~·~~~~IETF / CalConnect Calendar Resource Schema~~IETF / CalConnect Calendar Resource Schema specification in progress

~~Declared XML Namespace(s):~~

~~urn:ietf:params:xml:ns:icalendar-2.0~~

XML schemas: ws-calendar-spec/v1.0/csprd03/xsd/

Abstract:

WS-Calendar describes~~:~~

- A semantic (or information) model for exchange of calendar information to coordinate activities
- A means of synchronizing and maintaining calendars

The specification includes XML vocabularies for the interoperable and standard exchange of:

- Schedules, including sequences of schedules
- Intervals, including sequences of Intervals
- Other calendar information consistent with the IETF iCalendar standards

These vocabularies describe schedules and Intervals future, present, or past (historical).

~~In this Working Draft the means for synchronizing and maintaining calendars uses REST; in a future version a web services set of services will be defined. The document is divided into three parts; Parts 1 and 2 are in version 1.0; Part 3 will be in a later version.~~

The specification is divided into three parts.

1) The ~~semantic~~information model and XML vocabularies for exchanging schedule information

2) RESTful Services for calendar update and synchronization

3) Web services for calendar update and synchronization

The Technical Committee has decided not to publish Parts 2 and 3 until a later version.

Status:

This document was last revised or approved by the ~~OASIS Web Services Calendar (WS-Calendar) TC~~OASIS Web Services Calendar (WS-Calendar) TC on the above date. The level of approval is also listed above. Check the “Latest ~~Version~~version” location noted above for possible later revisions of this document.

Technical Committee members should send comments on this specification to the Technical Committee’s email list. Others should send comments to the Technical Committee by using the “~~Send A Comment~~Send A Comment” button on the Technical Committee’s web page at http://www.oasis-open.org/committees/ws-calendar/.

For information on whether any patents have been disclosed that may be essential to implementing this specification, and any offers of patent licensing terms, please refer to the Intellectual Property Rights section of the Technical Committee web page~~: http://www.oasis-open.org/committees/ws-calendar/ipr.php.~~ (http://www.oasis-open.org/committees/ws-calendar/ipr.php).

Citation ~~Format~~format:

When referencing this specification the following citation format should be used:

[~~ws-calendar-spec~~WS-Calendar]

WS-Calendar Version 1.0. ~~25 March~~17 June 2011. OASIS Committee Specification Draft 04 / Public Review Draft ~~02. http://docs.oasis-open.org/ws-calendar/ws-calendar-spec/v1.0/csprd02/ws-calendar-spec-v1.0-csprd02.html~~03. http://docs.oasis-open.org/ws-calendar/ws-calendar-spec/v1.0/csprd03/ws-calendar-spec-v1.0-csprd03.pdf.

Notices

Table of Contents

Tables

Index of Tables

Index of Examples

1 Introduction

The ~~semantic~~information model of WS-Calendar is intended to be used ~~by~~to define information payloads for Web services and ~~other service~~Service-style interactions~~.~~ **[SOA-RM]**. Placing these requirements in context requires a brief overview of service requirements.

~~One of the most~~ Agreement on when something should or did occur is fundamental ~~components of~~to negotiating service use. Negotiated services ~~is agreeing when something should occur, and in auditing when they did occur.~~must be audited to understand timely performance. Short running services traditionally have been handled as if they were instantaneous, and have handled scheduling through just-in-time requests. Longer running processes, including physical processes, may require significant lead~~-~~-times. When multiple long-running services participate in the same business process, it may be more important to negotiate a common completion time than a common start time. Pre-existing approaches that rely on direct control of such services by a central system increases integration costs and reduce interoperability as they require the controlling agent to know and manage multiple lead times.

Not all services are requested one time as needed. Processes may have multiple and periodic occurrences. An agent may need to request identical processes on multiple schedules. An agent may request services to coincide with or to avoid human interactions. Service performance may be required on the first Tuesday of every month, or in weeks in which there is no payroll, to coordinate with existing business processes. Service performance requirements may vary by local time zone. A common schedule communication must support diverse requirements.

Web services already coordinate a number of physical processes. Web services for building-based systems include the standards **[oBIX]**, BACnet/WS[1] LON-WS[2], OPC UA[3], as well as a number of proprietary systems. ~~LON-WS[4],~~ The European research and advanced development project SIRENA (Service Infrastructure for Real time Embedded Networked Applications) explored SOA for buildings, factories and devices, including SODA (Service Oriented Device Architecture). SOA4D[5] (Service-Oriented Architecture for Devices) offers a collaborative open source development web platform, including implementations (**[SOAP]** messaging, **[WS-Management]**, **[WS-Security]**, **[~~DP-WS~~DPWS]**) adapted to the specific constraints of embedded devices. There is a growing interest in coordinating the activities of things, building systems, industrial processes, homes, with human enterprise activities. In particular, if building systems coordinate with the schedules of the building's occupants, they can reduce energy use while improving performance.

An increasing number of specifications envision synchronization of processes through mechanisms including broadcast scheduling. Efforts to build an intelligent power grid (or smart grid) rely on coordinating processes in homes, offices, and industry with projected and actual power availability;

[1] BACnet® is a registered trademark of American Society of Heating, Refrigerating and Air-Conditioning Engineers (ASHRAE).

[2] LON is a registered trademark of Echelon Corporation.

[3] OPC UA is owned by the OPC Foundation

~~[4] LON is a registered trademark of Echelon Corporation.~~

[5] http://cms.soa4d.org/

mechanisms proposed include communicating different prices at different times. ~~Several active OASIS Technical Committees require a common means to specify schedule and interval: Energy Interoperation **[EITC]** and Energy Market Information Exchange **[EMIX]**. Emergency management coordinators wish to inform geographic regions of future events, such as a projected tornado touchdown, using **[EDXL]**. The open Building Information Exchange specification **[oBIX]** lacks a common schedule communications for interaction with enterprise activities. These and other efforts would~~These and other efforts can benefit from a common cross-domain, cross specification standard for communicating schedule and interval.

For human interactions and human scheduling, the well-known iCalendar format ~~is used to address~~addresses these problems. Prior to WS-Calendar, there has been no comparable standard for web services. As an increasing number of physical processes become managed by web services, the lack of a similar standard for scheduling and coordination of services becomes critical.

The ~~intent of the~~ WS-Calendar ~~technical committee was to adapt the existing specifications for calendaring~~Technical Committee (TC) based its work upon the iCalendar specification as updated in 2009 (IETF **[RFC5545]** and ~~apply them to develop~~ its the XML serialization **[XCAL]**, currently (2011-05) on a ~~standard for how schedule and event information is passed between and within services.~~standards track in the IETF. The ~~standard~~specification adopts the semantics and vocabulary of iCalendar for application to the completion of ~~web~~ service contracts~~. WS Calendar builds on work done and ongoing in The~~ and inter-process interactions. Members of the Calendaring and Scheduling Consortium (CalConnect~~), which works to increase interoperation between calendaring systems.~~ .org) developed both updates to IETF specifications and provided advice to this TC.

While this specification (WS-Calendar) defines the use of core semantic elements from iCalendar, no part of this document ~~prevents~~is intended to prevent the use of other semantic elements from iCalendar from being used. WS-Calendar describes the minimal use of that standard, not the maximal.

Everything with the exception of all examples, all appendices, and the introduction is normative unless otherwise specifically noted.

1.1 Terminology

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]**

1.2 Normative References

~~**Calendar Resource Schema**~~ **CalendarResource** C. Joy, C. Daboo, M ~~Douglas~~Douglass, *Schema for representing resources for calendaring and scheduling services*, ~~http://tools.ietf.org/html/draft-cal-resource-schema-00, (Internet-Draft), April~~http://tools.ietf.org/html/draft-cal-resource-schema-03, (Internet-Draft), November 2010.

~~**CalDAV** C. Daboo, B. Desruisseaux, L. Dusseault, *Calendaring Extensions to WebDAV (CalDAV)*, http://www.ietf.org/RFC/RFC4791.txt, IETF RFC4791, March 1997.~~

~~**DPWS** T Nixon, A Regnier, D Driscoll, Antoine Mensch, *Devices Profile for Web Services 1.1*, http://docs.oasis-open.org/ws-dd/ns/dpws/2009/01, July 2009~~

FreeBusy~~ **Read URL**~~ E York. *Freebusy read URL*, ~~http://www.calconnect.org/pubdocs/CD0903%20Freebusy%20Read%20URL%20V1.0.pdf~~ http://www.calconnect.org/pubdocs/CD0903%20Freebusy%20Read%20URL%20V1.0.pdf, April 2009

REST T Fielding, Architectural Styles and the Design of Network-based Software Architectures, http://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm., Dissertation, 2000

ISO8601 ISO (International Organization for Standardization). *Representations of dates and times, third edition*, December 2004, (ISO 8601:2004)

RFC2119 S. Bradner, *Key words for use in RFCs to Indicate Requirement Levels*, ~~http://www.ietf.org/RFC/RFC2119.txt~~http://www.ietf.org/RFC/RFC2119.txt, IETF RFC2119, March 1997.

RFC2616 R Fielding, et al. et al, Hypertext Transfer Protocol -- HTTP/1.1 ~~**RFC2447**~~ ~~F. Dawson, S. Mansour, S. Silverberg, iCalendar Message-Based Interoperability Protocol (iMIP), http://www.ietf.org/RFC/RFC2247.txt, IETF RFC2447, December 2009.~~

~~**RFC2616**~~ ~~R Fielding, et al. et al, Hypertext Transfer Protocol -- HTTP/1.1 http://tools.ietf.org/html/RFC2616, IETF RFC2616, November 1998~~

http://tools.ietf.org/html/RFC2616, IETF RFC2616, draft standard, June, 1999

RFC3339 G Klyne, C Newman, *Date and Time on the Internet: Timestamps* ~~http://tools.ietf.org/html/rfc3339~~ http://tools.ietf.org/html/rfc3339, IETF RFC3339 proposed standard, July 2002

RFC4791 Daboo, et al. *Calendaring Extensions to WebDAV (CalDAV*~~).~~ ~~http://www.ietf.org/rfc/rfc4791.txt~~), http://www.ietf.org/rfc/rfc4791.txt. IETF RFC ~~2119~~4791, proposed standard, March 2007

~~**RFC4918**~~ ~~L. Dusseault, HTTP Extensions for Web Distributed Authoring and Versioning (WebDAV). http://tools.ietf.org/html/rfc4918~~

RFC5545 B. Desruisseaux *Internet Calendaring and Scheduling Core Object Specification (iCalendar)*, ~~http://www.ietf.org/rfc/rfc5545.txt~~http://www.ietf.org/rfc/rfc5545.txt, IETF RFC5545, proposed standard, September 2009~~.~~

RFC5546 C. Daboo *iCalendar Transport-Independent Interoperability Protocol (iTIP)*, ~~http://www.ietf.org/rfc/rfc5546.txt~~http://www.ietf.org/rfc/rfc5546.txt, IETF RFC5546, proposed standard, December 2009.

~~**RFC5598**~~**RFC5988** M. Nottingham, *Web Linking*, ~~http://www.ietf.org/rfc/rfc5598.txt, IETF RFC5598~~http://tools.ietf.org/html/rfc5988, IETF RFC5998, proposed standard, October 2010.

RFC6047 A. Melnikov, *iCalendar Message-Based Interoperability Protocol (iMIP),* http://tools.ietf.org/html/rfc6047, IETF RFC6047, December 2010.

SOA-RM *OASIS Standard, Reference Model for Service Oriented Architecture 1.0*, http://docs.oasis-open.org/soa-rm/v1.0/soa-rm.pdf , October 2006. ~~http://docs.oasis-open.org/soa-rm/v1.0/soa-rm.pdf~~

SOAP M Gudgin, M Hadley, N Mendelsohn, J Moreau, H Nielsen, A Karmarkar, SOAP Version 1.2 Part 1: Messaging Framework, W3C Recommendation ~~http://www.w3.org/TR/soap12-part1/,~~http://www.w3.org/TR/soap12-part1/, W3C Recommendation, April 2007

Vavailability C. Daboo, B. Desruisseaux, Calendar Availability, ~~http://tools.ietf.org/html/draft-daboo-calendar-availability-01, IETF draft, November 2008~~http://tools.ietf.org/html/draft-daboo-calendar-availability-02, IETF Internet Draft, April 2011

~~**WSDL**~~ ~~E Christensen, F Curbera, G Meredith, S Weerawarana, Web Services Description Language (WSDL) 1.1, http://www.w3.org/TR/wsdl, W3C Note March 2001~~

~~**WS-Management**~~ ~~DMTF Standard, Web Services for Management, http://www.dmtf.org/sites/default/files/standards/documents/DSP0226_1.0.0.pdf April 2010~~

~~**WS-Security**~~ ~~K Lawrence, C Kaler, A Nadalin, R Monzillo, P Hallam-Baker, ***Web Services Security: 4 SOAP Message Security 1.1***, http://www.oasis-open.org/committees/download.php/16790/wss-v1.1-spec-os-SOAPMessageSecurity.pdf, February 2006~~

~~**draft**~~ **xCal** C. Daboo, M ~~Douglas~~Douglass, S Lees *xCal: The XML format for iCalendar*, ~~http://tools.ietf.org/html/draft-daboo-et-al-icalendar-in-xml-07, Internet-Draft,~~

~~October 2010~~http://tools.ietf.org/html/draft-daboo-et-al-icalendar-in-xml-08, IETF Internet-Draft, April 2011.

XPATH A Berglund, ~~-~~S Boag, D Chamberlin, ~~MF~~M Fernández, M Kay, J Robie, J Siméon, *XML Path Language (XPath) 2.0~~,~~ (Second Edition),* ~~http://www.w3.org/TR/xpath20/ January 2007~~http://www.w3.org/TR/xpath20/, W3C Recommendation, December 2010.

XLINK S DeRose, E Maler, D Orchard, N Walsh, *XML Linking Language (XLink) Version 1.1~~.~~,* ~~http://www.w3.org/TR/xlink11/~~ http://www.w3.org/TR/xlink11/ May 2010.

~~**XPOINTER**~~ ~~S DeRose, E Maler, R Daniel Jr. XPointer xpointer Scheme, http://www.w3.org/TR/xptr-xpointer/ December 2002.~~

XPTR P Grosso, E Maler, J Marsh, N Walsh, *XPointer Framework*, http://www.w3.org/TR/xptr-framework/ W3C Recommendation, March 2003
P Grosso, E Maler, J Marsh, N Walsh, *XPointer element() Scheme*, http://www.w3.org/TR/xptr-element/, W3C Recommendation March 2003

XML NAMES **T Bray, D Hollander, A Layman, R Tobin, HS Thompson *“Namespaces in XML 1.0 (Third Edition)“*** http://www.w3.org/TR/xml-names/ **W3C Recommendation, December 2009**

XML SCHEMA PV Biron, A Malhotra, XML Schema Part 2: Datatypes Second Edition, ~~http://www.w3.org/TR/xmlschema-2/~~http://www.w3.org/TR/xmlschema-2/ October 2004.

XRD OASIS XRI Committee Draft 01, Extensible Resource Descriptor (XRD) Version 1.0, ~~http://docs.oasis-open.org/xri/xrd/v1.0/cd01/xrd-1.0-cd01.pdf~~http://docs.oasis-open.org/xri/xrd/v1.0/cd01/xrd-1.0-cd01.pdf October 2009.

1.3 Non-Normative References

DPWS **T Nixon, A Regnier, D Driscoll, Antoine Mensch,** *Devices Profile for Web Services 1.1*, http://docs.oasis-open.org/ws-dd/ns/dpws/2009/01**, July 2009**

NIST Framework **NIST Framework and Roadmap for Smart Grid Interoperability Standards**, Office of the National Coordinator for Smart Grid Interoperability, Release 1.0, NIST Special Publication 1108, ~~http://www.nist.gov/public_affairs/releases/upload/smartgrid_interoperability_final.pdf~~http://www.nist.gov/public_affairs/releases/upload/smartgrid_interoperability_final.pdf.

NAESB Requirements **NAESB Smart Grid Requirements** (awaiting publication) (draft contributed) ~~http://lists.oasis-open.org/archives/ws-calendar-comment/201005/doc00000.doc~~http://lists.oasis-open.org/archives/ws-calendar-comment/201005/doc00000.doc, May 2010

~~**REST**~~ ~~T Fielding, Architectural Styles and the Design of Network-based Software Architectures, http://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm.~~

RFC4918 L. Dusseault, HTTP Extensions for Web Distributed Authoring and Versioning (WebDAV). http://tools.ietf.org/html/rfc4918 IETF RFC 4918, proposed standard, June 2007

TZDB P Eggert, A.D. Olson, "Sources for Time Zone and Daylight Saving Time Data", ~~http://www.twinsun.com/tz/tz-link.htm~~http://www.twinsun.com/tz/tz-link.htm

Time Zone Recommendations, CalConnect, CalConnect EDST (Extended Daylight Savings Time) Reflections and Recommendations, Version: 1.1, ~~http://www.calconnect.org/pubdocs/CD0707%20CalConnect%20EDST%20Reflections%20and%20Recommendations%20V1.1.pdf~~http://www.calconnect.org/pubdocs/CD0707%20CalConnect%20EDST%20Reflections%20and%20Recommendations%20V1.1.pdf
October 2010

Time Zone Service, M Douglas, C Daboo, Timezone Service Protocol, Draft RFC,IETF, ~~http://datatracker.ietf.org/doc/draft-douglass-timezone-service/~~http://datatracker.ietf.org/doc/draft-douglass-timezone-service/

WS-Management **DMTF Standard, Web Services for Management,** http://www.dmtf.org/sites/default/files/standards/documents/DSP0226_1.0.0.pdf **April 2010**

WS-Security **K Lawrence, C Kaler, A Nadalin, R Monzillo, P Hallam-Baker, *Web Services Security: 4 SOAP Message Security 1.1,*** http://www.oasis-open.org/committees/download.php/16790/wss-v1.1-spec-os-SOAPMessageSecurity.pdf**, February 2006**

WSDL R Chinnici, J Moreau, A Ryman, S Weerawarana, *Web Services Description Language (WSDL) Version 2.0 Part 1: Core Language* http://www.w3.org/TR/wsdl20/, W3C Recommendation, June 2007.
R Chinnici, H Haas, A Lewis, J Moreau, D Orchard, S Weerawarana, *Web Services Description Language (WSDL) Version 2.0 Part 2: Adjuncts,* http://www.w3.org/TR/wsdl20-adjuncts/, W3C Recommendation, June 2007.

1.4 Contributions

The NIST Roadmap for Smart Grid Interoperability Standards **[NIST Framework]** requested that many standards development organizations (SDOs) and trade associations work together closely in unprecedented ways. An extraordinary number of groups came together and contributed effort, and time, requirements, and documents. The North American Energy Standards Board (NAESB) oversaw meetings with many representatives from every energy sector to contribute requirements to the TC. These meetings were presided over by Jonathan Booe to support the Roadmap's Priority Action Plan 04 (PAP04), a common specification of time and schedule.

NAESB Smart Grid Standards Development Subcommittee:

The following documents are password protected. For information about obtaining access to these documents, please visit www.naesb.org or contact the NAESB office at (713) 356 0060.

Wholesale http://www.naesb.org/member_login_check.asp?doc=fa_2010_weq_api_6_b_ii.doc

Retail http://www.naesb.org/member_login_check.asp?doc=fa_2010_retail_api_9_b_ii.doc

~~1.4~~1.5 Namespace

The XML namespace ~~[XML-ns]~~**[XMLNAMES]** URI that MUST be used by implementations of this specification is:

```
urn:ietf:params:xml:ns:icalendar-2.0
```

Table 1-1 lists the XML schemas that are used in this specification. The choice of any namespace prefix is arbitrary and not semantically significant.

Table 1-1: Namespaces used in this specification

Prefix	Namespace
xs	~~http://www.w3.org/2001/XMLSchema~~http://www.w3.org/2001/XMLSchema
xcal	urn:ietf:params:xml:ns:icalendar-2.0
ts	http://docs.oasis-open.org/ns/ws-calendar/timestamp/201103

The Resource Directory Description Language [~~RDDL 2.0~~RDDL 2.0] document that describes this namespace can be found at ~~http://docs.oasis-open.org/ns/ws-calendar.~~http://docs.oasis-open.org/ns/ws-calendar. The normative schemas for WS-Calendar can be found linked from this namespace document. The schemas are listed in Table 1-2.

Table 1-2: Schemas and Extensions Used in this Specification

Schema	Description
iCalendar.xsd	Base Schema expressing core iCalendar information
iCalendar-params.xsd	Parameters used in iCalendar objects
iCalendar-props.xsd	Properties of iCalendar objects
iCalendar-valtypes.xsd	Values used by iCalendar
iCalendar-link-extension.xsd	Link extensions based on ~~[~~web linking **[RFC5998]** to define relationships between ~~components.~~Components.
iCalendar-wscal-extensions.xsd	Extensions to iCalendar to support service functionality
iCalendar-bw-extensions.xsd	Extensions to support integration with Bedeworks server.
iCalendar-ms-extensions.xsd	Extensions to support integration with MS Exchange Server
TimeStamp.xsd	An ancillary information model describing the elements needed to support event forensics

Reviewers can find the schemas at ~~http://docs.oasis-open.org/ws-calendar/ws-calendar-spec/v1.0/csprd02/xsd/.~~http://docs.oasis-open.org/ws-calendar/ws-calendar-spec/v1.0/csd03/xsd/.

~~1.5~~1.6 Naming Conventions

This specification follows some naming conventions for artifacts defined by the specification, as follows:

For the names of elements and the names of attributes within XSD files, the names follow the lower camelCase convention, with all names starting with a lower case letter. For example,

```
<element name="componentType" type="energyinterop:ComponentType"/>
```

For the names of types within XSD files, the names follow the lower CamelCase convention with all names starting with a lower case letter prefixed by “type-“. For example,

```
<complexType name="type-componentService">
```

For the names of intents, the names follow the lower camelCase convention, with all names starting with a lower case letter, EXCEPT for cases where the intent represents an established acronym, in which case the entire name is in upper case.

An example of an intent that is an acronym is the "SOAP" intent.

~~1.6~~1.7 Editing Conventions

For readability, element names in tables appear as separate words. The actual names are lowerCamelCase, as specified above, and as they appear in the XML schemas.

All elements in the tables not marked as “optional” are mandatory.

Information in the “Specification” column of the tables is normative. Information appearing in the note column is explanatory and non-normative.

All sections explicitly noted as examples are informational and are not to be considered normative.

~~1.7~~1.8 Architectural References

WS-Calendar assumes incorporation into services. Accordingly it assumes a certain amount of definitions of roles, names, and interaction patterns. This document relies heavily on roles and interactions as defined in the OASIS Standard *Reference Model for Service Oriented Architecture* ***[SOA-RM]***.

~~1.8~~1.9 Semantics

Certain terms appear throughout this document, some with extensive definitions. ~~The table~~Table 1-3 provides ~~summary~~ definitions for the convenience of the reader and reviewer. ~~When full definitions of the~~ Many terms ~~below appear~~require fuller discussion than is in this section, and are discussed in greater depth in later sections~~ of this document, with the exception of in the appendices, then that later~~. In all cases, the normative actual definition is ~~normative~~the one in this section.

WS-Calendar terminology begins with a specialized terminology for the segments of time, and for groups of related segments of time. These terms are defined in Table 1-3 through Table 1-6 below.

Table 1-3: Semantics: Foundational Elements

Time Segment	Definition
Component	In iCalendar, the primary information structure is a Component. Intervals and Gluons are new Components defined in this specification.
Duration	Well-known element from iCalendar and **[XCAL]**, Duration is the length of an event scheduled using iCalendar or any of its derivatives. The **[XCAL]** duration is a data type -using the string representation defined in the iCalendar duration.~~ The Duration is the sole descriptive element of the VTODO object that is mandatory in the Interval.~~
Interval	The Interval is a single ~~duration~~Duration derived from the common calendar ~~components~~Components as defined in iCalendar (**[RFC5545**~~**]**) and refined in **[XCAL]**. In Calendar systems, it~~**]**). An Interval is ~~processed as~~part of a ~~vtodo, but the constraints and conformance~~Sequence. An entire Sequence can be scheduled by scheduling a single Interval in that sequence. For this reason, Intervals are ~~different.~~defined through Duration rather than through dtStart or dtEnd.
Sequence	A Sequence is a set of Intervals with defined temporal relationships. Sequences may have gaps between Intervals, or even simultaneous activities. A Sequence is re-locatable, i.e., it does not have a specific date and time. A Sequence may consist of a single Interval. A Sequence may optionally include a Lineage. A Sequence can be scheduled multiple times through repeated reference by different Gluons. Intervals are defined through their Duration, and the schedule, dtEnd or dtStart, is applied to the Sequence as a whole.
Partition	A Partition is a set of consecutive Intervals. The Partition includes the trivial case of a single Interval. Partitions are used to define a single service or behavior that varies over time. Examples include energy prices over time and energy usage over time.
Gluon	A gluon is influences the serialization of Intervals in a Sequence, though inheritance and through schedule setting. The Gluon is similar to the Interval, but has no service or schedule effects until applied to an Interval or Sequence.
Artifact	An Artifact is the thing that occurs during an Interval. WS-Calendar ~~extends the **[XCAL]** attach object to contain this~~uses the Artifact as a placeholder. The contents of the Artifact are not specified in WS-Calendar~~,~~; rather the Artifact provides an extension base for the use of WS-Calendar in other specifications. Artifacts may inherit elements as do Intervals within a Sequence.

WS-Calendar works with groups of Intervals that have relationships between them. These relations constrain the final instantiation of a schedule-based service. Relations can control the ordering of

Intervals in a Sequence. They can describe when a service can be, or is prevented from, being invoked. They establish the parameters for how information will be shared between elements using Inheritance. The terminology for these relationships is defined in Table 1-4.

Table 1-4: Semantics: Relations, Limits, and Constraints

Term	Definition
Link	The Link is used by one WS-Calendar object to reference another. A link can reference either an internal object, within the same calendar, or an external object in a remote system.
Relationship	Relationships link between ~~components~~Components for Binding. ICalendar defines several relationships, but WS-Calendar uses only the CHILD relationship, and that only to bind Gluons to each other and to Intervals.
Temporal Relationship	Temporal Relationships extend the **[RFC5545]** Relationships to define how Intervals become a Sequence by creating an order between Intervals. The Predecessor Interval includes a Temporal Relation, which references the Successor Interval. When the start time and ~~duration~~Duration of one Interval is known, the start time of the others can be computed through applying Temporal Relations.
Availability	Availability expresses the range of times in which an Interval or Sequence can be Scheduled. Availability ~~is~~ often overlays or is overlaid by Busy. Availability can be Inherited.
Busy	Busy expresses the range of times in which an Interval or Sequence cannot be Scheduled. Busy ~~is~~ often ~~used to overlay or be~~ overlays is overlaid by Availability. Busy can be Inherited.
Child, Children	The CHILD relationship type (rel_type) defines a logical link (via URI or UID) from parent object to a child object. A Child object is the target of one or more CHILD relationships and may have ~~zero~~one to many Parent objects.
Parent [Gluon]	A Gluon (in a Sequence) that includes a CHILD relationship parameter type (rel_type) defines a logical link (via URI or UID) from parent object to a child object. A Parent Component contains one or more CHILD Relationships

WS-Calendar describes how to modify and complete the specification of Sequences. WS-Calendar calls this process Inheritance and specifies a number of rules that govern inheritance. Table 1-5 defines the terms used to describe inheritance.

Table 1-5: Semantics: Inheritance

Term	Definition
Lineage	The ordered set of Parents that results in a given inheritance or execution context for a Sequence.
Inheritance	Parents bequeath information to Children that inherit them. If a child does not already possess that information, then it accepts the inheritance. WS-Calendar specifies rules whereby information specified in one informational object is considered present in another that is itself lacking expression of that information. This information is termed the Inheritance of that object.
Bequeath	A Parent Bequeaths attributes (Inheritance) to its Children.
Inherit	A Child Inherits attributes (Inheritance) from its Parent.

Term	Definition
Covarying Attributes	Some attributes are inherited as a group. If any member of that group is expressed in a Child, all members of that group are deemed expressed in that Child, albeit some may be default values. These characteristics are called covarying or covariant. A parent bequeaths covarying characteristics as a group and a child accepts or refuses them as a group.
Decouplable Attributes	Antonym for Covarying Attributes. Decouplable Attributes can be inherited separately.

As Intervals are processed, as Intervals are assembled, and as inheritance is processed, the information conveyed about each element changes. When WS-Calendar is used to describe a business process or service, it may pass through several stages in which the information is not yet complete or actionable, but is still a conforming expression of time and Sequence. Table 1-6 defines the terms used when discussing the processing or processability of Intervals and Sequences.

During the life-cycle of communications concerning Intervals, different information may be available or required. For service performance, Start Duration and the Attachment Payload must be complete. These may not be available or required during service advertisement or other pre-execution processes. Table 1-6 defines the language used to discuss how the information in an Interval is completed.

Table 1-6: Semantics: Describing Intervals

Term	Definition
~~Anchored~~Designated Interval	~~An Interval is Anchored [in time] if it is Bound to a full date and time. A Sequence or Partition is Anchored if it contains an Anchored Interval, and when Fully Bound, the specific date, time, and duration of all Intervals can be determined unambiguously. Specific performance of a Service Contract always occurs in an Anchored Sequence.,~~An Interval that is referenced by a Gluon is the Designated Interval for a Series. An Interval can be Designated and still not Anchored.
~~Partially~~ Anchored	An Interval is ~~Partially~~ Anchored ~~if EITHER its Date OR its Time is Bound.~~when it includes a Start or End, either directly or through Binding. A Sequence ~~or Partition~~ is ~~Partially~~ Anchored ~~if~~when its Designated Interval is ~~Partially~~ Anchored.
Unanchored	~~An Interval is Unanchored if NEITHER its Begin Date nor its Begin Time are known.~~An Interval is Unanchored when it includes neither a Start or an End, either internally, or through Binding. A Sequence is Unanchored if its Designated Interval Unanchored. *Note: a Sequence that is re-used may be Unanchored in one context even while it is Anchored in another.*
Binding	Binding is the application of information to an Interval or Gluon, information derived through Inheritance or through Temporal Assignment.
Bound Element	A Bound Element refers to an Element and its Value after Binding, e.g., a Bound Duration.
Bound Interval	~~As in mathematical logic where a metasyntactic variable is called "bound", an Interval, Sequence, or Partition is said to be Bound when the values necessary to execute it (as a service) are completely filled in.~~A Bound Interval refers to an Interval and the values of its Elements after Binding.

Term	Definition
~~Partially~~ Bound Sequence	~~A Partially Bound Interval is one that is still not Bound after receiving its Inheritance. A Sequences or Partitions is Partially Bound if it contains at least one Interval that is Partially Bound.~~A Bound Sequence refers to a Sequence and the values of its Intervals after Binding.
Partially Bound~~Unbound~~	~~An Unbound~~Partially Bound refers to an Interval or a Sequence which is not ~~itself~~yet complete~~, but must still receive inheritance to~~ following Binding, i.e., the processes cannot yet be ~~fully specified. A Sequences or Partitions is Unbound if it contains at least one Interval that is Unbound.~~executed.
Fully Bound	Fully Bound refers to an Interval or Sequence that is complete after Binding, i.e., the process can be unambiguously executed when Anchored.
Unbound~~Fully Bound~~	~~A synonym for Bound~~An Unbound Interval or Sequence is not itself complete, but must still receive inheritance to be fully specified. A Sequence or Partition is Unbound if it contains at least one Interval that is Unbound.
Constrained	An Interval is Constrained if it is not Anchored and it is bound to one or more Availability or Free/Busy elements
Temporal Assignment	Temporal Assignment determines the start times of Intervals in a Sequence through processing of their Durations and Temporal Relations.
Scheduled	A Sequence or Partition is said to be Scheduled when it is Anchored, Fully Bound, and service performance has been requested.
Unscheduled	An Interval is Unscheduled if ~~its neither its begin date and time~~it is not Anchored, nor ~~its end date and time have been set.~~is any Interval in its Sequence Anchored. A Sequence or Partition is Unscheduled if none of its Intervals, ~~after~~ when Fully Bound, is Scheduled.
~~Designated Interval~~	~~In a Sequence the Designated Interval is either (a) (if there are no Gluons related to the Sequence) one of the Earliest Interval(s), or (b) (if there is at least one Gluon related to the Sequence) the single Interval referenced by a Gluon as CHILD.~~
Predecessor Interval	A Predecessor Interval includes a Temporal Relation which references a Successor Interval.
Successor Interval	A Successor Interval is one referred to by a Temporal Relationship in a Predecessor Interval.
Antecedent Interval(s)	~~An~~Antecedents are an Interval or set of Intervals that precede a given Interval within the same Sequence
Earliest Interval	The set of Intervals at the earliest time in a given Sequence
Composed Interval	A Composed Interval is the virtual Interval specified by applying inheritance through the entire lineage and into the Sequence in accord with the inheritance rules. A Composed Interval may be Bound, Partially Bound, or Unbound.
Composed Sequence	A Composed Sequence is the virtual Sequence specified by applying inheritance through the entire lineage and into the Sequence in accord with the inheritance rules. A Composed Sequence may be Bound, Partially Bound, or Unbound.

Term	Definition
Comparable Sequences	Two Sequences are Comparable if and only if ~~there exists a~~ the Composed version of each ~~that~~ defines the same schedule.

2 Overview of WS-Calendar

A calendar communication without a real world effect[6] is of little interest. That real world effect is the result of a service execution context within a policy context. Practitioners can use WS-Calendar to add communication of schedule and Interval to the execution context of a service. Use of WS-Calendar will align the performance expectations between execution contexts in different domains. The Technical Committee intends for other specifications and standards to normatively reference and claim conformance to WS-Calendar, bringing a common scheduling context to diverse interactions in different domains

2.1 Approach taken by the WS-Calendar Technical Committee

The Technical Committee (TC) based its work upon the iCalendar specification as updated in 2009 (IETF **[RFC5545]** and its the XML serialization **[XCAL]**, currently (~~2010-07~~2011-05) on a standards track in the IETF. Members of the Calendaring and Scheduling Consortium (CalConnect.org) developed both updates to IETF specifications and provided advice to this TC. ~~This work~~**[RFC5545]** provides the normative vocabulary for use in this specification.

This committee developed the normative schema (XSD) for iCalendar. This schema, including the schema extensions necessary for the services defined herein, is part of the WS-Calendar specification.

The committee solicited requirements from a range of interests, notably the NIST Smart Grid Roadmap **[NIST Framework]** and the requirements of the Smart Grid Interoperability Panel (SGIP) as developed by the North American Energy Standards Board (NAESB~~).~~) **[NAESB Requirements]**. Others submitting requirements included members of the oBIX technical committee and ~~representative~~representatives of the FIX Protocol Association. These requirements are reflected in the semantic elements described in Chapters 3 and 4.

In a parallel effort, the CalConnect TC-XML committee developed a number of schedule and calendar-related services. CalConnect drew on its experience in interoperability between enterprise calendaring systems as well as interactions with web-based calendars and personal digital assistants (PDAs). These services were developed as RESTfull (using **[REST]**) services by CalConnect and contributed to the WS-Calendar TC. CalConnect also developed and contributed **[SOAP]** and **[WSDL]** definitions to this TC.

2.2 ~~Scheduling~~Communicating Schedules and Service Performance

Time semantics are critical to ~~WS-Calendar.~~process interactions. Services requested differently can have different effects on performance even though they appear to request the same time interval. This is inherent in the concept of a service-oriented architecture.

[6] ~~This paragraph includes a number of terms of art used in service oriented architecture (SOA). In all cases, the terms are as defined in the *Reference Model for Service Oriented Architecture*, found in the normative references.~~

As defined in the OASIS Reference Model for Service Oriented Architecture 1.0~~[7],~~ **[SOA-RM]**, service requests access the capability of a remote system.

> *The purpose of using a capability is to realize one or more real world effects. At its core, an interaction is "an act" as opposed to "an object" and the result of an interaction is an effect (or a set/series of effects). This effect may be the return of information or the change in the state of entities (known or unknown) that are involved in the interaction.*
>
> *We are careful to distinguish between public actions and private actions; private actions are inherently unknowable by other parties. On the other hand, public actions result in changes to the state that is shared between at least those involved in the current execution context and possibly shared by others. Real world effects are, then, couched in terms of changes to this shared state*

A request for remote service performance is a request for specific real world effects. For ~~WS-Calendar~~process interaction, these effects are expected to occur during a given period. Consider two service providers that offer the same service. One must start planning an hour or more in advance. The second may be able to achieve the service in five minutes. The service start time is the time when that service becomes fully available; that is the time specified in service interactions. Because this service start time and service period are all that matters, the same service can be offered by different providers using quite different technologies.

~~The complement of this is the scheduled end time. The party offering the service may need to ramp down long running processes. Using for example energy demand response, if a system contracts to end energy use by 3:00, it assumes the onus of turning everything off before 3:00.~~

~~Duration is how long a behavior is continued. If a service contracts to provide shed load for an hour, it is not necessary for it to stop shedding load 65 minutes later (which may be the end of the work day). It must, however, shed the agreed upon load during all of the 60 minutes.~~

~~In this way, the service scheduled to shed load from 4:00 ending at 5:00 may be quite different than the one scheduled to shed load for an hour beginning at 4:00.~~

2.2.1 Which Time? UTC vs. Local Time

Coordinated Universal Time (abbreviated UTC) is a time standard based on International Atomic Time (TAI) with leap seconds added at irregular intervals to compensate for the Earth's slowing rotation.Time zones around the world can be expressed as positive or negative offsets from UTC.

When 2 or more parties attempt to agree on a time, e.g., for a meeting, or when to provide a service, they agree to start at a particular instant of time UTC. They agree on that instant in time by converting from local time, e.g., they want a meeting to start at 13:00 Eastern, 18:00 UK. Our lives and the use of services are bound by local time not by UTC. ~~To humans~~Experientially, local time is the invariant and UTC is mapped on to it. If a government modifies the rules we adjust the mappings and we shift the UTC time. We still want to meet at 13:00 local or have the heating start at 07:00.

As long as the rules never change this causes no confusion—but they do. Recent experience has included considerable efforts when the rules for the start of Daylight ~~Savings~~Saving Time (DST) have changed. If all information is in UTC, and no record of the ~~events~~event's basis in the local time and time

~~[7] See normative references in section 1.2~~

zone remains, there is no way to re-compute existing contracts. ~~We don't~~It is often necessary to know if ~~that~~ UTC was calculated based on an old or a new rule.

A triplet of Local time + timezoneid + (UTC or offset) always allows ~~you to determine~~the determination if ~~the~~a time is valid. If a recalculation of UTC for that local time + tzid results in a different value from that stored then presumably the DST rules have changed since the data was stored. If ~~you~~one can detect that the scheduled time is no longer valid~~ you~~, one can take corrective action.

~~For simplicity, all examples and discussion in this document are based on Greenwich Mean Time also known as Coordinated Universal Time (UTC).~~ The Technical Committee makes no representation as whether UTC or local time are more appropriate for a given interaction. Because WS-Calendar is based on **[iCalendar]**, business practices built upon WS-Calendar can support either. Specifications that claim conformance with this specification may require choices to support their particular business processes.

~~Practitioners should~~For a fuller discussion of time zones, consult **[Time Service Recommendations]** and **[Time Zone Service]** in the non-normative references.

2.3 Overview of This Document

The specification consists of a standard schema and semantics for schedule and interval information. Often the most important service schedule communications involve series of related services over time, which WS-Calendar defines as a Sequence. These semantic elements are defined and discussed in Section 3. While this specification describes only the use of core semantic elements from iCalendar, no part of this document prevents other semantic elements from iCalendar from also being used.

Section 3.2 introduces notions of tolerance, i.e. what does it mean to be “on time”. This section also describes the different ways to associate a service request with each Interval in a Sequence.

Managing information exchanges about a Sequence of events can easily become cumbersome, or prone to error. WS-Calendar defines the~~ Calendar~~ Gluon, a mechanism for making assertions about all or most of the Intervals in a Sequence. Intervals can inherit from a ~~Calendar~~ Gluon, or they can override locally assertions inherited from the ~~Calendar~~ Gluon. Section 3.3 discusses inheritance and parsimony of communication and introduces contract scheduling.

~~In Sections 4-4.9, this document describes **[REST]**-based, (RESTfull) web services for interacting with remote calendars. These interactions are derived from the well-known interactions defined in **[CalDAV]**, although they do not specify any interaction with **[CalDAV]** servers. This specification defines services for calendar inquiries, event scheduling, event updating, and event cancelation.~~

~~In Sections n-n, this document describes **[SOAP]**-based interactions for Calendar services. As with REST, the specification defines services for calendar inquiries, event scheduling, event updating, and event cancelation using the iCalendar schema.~~

~~With incompatible communications defined (REST, SOAP), the specification is not prescriptive of the communications used.~~ The practitioner must decide whether to use ~~either~~one or the other of these communication protocols, or whether WS-Calendar artifacts are better used when embedded within other messages. These decisions~~, along with decisions about the specific security needed by the communication~~ must be based upon the specific application and message content. Specifications that claim conformance to this specification may wish to provide guidance appropriate for the business purposes of that specification.

2.4 Security Considerations

Part 1 describes an information model. The information models can be expressed in any interaction, using any protocol. There are no security aspects of the information model.

Specifications which claim conformance with WS-Calendar may wish to specify security approaches or techniques. Security choices must be based on the business requirements and operational risks of the interaction that those specifications define. As this specification defines a general information model, for use in many interactions, it specifies no security approach.

3 PART ONE: ~~Semantic Model~~Information model for WS-Calendar

3.1 Intervals, Temporal Relations, and Sequences

WS-Calendar Elements are semantic elements derived from the **[XCAL]** specification. ~~These~~ This set of elements ~~are~~is smaller than ~~a~~those needed for full schedule interaction, and describe the ~~intervals, durations~~Intervals, Durations, and time-related events that are relevant to service interactions. ~~The~~WS-Calendar uses the elements ~~are used~~ to build a precise vocabulary of time, ~~duration~~Duration, Sequence, and ~~schedule~~Schedule.

WS-Calendar elements adapt the iCalendar objects to make interaction requirements explicit. For example, in human schedule interactions, different organizations have their own expectations. Meetings may start on the hour or within 5 minutes of the hour. As agents scheduled in those organizations, people learn the expected precision. ~~WS-Calendar elements adapt the iCalendar objects to make interaction requirements explicit. For example, in~~ This~~human schedule interactions, different organizations have their own expectations. Meetings may start on the hour or within 5 minutes of the hour. As agents scheduled in those organizations, people learn the expected precision. In WS-Calendar, that~~ precision expectation must be explicit to prevent interoperation problems. ~~WS-Calendar~~This specification defines a performance element to elaborate the simple specification of **[XCAL]** to make explicit the performance expectations within a scheduled event.

~~WS-Calendar~~This specification defines common semantics for recording and exchanging event information~~.~~ (Time Stamps).

3.1.1 Core Semantics derived from [XCAL]

~~3.1.1~~1.1.1 The iCalendar data format [RFC5545] is a widely deployed interchange format for calendaring and ~~Core Semantics derived from [XCAL]~~

~~The iCalendar data format **[RFC5545]** is a widely deployed interchange format for calendaring and scheduling~~schedule data. The **[XCAL]** specification ~~(in process)~~ standardizes the XML representation of iCalendar information. WS-Calendar relies on **[XCAL]** standards and data representation to develop its semantic ~~components~~Components.

~~3.1.1.1 Time~~

3.1.1.1 Time ~~is an ISO 8601 compliant time~~

[ISO8601] defines string ~~with~~formats for the ~~optional accompaniment~~expression of ~~a~~ date, time, and duration ~~interval~~. **[ISO8601]** also defines string formats to ~~define times~~express the passage of ~~less than 1 second.~~time, herein a Duration. This specification relies extensively on **[ISO8601]**. Examples of date and time representations ~~the from the ISO 8601 standard~~ include:

```
Year:
   YYYY (eg 1997)
Year and month:
   YYYY-MM (eg 1997-07)
Complete date:
   YYYY-MM-DD (eg 1997-07-16)
Complete date plus hours and minutes:
   YYYY-MM-DDThh:mmTZD (eg 1997-07-16T19:20+01:00)
Complete date plus hours, minutes and seconds:
   YYYY-MM-DDThh:mm:ssTZD (eg 1997-07-16T19:20:30+01:00)
Complete date plus hours, minutes, seconds and a decimal fraction of a second
   YYYY-MM-DDThh:mm:ss.sTZD (eg 1997-07-16T19:20:30.45+01:00)
```

~~Normative information on **[ISO 8601]** is found in section 1.2.~~

This specification is general purpose. Standards that claim conformance to this specification may need to restrict the variability above to improve interoperation within their own interactions.

3.1.1.2 The iCalendar Components (VComponents)

iCalendar and **[XCAL]** have a number of long defined ~~component~~Component objects that comprise the payload inside of an iCalendar message. These include the VTODO, the VALARM, the VEVENT. (The “v” that begins each element name is there for historic ~~purposes~~reasons.) The definitions and use of each of the ~~vObjects~~vComponents can be found in **[RFC5545]**.

The ~~vObjects~~vComponents share the same parameters and properties. The distinctions between these informational ~~objects is in which~~types are ~~permitted,~~ones of purpose and ~~which are required. Because of its flexibility,~~ conformance. The Interval and Gluon are new vComponents; each is derived from the ~~VTODO object is~~ same base type as the ~~basis~~other vComponents.

This specification in no way deprecates the pre-existing vComponents. The new components are introduced to support stored sequences of operations and remote invocation. The existing vComponents are extended to support informational payloads for ~~WS-Calendar objects for service performance. Because WS-Calendar services~~process interaction. A conforming specification can use both old and new vComponents where each makes sense.

The RESTful and SOAP in Parts Two and Three of a future version of this specification support all traditional vComponents as well as the new ones defined here. Conforming information elements MAY be processed using traditional iCalendar-based interactions (CalDAV, et al~~.), all VComponents SHALL be supported.~~ .) and managed in traditional iCalendar stores.

~~The Interval and Gluon are new vObjects, and each is derived from vtodo.~~

3.1.1.3 Duration and the ~~granularity~~Granularity of Time

This specification uses Duration as defined in **[ISO 8601]** as a data-type throughout. iCalendar makes a number of assumptions about the meaning of time when expressed as duration, ~~based on guidance in **[ISO 8601]**. These become~~i.e., a duration is over when the same common metric is reached in the next such unit. For example, a duration of one day starting at 6:00 AM lasts until 6:00 AM the next day. This becomes important during ~~times~~periods when the meaning of a duration changes. The passage of a month that begins on January 5 is complete on February 5. Another month comes to March 5. Each is expressed using the format ~~(1M).~~*“P1M”*. These durations are, respectively, 31, 28 or 29, and 31 days. In a similar way, Years ~~(1Y)~~*“P1Y”* may be 365 or 366 days long, days ~~(1D)~~*“P1D”* may be 23, 24, or 25 hours long.~~ A duration is over, when the same common metric is reached in the next such unit~~

~~The meaning of a communication is based upon the granularity of the communication.~~ If the intention is to express 30 days, then one should use ~~(30D)~~*“P30D”* and not ~~(~~*“P1M*~~).~~*”*. Similarly, if the intent is to express from now until the same time tomorrow, use ~~(1D)~~*“P1D”* rather than 24 hours ~~(24H).~~*“PT24H”*.

3.1.2 Intervals

~~Time Segments, i.e., increments of~~ Clear communication of the continuous passage of time~~, are a~~ is critical ~~component of~~ to defining service ~~alignment using WS-Calendar. There are many overloaded uses of terms about time, and within a particular time segment, there may be many of them.~~ coordination.

The building block for ~~the WS-Calendar~~this information model is the Interval. The Interval is a time segment whose length is specified by ~~the~~ a Duration. ~~The Duration is represented by a string as defined in the iCalendar specification **[RFC5545]**. The Committee listened to arguments that we should redefine the use and meaning of Duration. Whatever their merit, the iCalendar Duration has a pre-existing meaning of the length of time of scheduled within an event.~~

~~An Interval is a~~ The Interval is a unit of time, and can be bound to service delivery~~, and can be bound to time.~~. An Unscheduled Interval ~~is not linked to a specific~~has no specified date and time. A Scheduled Interval has a ~~known~~specified start date and time. Intervals can legally contain all elements ~~of the VTODO~~properties as defined in **[RFC5545]**. For convenience, the elements essential to coordinating service operations using Intervals are listed in Table 3-1.

An Interval is part of a Sequence. An entire Sequence can be scheduled by scheduling a single Interval in a Sequence. A single Sequence can be scheduled multiple times through repeated reference by different Gluons. It may be useful to consider the Unanchored Sequence as a process subroutine and that a Gluon can be used to invoke that subroutine. For this reason, of the three primary temporal elements (dtStart, dtEnd, and Duration) in a Component, the Duration has primacy in Intervals. Within a Sequence, a maximum of a single Interval MAY have a dtStart or a dtEnd.

Nothing in this section supersedes **[RFC5545].** Implementers SHALL refer to those respective specifications **[RFC5545]** and the **[XCAL]** specifications for the normative description of each element~~.~~ with the exception of Duration, which is as defined as in **[ISO8601]**.

Table 3-1: ~~Properties~~Elements of Intervals

Elements		**Description~~Use~~** ~~**Use in WS-Calendar**~~
Dtstamp	~~xcal:dtstamp~~ ~~Mandatory~~	Identifies when Interval object was created
Uid	~~Mandatory~~	Used to enable unambiguous referencing by other ~~components~~Components
Duration	~~xcal:Duration~~ ~~Optional~~	Identifies length of time for Interval~~.~~. Duration must be known before an Interval can be transacted, but the Duration may only come through Binding.
DtStart	~~Xcal:dateTime~~ ~~Optional~~	Scheduled start date and time for Interval. The Start must be known before an Interval can be transacted, but the Duration may only come through Binding.
Attach~~attach~~	~~Mandatory~~	In **[XCAL]**, any attachment. In WS-Calendar, the Attach contains the informational payload used by ~~incorporating~~conforming specifications. ~~Defined in~~See section 3.2.
~~Relations~~Relation	~~As defined in~~ ~~[RFC5998]~~ ~~Optional~~	Relations contain the temporal relations between Intervals that create Sequences. Section 0. describes Temporal Relations and their use.

An Interval specifies how long an activity lasts. ~~An Unanchored Interval is not linked to a specific date and time.~~ The example below (Example 3-1) shows ~~the components section~~a fragment of a WS-Calendar-based message containing a single Unanchored Interval, i.e., it contains neither a dtStart nor a dtEnd. Note that there is no Relationship; there is no need for Relationships until an Interval is incorporated into a Sequence.

Example 3-1: An Unanchored Interval

```
<xcal:interval~~ xmlns:xcal="urn:ietf:params:xml:ns:icalendar-2.0">~~>
   <xcal:properties>
          <xcal:uid>
                 <xcal:text>~~6fa8b9c5-c9b1-4ba1-bf9c-
5e5da03cb943@examples.oasis-open.org~~bfe8069a-0c94-4b53-852f-
6a2f12639c7e</xcal:text>
          </xcal:uid>
          <xcal:duration>
                 <xcal:duration>~~T10H~~PT10H</xcal:duration>
```

```
        </xcal:duration>
   </xcal:properties>
   ~~<xcal:components/>~~
</xcal:interval>
```

~~Note that no start time is specified, and no relationship. Relationships are not needed until an Interval is incorporated into a Sequence.~~

3.1.3 Connecting the Intervals

Many iCalendar communications involve more than one Interval. Classic iCalendar **[RFC5545]** defines relationships internally. ~~WS-Calendar instead~~**[xCAL]** uses the extensible expression pattern of Web ~~Link **[RFC5998]**, both for~~Links (as described in **[RFC5588]**) to express the ~~traditional Relationships (parent, child, sibling)~~iCalendar relationships PARENT, CHILD, and ~~for the~~SIBLING. This specification extends these relationships by adding Temporal ~~Relationships. Relationships include~~Relations. Temporal Relations consist of a reference, a relation, and ~~optional Tolerance parameters~~a Gap that specifies any Duration between Predecessor and Successor.

~~Temporal Relationships, new in WS-Calendar, use Web Linking **[RFC5998]** in an Interval (the Predecessor) to reference another Interval (the Successor). Temporal Relationships optionally include a Gap that specifies any lag between Predecessor and Successor.~~

Unlike most semantic elements in this specification, Temporal Relations are defined in this specification, rather than defined elsewhere and used herein.

Table 3-2: Temporal Relationships

Temporal Relationship	Short Form	Definition	Example
~~finishToStart~~Gap	~~FS~~	~~As soon as~~ Duration indicating the time between the predecessor ~~Interval finishes,~~and the successor ~~Interval starts.~~ . Optional, where missing, Gap is treated as a zero duration	~~When sanding is complete, painting begins.~~Gap may be positive or negative. In the examples below, the Gap, when present, is 20 minutes.
~~finishToFinish~~ Finish To Start	~~FF~~FS	~~The successor Interval continues as long~~As soon as the predecessor Interval finishes, the successor Interval starts.	~~The concession stand stops serving 20 minutes after the end of the game.~~When sanding is complete, painting begins.
~~startToFinish~~Finish To Finish	~~SF~~FF	The ~~start of~~successor Interval continues as long as the predecessor ~~controls the finish of the successor.~~Interval.	The ~~start of Attendee Check-in controls~~concession stand stops serving 20 minutes after the end of the ~~Interval "Set up registration booth."~~game.
~~startToStart~~Start To Finish	~~SS~~SF	The ~~Predecessor Interval triggers the~~ start of the ~~second task. The Gap indicates~~predecessor controls the finish of the ~~lag time~~successor.	The start of Attendee Check-in controls the end of the Interval "Set up registration booth."~~20 minutes after the caterer begins work, the dining lines are open.~~

~~Gap~~Start To Start	SS	~~Duration indicating the time between the predecessor and the successor. Optional, where missing, Gap is treated as a zero duration~~The Predecessor Interval triggers the start of the second task.	20 minutes after the caterer begins work, the dining lines are open.~~Gap may be positive or negative.~~

While simple relationships may be ordered based on which task occurs first (finishToStart), if a later Interval is controlling, other choices may make more useful. For example, if ramp-up time must be completed before run-time, and run-time start is indicated in a contract, it may be useful to specify that the Ramp Interval (Successor) must complete before (startToFinish) the Designated Interval's (Predecessor) scheduled start time. ~~Referencing specifications~~Specifications claiming conformance should consider statements of conformance around Temporal Relationships.

The relationship below indicates that this Interval is to start ten minutes following the finish of the Interval specified.

Example 3-2: Temporal Relationship

```
<xcal:related-to~~ xmlns:xcal="urn:ietf:params:xml:ns:icalendar-2.0">~~>
   <xcal:parameters>
          <xcal:reltype>
                 <xcal:text>FS</xcal:text>
          </xcal:reltype>
          <xcal:gap>
                 <xcal:duration~~ xs:type="xcal:DurationPropType">~~
                        ~~<xcal:parameters/>~~
                        ~~<xcal:duration>T10M</xcal:duration>~~
                 >PT10M</xcal:duration>
          </xcal:gap>
   </xcal:parameters>
   <xcal:uid>~~05782926-1d71-4a55-ae3b-cba5ebf419d3@examples.oasis-open.org~~07fb177d-54ea-44ea-8ef5-5b763dc9f0c6</xcal:uid>
</xcal:related-to>
```

If there is no temporal separation between Intervals, the gap element is optional. The following examples are equivalent expressions to express a relationship wherein both Intervals must start at the same moment.

Example 3-3: Temporal Relationship with Gap

```
~~<xcal:related-to xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xmlns:xcal="urn:ietf:params:xml:ns:icalendar-2.0">~~
<xcal:related-to>
   <xcal:parameters>
          <xcal:reltype>
                 <xcal:text>FS</xcal:text>
          </xcal:reltype>
          <xcal:gap>
                 <xcal:duration~~ xs:type="xcal:DurationPropType">~~
                        ~~<xcal:parameters/>~~
                        ~~<xcal:duration>T0M</xcal:duration>~~
                 >PT10M</xcal:duration>
          </xcal:gap>
   </xcal:parameters>
   <xcal:uid>~~5decdb30-7278-4e96-9f81-c20c81f283c3@examples.oasis-open.org~~07fb177d-54ea-44ea-8ef5-5b763dc9f0c6</xcal:uid>
</xcal:related-to>
```

Leaving out the optional Gap element, we have:

Example 3-4: Temporal Relationship without Gap

```
~~<xcal:related-to xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"~~
~~xmlns:xcal="urn:ietf:params:xml:ns:icalendar-2.0">~~
<xcal:related-to>
   <xcal:parameters>
          <xcal:reltype>
                 <xcal:text>FS</xcal:text>
          </xcal:reltype>
   </xcal:parameters>
   <xcal:uid>~~5decdb30-7278-4e96-9f81-c20c81f283c3@examples.oasis-~~
~~open.org~~07fb177d-54ea-44ea-8ef5-5b763dc9f0c6</xcal:uid>
</xcal:related-to>
```

The ~~two~~ expressions of a Temporal Relationship ~~above~~in Example 3-3and Example 3-4 are equivalent.

Intervals with Temporal Relationships enable the message to express complex temporal relations to form a Sequence~~, as well as express the simple~~. A Sequence consisting of identical consecutive Intervals is named a Partition

~~A Sequence describes a coherent set of Intervals that can be assembled from a collection of Intervals.~~.

As the rules for parsing XML do not mandate preservation of order within a sub-set, we cannot assume that order is preserved when parsing a set of ~~Components.~~Intervals. For Sequences in WS-Calendar, then, mere order is not enough—a Sequence is a collection of Intervals each of which Interval either refers to or is referred by at least one Interval. Using the references, expressed as Temporal Relations, WS-Calendar describes a single coherent Sequence ~~that is~~ assembled from a set of Intervals in a collection.

3.1.4 Sequences: Combining Intervals

A Sequence is a collection of Intervals with a coherent set of Temporal Relationships~~.~~ (Table 1-3). Temporal Relationships are transitive, so that if Interval A is related to Interval B, and Interval B is related to Interval C, then Interval A is related to Interval C. Sequences can also include Gluons *(see section ~~3.3.1, References and Inheritance.~~3.3.1, References and Inheritance.*, but for this section, we will discuss Sequences only as a set of Intervals.

~~Table~~Example 3-~~3~~5: Introducing the Sequence

```
~~<xcal:vcalendar xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"~~
~~xmlns:xcal="urn:ietf:params:xml:ns:icalendar-2.0" xs:type="xcal:VcalendarType">~~
<xcal:vcalendar>
   <xcal:components>
          <xcal:interval>
                 <xcal:properties>
                        <xcal:uid>
                               <xcal:text>~~6bf5b06f-0418-4fd7-b861-~~
~~d3a2b9b0292a@examples.oasis-open.org~~69343fc9-c1da-4cd0-abbd-889716a401d2</xcal:text>
                        </xcal:uid>
                        <xcal:duration>
                               ~~<xcal:parameters/>~~
                               <xcal:duration>~~T1H~~PT1H</xcal:duration>
                        </xcal:duration>
                        ~~</xcal:duration>~~
                 ~~</xcal:properties>~~
          ~~</xcal:interval>~~
          ~~<xcal:interval>~~
                 ~~<xcal:properties>~~
                        ~~<xcal:uid>~~
                 </xcal:properties>
          </xcal:interval>
          <xcal:interval>
                 <xcal:properties>
                        <xcal:uid>
                               <xcal:text>~~a40a85bb-3052-4e52-ad20-~~
~~4d19cb76d9e7@examples.oasis-open.org~~0ba5a8c0-4eb2-49db-8514-5da18f53caaa</xcal:text>
                        </xcal:uid>
                        ~~<xcal:duration>~~
                               ~~<xcal:duration>T2H</xcal:duration>~~
```

```
                           ~~</xcal:duration>~~
                           <xcal:related-to>
                                   <xcal:parameters>
                                           <xcal:reltype>
                                                   <xcal:text>FS</xcal:text>
                                           </xcal:reltype>
                                   </xcal:parameters>
                                   <xcal:uid>~~6bf5b06f-0418-4fd7-b861-
d3a2b9b0292a@examples.oasis-open.org~~69343fc9-c1da-4cd0-abbd-889716a401d2</xcal:uid>
                           ~~</xcal:related-to>~~
                   ~~</~~      </xcal:related-to>
                           <xcal:~~properties~~duration>
           ~~</~~                      <xcal:~~interval>~~
           <duration>PT2H</xcal:~~interval~~duration>
                           </xcal:duration>
                   </xcal:properties>
           </xcal:interval>
           <xcal:interval>
                   <xcal:properties>
                           <xcal:uid>
                   ~~<xcal:properties>~~
                           ~~<xcal:uid>~~
                                   <xcal:text>~~9aa279d1-78d0-4ef7-a737-
9519bec09007@examples.oasis-open.org~~8edc5ef8-c269-4897-be10-c8a3eb6c8fb6</xcal:text>
                           </xcal:uid>
                           ~~<xcal:duration>~~
                                   ~~<xcal:duration>T3H</xcal:duration>~~
                           ~~</xcal:duration>~~
                           <xcal:related-to>
                                   <xcal:parameters>
                                           <xcal:reltype>
                                                   <xcal:text>FS</xcal:text>
                                           </xcal:reltype>
                                           <xcal:gap>
                                                   <xcal:duration
~~xs:type="xcal:DurationPropType">~~

   ~~<xcal:duration>T10M</xcal:duration>~~
                                                   >PT10M</xcal:duration>
                                           </xcal:gap>
                                   </xcal:parameters>
                                   <xcal:uid>~~a40a85bb-3052-4e52-ad20-
4d19cb76d9e7@examples.oasis-open.org~~0ba5a8c0-4eb2-49db-8514-5da18f53caaa</xcal:uid>
                           </xcal:related-to>
                           ~~</xcal:related-to>~~
                           <xcal:duration>
                                   <xcal:duration>PT3H</xcal:duration>
                           </xcal:duration>
                   </xcal:properties>
           </xcal:interval>
   </xcal:components>
</xcal:vcalendar>
```

In ~~this~~ the example above, the Intervals are ~~one hour,~~ 1 hour, 2 hours, and ~~three~~3 hours long. There is a ten minute period between the second and third periods.

3.1.4.1 Anchoring a Sequence

A Sequence becomes an Anchored Sequence whenever ~~a single~~the Designated Interval within the Sequence ~~is~~becomes Anchored. An Interval is Anchored when it has a specific starting date and time (dtstart). A Sequence may become Anchored when a Designated Interval becomes Anchored through Binding. A Gluon may reference a Designated Interval through an external reference, i.e., through referring to a resolvable Uid. A given Sequence may remain Unanchored while being incorporated into many Anchored Sequences through multiple Gluon references each creating a different Bound dtStart.

Example 3-6: An Anchored Sequence

```
~~<xcal:vcalendar xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"~~
~~xmlns:xcal="urn:ietf:params:xml:ns:icalendar-2.0"~~
~~xs:type="xcal:VcalendarType">~~
<xcal:vcalendar>
  <xcal:components>
    <xcal:interval>
      <xcal:properties>
        <xcal:uid>
          <xcal:text>~~12b59df6-cac2-41e7-a4c8-~~
~~41a9d347e54c@examples.oasis-open.org~~86879372-6d90-4de3-8267-
eae50c774f82</xcal:text>
        </xcal:uid>
        <xcal:duration>
          <xcal:duration>~~T15M~~PT15M</xcal:duration>
        </xcal:duration>
      </xcal:properties>
      ~~<xcal:components/>~~
    </xcal:interval>
    <xcal:interval>
      <xcal:properties>
        <xcal:uid>
          <xcal:text>~~5dcc9e77-8afa-4371-9437-~~
~~11d673f7f901@examples.oasis-open.org~~c212aa90-6fc4-41e1-b8fb-
a23308610247</xcal:text>
        </xcal:uid>
        ~~<xcal:duration>~~
          ~~<xcal:duration>T2H</xcal:duration>~~
        ~~</xcal:duration>~~
        <xcal:related-to>
          <xcal:parameters>
            <xcal:reltype>
              <xcal:text>FS</xcal:text>
            </xcal:reltype>
          </xcal:parameters>
          <xcal:uid>~~12b59df6-cac2-41e7-a4c8-~~
~~41a9d347e54c@examples.oasis-open.org~~86879372-6d90-4de3-8267-
eae50c774f82</xcal:uid>
        </xcal:related-to>
        ~~</xcal:related-to>~~
        <xcal:duration>
          <xcal:duration>PT2H</xcal:duration>
        </xcal:duration>
        <xcal:dtstart>
          <xcal:parameters>
            <xcal:tzid>

  <xcal:text>America/New_York</xcal:text>
            </xcal:tzid>
          </xcal:parameters>
          <xcal:date-time>~~20110315T090000~~2011-05-
28T09:00:00</xcal:date-time>
        </xcal:dtstart>
      </xcal:properties>
      ~~<xcal:components/>~~
    </xcal:interval>
    <xcal:interval>
      <xcal:properties>
        <xcal:uid>
          <xcal:text>~~ec72e7df-c837-4cba-afbb-~~
~~aa54b9043158@examples.oasis-open.org~~50970789-1a4b-4fe5-a7e3-
2da2c6db43ef</xcal:text>
        </xcal:uid>
        ~~<xcal:duration>~~
```

```
                             ~~<xcal:duration>T30M</xcal:duration>~~
                       ~~</xcal:duration>~~
                       <xcal:related-to>
                             <xcal:parameters>
                                    <xcal:reltype>
                                           <xcal:text>FS</xcal:text>
                                    </xcal:reltype>
                                    <xcal:gap>
                                           <xcal:duration
~~xs:type="xcal:DurationPropType">~~
                >PT10M                             ~~<xcal:duration>T10M</xcal:duration>~~
                                           </xcal:duration>
                                    </xcal:gap>
                             </xcal:parameters>
                             <xcal:uid>~~5dce9e77-8afa-4371-9437-~~
~~11d673f7f901@examples.oasis-open.org~~c212aa90-6fc4-41e1-b8fb-
a23308610247</xcal:uid>
                       </xcal:related-to>
                       ~~</~~<xcal:~~related-to~~duration>
                             <xcal:duration>PT30M</xcal:duration>
                       </xcal:duration>
                </xcal:properties>
         </xcal:interval>
   </xcal:components>
</xcal:vcalendar>
```

~~Note that the entire Sequence is Anchored when a single Interval within the Sequence is Anchored.~~

3.1.5 State Changes

A common service interaction is to request that, at a certain time, a discrete state change will occur. It could be that the price will rise. It could be that a report will be run. Such a communication has no ~~logical Duration. WS-Calendar communicates state changes through use of an Interval with the Duration explicitly set to zero time. Because the Duration is explicit, it will not be over-ridden through inheritance.~~inherent Duration.

~~Specifications that normatively reference and claim conformance with WS-Calendar SHALL define the business meaning of zero duration Intervals.~~

While this specification extends iCalendar through the use of Intervals in Sequences, the pre-existing elements of iCalendar remain in place, and more are defined periodically. State Changes can be handled in one of two ways today. As iCalendar is continually extended, other ways may become available tomorrow. Specifications that claim conformance to WS-Calendar SHALL state how they will communicate state changes.

[RFC5545] specified the use of a VEVENT with a start date and time, but no end date and time and no duration. WS-Calendar introduces the communication of state changes through use of an Interval with the Duration explicitly set to zero time *"P"*. Because the Duration is explicit, it is not be over-ridden through inheritance.

Example 3-7 State Change communication using Zero Duration Interval

```
~~<xcal:interval xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"~~
~~xmlns:xcal="urn:ietf:params:xml:ns:icalendar-2.0">~~
<xcal:interval>
   <xcal:properties>
         <xcal:uid>
                <xcal:text>~~f1bac9f0-cdd4-4f78-9a83-~~
~~e8f2446fe205@examples.oasis-open.org~~04998c75-86fb-429b-8206-
0a95559feb96</xcal:text>
         </xcal:uid>
         <xcal:duration>
```

```
                <xcal:duration>~~T0~~PT0M</xcal:duration>
         </xcal:duration>
         <xcal:dtstart>
                <xcal:parameters>
                       <xcal:tzid>
                              <xcal:text>America/New_York</xcal:text>
                       </xcal:tzid>
                </xcal:parameters>
                <xcal:date-time>~~20110315T161500~~2011-05-28T16:15:00</xcal:date-time>
         </xcal:dtstart>
   </xcal:properties>
</xcal:interval>
```

Example 3-8 State Change communication using Event without Duration or End

```
<xcal:vevent>
   <xcal:properties>
         <xcal:uid>
                <xcal:text>abfccdb0-41b5-46c5-a71d-226cea632034</xcal:text>
         </xcal:uid>
         <xcal:dtstart>
                <xcal:parameters>class
                       <xcal:tzid>
                              <xcal:text>America/New_York</xcal:text>
                       </xcal:tzid>
                </xcal:parameters>
                <xcal:date-time>2011-05-28T16:15:00</xcal:date-time>
         </xcal:dtstart>
   </xcal:properties>
   <xcal:components/>
~~</xcal:interval>~~
</xcal:vevent>
```

3.2 Attachments and ~~Timely Performance~~Tolerance

While iCalendar expresses time and intervals, WS-Calendar associates those intervals with specific services and service performance characteristics. In iCalendar ~~components~~Components, the ~~ATTACH component~~Attachment is used to include information outside the scope of traditional Calendar services. WS-Calendar extends the ~~ATTACH element~~Attachment to support payloads developed in other specifications. WS-Calendar also defines a new ~~class of parameters~~Property for iCalendar ~~components~~Components that ~~specify~~specifies the Tolerance for variation in temporal performance ~~requirements of the~~that still results in successful delivery of service.

3.2.1 Attachment and the Artifact

~~The WS-Calendar~~ Each Interval contains an Attachment to provide**~~Attach~~**~~ component provides~~ a container for delivering a payload or for referencing an external service. This payload ~~would be~~is transported within ~~WS-Calendar~~the Interval either because it describes a service that is or can be provided over an Interval, or whose service qualities vary over several Intervals in a Sequence. As the Technical Committee cannot know all the specifications that may incorporate WS-Calendar, this specification cannot discuss the contents of this payload. WS-Calendar does expect, however, that these payloads will respect and extend the inheritance and conformance rules herein specified.

The payload may be in-line, i.e., contained within the WS-Calendar Attach, or it may be found by reference. WS-Calendar supports references either to another section of the same XML document sharing the same message as WS-Calendar element, or to an external service or specification. The WS-Calendar Attach can be ~~thought~~thought of as having three options: “perform as described here”, or “perform as described below”, or “perform as described elsewhere.”

~~The WS-Calendar Attach has three options for communicating interval-based information as below.~~

Table 3-3: Elements of a WS-Calendar Attachment

Attachment Element	~~Use~~	Discussion
Artifact	~~any in-line XML (xs:any)~~ ~~An attachment must have at least one artifact or reference~~	Unevaluated (by WS-Calendar) container for payload describing service.
~~**uri**~~**Uri**	~~[XPOINTER]~~ ~~An attachment must have at least one of artifact or reference~~	Points to external XML, or XML located elsewhere in the document
Text	~~Any text (xs:text)~~	The use of text in ~~WS-Calendar~~an Attachment is allowed by but not defined~~.~~ in this specification.

Deleted

Specifications that incorporate WS-Calendar may wish to restrict these choices through conformance requirements.

Example 3-9: Use of an Attachment with inline XML artifact

```
<xcal:interval xmlns:~~xs="http://www.w3.org/2001/XMLSchema-instance"~~
~~xmlns:xcal~~payload="urn:~~ietf:params:xml:ns:icalendar-2.0">~~
~~  <xcal:properties>~~
~~    <xcal:uid>~~
~~      <xcal:text>38db45b7-0e95-4034-af56-90901cc5b892@examples.oasis-~~
~~open.org</xcal:text>~~
~~    </xcal:uid>~~
~~    <xcal:duration>~~
~~      <xcal:duration>T10H</xcal:duration>~~
~~    </xcal:duration>~~
~~    <xcal:x-wscalendar-attach>~~
~~      <xcal~~not:a:real:artifact~~>~~
~~        <xx:payload xmlns:xx="urn:externally:defined:artifact">~~
~~          <xx:units>furlongs</xx:units>~~
~~          <xx:quantity>14</xx:quantity>~~
~~        </xx:payload>~~
~~      </xcal:artifact>~~
~~    </xcal:x-wscalendar-attach>~~
~~  </xcal:properties>~~
~~  <xcal:components />~~
~~</xcal:interval>~~
```

~~The Artifact is of type xs:any, allowing compliant XML from any namespace to be submitted as a payload. Per the conformance rules, the payload should be Fully Bound before evaluation.~~

~~*Example 3-8: Use of an Attachment with external reference*~~

```
~~<xcal:interval xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"~~
~~xmlns:xcal="urn:ietf:params:xml:ns:icalendar-2.0~~">
   <xcal:properties>
          <xcal:uid>
~~                <xcal:parameters/>~~
                <xcal:text>~~d3c868ad-91e4-46ab-9281-~~
~~2505d03421e0@examples.oasis-open.org~~9c829c35-061a-466e-98f5-
ec1fe7b49d6a</xcal:text>
          </xcal:uid>
          <xcal:duration>
~~                <xcal:parameters/>~~
                <xcal:duration>~~T10H~~PT10H</xcal:duration>
          </xcal:duration>
```

```
~~            </xcal:duration>~~
            <xcal:x-~~wscalendar~~wsCalendar-attach>
                    <~~xcal:parameters/>~~payload:payload>
~~                    <xcal:uri>http://examples.oasis-~~
~~open.org/reference/external</xcal:uri>~~
                            <payload:units>furlongs</payload:units>
                            <payload:quantity>11</payload:quantity>
                    </payload:payload>
            </xcal:x-~~wscalendar~~wsCalendar-attach>
    </xcal:properties>
</xcal:interval>
~~    <xcal:components/>~~
```

~~</~~The Artifact is any element derived from the Attach, allowing compliant XML from any namespace to be submitted as a payload. As per the rules of any specification claiming conformance, the payload should be Fully Bound before evaluation for completeness.

Example 3-10: Use of an Attachment with external reference

```
<xcal:interval>
    <xcal:properties>
            <xcal:uid>
                    <xcal:text>ad289a5e-44b0-4e28-9cbc-d61b715f5427</xcal:text>
            </xcal:uid>
            <xcal:duration>
                    <xcal:duration>PT10H</xcal:duration>
            </xcal:duration>
            <xcal:x-wsCalendar-attach>
                    <xcal:uri>http://examples.oasis-
open.org/ref/external</xcal:uri>
            </xcal:x-wsCalendar-attach>
    </xcal:properties>
```

```
~~Specifying~~</xcal:interval>
```

3.2.2 Tolerance: What is Timely Performance

~~WS-Calendar elements adapt the iCalendar objects to make interaction requirements explicit. For example, in human schedule interactions, different organizations have their own expectations. Meetings may start on the hour or within 5 minutes of the hour. As agents scheduled in those organizations, people learn the expected precision.~~ The Tolerance parameter in WS-Calendar make interaction requirements explicit. In human schedule interactions, different organizations have their own expectations. Meetings may start on the hour or within 5 minutes of the hour. As agents scheduled in those organizations, people learn the expected precision. ~~WS-Calendar~~For services, that precision must be explicit to prevent interoperation problems.

~~Service~~Action coordination between systems requires precise communication about expectations for the timeliness of performance. ~~WS-Calendar defines~~The Tolerance ~~parameters. Tolerance parameters are~~parameter added to any iCalendar ~~component to make~~Component makes explicit the tolerance for time imprecision within a scheduled event. Tolerance can be ~~set for~~ applied to each Interval or ~~for~~to an entire Sequence.

The Tolerance ~~component~~Property refines the meaning of time-related ~~service~~ communication~~.~~ between services. All elements of the Tolerance ~~parameter~~Property use the Duration element as defined in **[RFC5545]**.

Table 3-4:~~ Performance~~ Tolerance Elements

~~Performance~~ Tolerance ~~Element~~	~~Definition~~	Discussion	
~~startBeforeToleranceS~~Start Before Tolerance	~~A Duration enumerating how far before the requested start time the requested service may commence.~~	Indicates how far before the requested start time the requested service may commence, for example if a service that begins at 1:57 is compliant with a request to start at 2:00	
~~startAfterTolerance~~Start After Tolerance	~~A Duration enumerating how far after the requested start time the requested service may commence.~~	Indicates how far after the requested start time the requested service may commence , for example, if a service that begins at 2:01 is compliant with a request to start at 2:00	
~~endBeforeTolerance~~End Before Tolerance	~~A Duration enumerating how far before scheduled end time may end.~~	Indicates how far before scheduled end time may end, for example, if a service that ends at 1:57 is compliant with a request to end at 2:00	
~~endAfterTolerance~~End After Tolerance	~~A Duration enumerating how far after the scheduled end time the requested service may commence.~~	Indicates how far after the scheduled end time the requested service may commence., for example, if a service that ends at 2:01 is compliant with a request to end at 2:00	
~~durationLongTolerance~~Duration Long Tolerance		~~A Duration indicating~~Indicates by how much the performance Duration may exceed the Duration specified in the ~~Interval . It may~~information exchange. Duration Long Tolerance SHALL NOT be ~~0.~~used when Start and End Tolerances are both specified.	~~Used when run time is more important than start and stop time. DurationLongTolerance SHALL NOT be used when Start and End Tolerances are both specified.~~
~~durationShortTolerance~~Duration Short Tolerance		~~A Duration indicating~~Indicates by how much the performance Duration may ~~fall short of~~exceed the Duration specified in the ~~Interval . It may~~information exchange. Duration Short Tolerance SHALL NOT be ~~0.~~used when Start and End Tolerances are both specified.	~~Used when run time is more important than start and stop time. DurationShortTolerance SHALL NOT be used when Start and End Tolerances are both specified.~~

Deleted

Deleted

~~Performance~~ Tolerance Element	~~Definition~~	Discussion
~~granularity~~Granularity	~~A Duration enumerating the smallest unit of time measured or tracked~~	Whatever the time tolerance above, there is some minimum time that is considered ~~insignificant. A~~significant. When used in Tolerance, Granularity ~~of 1 second~~ defines the tracking and reporting requirements for a service.

Tolerance is part of the core WS-Calendar service definition. Similar products or services, identical except for different Tolerance characteristics may appear in different markets. The ability to perform within Tolerance influences the price offered and the service selected.

Note that Tolerance parameter does not indicate time, but only Duration. A Tolerance parameter associated with an ~~unscheduled~~Unscheduled Interval does not change when that Interval is scheduled.

~~Tolerance parameters are optional components of each WS-Calendar attachment.~~

~~*Example 3-9: Performance Component*~~

```
<xcal:interval xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xcal="urn:ietf:params:xml:ns:icalendar-2.0">
   <xcal:properties>
          <xcal:uid>
                <xcal:parameters>
                       <xcal:startbeforetolerance>
                              <xcal:duration xs:type="xcal:DurationPropType">
                                     <xcal:duration>T1M</xcal:duration>
                              </xcal:duration>
                       </xcal:startbeforetolerance>
                       <xcal:startaftertolerance>
                              <xcal:duration xs:type="xcal:DurationPropType">
                                     <xcal:duration>T0M</xcal:duration>
                              </xcal:duration>
                       </xcal:startaftertolerance>
                       <xcal:durationlongtolerance>
                              <xcal:duration xs:type="xcal:DurationPropType">
                                     <xcal:duration>T0M</xcal:duration>
                              </xcal:duration>
                       </xcal:durationlongtolerance>
                       <xcal:durationshorttolerance>
                              <xcal:duration xs:type="xcal:DurationPropType">
                                     <xcal:duration>T0M</xcal:duration>
                              </xcal:duration>
                       </xcal:durationshorttolerance>
                </xcal:parameters>
                <xcal:text>d79c8b20-68db-43bf-8919-
4c397264a654@examples.oasis-open.org</xcal:text>
          </xcal:uid>
          <xcal:duration>
                <xcal:duration>T30M</xcal:duration>
          </xcal:duration>
   </xcal:properties>
   <xcal:components/>
</xcal:interval>
```

In the example, the service can start as much as 1 ~~minutes~~minute earlier than the scheduled time, and must start no later than the scheduled time. Whenever the service starts, the service must execute for exactly the Duration indicated.

Generally, the implementer should refrain from expressing unnecessary or redundant Tolerance characteristics.

~~3.2.3 Expressing Service and Tolerance~~

~~Services, references and Tolerance each appear in the example below~~

Example 3-11: Interval with inline XML artifact and optional specified Performance

```
<xcal:interval xmlns:~~xs="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xcal="urn:ietf:params:xml:ns:icalendar-
2.0~~payload="urn:not:a:real:artifact">
  <xcal:properties>
            <xcal:uid>
                  <xcal:text>~~70e487fc-45c6-40d3-a2ac-
51749c7e8c8f@examples.oasis-open.org~~030c603c-9e06-4dd4-8354-
69dc3fa4d253</xcal:text>
            </xcal:uid>
            <xcal:duration>
~~      <xcal:parameters>~~
~~        <xcal:startbeforetolerance>~~
                        <xcal:duration~~ xs:type="xcal:DurationPropType">~~
~~              <xcal:parameters />~~
~~              <~~>PT3H30M</xcal:duration>~~T10M~~
         </xcal:duration>
~~            </xcal:duration>~~
~~        </xcal:startbeforetolerance>~~
~~        <xcal:startaftertolerance>~~
~~          <xcal:duration xs:type="xcal:DurationPropType">~~
~~            <xcal:parameters />~~
~~            <xcal:duration>T0M</xcal:duration>~~
~~          </xcal:duration>~~
~~        </xcal:startaftertolerance>~~
~~      </xcal:parameters>~~
~~      <xcal:duration>T3H30M</xcal:duration>~~
~~    </xcal:duration>~~
            <xcal:x-~~wscalendar~~wsCalendar-attach>
~~      <xcal:artifact>~~
~~        <xx:payload xmlns:xx="urn:externally:defined:artifact">~~
~~          <xx~~                <payload:payload>
                      <payload:units>furlongs</~~xx~~payload:units>
~~          <xx~~
   <payload:quantity>~~14</xx~~11</payload:quantity>
~~        </xx:~~                </payload:payload>
~~      </xcal:artifact>~~
            </xcal:x-~~wscalendar~~wsCalendar-attach>
         <xcal:tolerance>
            <xcal:tolerate>
                  <xcal:startbefore>PT10M</xcal:startbefore>
                  <xcal:startafter>PT0M</xcal:startafter>
            </xcal:tolerate>
         </xcal:tolerance>
  </xcal:properties>
~~  <xcal:components />~~
</xcal:interval>
```

3.3 Using Sequences: referencing, modifying, and remote access

Sequences can define specific progressions of performance or state within a wide range of services and specifications. They become more useful as they can be re-used or modified. A Sequence that is not fully specified can be adapted and re-used without re-statement. An abstract Sequence can become a service through iterative referencing.

An entire Sequence can become scheduled by scheduling a single Interval in a Sequence. A single Sequence can become scheduled multiple times by repeated reference through different Gluons. The terminology describing this was introduced in Table 1-6.

As a Sequence is reified through reference, WS-Calendar specifies how additional information is applied or not applied to each Interval through a chain of references. We refer to this process as inheritance. Derivative specification can take advantage of inheritance by defining specific rules that conform to the WS-Calendar inheritance pattern.

This section describes how to create ~~references~~References to Sequences, including remote ~~references~~References, the rules that allow schedule-related information to become more complete through those references, and how to specify conforming rules in derivative specifications.

3.3.1 References and Inheritance.

Sequences are composed of Intervals for which a set of temporal relations have been defined. ~~**[RFC5545]** also defines the “PARENT”, “CHILD” and “SIBLING” relationships, in which one component references another by UID.~~ In WS-Calendar, we ~~reference~~ refer to a Sequence by creating a ~~relationship with~~Relation of type “CHILD” that references the UID of any ~~single~~ Interval in the Sequence. ~~We refer to~~ As defined in Table 1-6, the Interval within a Sequence that ~~has~~is the target of this ~~relationship as~~reference is the Designated Interval. The referring Component is named the Parent.

Wherever ~~there is “missing” information in~~ the Designated Interval, it can ~~be inherited is inherited~~ inherit that information from the referring ~~component; we use the “CHILD” relationship to reference the designated Interval.~~Component. These references may be local or remote. Some, but not all, of the information can be inherited by the other Intervals in the Sequence.

Adding additional references can further specify information in the Sequence through inheritance; these additional references are created by specifying an additional ~~component~~Gluon that has a ~~parent relation to~~Relation that references the previous referring ~~component.~~Component as a CHILD. In this way, we can create a grand-parent and a great grand-parent.

A Remote Reference is a Relation to a Component external to the conveying message. A Component in a message may reference a component already known to the receiving system. In this way, a remote Sequence can be invoked (and scheduled) without re-definition or re-transmission.

Each ~~parent~~Parent bequeaths information to its ~~child.~~Child. A ~~child~~Child inherits this information in accord with the inheritance rules. If the child is itself a parent, it bequeaths its information, the ~~bound~~Bound result of its internal information and its inheritance, to its child. Information to complete the specification of a Sequence flows in this way from parent to child, from the outer reference to the inner Sequence.

Inheritance by the ~~designated~~Designated Interval is governed by slightly different inheritance rules than the other Intervals in the Sequence. In particular, only the ~~designated~~Designated Interval can inherit the start date and time from its parent. The starting date and times if other Intervals in a Sequence are computed using the temporal relationships within the Sequence. Other information can be inherited by all Intervals in a Sequence. ~~Full inheritance rules are specified at [reference].~~ The semantics used for inheritance is in *Table 1-5: Semantics: Inheritance* and conformance rules for Inheritance are found in *Section 0*.

3.3.1.1 Introducing the Gluon

The referring ~~components~~Components described in 3.3.1 are named Gluons. In physics, gluons are particles that affect the exchanges of force between quarks, but are not themselves quarks. By analogy, ~~WS-Calendar~~ Gluons affect the referencing and binding of Intervals in a Sequence, but are not themselves Intervals or part of Sequences. Because Intervals can inherit almost any property from a Gluon, Gluons ~~must~~ contain most of the same information elements as Intervals. Because Intervals can contain information payloads for specifications that use WS-Calendar, ~~and these payloads can inherit information from gluons in the same way Intervals do,~~ Gluons ~~must be able to~~can contain information payloads from those specifications as well. ~~Gluons are described in the next section.~~

~~3.3.1.11.1.1.1 Introducing the Gluon~~

~~WS-Calendar~~ Gluons ~~are used to referencing~~reference and bind the Intervals in a Sequence, but are not themselves Intervals or part of Sequences. ~~Gluons must contain most of the same information elements as Intervals, because Intervals~~Gluons can ~~inherit almost any property from a Gluon. When Intervals are used in other specifications, they~~ contain payloads ~~for that~~, or portions of payloads, which are not defined in ~~WS-Calendar.~~ this specification. Information from Gluons ~~can also hold the same payloads, and conforming specifications MUST define inheritance rules that govern inheritance within these payloads. Conformance rules, including those for inheritance conformance, are discussed~~is inherited by their Children as described in section 0 ~~*Conformance and Rules for WS-Calendar and Referencing Specifications*~~*Conformance and Rules for WS-Calendar and Referencing Specifications.*

The ~~WS-Calendar~~ Gluon is in essence an the Interval ~~component~~Component profiled down to minimal elements for which inheritance rules defined, and able to carry a conforming informational payload. (See Appendix ~~*Overview of WS-Calendar, its Antecedents and its Use*) Calendar~~*Overview of WS-Calendar, its Antecedents and its Use*) Gluons use iCalendar relations to apply service information to Sequences.

Table 3-5: ~~Calendar~~ Gluon Elements

~~Calendar~~ **Gluon Element**	~~**Use**~~	**Discussion**	
~~dtStamp~~DtStamp	~~[XCAL]:dtstamp~~ ~~*Mandatory*~~	Time and date that ~~Calendar~~ Gluon ~~object~~artifact was created	
Uid	~~*Mandatory*~~	Used to enable unambiguous referencing of each Gluon object	
Summary	~~Text~~ ~~*Optional*~~	Text describing the ~~Calendar~~ Gluon	
~~child~~Child	~~As defined in **[RFC5998]**~~	A ~~Calendar~~ Gluon must have a link to at least one CHILD.	
~~dtStart~~Duration		~~dateTime~~ ~~Start time for the Designated Interval.~~ ~~*Optional*~~If specified, a Duration is potentially inherited by all Intervals in the referred-to Sequence.	~~A Calendar Gluon may either have a dtStart or a dtEnd, but may not have both.~~
~~dtEnd~~DtStart		~~DateTime.~~ ~~Scheduled completion time for~~A Gluon may either have a dtStart or a dtEnd, but may not have both. DtStart is inherited by the Designated Interval. ~~*Optional*~~	~~An Calendar Gluon may either have a dtStart or a dtEnd, but may not have both.~~

Deleted

Deleted

~~Calendar~~ Gluon Element	~~Use~~	Discussion
~~duration~~DtEnd	~~Duration~~ ~~*Optional*~~	~~If specified, a Duration is potentially inherited by all Intervals in the referred-to Sequence,~~A Gluon may have either a dtStart or a dtEnd, but may not have both. DtEnd is inherited by the Designated Interval, in which it is used with the Bound Duration to compute the Bound dtStart.
~~WsCalendar~~ Attach	~~WSCalendar:Attachment~~ ~~Optional~~	The ~~Attach~~used as a base class for extension by conforming specifications. Each contains the informational payload ~~used by incorporating specifications.~~ defined in that specification. Defined in section 3.2~~.~~.
Availability~~Vavailability~~	~~Vavailability, Optional~~ ~~See Table 3-7~~	~~Provides~~Referred to as Availability, provides information as to when ~~information the service~~a process can be scheduled.

Deleted

Deleted

It is important to distinguish between the general model of the Gluon in WS-Calendar and the more specific requirements of an incorporating specification. At its minimum, a Gluon may be only a pointer to a sequence, containing only a link to its child. A Gluon may alternately include information completing (or partially completing) the information in a Sequence; that information may vary based on ~~and~~ what is required to make the information payload actionable within any particular transaction.

Because the properties of the ~~Calendar~~ Gluon are bequeathed to the child Sequence, they can stand for the elements in any Interval in the Sequence, as defined in the Conformance Section. An inherited element can even serve as a substitute for an Interval mandatory element. For example, Duration is mandatory for all Intervals. Intervals are able to inherit Duration from a parent. A single Duration in the Parent can be inherited by each Interval in a Sequence.

In this way, a Sequence in which every Interval does not have a Duration, could be made complete through inheritance. If one of those Intervals does include a Duration, the Bound Duration would be its own, rather than that it inherited from a Parent of the Sequence.

There is a critical distinction between an individual Gluon, which may be only a pointer to a sequence, or may have information completing (or partially completing) the information in a Sequence, and what is required to make the information payload actionable within any particular transaction.

3.3.1.2 Availability

~~One~~An additional use for gluons is to expose a Sequence for remote invocation. The service offered may be only sometimes ~~unavailable.~~available. WS-Calendar incorporates the iCalendar extension **[Vavailability~~].~~]** to expose this schedule.

~~It~~**[Vavailability]** offers a means to describe recurring temporal patterns, such a weekdays from 9:00-5:00, Thursday mornings until July, and thereafter Tuesday evening as well. A Vavailability component is ~~likely that~~a collection of Availability components, each with its data boundaries and its recurrence patterns. The parameters and properties are those defined in iCalendar, the structure is defined in the referenced **[Vavailability]**, and the artifact is an optional Component of a Gluon.

A requestor may not be aware of all aspects of the Sequence. A service ~~requester is aware only of when he wants the service and for how long. These are properties~~ requestor does know, however, the desired Start and Duration of the Designated Interval. **[Vavailability]** in a Gluon is **~~Availability~~** ~~will be~~ interpreted as a filter only on the Designated Interval~~, but on no others.~~ .

~~*Table 3-7:Vavailability elements with specified use in WS-Calendar*~~

~~Vavailability Element~~	~~Use~~	~~Discussion~~

~~Vavailability Element~~	~~Use~~	~~Discussion~~
~~**dtStamp**~~	~~[XCAL]:dtstamp~~ ~~*Mandatory*~~	~~Time and date that Vavailability object was created~~
~~**Uid**~~	~~*Mandatory*~~	~~Used to enable unambiguous referencing of each Vavailability object~~
~~**Summary**~~	~~Text~~ ~~*Optional*~~	~~Text describing the Vavailability~~
~~**dtStart**~~	~~xcal:dateTime~~ ~~Start time for the Availability.~~ ~~*Mandatory*~~	~~All time before the dtStart is considered unavailable..~~
~~**dtEnd**~~	~~xcal:dateTime.~~ ~~End point for availability.~~ ~~*Optional*~~	~~If present, all recurrence and other patterns inside the Available objects ends with the dtEnd. Either a dtStart or a dtEnd, may be present, but not both.~~
~~**duration**~~	~~Duration~~ ~~*Optional*~~	~~If present, the duration beginning with the dtStart is time for which availability is specified. Either a dtStart or a dtEnd, may be present, but not both,~~
~~**Availability**~~	~~Availability, one to many occurrences. See Table 3-8~~	~~At least one Availability is required to state the pattern when the service is available~~
~~**Granularity**~~	~~xcal:duration~~ ~~Optional~~	~~Granularity, when used in Availability, limits when a service can be scheduled. For example, a 15 minute granularity on a 9:00 dtStart, implies that legal dtStarts are 9:99, 9:15, 9:30, 9:45, …~~

~~If~~WS-Calendar adds a single optional parameter to the **[Vavailability]** component. When a Granularity component is applied, it further defines the acceptable service invocation. Granularity is discussed in the next section.

3.3.1.3 Granularity used as part of Availability

Granularity can be applied both to Availability ~~is not terminated by either an dtEnd or bounded by a duration, then the end of the Vavailability is undefined. If Vavailability is so terminated, then that termination bounds any recurrence patterns defined in the Available elements.~~

~~Available elements define the actual times during which the resource or service is available for invocation or scheduling.~~

~~*Table 3-8:Availability elements contained within Vavailabilty.*~~

~~Availability Element~~	~~Use~~	~~Discussion~~
~~**Uid**~~	~~*Mandatory*~~	~~Used to enable unambiguous referencing of each Availability object~~
~~**Summary**~~	~~Text~~ ~~*Optional*~~	~~Text describing the Availability~~

~~Availability Element~~	~~Use~~	~~Discussion~~
~~dtStart~~	~~xcal:dateTime~~ ~~Start time for the Availability.~~ *~~Mandatory~~*	~~All time before the dtStart is considered unavailable..~~
~~dtEnd~~	~~xcal:~~d~~ateTime.~~ ~~End point for availability.~~ *~~Optional~~*	~~If present, all recurrence and other patterns inside the Available objects ends with the dtEnd. Either a dtStart or a dtEnd, may be present, but not both.~~
~~duration~~	~~Duration~~ *~~Optional~~*	~~If present, the duration beginning with the dtStart is time for which availability is specified. Either a dtStart or a dtEnd, may be present, but not both,~~
~~RRule~~	~~xcal:rrule~~	~~Defines how often the avalablility interval, defined as either the period bounded by the dtStart and dtEnd OR beginning with the dtStart and lasting for the duration, repeats~~
~~Granularity~~	~~xcal:duration~~ ~~Optional~~	~~Granularity, when used in Availability, limits when a service can be scheduled. For example, a 15 minute granularity on a 9:00 dtStart, implies that legal dtStarts are 9:99, 9:15, 9:30, 9:45, …~~

~~3.3.1.3 Granularity~~

~~Granularity affects both Vavailability and Availability.~~(the collection) and to Availability (the individual rule). If Granularity is specified, then it communicates the ~~-~~expectation that services that invoke WS-Calendar ~~specifications~~conforming services should request ~~them in multiples of~~only Start times that match the Granularity.

For example, ~~assume~~ the Designated Interval of a ~~service~~ Sequence has a Duration of One Hour, and is available on weekdays from ~~9:00 to~~8:30 until 11:00. Without Granularity~~ suggests that if~~, the ~~duration~~Service can be ~~specified, then it should be specified as one of 15M, 30M, 45M, 1H, 1H15M, 1H30M, 1H45M or 2H. The same~~Scheduled at any time that does not start before 8:30, nor end after 11:00. If a Granularity ~~indicates that the service can only be specified in~~ of 30 minutes *"PT30M"* is applied, the Scheduled Starts are limited to 8:30, 9:00, 8:30, and 10:00, i.e., integral multiples of the Duration of the Granularity ~~following the dtStart, i.e., at 9:00 (0x), 9:15 (1x), 9:30 (2x), 9:45(3x), and so on~~beginning at the beginning of the available window.

3.3.2 ~~Calendar~~ Gluons and Sequences

WS-Calendar Gluons express common service requirements for an entire Sequence. If a Gluon is parent to an Interval in a Sequence, then the Gluon's Attachment expresses service attributes inheritable by all Intervals in the Sequence.

In this example, the Sequence in the previous example is expressed using a ~~Calendar~~ Gluon.

Example 3-12: Sequence with Performance defined in the ~~Calendar~~ Gluon

```
~~<xcal:vcalendar xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"~~
~~xmlns:xcal="urn:ietf:params:xml:ns:icalendar-2.0"~~
~~xs:type="xcal:VcalendarType">~~
~~    <xcal:properties/>~~
<xcal:vcalendar>
```

```
    <xcal:components>
        <xcal:gluon>
            <xcal:properties>
                <xcal:uid>
                    <xcal:text>~~2f9d675e-88b3-457d-a6e1-3045ac1816d6@examples.oasis-open.org~~5ffaa487-206f-46e8-b3e5-958b37477cab</xcal:text>
                </xcal:uid>
                <xcal:related-to>
                    <xcal:parameters>
                        <xcal:reltype>
                            <xcal:text>CHILD</xcal:text>
                        </xcal:reltype>
                    </xcal:parameters>
                ~~</xcal:uid>~~
                ~~<xcal:related-to>~~
                    ~~<xcal:parameters>~~
                        ~~<xcal:reltype>~~
                            ~~<xcal:text>FS</xcal:text>~~
                        ~~</xcal:reltype>~~
                    ~~</xcal:parameters>~~
                    <xcal:uid>~~e53d40cc-5c9c-44a4-9674-6ad492e76021@examples.oasis-open.org~~2a7de3f0-54c5-4a31-9856-6a94e6c82902</xcal:uid>
                </xcal:related-to>
                <xcal:dtstart>
                    <xcal:parameters>
                        <xcal:tzid>
                            <xcal:text>America/New_York</xcal:text>
                        </xcal:tzid>
                ~~</xcal:related-to>~~
                ~~<xcal:dtstart>~~
                    ~~<xcal:parameters>~~
                        ~~<xcal:tzid>~~
                            ~~<xcal:text>America/New_York</xcal:text>~~
                        ~~</xcal:tzid>~~
                        ~~<xcal:startbeforetolerance>~~
                            ~~<xcal:duration xs:type="xcal:DurationPropType">~~
                                ~~<xcal:duration>T0M</xcal:duration>~~
                            ~~</xcal:duration>~~
                        ~~</xcal:startbeforetolerance>~~
                        ~~<xcal:startaftertolerance>~~
                            ~~<xcal:duration xs:type="xcal:DurationPropType">~~
                                ~~<xcal:duration>T0M</xcal:duration>~~
                            ~~</xcal:duration>~~
                        ~~</xcal:startaftertolerance>~~
                        ~~<xcal:durationlongtolerance>~~
                            ~~<xcal:duration xs:type="xcal:DurationPropType">~~
                                ~~<xcal:duration>T5M</xcal:duration>~~
                            ~~</xcal:duration>~~
                        ~~</xcal:durationlongtolerance>~~
                        ~~<xcal:durationshorttolerance>~~
                            ~~<xcal:duration xs:type="xcal:DurationPropType">~~
```

```
                                ~~<xcal:duration>T0M</xcal:duration>~~
                            ~~</xcal:duration>~~
                        ~~</xcal:durationshorttolerance>~~
                        ~~<xcal:granularity>~~
                            ~~<xcal:duration xs:type="xcal:DurationPropType">~~
                                ~~<xcal:duration>T5S</xcal:duration>~~
                            ~~</xcal:duration>~~
                        ~~</xcal:granularity>~~
                    </xcal:parameters>
                    <xcal:date-time>~~20110315T08450000~~2011-05-28T08:45:00</xcal:date-time>
                </xcal:dtstart>
                <xcal:tolerance>
                    <xcal:tolerate>
                        <xcal:durationlong>PT5M</xcal:durationlong>
                        <xcal:durationshort>PT0M</xcal:durationshort>
                        <xcal:granularity>PT5S</xcal:granularity>
                    </xcal:tolerate>
                </xcal:tolerance>
                <xcal:x-~~wscalendar~~wsCalendar-attach>
                    <~~xcal:artifact/>~~payload:payload>
                        <payload:units>furlongs</payload:units>
                        <payload:quantity>11</payload:quantity>
                    </payload:payload>
                </xcal:x-~~wscalendar~~wsCalendar-attach>
            ~~</xcal:properties>~~
        ~~</xcal:gluon>~~
        ~~<xcal:interval>~~
            ~~<xcal:properties>~~
                ~~<xcal:uid>~~
                    ~~<xcal:text>c53d40cc-5e9e-44a4-9674-6ad492e76021@examples.oasis-open.org</xcal:text>~~
                ~~</xcal:uid>~~
            ~~</xcal:properties>~~
        ~~</xcal:interval>~~
        ~~<xcal:interval>~~
            ~~<xcal:properties>~~
                ~~<xcal:uid>~~
                    ~~<xcal:parameters/>~~
                    ~~<xcal:text>67319fa7-28b3-4abc-91b8-c595fc2948a8@examples.oasis-open.org</xcal:text>~~
                ~~</xcal:uid>~~
                ~~<xcal:related-to>~~
                    ~~<xcal:parameters>~~
                        ~~<xcal:reltype>~~
                            ~~<xcal:text>FS</xcal:text>~~
                        ~~</xcal:reltype>~~
                    ~~</xcal:parameters>~~
                    ~~<xcal:uid>c53d40cc-5e9e-44a4-9674-6ad492e76021@examples.oasis-open.org</xcal:uid>~~
                ~~</xcal:related-to>~~
            </xcal:properties>
            <xcal:components/>
        </xcal:gluon>
        <xcal:interval>
            <xcal:properties>
                <xcal:uid>
        ~~</xcal:interval>~~
```

```
          ~~<xcal:interval>~~
                 ~~<xcal:properties>~~
                        ~~<xcal:uid>~~
                               <xcal:text>2a7de3f0-54c5-4a31-9856-
6a94e6c82902</xcal:text>
                        </xcal:uid>
                 </xcal:properties>
          </xcal:interval>
          <xcal:interval>
                 <xcal:properties>
                        <xcal:uid>
                               <xcal:text>1886ebd8-a5a9-4aa8-9b6b-
2869e4b711de</xcal:text>
                        </xcal:uid>
                        <xcal:related-to>
                               <xcal:parameters>
                                      <xcal:reltype>
                                             <xcal:text>FS</xcal:text>
                                      </xcal:reltype>
                               </xcal:parameters>
                               <xcal:uid>2a7de3f0-54c5-4a31-9856-
6a94e6c82902</xcal:uid>
                        </xcal:related-to>
                 </xcal:properties>
          </xcal:interval>
          <xcal:interval>
                 <xcal:properties>
                        <xcal:uid>
                               <xcal:text>~~e6c3e351-77ee-4c27-abce-~~
~~8e5c1d9ef6db@examples.oasis-open.org~~62e518b4-65df-4fea-b64a-
91a762ae173a</xcal:text>
                        </xcal:uid>
                        <xcal:related-to>
                               <xcal:parameters>
                                      <xcal:reltype>
                                             <xcal:text>FS</xcal:text>
                                      </xcal:reltype>
                               </xcal:parameters>
                        ~~</xcal:uid>~~
                        ~~<xcal:duration>~~
                               ~~<xcal:duration>T30M</xcal:duration>~~
                        ~~</xcal:duration>~~
                        ~~<xcal:related-to>~~
                               ~~<xcal:parameters>~~
                                      ~~<xcal:reltype>~~
                                             ~~<xcal:text>FS</xcal:text>~~
                                      ~~</xcal:reltype>~~
                               ~~</xcal:parameters>~~
                               <xcal:uid>~~67319fa7-28b3-4abe-91b8-~~
~~e595fc2948a8@examples.oasis-open.org~~1886ebd8-a5a9-4aa8-9b6b-
2869e4b711de</xcal:uid>
                        </xcal:related-to>
                        ~~</~~<xcal:~~related-to~~duration>
                               <xcal:duration>PT30M</xcal:duration>
                        </xcal:duration>
                 </xcal:properties>
                 ~~<xcal:components/>~~
          </xcal:interval>
   </xcal:components>
</xcal:vcalendar>
```

Note that the performance expectations, identical for each Interval, have moved into the ~~Calendar~~ Gluon. Not also that while the duration for all Intervals in the partition is set in the ~~Calendar~~ Gluon, Interval 3 overrides that with a half~~-~~-hour duration assigned locally. This~~ Calendar~~ Gluon happens to be related to the first Interval in the Sequence; there are specific use cases (discussed below) which require it to be linked to other Intervals.

3.3.3 Inheritance rules for ~~Calendar ~~Gluons

In general, the rule is that anything specified in the Parent~~ Calendar~~ Gluon applies to each Child. The Parent of an Interval in a Sequence is parent to all Intervals in the Sequence. As a Sequence creates single temporal relationship, assigning a start time (dtstart) to any Interval allows computation of the starting time for each of them.

Table 3-6 Gluon Inheritance rules

Attribute	Inheritance Rules
General	A Interval or~~ Calendar~~ Gluon inherits its attributes through it's the parent. Local specification of an attributes overrides any inheritance.
Duration	Follows general rules
Temporal Relation	Relationship Type and Gap only are inherited. Either may be overridden locally. To specify no gap when a parent specifies a gap, an explicit zero duration gap must be specified. Related-to is not inherited.
Performance	Performance is either inherited intact or overridden completely. There are no rules for recombining partial Performance objects through inheritance.
Artifacts	Artifacts hold payload from other specifications. Elements within Artifacts are inherited in accord with the rules in those specifications, which must be consistent the inheritance rules in WS-Calendar. Artifacts are evaluated for completeness and conformance only after processing inheritance.
~~Schedules~~Schedule	~~Schedules~~Schedule, i.e., the start date and time, are inherited only by the ~~designated~~Designated Interval. The start date and times of other Intervals are computed by reference to the ~~designated~~Designated Interval. Between the Gluon bequeathing a schedule and the Designated Interval, an intervening Gluon may set ~~availability.~~Availability. It is up to the application or to the specification incorporating WS-Calendar to assert whether an Interval that is outside the ~~availability~~Availability is conforming or not.
Availability	Availability communicates restrictions on when a service is offered. Service availability is interpreted for the Designated Interval only. If there are two ~~availability~~Availability objects, they are evaluated for the union of the two availabilities. For example, if I am available all week from 2:00 to 6:00 in one, and available all day Tuesday in the other, then after inheritance, there remains only 2:00 to 6:00 on Tuesday,

3.3.4 Optimizing the expression of a Partition

~~Partitions are Sequences composed~~A Partition is a set of consecutive Intervals. ~~A~~The expressions of a Partition can be ~~further ~~optimized by bringing the ~~relationship ~~Relation and Duration into the Gluon. Notice that while the type of the relationship is defined in the ~~Calendar ~~Gluon, the Temporal Relation for each Interval must still be expressed within the Interval.

Example 3-13: Partition with Duration and Relationship defined in the ~~Calendar ~~Gluon

```
~~<xcal:vcalendar xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"~~
~~xmlns:xcal="urn:ietf:params:xml:ns:icalendar-2.0"~~
~~xs:type="xcal:VcalendarType">~~
   ~~<xcal:properties/>~~
<xcal:vcalendar>
   <xcal:components>
          <xcal:gluon>
                 <xcal:properties>
                        <xcal:uid>
                               ~~<xcal:parameters/>~~
                               <xcal:text>~~97c504ed-263e-447d-95a6-~~
~~d59b97422edc@examples.oasis-open.org~~10795fba-5c0d-406e-b9a1-
6e8448d8e125</xcal:text>
                        ~~</xcal:uid>~~
                        ~~<xcal:related-to>~~
                               ~~<xcal:parameters>~~
                                      ~~<xcal:reltype>~~
                                             ~~<xcal:text>CHILD</xcal:text>~~
                                      ~~</xcal:reltype>~~
                               ~~</xcal:parameters>~~
                               ~~<xcal:uid>9b1c1ae8-ea4f-4065-9cf6-~~
~~45c53e709e55@examples.oasis-open.org</xcal:uid>~~
                        ~~</xcal:related-to>~~
                        ~~<xcal:related-to>~~
                               ~~<xcal:parameters>~~
                                      ~~<xcal:reltype>~~
                                             ~~<xcal:text>FS</xcal:text>~~
                                      ~~</xcal:reltype>~~
                               ~~</xcal:parameters>~~
                               ~~<xcal:uid>9b1c1ae8-ea4f-4065-9cf6-~~
~~45c53e709e55@examples.oasis-open.org</xcal:uid>~~
                        ~~</xcal:related-to>~~
                        ~~<xcal:duration>~~
                               ~~<xcal:duration>T15M</xcal:duration>~~
                        ~~</xcal:duration>~~
                        ~~<xcal:x-wscalendar-attach>~~
                               ~~<xcal:parameters/>~~
                               ~~<xcal:artifact/>~~
                        ~~</xcal:x-wscalendar-attach>~~
                 ~~</xcal:properties>~~
                 ~~<xcal:components/>~~
          ~~</xcal:gluon>~~
          ~~<xcal:interval>~~
                 ~~<xcal:properties>~~
                        ~~<xcal:uid>~~
                               ~~<xcal:text>9b1c1ae8-ea4f-4065-9cf6-~~
~~45c53e709e55@examples.oasis-open.org</xcal:text>~~
                        ~~</xcal:uid>~~
                        ~~<xcal:related-to>~~
                               ~~<xcal:uid>9b1c1ae8-ea4f-4065-9cf6-~~
~~45c53e709e55@examples.oasis-open.org</xcal:uid>~~
                        ~~</xcal:related-to>~~
                 ~~</xcal:properties>~~
                 ~~<xcal:components/>~~
          ~~</xcal:interval>~~
          ~~<xcal:interval>~~
                 ~~<xcal:properties>~~
                        ~~<xcal:uid>~~
                               ~~<xcal:text>50149441-18e8-4e8b-9e0f-~~
~~e8da3e671895@examples.oasis-open.org</xcal:text>~~
                        ~~</xcal:uid>~~
                        ~~<xcal:related-to>~~
```

```
                              ~~<xcal:uid>50149441-18e8-4e8b-9e0f-~~
~~e8da3e671895@examples.oasis-open.org</xcal:uid>~~
                       ~~</xcal:related-to>~~
                ~~</xcal:properties>~~
         ~~</xcal:interval>~~
         ~~<xcal:interval>~~
                ~~<xcal:properties>~~
                       ~~<xcal:uid>~~
                              ~~<xcal:parameters/>~~
                              ~~<xcal:text>661e6127-9e06-429c-b641-~~
~~205a31df64d1@examples.oasis-open.org</xcal:text>~~
                       ~~</xcal:uid>~~
                       ~~<xcal:related-to>~~
                              ~~<xcal:uid>661e6127-9e06-429c-b641-~~
~~205a31df64d1@examples.oasis-open.org</xcal:uid>~~
                       ~~</xcal:related-to>~~
                ~~</xcal:properties>~~
         ~~</xcal:interval>~~
         ~~<xcal:interval>~~
                ~~<xcal:properties>~~
                       ~~<xcal:uid>~~
                              ~~<xcal:parameters/>~~
                              ~~<xcal:text>c4457c5d-a848-4878-8571-~~
~~2f35ed02c594@examples.oasis-open.org</xcal:text>~~
                       ~~</xcal:uid>~~
                       ~~<xcal:related-to>~~
                              ~~<xcal:uid>c4457c5d-a848-4878-8571-~~
~~2f35ed02c594@examples.oasis-open.org</xcal:uid>~~
                       ~~</xcal:related-to>~~
                ~~</xcal:properties>~~
         ~~</xcal:interval>~~
         ~~<xcal:interval>~~
                ~~<xcal:properties>~~
                       ~~<xcal:uid>~~
                              ~~<xcal:text>13ffa401-dd7a-48cc-980a-~~
~~8f19aa91fd58@examples.oasis-open.org</xcal:text>~~
                       ~~</xcal:uid>~~
                       ~~<xcal:related-to>~~
                              ~~<xcal:uid>13ffa401-dd7a-48cc-980a-~~
~~8f19aa91fd58@examples.oasis-open.org</xcal:uid>~~
                       ~~</xcal:related-to>~~
                ~~</xcal:properties>~~
         ~~</xcal:interval>~~
         ~~<xcal:interval>~~
                ~~<xcal:properties>~~
                       ~~<xcal:uid>~~
                              ~~<xcal:text>c5cf94b3-4514-4093-b8a7-~~
~~70cd7bd174b1@examples.oasis-open.org</xcal:text>~~
                       ~~</xcal:uid>~~
                       ~~<xcal:related-to>~~
                              ~~<xcal:parameters>~~
                              ~~<xcal:uid>c5cf94b3-4514-4093-b8a7-~~
~~70cd7bd174b1@examples.oasis-open.org</xcal:uid>~~
                       ~~</xcal:related-to>~~
                ~~</xcal:properties>~~
         ~~</xcal:interval>~~
         ~~<xcal:interval>~~
                ~~<xcal:properties>~~
                       ~~<xcal:uid>~~
                              ~~<xcal:text>38f4e770-5f3d-466f-9222-~~
~~f1c801a43657@examples.oasis-open.org</xcal:text>~~
                       ~~</xcal:uid>~~
                       ~~<xcal:related-to>~~
```

```
                        ~~<xcal:uid>38f4e770-5f3d-466f-9222-f1c801a43657@examples.oasis-open.org</xcal:uid>~~
                    ~~</xcal:related-to>~~
                ~~</xcal:properties>~~
            ~~</xcal:interval>~~
            ~~<xcal:interval>~~
                ~~<xcal:properties>~~
                    ~~<xcal:uid>~~
                        ~~<xcal:text>daa6c916-de0a-4c1a-a852-1670fd97d47c@examples.oasis-open.org</xcal:text>~~
                    ~~</xcal:uid>~~
                    ~~<xcal:related-to>~~
                        ~~<xcal:uid>daa6c916-de0a-4c1a-a852-1670fd97d47c@examples.oasis-open.org</xcal:uid>~~
                    ~~</xcal:related-to>~~
                ~~</xcal:properties>~~
            ~~</xcal:interval>~~
    ~~</xcal:components>~~
~~</xcal:vcalendar>~~
```

~~This Partition shows 8 consecutive 15 minute intervals as part of a 2 hour partition.~~

~~*Example 3-13: Partition with Duration, Relationship, and Gap defined in the Calendar Gluon*~~

```
~~<xcal:vcalendar xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xmlns:xcal="urn:ietf:params:xml:ns:icalendar-2.0" xs:type="xcal:VcalendarType">~~
    ~~<xcal:components>~~
            ~~<xcal:gluon>~~
                ~~<xcal:properties>~~
                    ~~<xcal:uid>~~
                        ~~<xcal:text>c7496e78-6d71-4118-b42a-641f1efe02a9@examples.oasis-open.org</xcal:text>~~
                    </xcal:uid>
                    <xcal:related-to>
                        <xcal:parameters>
                            <xcal:reltype>
                                <xcal:text>CHILD</xcal:text>
                            </xcal:reltype>
                        </xcal:parameters>
                        <xcal:uid>~~5b7b5f46-fbc4-455e-9c60-7639463aca4e@examples.oasis-open.org~~bfe0040e-a5e0-4558-bbe1-a9207004a4cc</xcal:uid>
                        <xcal:gap>
                            <xcal:duration>PT10M</xcal:duration>
                        </xcal:gap>
                    </xcal:related-to>
                    <xcal:duration>
                        <xcal:duration>PT50M</xcal:duration>
                    </xcal:duration>
                    <xcal:x-wsCalendar-attach>
                        <payload:payload>
                            <payload:units>students</payload:units>
                            <payload:quantity>23</payload:quantity>
                        </payload:payload>
                    </xcal:x-wsCalendar-attach>
                    <xcal:tolerance>
                        <xcal:tolerate>
                            <xcal:startbefore>P</xcal:startbefore>
                            <xcal:startafter>PT5M</xcal:startafter>
                        </xcal:tolerate>
                    </xcal:tolerance>
                </xcal:properties>
                <xcal:components/>
            </xcal:gluon>
            <xcal:interval>
                <xcal:properties>
                    <xcal:uid>
                        <xcal:text>bfe0040e-a5e0-4558-bbe1-
    a9207004a4cc</xcal:text>
                    </xcal:uid>
                </xcal:properties>
            </xcal:interval>
            <xcal:interval>
                <xcal:properties>
                    <xcal:uid>
                        <xcal:text>7dabb353-8d0a-44c5-a81e-
    e49268fd85f3</xcal:text>
                    </xcal:uid>
                    <xcal:related-to>
                        <xcal:parameters>
                            <xcal:reltype>
                                <xcal:text/>
                            </xcal:reltype>
                        </xcal:parameters>
                        <xcal:uid>bfe0040e-a5e0-4558-bbe1-
    a9207004a4cc</xcal:uid>
                    </xcal:related-to>
                </xcal:properties>
            </xcal:interval>
            <xcal:interval>
                <xcal:properties>
                    <xcal:uid>
                        <xcal:text>4c9ca94b-0b47-4244-b72d-
    e5333e1f7e43</xcal:text>
                    </xcal:uid>
                    <xcal:related-to>
                        <xcal:parameters>
                            <xcal:reltype>
                                <xcal:text/>
                            </xcal:reltype>
                        </xcal:parameters>
                        <xcal:uid>bfe0040e-a5e0-4558-bbe1-
    a9207004a4cc</xcal:uid>
                    </xcal:related-to>
                </xcal:properties>
            </xcal:interval>
            <xcal:interval>
                <xcal:properties>
                    <xcal:uid>
                        <xcal:text>397068e3-ca9a-4b3a-a1e0-
    76a45671bb2f</xcal:text>
                    </xcal:uid>
                    <xcal:related-to>
                        <xcal:parameters>
                            <xcal:reltype>
                                <xcal:text~~>FS</xcal:text>~~/>
                            </xcal:reltype>
                            ~~<xcal:gap>~~
                                ~~<xcal:duration~~
    ~~xs:type="xcal:DurationPropType">~~
                                ~~<~~</xcal:parameters~~/>~~>
                                <xcal:~~duration>T10M~~uid>bfe0040e-
    a5e0-4558-bbe1-a9207004a4cc</xcal:~~duration~~uid>
                    </xcal:related-to>
                </xcal:properties>
```

```
         </xcal:interval>
         <xcal:interval>
                <xcal:properties>
                       <xcal:uid>
                              ~~</~~<xcal:~~duration~~text>c97dfe10-1e2d-
42c2-bdc5-41a5ca17c98a</xcal:text>
                       </xcal:uid>
                       <xcal:related-to>
                              ~~</xcal:gap>~~
                              ~~</~~<xcal:parameters>
                       ~~</~~      <xcal:~~related-to~~reltype>
                                          <xcal:~~duration>~~text/>
                              ~~<~~      </xcal:reltype>
                              </xcal:parameters~~/>~~>
                              <xcal:~~duration>T50M~~uid>bfe0040e-a5e0-4558-bbe1-
a9207004a4cc</xcal:~~duration~~uid>
                       </xcal:related-to>
                </xcal:properties>
         </xcal:interval>
         <xcal:interval>
                <xcal:properties>
                       <xcal:uid>
                              <xcal:text>bf22e5e1-891f-40eb-9012-
89577482f38d</xcal:text>
                       </xcal:uid>
                       <xcal:related-to>
                       ~~</xcal:duration>~~
                       ~~<xcal:x-wscalendar-attach>~~
                              <xcal:parameters~~/>~~>
                                     <xcal:~~artifact/>~~reltype>
                       ~~</~~             <xcal:~~x-wscalendar-attach>~~text/>
                                     </xcal:~~properties~~reltype>
         ~~</xcal:gluon>~~
         ~~<xcal:interval>~~
                ~~<xcal:properties>~~
                       ~~<xcal:uid>~~
                              ~~<~~</xcal:parameters~~/>~~>
                              <xcal:uid>bfe0040e-a5e0-4558-bbe1-
a9207004a4cc</xcal:uid>
                       </xcal:related-to>
                </xcal:properties>
         </xcal:interval>
         <xcal:interval>
                <xcal:properties>
                       <xcal:uid>
                              <xcal:text>~~5b7b5f46-fbc4-455e-9c60-
7639463aca4e@examples.oasis-open.org~~2fbe48fa-3539-4167-8db6-
2dae1d86f10b</xcal:text>
                       </xcal:uid>
                       <xcal:related-to>
                       ~~</xcal:uid>~~
                       ~~<xcal:related-to>~~
                              ~~<xcal:uid>5b7b5f46-fbc4-455e-9c60-
7639463aca4e@examples.oasis-open.org</xcal:uid>~~
                       ~~</xcal:related-to>~~
                ~~</xcal:properties>~~
         ~~</xcal:interval>~~
         ~~<xcal:interval>~~
                ~~<xcal:properties>~~
                       ~~<xcal:uid>~~
                              <xcal:parameters~~/>~~>
                              ~~<xcal:text>43da0574-d00b-41e8-8a47-
70767f63da78@examples.oasis-open.org</xcal:text>~~
```

```
                ~~</xcal:uid>~~
                ~~<xcal:related-to>~~
                    ~~<xcal:uid>43da0574-d00b-41e8-8a47-70767f63da78@examples.oasis-open.org</xcal:uid>~~
                ~~</xcal:related-to>~~
            ~~</xcal:properties>~~
        ~~</xcal:interval>~~
        ~~<xcal:interval>~~
                        <xcal:~~properties~~reltype>
                            <xcal:~~uid>~~text/>
                    ~~<~~    </xcal:reltype>
                    </xcal:parameters~~/>~~>
                    ~~<xcal:text>d586e62f-617b-4207-a937-9a0ec8d45b5e@examples.oasis-open.org</xcal:text>~~
                ~~</xcal:uid>~~
                ~~<xcal:related-to>~~
                    ~~<xcal:uid>d586e62f-617b-4207-a937-9a0ec8d45b5e@examples.oasis-open.org</xcal:uid>~~
                ~~</xcal:related-to>~~
            ~~</xcal:properties>~~
        ~~</xcal:interval>~~
        ~~<xcal:interval>~~
            ~~<xcal:properties>~~
                    <xcal:uid>bfe0040e-a5e0-4558-bbe1-a9207004a4cc</xcal:uid>
                </xcal:related-to>
            </xcal:properties>
        </xcal:interval>
        <xcal:interval>
            <xcal:properties>
                <xcal:uid>
                    ~~<xcal:parameters/>~~
                    <xcal:text>~~848af3d9-5b1b-4b4f-a353-39a4b2c857f9@examples.oasis-open.org~~5b1e382f-4ff0-43fe-9535-1d98ca2a852d</xcal:text>
                </xcal:uid>
                <xcal:related-to>
                    <xcal:~~uid>848af3d9-5b1b-4b4f-a353-39a4b2c857f9@examples.oasis-open.org</~~parameters>
                        <xcal:~~uid~~reltype>
                ~~</~~          <xcal:~~related-to>~~text/>
                        </xcal:~~properties~~reltype>
        ~~</xcal:interval>~~
        ~~<xcal:interval>~~
            ~~<xcal:properties>~~
                ~~<xcal:uid>~~
                    ~~<~~                 </xcal:parameters~~/>~~>
                    ~~<xcal:text>c088de06-770c-44b8-9abf-8dbfc6448ce8@examples.oasis-open.org</xcal:text>~~
                                    <xcal:uid>bfe0040e-a5e0-4558-bbe1-a9207004a4cc</xcal:uid>
                </xcal:related-to>
            </xcal:properties>
        </xcal:interval>
    </xcal:components>
</xcal:vcalendar>
                ~~<xcal:related-to>~~
                    ~~<xcal:uid>c088de06-770c-44b8-9abf-8dbfc6448ce8@examples.oasis-open.org</xcal:uid>~~
                ~~</xcal:related-to>~~
            ~~</xcal:properties>~~
        ~~</xcal:interval>~~
        ~~<xcal:interval>~~
```

```
~~                <xcal:properties>~~
~~                        <xcal:uid>~~
~~                                <xcal:parameters/>~~
~~                                <xcal:text>c0bc8725-383d-4019-96a0-~~
~~3a3a7e19fe83@examples.oasis-open.org</xcal:text>~~
~~                        </xcal:uid>~~
~~                        <xcal:related-to>~~
~~                                <xcal:uid>c0bc8725-383d-4019-96a0-~~
~~3a3a7e19fe83@examples.oasis-open.org</xcal:uid>~~
~~                        </xcal:related-to>~~
~~                </xcal:properties>~~
~~        </xcal:interval>~~
~~        <xcal:interval>~~
~~                <xcal:properties>~~
~~                        <xcal:uid>~~
~~                                <xcal:parameters/>~~
~~                                <xcal:text>2a4b8df8-47da-474b-9a2e-~~
~~771c3f8ef915@examples.oasis-open.org</xcal:text>~~
~~                        </xcal:uid>~~
~~                        <xcal:related-to>~~
~~                                <xcal:uid>2a4b8df8-47da-474b-9a2e-~~
~~771c3f8ef915@examples.oasis-open.org</xcal:uid>~~
~~                        </xcal:related-to>~~
~~                </xcal:properties>~~
~~        </xcal:interval>~~
~~        <xcal:interval>~~
~~                <xcal:properties>~~
~~                        <xcal:uid>~~
~~                                <xcal:parameters/>~~
~~                                <xcal:text>fad65470-0c03-4dca-8669-~~
~~90d1b49769c2@examples.oasis-open.org</xcal:text>~~
~~                        </xcal:uid>~~
~~                        <xcal:related-to>~~
~~                                <xcal:uid>fad65470-0c03-4dca-8669-~~
~~90d1b49769c2@examples.oasis-open.org</xcal:uid>~~
~~                        </xcal:related-to>~~
~~                </xcal:properties>~~
~~        </xcal:interval>~~
~~   </xcal:components>~~
~~</xcal:vcalendar>~~
```

~~This~~The Partition above shows a school schedule in which classes start one hour apart. Each class is for 50 minutes, and there is a 10 minute gap between each as students move between classes. Classes may not begin before the schedule, but they may start up to five minutes late.

~~3.3.5 Controlling Start Times in Service Advertisements~~

3.3.5 Notifying Partners of Process Availability

A Sequence has not been scheduled until it has a start ~~time~~date and ~~date.~~time. Sometimes it is useful to ~~control~~limit the possible start-times. For example, consider a service that is only available at 9:00 AM each day. It has not yet been scheduled, so ~~it's~~its dtStart is empty. The Vavailability object, expressed either in the ~~designated interval~~Designated Interval, or in the lineage of Gluons, is used to restrict this offering.

Example 3-14: Vavailability

```
<xcal:vavailability~~ xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"~~
~~xmlns:xcal="urn:ietf:params:xml:ns:icalendar-2.0">~~>
   <xcal:properties>
           <xcal:uid>
```

```
                <xcal:text>~~eed68bda-ec20-4814-b48b-
2ff75ee56821@examples.oasis-open.org~~9ae65e93-4c68-4811-aa10-
fcdbacb7ba79</xcal:text>
         </xcal:uid>
         <xcal:dtstart>
                <xcal:parameters>
                       <xcal:tzid>
                              <xcal:text>America/New_York</xcal:text>
                       </xcal:tzid>
                </xcal:parameters>
                <xcal:date-time>~~20110301T000000~~2011-03-
01T00:00:00</xcal:date-time>
         </xcal:dtstart>
         <xcal:dtend>
                <xcal:parameters>
                       <xcal:tzid>
                              <xcal:text>America/New_York</xcal:text>
                       </xcal:tzid>
                </xcal:parameters>
                <xcal:date-time>~~20110331T000000~~2011-03-
31T00:00:00</xcal:date-time>
         </xcal:dtend>
   </xcal:properties>
   <xcal:components>
         <xcal:available~~>~~
                ~~<~~ xs:type="xcal:~~properties>~~AvailableType">
                <xcal:properties>
                       <xcal:dtstart>
                       ~~<xcal:dtstart>~~
                              ~~<xcal:parameters>~~
                                    ~~<xcal:tzid>~~

   ~~<xcal:text>America/New_York</xcal:text>~~
                                    ~~</xcal:tzid>~~
                              ~~</xcal:parameters>~~
                              <xcal:date~~>20110301T090000~~-time>2011-03-
01T09:00:00</xcal:date-time>
                       </xcal:dtstart>
                       <xcal:dtend>
                       ~~</xcal:dtstart>~~
                       ~~<xcal:dtend>~~
                              ~~<xcal:parameters>~~
                                    ~~<xcal:tzid>~~

   ~~<xcal:text>America/New_York</xcal:text>~~
                                    ~~</xcal:tzid>~~
                              ~~</xcal:parameters>~~
                              <xcal:date~~>20110301T110000~~-time>2011-03-
01T11:00:00</xcal:date-time>
                       </xcal:dtend>
                       <xcal:rrule>
                              <xcal:recur>
                                    <xcal:freq>WEEKLY</xcal:freq>
                                    <xcal:byday>MO</xcal:byday>
                                    <xcal:byday>TU</xcal:byday>
                                    <xcal:byday>WE</xcal:byday>
                                    <xcal:byday>TH</xcal:byday>
                              </xcal:recur>
                       </xcal:rrule>
                </xcal:properties>
         </xcal:available>
         <xcal:available~~>~~ xs:type="xcal:AvailableType">
                <xcal:properties>
```

```
                        <xcal:dtstart>
                ~~<xcal:properties>~~
                        ~~<xcal:dtstart>~~
                                ~~<xcal:parameters>~~
                                        ~~<xcal:tzid>~~

   ~~<xcal:text>America/New_York</xcal:text>~~
                                        ~~</xcal:tzid>~~
                                ~~</xcal:parameters>~~
                                <xcal:date~~>20110301T150000~~-time>2011-03-
01T15:00:00</xcal:date-time>
                        </xcal:dtstart>
                        <xcal:dtend>
                                ~~<xcal:parameters>~~
                                        ~~<xcal:tzid>~~

   ~~<xcal:text>America/New_York</xcal:text>~~
                                        ~~</xcal:tzid>~~
                                ~~</xcal:parameters>~~
                                <xcal:date~~>20110301T160000~~-time>2011-03-
01T16:00:00</xcal:date-time>
                        </xcal:dtend>
                        <xcal:rrule>
                                <xcal:recur>
                                        <xcal:freq>WEEKLY</xcal:freq>
                                        <xcal:byday>FR</xcal:byday>
                                </xcal:recur>
                        </xcal:rrule>
                </xcal:properties>
        </xcal:available>
   </xcal:components>
</xcal:vavailability>
```

The ~~Vavailbility~~Vavailability above describes service availability for the month of March, 2011, i.e., it has a start date of March 1 and an end date of March 31. Within that period, there are two schedules, described by the two availability artifacts. The first specifies that starting on March 1, there is a window of 9-11 am, Eastern Time, on Monday, Tuesday, Wednesday, and Thursday each week. The second specifies another window of availability from 3:00 PM (15:00) to 4:00 PM (16:00) on Fridays. These schedules are each valid only through March 31, the dtEnd of the encompassing Vavailability. If neither date nor duration were specified, then the end of the schedules would be indefinite.

The example above uses daily schedules with a weekly recurrence. The full breadth of recurrence rules is described in **[iCalendar]**.

3.3.5.1 Combining a Gluon and Availability.

Consider the school schedule in the partition example in ~~Example 3-13~~Section 3.3.4 Optimizing the expression of a Partition that is used in several examples. The school has a single valid start time, at 8:00. The service can be refined by advertising its Availability as beginning at 9:00 on the first day. Availability re-occurs on a weekly schedule, only on the weekdays Monday, Tuesday, Thursday, and Friday. Furthermore, the schedule can only be invoked during the Fall semester, from September 1, to December 15.

With a ~~granularity~~Granularity of one hour set, the schedule can only begin on the time that the Availability begins, or at one hour intervals thereafter. If the Availability Window is only from 8:00 with a Duration of one hour, then the service is advertised only for a start at this hour.

The example below illustrates how to use the Vavailability object contained in a gluon to publish ~~availability~~Availability on a pre-existing sequence.

Example 3-15 Gluon publishing availability of pre-existing sequence

```
<xcal:gluon>
```

```
~~<xcal:gluon xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"~~
~~xmlns:xcal="urn:ietf:params:xml:ns:icalendar-2.0">~~
   <xcal:properties~~>~~/>
          ~~<xcal:uid>~~
                 ~~<xcal:parameters/>~~
                 ~~<xcal:text>4b8bcc8a-570a-4d23-8059-~~
~~3f55b090da35@examples.oasis-open.org</xcal:text>~~
          ~~</xcal:uid>~~
          ~~<xcal:related-to>~~
                 ~~<xcal:parameters>~~
                        ~~<xcal:reltype>~~
                               ~~<xcal:text>CHILD</xcal:text>~~
                        ~~</xcal:reltype>~~
                 ~~</xcal:parameters>~~
                 ~~<xcal:uid>c7496e78-6d71-4118-b42a-641f1efc02a9@examples.oasis-~~
~~open.org</xcal:uid>~~
          ~~</xcal:related-to>~~
   ~~</xcal:properties>~~
   <xcal:components>
          <xcal:vavailability>
                 <xcal:properties>
                        <xcal:uid>
                               <xcal:text>~~c83cf824-acb6-45ab-8bb0-~~
~~8648a37e92f6@examples.oasis-open.org~~7cf50215-e50e-46a9-9377-
797aff409ae3</xcal:text>
                        </xcal:uid>
                        <xcal:dtstart>
                               <xcal:parameters>
                                      <xcal:tzid>

   <xcal:text>America/New_York</xcal:text>
                                      </xcal:tzid>
                               </xcal:parameters>
                               <xcal:date-time>~~20110901T00000000~~2011-09-
01T00:00:00</xcal:date-time>
                        </xcal:dtstart>
                        <xcal:dtend>
                               <xcal:parameters>
                                      <xcal:tzid>

   <xcal:text>America/New_York</xcal:text>
                                      </xcal:tzid>
                               </xcal:parameters>
                               <xcal:date-time>~~201112017T00000000~~2011-12-
17T00:00:00</xcal:date-time>
                        </xcal:dtend>
                 </xcal:properties>
                 <xcal:components>
                        <xcal:available~~>~~ xs:type="xcal:AvailableType">
                               <xcal:properties>
                                      <xcal:dtstart>
                                             ~~<xcal:parameters>~~
                                                    ~~<xcal:tzid>~~

   ~~<xcal:text>America/New_York</xcal:text>~~
                                                    ~~</xcal:tzid>~~
                                             ~~</xcal:parameters>~~
                                             <xcal:date~~>20110901T08000000~~-
time>2011-09-01T08:00:00</xcal:date-time>
                                      </xcal:dtstart>
                                      <xcal:dtend>
                                             ~~<xcal:parameters>~~
                                                    ~~<xcal:tzid>~~

   ~~<xcal:text>America/New_York</xcal:text>~~
                                                    ~~</xcal:tzid>~~
                                             ~~</xcal:parameters>~~
                                             <xcal:date~~>20110901T09000000~~-
time>2011-09-01T09:00:00</xcal:date-time>
                                      </xcal:dtend>
                                      <xcal:rrule>
                                             <xcal:recur>
                                                    <xcal:freq>WEEKLY</xcal:freq>
                                                    <xcal:byday>MO</xcal:byday>
                                                    <xcal:byday>WE</xcal:byday>
                                                    <xcal:byday>FR</xcal:byday>
                                             </xcal:recur>
                                      </xcal:rrule>
                               </xcal:properties>
                        </xcal:available>
                 </xcal:components>
          </xcal:vavailability>
   </xcal:components>
</xcal:gluon>
```

In the example above, the general classroom schedule has been referenced by a new gluon, and established the availability for the Fall semester. The new gluon references the pre-existing gluon that establishes the sequence as a partition.

This double inheritance, in which a Sequence inherits from a ~~Calendar~~ Gluon which inherits from a ~~Calendar~~ Gluon is a useful pattern for advertising or scheduling a service.

3.3.6 Other Scheduling Scenarios

Sometimes, the invoker of a service is interested only in single Interval of the Sequence, but the entire Sequence is required. In the example below, the second Interval is advertised, i.e., the ~~Calendar~~ Gluon points to the second Interval. The first Interval might be a required ramp-period, during which the underlying process is “warming up”, and which may bring some lesser service to market during that ramp time. The ramp-down time at the end is similarly fixed. The entire Service offering is represented by the exposed (it has a public URI) ~~Calendar~~ Gluon.

Example 3-16: Standard Sequence with Ramp-Up and Ramp Down

```
~~<?xml version="1.0" encoding="utf-16"?>~~
~~<xcal:icalendar xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"~~
~~xmlns:xcal="urn:ietf:params:xml:ns:icalendar-2.0">~~
  <xcal:vcalendar~~ xs:type="xcal:VcalendarType">~~>
         <xcal:components>
                <xcal:gluon>
                       <xcal:properties>
                                    <xcal:uid>
                                           <xcal:text>~~26e1fa7e-aeac-429d-ab7a-~~
~~f6d92cf9afc2@examples.oasis-open.org~~fcd9ebc7-32f4-4a8c-9dbd-
23749e6b324e</xcal:text>
                       </xcal:uid>
                       <xcal:related-to>
                              <xcal:parameters>
          ~~</xcal:uid>~~
          ~~<xcal:related-to>~~
            ~~<xcal:parameters>~~
                                                  <xcal:reltype>

   <xcal:text>CHILD</xcal:text>
                                                  </xcal:reltype>
                                           </xcal:parameters>
```

```
                    <xcal:uid>~~429dddae-d6b8-418e-a897-d57c6c83052b@examples.oasis-open.org~~a4cde8b9-ed43-4ca7-9eb0-5a78a02d30b8</xcal:uid>
                  </xcal:related-to>
                  <xcal:dtstart>
                <xcal:parameters>
                  <xcal:tzid>
                    <xcal:text>America/New_York</xcal:text>
                  </xcal:tzid>
                </xcal:parameters>
~~            <xcal:parameters>~~
~~              <xcal:tzid>~~
~~                <xcal:text>America/New_York</xcal:text>~~
~~              </xcal:tzid>~~
~~            </xcal:parameters>~~
                    <xcal:date-time>~~20110315T08450000~~2011-05-28T08:45:00</xcal:date-time>
                  </xcal:dtstart>
                  <xcal:duration>
                    <xcal:duration>~~T2H~~PT2H</xcal:duration>
                  </xcal:duration>
~~          </xcal:duration>~~
                  <xcal:x-~~wscalendar~~wsCalendar-attach>
~~            <xcal:artifact>~~
~~              <xx:payload xmlns:xx="urn:not:a:real:artifact">~~
~~                <xx~~                    <payload:payload>
                    <payload:units>fortnights</payload:units>
                    <payload:quantity>~~14</xx~~34</payload:quantity>
~~              </xx:~~                  </payload:payload>
~~            </xcal:artifact>~~
                  </xcal:x-~~wscalendar~~wsCalendar-attach>
              </xcal:properties>
            <xcal:components/>
          </xcal:gluon>
          <xcal:interval>
            <xcal:properties>
              <xcal:uid>
                <xcal:text>~~00564cfe-975d-42a1-9f3a-eb4267e89350@examples.oasis-open.org~~c5473482-2ecb-4a9b-b53c-91ff5614dc40</xcal:text>
              </xcal:uid>
              <xcal:duration>
                <xcal:duration>~~T10M~~PT10M</xcal:duration>
              </xcal:duration>
            </xcal:properties>
          </xcal:interval>
          <xcal:interval>
            <xcal:properties>
              <xcal:uid>
                <xcal:text>a4cde8b9-ed43-4ca7-9eb0-5a78a02d30b8</xcal:text>
              </xcal:uid>
              <xcal:related-to>
                <xcal:parameters>
                  <xcal:reltype>
                    <xcal:text>FS</xcal:text>
                  </xcal:reltype>
                </xcal:parameters>
```

```
                             <xcal:uid>c5473482-2ecb-4a9b-b53c-
91ff5614dc40</xcal:uid>
                      </xcal:related-to>
                                    <xcal:x-~~wscalendar~~wsCalendar-attach>
            ~~<xcal:parameters />~~
            ~~<xcal:artifact>~~
              ~~<xx:~~                               <payload
~~xmlns:xx="urn:not:a:real:artifact">~~
                ~~<xx:warmUp>fixed content</xx:warmUp>~~
              ~~</xx~~:payload>
            ~~</xcal:artifact>~~

   <payload:units>furlongs</payload:units>
                                    <payload:quantity>11</payload:quantity>
                             </payload:payload>
                      </xcal:x-~~wscalendar~~wsCalendar-attach>
                </xcal:properties>
         </xcal:interval>
         <xcal:interval>
                <xcal:properties>
                      <xcal:uid>
        ~~</xcal:properties>~~
      ~~</xcal:interval>~~
      ~~<xcal:interval>~~
        ~~<xcal:properties>~~
          ~~<xcal:uid>~~
                                          <xcal:text>~~429dddac-d6b8-418e-a897-
d57c6c83052b@examples.oasis-open.org~~de2aa95b-f930-44d5-bc2b-
1356f8732879</xcal:text>
                      </xcal:uid>
                      <xcal:related-to>
                             <xcal:parameters>
                                    <xcal:reltype>
                                          <xcal:text>FS</xcal:text>
                                    </xcal:reltype>
                             </xcal:parameters>
          ~~</xcal:uid>~~
          ~~<xcal:related-to>~~
            ~~<xcal:parameters>~~
                                          <xcal:~~reltype>~~
                ~~<xcal:text>FS</xcal:text>~~
              ~~</xcal:reltype>~~
            ~~</xcal:parameters>~~
            ~~<~~uid>a4cde8b9-ed43-4ca7-9eb0-5a78a02d30b8</xcal:uid>~~00564cfe-975d-
42a1-9f3a-cb4267e89350@examples.oasis-open.org</xcal:uid>~~
                                    </xcal:related-to>
                                    <xcal:duration>
                             <xcal:duration>PT5M</xcal:duration>
                      </xcal:duration>
                      <xcal:x-~~wscalendar~~wsCalendar-attach>
            ~~<xcal:artifact>~~
              ~~<xx:payload xmlns:xx="urn:not:a:real:artifact">~~
                ~~<xx~~                              <payload:payload>
                                    <payload:units>furlongs</~~xx~~payload:units>
              ~~</xx:~~
   <payload:quantity>11</payload:quantity>
            ~~</xcal:artifact>~~
                                          </payload:payload>
                      </xcal:x-~~wscalendar~~wsCalendar-attach>
                </xcal:properties>
         </xcal:interval>
   </xcal:components>
</xcal:vcalendar>
```

```
        ~~</xcal:properties>~~
      ~~</xcal:interval>~~
      ~~<xcal:interval>~~
        ~~<xcal:properties>~~
          ~~<xcal:uid>~~
            ~~<xcal:text>59e717c3-7330-4cf3-8d57-f3239c4bc254@examples.oasis-~~
~~open.org</xcal:text>~~
          ~~</xcal:uid>~~
          ~~<xcal:duration>~~
            ~~<xcal:duration>T5M</xcal:duration>~~
          ~~</xcal:duration>~~
          ~~<xcal:related-to>~~
            ~~<xcal:parameters>~~
              ~~<xcal:reltype>~~
                ~~<xcal:text>FS</xcal:text>~~
              ~~</xcal:reltype>~~
            ~~</xcal:parameters>~~
            ~~<xcal:uid>429dddac-d6b8-418e-a897-d57c6c83052b@examples.oasis-~~
~~open.org</xcal:uid>~~
          ~~</xcal:related-to>~~
          ~~<xcal:x-wscalendar-attach>~~
            ~~<xcal:artifact>~~
              ~~<xx:payload xmlns:xx="urn:not:a:real:artifact">~~
                ~~<xx:coolDown>fixed content</xx:coolDown>~~
              ~~</xx:payload>~~
            ~~</xcal:artifact>~~
          ~~</xcal:x-wscalendar-attach>~~
        ~~</xcal:properties>~~
      ~~</xcal:interval>~~
    ~~</xcal:components>~~
  ~~</xcal:vcalendar>~~
~~</xcal:icalendar>~~
```

The underlying sequence has a fixed warm up and cool down (intervals 1 and 3). The Gluon shares a payload with Interval 2, which has no duration. Interval 2 inherits the quantity (14) and the duration (2H) from the Gluon.

If expressed all at once, the Gluon merely provides a handle for the Sequence. A more useful expression would have the Gluon separate, or perhaps inheriting its information from a market agreement. This enables the service interaction to express that Start Time, Duration and Quantity. All three are inherited, in this case, only by the Designated Interval.

3.4 Time Stamps

Time stamps are used everywhere in inter-domain service performance analysis and have particular use ~~in smart grids ~~to support event forensics. Time stamps ~~are often~~may be assembled and collated from events across multiple time zones and from multiple systems.

Different systems may track time and therefore record events with different levels of Tolerance. It is not unusual for a time~~-stamp~~-stamped event from a domain with ~~a ~~low Tolerance to appear to have occurred after one or more time-stamped events from a domain with high~~-~~ Tolerance~~ time-stamps that it caused.~~. A fully qualified time-stamp includes the ~~granularity~~Granularity measure.

Table 3-7: ~~Aspects~~Elements of Time Stamps

Time Stamp Element	~~Definition (Normative)~~	Note (Non-Normative)

Deleted

Time Stamp Element	~~Definition (Normative)~~	Note (Non-Normative)
~~timestamp~~Time Stamp		~~WS-Calendar:time~~ ~~A fully~~Fully qualified date and time of event~~.~~ ~~Mandatory.~~. ~~May include two objects as defined above.~~
~~accuracy~~Accuracy	~~A Duration defining the accuracy of the TimeStamp value.~~ ~~Mandatory.~~	Identifies whether an interval of a particular duration~~ (resp. starting at a particular time)~~ is indeed an interval of the mentioned duration plus or minus some number of milliseconds, seconds and minutes ~~(resp. an interval starting at the mentioned time plus or minus some number of milliseconds, seconds and minutes.~~.
~~timeStampRealm~~Time Stamp Realm	~~Of type Uri, shall identify the system where the TimeStamp value originated.~~ ~~The value of this element shall be set by:~~ ~~• The component at the realm border in a particular inter-domain interaction or,~~ ~~• By any component able to accurately set it within a system or sub-system.~~ ~~In the latter case, nothing prevents the component at the realm border to overwrite it without any notice.~~ ~~Optional.~~	~~A set of points~~Identifies the system where the TimeStamp value originated. A set of recordings originating from the same realm are reasonably synchronized. Within a realm, one can assume that time-stamped objects sorted by time are in the order of their occurrence. Between realms, this assumption is rebuttable. A system border is crossed in an interaction when the 2 communication partners are not synchronized based on the same time source. See the example below for more information.

Deleted

Deleted

Deleted

Time Stamp Element	~~Definition (Normative)~~	Note (Non-Normative)
~~leapSecondsKnown~~Leap Seconds Known	~~Xs:bool~~ ~~If True, shall indicate that the TimeStamp value takes into account all leap seconds occurred.~~ ~~Otherwise False. Optional.~~	Indicates that the time source of the sending device support leap seconds adjustments.
~~clockFailure~~Clock Failure	~~xs:bool~~ ~~If True, shall indicate a failure on the time source preventing the TimeStamp value issuer from setting accurate timestamps. Otherwise False.~~ ~~Mandatory.~~	Indicates that the time source of the sending device is unreliable. ~~The~~ This may put in doubt the advisability of direct comparison of this timestamp ~~should be ignored.~~information from this system and from foreign systems,
~~clockNotSynchronized~~Clock Not Synchronized	~~xs:bool~~ ~~If True, shall indicate the time source of the TimeStamp value issuer is not synchronized correctly, putting in doubt the accuracy of the timestamp.~~ ~~Mandatory.~~	Indicates that the time source of the sending device is not synchronized with the external UTC time source.
~~timeSourceAccuracy~~Time Source Accuracy	~~A Duration defining the accuracy of the time source used in the TimeStampRealm system.~~ ~~Optional.~~	Represents the time accuracy class of the time source of the sending device relative to the external UTC time source.

Deleted (

3.4.1 Time Stamp Realm Discussion

Within a single system, or synchronized system of systems, one can sort the temporal order of event by sorting them by TimeStamp. Determining the order of events is the first step of event forensics. This assumption does not apply when events are gathered across systems.

Different systems may not have synchronized time, or may synchronize time against different sources. This means different system clocks may drift apart. It may be that a later timestamp from one system occurred before an earlier timestamp in another. As this drift is unknown, it cannot be automatically corrected for without additional information.

The TimeStampRealm element identifies which system created an event time~~-~~ stamp. The TimeStampRealm identifies a source system in inter-domain interactions (a system of systems). For example: ~~http://SystemA.com and http://SystemB.com~~http://SystemA.com and http://SystemB.com identify 2 systems. This example assumes SystemA and SystemB do not have a common time source.

The TimeStampRealm can also be used to identify sub-systems in intra-domain interactions (sub-systems of a system). For example: ~~http://SystemA.com/SubSystem1~~http://SystemA.com/SubSystem1 and ~~http://SystemA.com/SubSystem2~~http://SystemA.com/SubSystem2 identify 2 subsystems of the same higher level system. In ~~case~~cases where the upper level SystemA does not have a global time source for synchronizing all of its sub-~~system~~systems, it can be useful to identify sub-systems in ~~such a way~~this manner.

~~4 PART TWO: Calendar Update and Synchronization with RESTful Services~~

~~4.1 Calendar Services~~

~~The Service interactions are built upon and make the same assumptions about structure as the CalDAV protocol defined in **[RFC4791]** and related specifications. It does NOT require nor assume the WebDAV nor CalDAV protocol but does make use of some of the same elements and structures in the CalDAV XML namespace.~~

~~Calendar resources, for example events and tasks are stored as named resources (files) inside special collections (folders) known as "**Calendar Collections**".~~

~~These services can be looked upon as a layer built on top of CalDAV and defines the basic operations which allow creation, retrieval, update and deletion. In addition, query, and free-busy operations are defined to allow efficient, partial retrieval of calendar data.~~

~~These services assume a degree of conformity with CalDAV is established such that services built in that manner do not have a significant mismatch. It is assumed that some WS-Calendar services will be built without any CalDAV support.~~

~~4.1.1 Overview of the protocol~~

~~The protocol is an HTTP based RESTfull protocol using a limited set of methods. Each request may be followed by a response containing status information.~~

~~The following methods are specified in the protocol description, PUT, POST, GET, DELETE. To avoid various issues with certain methods being blocked clients may use the X-HTTP-Method-Override: header to specify the intended operation. Servers SHOULD behave as if the named method was used.~~

```
~~POST /user/fred/calendar/ HTTP/1.1~~
~~...~~
~~X-HTTP-Method-Override: PUT~~
~~Properties~~
```

~~A service or resource will have a number of properties which describe the current state of that service or resource. These properties are accessed through a GET on the target resource or service with an ACCEPT header specifying application/xrd+xml. See Section 4.1.1.3.6~~

~~The following operations are defined by this specification:~~

- ~~Retrieval and update of service and resource properties~~
- ~~Creation of a calendar object~~
- ~~Retrieval of a calendar object~~
- ~~Update of a calendar object~~
- ~~Deletion of a calendar object~~
- ~~Query~~
- ~~Free-busy query~~

~~4.1.1.1 Calendar Object Resources~~

~~The same restrictions apply to Calendar Object Resources as specified in CalDAV **[RFC4791]** section 4.2. An additional constraint for CalWS is that no timezone specifications are transferred.~~

~~4.1.1.2 Timezone information~~

~~It is assumed that the client and server each have access to a full set of up to date timezone information. Timezones will be referenced by a timezone identifier from the full set of Olson data together with a set of well-known aliases defined **[TZDB]**. CalWS services may advertise themselves as timezone servers through the server properties object.~~

~~4.1.1.3 Issues not addressed by this specification.~~

~~A number of issues are not addressed by this version of the specification, either because they should be addressed elsewhere or will be addressed at some later date.~~

~~4.1.1.3.1 Access Control~~

~~It is assumed that the targeted server will set an appropriate level of access based on authentication. This specification will not attempt to address the issues of sharing or Access Control Lists (ACLs).~~

~~4.1.1.3.2 Provisioning~~

~~The protocol will not provide any explicit provisioning operations. If it is possible to authenticate or address a principal's calendar resources then they MUST be automatically created if necessary or appropriate~~

~~4.1.1.3.3 Copy/Move~~

~~These operations are not yet defined for this version of the CalWS protocol. Both operations raise a number of issues. In particular implementing a move operation through a series of retrievals, insertions and deletions may cause undesirable side-effects. Both these operations will be defined in a later version of this specification.~~

~~4.1.1.3.4 Creating Collections~~

~~We will not address the issue of creating collections within the address space. The initial set is created by provisioning.~~

~~4.1.1.3.5 Retrieving collections~~

~~This operation is currently undefined. A GET on a collection may fail or return a complete calendar object representing the collection.~~

~~4.1.1.3.6 Setting service and resource properties.~~

~~These operations are not defined in this version of the specification. In the future it will be possible to define or set the properties for the service or resources within the service.~~

~~4.1.1.4 CalWS Glossary~~

~~4.1.1.4.1 Hrefs~~

~~An href is a URI reference to a resource, for example~~

~~"http://example.org/user/fred/calendar/event1.ics".~~

~~The URL above reflects a possible structure for a calendar server. All URLs should be absolute or path-absolute following the rules defined in RFC4918 Section 8.3.~~

~~4.1.1.4.2 Calendar Object Resource~~

~~A calendar object resource is an event, meeting or a task. Attachments are resources but NOT calendar object resources. An event or task with overrides is a single calendar resource entity.~~

~~4.1.1.4.3 Calendar Collection~~

~~A folder only allowed to contain calendar object resources.~~

~~4.1.1.4.4 Scheduling Calendar Collection~~

~~A folder only allowed to contain calendar resources which is also used for scheduling operations. Scheduling events placed in such a collection will trigger implicit scheduling activity on the server.~~

~~4.1.1.4.5 Principal Home~~

~~The collection under which all the resources for a given principal are stored. For example, for principal "fred" the principal home might be "/user/fred/"~~

~~4.1.2 Error conditions~~

~~Each operation on the calendar system has a number of pre-conditions and post-conditions that apply.~~

~~A "precondition" for a method describes the state of the server that must be true for that method to be performed. A "post-condition" of a method describes the state of the server that must be true after that method has been completed. Any violation of these conditions will result in an error response in the form of a CalWS XML error element containing the violated condition and an optional description. \~~

~~Each method specification defines the preconditions that must be satisfied before the method can succeed. A number of post-conditions are generally specified which define the state that must exist after the execution of the operation. Preconditions and post-conditions are defined as error elements in the CalWS XML namespace.~~

~~4.1.2.1 Example: error with CalDAV error condition~~

```
~~<?xml version="1.0" encoding="utf-8"~~
~~      xmlns:CW="Error! Reference source not found.""~~
~~      xmlns:C="urn:ietf:params:xml:ns:caldav" ?>~~
~~<CW:error>~~
~~  <C:supported-filter>~~
~~    <C:prop-filter name="X-ABC-GUID"/>~~
~~  </C:supported-filter>~~
~~  <CW:description>Unknown property </CW:description>~~
~~</CW:error>~~
```

~~4.2 Properties and link relations~~

~~4.2.1 Property and relation-type URIs~~

~~In the XRD entity returned properties and related services and entities are defined by absolute URIs which correspond to the extended relation type defined in **[web linking]** Section 4.2. These URIs do NOT correspond to any real entity on the server and clients should not attempt to retrieve any data at that target.~~

~~Certain of these property URIs correspond to CalDAV preconditions. Each URL is prefixed by the CalWS relations and properties namespace http://docs.oasis-open.org/ns/wscal/calws. Those properties which correspond to CalDAV properties have the additional path element "caldav/", for example~~

```
~~http://docs.oasis-open.org/ns/wscal/calws/caldav/supported-calendar-data~~
```

~~corresponds to~~

```
~~CalDAV:supported-calendar-data~~
```

~~In addition to those CalDAV properties, the CalWS specification defines a number of other properties and link relations with the URI prefix of http://docs.oasis-open.org/ns/wscal/calws.~~

~~4.2.2 supported-features property.~~

~~http://docs.oasis-open.org/ns/wscal/calws/supported-features~~

~~This property defines the features supported by the target. All resources contained and managed by the service should return this property. The value is a comma separated list containing one or more of the following~~

- ~~calendar-access - the service supports all MUST requirements in this specification~~

```
~~<Property type="http://docs.oasis-open.org/ns/wscal/calws/supported-features"
          >calendar-access</Property>~~
```

~~4.2.3 max-attendees-per-instance~~

~~http://docs.oasis-open.org/ns/wscal/calws/max-attendees-per-instance~~

~~Defines the maximum number of attendees allowed per event or task.~~

~~4.2.4 max-date-time~~

~~http://docs.oasis-open.org/ns/wscal/calws/max-date-time~~

~~Defines the maximum date/time allowed on an event or task~~

~~4.2.5 max-instances~~

~~http://docs.oasis-open.org/ns/wscal/calws/max-instances~~

~~Defines the maximum number of instances allowed per event or task~~

~~4.2.6 max-resource-size~~

~~http://docs.oasis-open.org/ns/wscal/calws/max-resource-size~~

~~Provides a numeric value indicating the maximum size of a resource in octets that the server is willing to accept when a calendar object resource is stored in a calendar collection.~~

~~4.2.7 min-date-time~~

~~http://docs.oasis-open.org/ns/wscal/calws/min-date-time~~

~~Provides a DATE-TIME value indicating the earliest date and time (in UTC) that the server is willing to accept for any DATE or DATE-TIME value in a calendar object resource stored in a calendar collection.~~

~~4.2.8 description~~

~~http://docs.oasis-open.org/ns/wscal/calws/description~~

~~Provides some descriptive text for the targeted collection.~~

~~4.2.9 timezone-service relation.~~

~~http://docs.oasis-open.org/ns/wscal/calws/timezone-service~~

~~The location of a timezone service used to retrieve timezone information and specifications. This may be an absolute URL referencing some other service or a relative URL if the current server also provides a timezone service.~~

```
~~<Link rel="http://docs.oasis-open.org/ns/wscal/calws/calws/timezone-service"
             href="http://example.com/tz" />~~
```

~~4.2.10 principal-home relation.~~

~~http://docs.oasis-open.org/ns/wscal/calws/principal-home~~

~~Provides the URL to the user home for the currently authenticated principal.~~

```
~~<Link rel="http://docs.oasis-open.org/ns/wscal/calws/principal-home"
             href="http://example.com/user/fred" />~~
```

~~4.2.11 current-principal-freebusy relation.~~

~~http://docs.oasis-open.org/ns/wscal/calws/current-principal-freebusy~~

~~Provides the URL to use as a target for freebusy requests for the current authenticated principal.~~

```
~~<Link rel="http://docs.oasis-open.org/ns/wscal/calws/current-principal-freebusy"
             href="http://example.com/freebusy/user/fred" />~~
```

~~4.2.12 principal-freebusy relation.~~

~~http://docs.oasis-open.org/ns/wscal/calws/principal-freebusy~~

~~Provides the URL to use as a target for freebusy requests for a different principal.~~

```
~~<Link rel="http://docs.oasis-open.org/ns/wscal/calws/principal-freebusy"
             href="http://example.com/freebusy" />~~
```

~~4.2.13 child-collection relation.~~

~~http://docs.oasis-open.org/ns/wscal/calws/child-collection~~

~~Provides information about a child collections for the target. The href attribute gives the URI of the collection. The element should only have CalWS child elements giving the type of the collection, that is the CalWS:collection link property and the CalWS-calendar-collection link property. This allows clients to determine the structure of a hierarchical system by targeting each of the child collections in turn.~~

~~The xrd:title child element of the link element provides a description for the child-collection.~~

```
~~<Link rel="http://http://docs.oasis-open.org/ns/wscal/calws/child-collection"
        href="http://example.com/calws/user/fred/calendar">
  <Title xml:lang="en">Calendar</Title>
  <Property type="http://docs.oasis-open.org/ns/wscal/calws/collection"
               xsi:nil="true" />
  <Property type="http://docs.oasis-open.org/ns/wscal/calws/calendar-
collection"
               xsi:nil="true" />
</Link>~~
```

~~4.2.14 created link property~~

~~http://docs.oasis-open.org/ns/wscal/calws/created~~

~~Appears within a link relation describing collections or entities. The value is a date-time as defined in **RFC3339** Section 5.6~~

```
~~<Property type="http://docs.oasis-open.org/ns/wscal/calws/created"
             >1985-04-12T23:20:50.52Z</Property>~~
```

~~4.2.15 last-modified property~~

~~http://docs.oasis-open.org/ns/wscal/calws/last-modified~~

~~Appears within an xrd object describing collections or entities. The value is the same format as would appear in the Last-Modified header and is defined in **[RFC2616]**, Section 3.3.1~~

```
~~<Property type="http://docs.oasis-open.org/ns/wscal/calws/last-modified"
           >Mon, 12 Jan 1998 09:25:56 GMT</Property>~~
```

~~4.2.16 displayname property~~

~~http://docs.oasis-open.org/ns/wscal/calws/displayname~~

~~Appears within an xrd object describing collections or entities. The value is a localized name for the entity or collection.~~

```
~~<Property type="http://docs.oasis-open.org/ns/wscal/calws/displayname"
           >My Calendar</Property>~~
```

~~4.2.17 timezone property~~

~~http://docs.oasis-open.org/ns/wscal/calws/timezone~~

~~Appears within an xrd object describing collections. The value is a text timezone identifier.~~

```
~~<Property type="http://docs.oasis-open.org/ns/wscal/calws/timezone"
           >America/New_York</Property>~~
```

~~4.2.18 owner property~~

~~http://docs.oasis-open.org/ns/wscal/calws/owner~~

~~Appears within an xrd object describing collections or entities. The value is a server specific uri.~~

```
~~<Property type="http://docs.oasis-open.org/ns/wscal/calws/owner"
          >/principals/users/mike</Property>~~
```

~~4.2.19 collection link property~~

~~http://docs.oasis-open.org/ns/wscal/calws/collection~~

~~Appears within a link relation describing collections or entities. The property takes no value and indicates that this child element is a collection.~~

```
~~<Property type="http://docs.oasis-open.org/ns/wscal/calws/collection"
           xsi:nil="true" />~~
```

~~4.2.20 calendar-collection link property~~

~~http://docs.oasis-open.org/ns/wscal/calws/calendar-collection~~

~~Appears within a link relation describing collections or entities. The property takes no value and indicates that this child element is a calendar collection.~~

```
~~<Property type="http://docs.oasis-open.org/ns/wscal/calws/calendar-collection"
            xsi:nil="true" />~~
```

~~4.2.21 CalWS:privilege-set XML element~~

~~http://docs.oasis-open.org/ns/wscal/calws:privilege-set~~

~~Appears within a link relation describing collections or entities and specifies the set of privileges allowed to the current authenticated principal for that collection or entity.~~

```
~~<!ELEMENT calws:privilege-set (calws:privilege*)>
<!ELEMENT calws:privilege ANY>~~
```

~~Each privilege element defines a privilege or access right. The following set is currently defined~~

- ~~CalWS: Read - current principal has read access~~

- ~~CalWS: Write - current principal has write access~~

```
~~<calWS:privilege-set>~~
~~  <calWS:privilege><calWS:read></calWS:privilege>~~
~~  <calWS:privilege><calWS:write></calWS:privilege>~~
~~</calWS:privilege-set>~~
```

~~4.3 Retrieving Collection and Service Properties~~

~~Properties, related services and locations are obtained from the service or from service resources in the form of an XRD document as defined by [XRD-1.0].~~

~~Given the URL of a CalWS service a client retrieves the service XRD document through a GET on the service URL with an ACCEPT header specifying application/xrd+xml.~~

~~Retrieving resource properties is identical to obtaining service properties, that is, execute a GET on the target URL with an ACCEPT header specifying application/xrd+xml.~~

~~The service properties define the global limits and defaults. Any properties defined on collections within the service hierarchy override those service defaults. The service may choose to prevent such overriding of defaults and limits when appropriate.~~

~~4.3.1 Request parameters~~

- ~~None~~

~~4.3.2 Responses:~~

- ~~200: OK~~
- ~~403: Forbidden~~
- ~~404: Not found~~

~~4.3.3 Example - retrieving server properties:~~

```
~~>>Request~~

~~GET / HTTP/1.1~~
~~Host: example.com~~
~~ACCEPT:application/xrd+xml~~

~~>>Response~~
~~<XRD xmlns="http://docs.oasis-open.org/ns/xri/xrd-1.0"~~
~~          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">~~
~~   <Expires>1970-01-01T00:00:00Z</Expires>~~
~~   <Subject>http://example.com/calws</Subject>~~
~~   <Property type="http://docs.oasis-open.org/ns/wscal/calws/created"~~
~~         >1970-01-01</Property>~~

~~   <Link rel="http://docs.oasis-open.org/ns/wscal/calws/timezone-service"~~
~~          href="http://example.com/tz" />~~

~~   <calWS:privilege-set>~~
~~   <calWS:privilege><calWS:read></calWS:privilege>~~
~~   </calWS:privilege-set>~~

~~   <Link rel="http://docs.oasis-open.org/ns/wscal/calws/principal-home"~~
~~          type="collection"~~
~~          href="http://example.com/calws/user/fred">~~
~~   <Title xml:lang="en">Fred's calendar home</Title>~~
~~   </Link>~~

~~  <Link rel="http://docs.oasis-open.org/ns/wscal/calws/child-collection"~~
~~          type="calendar,scheduling"~~
~~          href="http://example.com/calws/user/fred/calendar">~~
~~  <Title xml:lang="en">Calendar</Title>~~
~~  </Link>~~

~~  <Property type="http://docs.oasis-open.org/ns/wscal/calws/max-instances"~~
~~          >1000</Property>~~

~~  <Property type="http://docs.oasis-open.org/ns/wscal/calws/max-attendees-~~
~~per-instance"~~
~~          >100</Property>~~

~~</XRD>~~
```

~~4.4 Creating Calendar Object Resources~~

~~Creating calendar object resources is carried out by a POST on the parent collection. The body of the request will contain the resource being created. The request parameter "action=create" indicates this POST is a create. The location header of the response gives the URL of the newly created object.~~

~~4.4.1 Request parameters~~

- ~~action=create~~

~~4.4.2 Responses:~~

- ~~201: created~~
- ~~403: Forbidden - no access~~

~~4.4.3 Preconditions for Calendar Object Creation~~

- ~~**CalWS:target-exists**: The target of a PUT must exist. Use POST to create entities and PUT to update them.~~
- ~~**CalWS:not-calendar-data:** The resource submitted in the PUT request, or targeted by a COPY or MOVE request, MUST be a supported media type (i.e., iCalendar) for calendar object resources;~~
- ~~**CalWS:invalid-calendar-data:** The resource submitted in the PUT request, or targeted by a COPY or MOVE request, MUST be valid data for the media type being specified (i.e., MUST contain valid iCalendar data);~~
- ~~**CalWS:invalid-calendar-object-resource:** The resource submitted in the PUT request, or targeted by a COPY or MOVE request, MUST obey all restrictions specified in Calendar Object Resources (e.g., calendar object resources MUST NOT contain more than one type of calendar component, calendar object resources MUST NOT specify the iCalendar METHOD property, etc.);~~
- ~~**CalWS:unsupported-calendar-component:** The resource submitted in the PUT request, or targeted by a COPY or MOVE request, MUST contain a type of calendar component that is supported in the targeted calendar collection;~~
- ~~**CalWS:uid-conflict:** The resource submitted in the PUT request, or targeted by a COPY or MOVE request, MUST NOT specify an iCalendar UID property value already in use in the targeted calendar collection or overwrite an existing calendar object resource with one that has a different UID property value. Servers SHOULD report the URL of the resource that is already making use of the same UID property value in the CalWS:href element~~
  ~~<!ELEMENT uid-conflict (CalWS:href)>~~

- ~~**CalWS:invalid-calendar-collection-location:** In a COPY or MOVE request, when the Request-URI is a calendar collection, the Destination-URI MUST identify a location where a calendar collection can be created;~~

- ~~**CalWS:exceeds-max-resource-size:** The resource submitted in the PUT request, or targeted by a COPY or MOVE request, MUST have an octet size less than or equal to the value of the CalDAV:max-resource-size property value on the calendar collection where the resource will be stored;~~

- ~~**CalWS:before-min-date-time:** The resource submitted in the PUT request, or targeted by a COPY or MOVE request, MUST have all of its iCalendar DATE or DATE-TIME property values (for each recurring instance) greater than or equal to the value of the CalDAV:min- date-time property value on the calendar collection where the resource will be stored;~~

- ~~**CalWS:after-max-date-time:** The resource submitted in the PUT request, or targeted by a COPY or MOVE request, MUST have all of its iCalendar DATE or DATE-TIME property values (for each recurring instance) less than the value of the CalDAV:max-date-time property value on the calendar collection where the resource will be stored;~~

- ~~**CalWS:too-many-instances:** The resource submitted in the PUT request, or targeted by a COPY or MOVE request, MUST generate a number of recurring instances less than or equal to the value of the CalDAV: max-instances property value on the calendar collection where the resource will be stored;~~

- ~~**CalWS:too-many-attendees-per-instance:** The resource submitted in the PUT request, or targeted by a COPY or MOVE request, MUST have a number of ATTENDEE properties on any one instance less than or equal to the value of the CalDAV:max-attendees-per-instance property value on the calendar collection where the resource will be stored;~~

~~4.4.4 Example - successful POST:~~

```
   ~~>>Request~~

   ~~POST /user/fred/calendar/?action=create HTTP/1.1~~
   ~~Host: example.com~~
   ~~Content-Type: application/xml+calendar; charset="utf-8"~~
   ~~Content-Length: ?~~

   ~~<?xml version="1.0" encoding="utf-8" ?>~~
   ~~<icalendar xmlns="urn:ietf:params:xml:ns:icalendar-2.0">~~
     ~~<vcalendar>~~
     ~~...~~
     ~~</vcalendar>~~
   ~~</icalendar>~~

   ~~>>Response~~

   ~~HTTP/1.1 201 Created~~
   ~~Location: http://example.com/user/fred/calendar/event1.ics~~
```

~~4.4.5 Example - unsuccessful POST:~~

```
   ~~>>Request~~

   ~~POST /user/fred/readcalendar/?action=create HTTP/1.1~~
   ~~Host: example.com~~
   ~~Content-Type: text/text; charset="utf-8"~~
   ~~Content-Length: ?~~

   ~~This is not an xml calendar object~~

```

```
~~   >>Response~~

~~   HTTP/1.1 403 Forbidden~~
~~   <?xml version="1.0" encoding="utf-8"~~
~~         xmlns:D="DAV:"~~
~~         xmlns:C="urn:ietf:params:xml:ns:caldav" ?>~~
~~   <D:error>~~
~~         <C:supported-calendar-data/>~~
~~         <D:description>Not an icalendar object</D:description>~~
~~   </D:error>~~
```

~~4.5 Retrieving resources~~

~~A simple GET on the href will return a named resource. If that resource is a recurring event or task with overrides, the entire set will be returned. The desired format is specified in the ACCEPT header. The default form is application/xml+calendar~~

~~4.5.1 Request parameters~~

- ~~none~~

~~4.5.2 Responses:~~

- ~~200: OK~~
- ~~403: Forbidden - no access~~
- ~~406 The requested format specified in the accept header is not supported.~~

~~4.5.3 Example - successful fetch:~~

```
~~>>Request~~

~~GET /user/fred/calendar/event1.ics HTTP/1.1~~
~~Host: example.com~~

~~>>Response~~

~~HTTP/1.1 200 OK~~
~~Content-Type: application/xml+calendar; charset="utf-8"~~
~~Content-Length: ?~~

~~<?xml version="1.0" encoding="utf-8" ?>~~
~~<icalendar xmlns="urn:ietf:params:xml:ns:icalendar-2.0">~~
~~  <vcalendar>~~
~~   ...~~
~~  </vcalendar>~~
~~</icalendar>~~
```

~~4.5.4 Example - unsuccessful fetch:~~

```
~~>>Request~~

~~PUT /user/fred/calendar/noevent1.ics HTTP/1.1~~
~~Host: example.com~~

~~>>Response~~

~~HTTP/1.1 404 Not found~~
```

~~4.6 Updating resources~~

~~Resources are updated with the PUT method targeted at the resource href. The body of the request contains a complete new resource which effectively replaces the targeted resource. To allow for optimistic locking of the resource use the if-match header.~~

~~When updating a recurring event all overrides and master must be supplied as part of the content.~~

~~Preconditions as specified in Section 4.4.3 are applicable.~~

~~4.6.1 Responses:~~

- ~~200: OK~~
- ~~304: Not modified - entity was modified by some other request~~
- ~~403: Forbidden - no access, does not exist etc. See error response~~

~~Example 4-1: Successful Update~~

```
~~>>Request~~

~~PUT /user/fred/calendar/event1.ics HTTP/1.1~~
~~Host: example.com~~
~~Content-Type: application/xml+calendar; charset="utf-8"~~
~~Content-Length: ?~~

~~<?xml version="1.0" encoding="utf-8" ?>~~
~~<icalendar xmlns="urn:ietf:params:xml:ns:icalendar-2.0">~~
~~  <vcalendar>~~
~~     ...~~
~~  </vcalendar>~~
~~</icalendar>~~

~~>>Response~~

~~HTTP/1.1 200 OK~~
```

~~Example 4-2: Unsuccessful Update~~

```
~~>>Request~~

~~PUT /user/fred/readcalendar/event1.ics HTTP/1.1~~
~~Host: example.com~~
~~Content-Type: application/xml+calendar; charset="utf-8"~~
~~Content-Length: ?~~

~~<?xml version="1.0" encoding="utf-8" ?>~~
~~<icalendar xmlns="urn:ietf:params:xml:ns:icalendar-2.0">~~
~~  <vcalendar>~~
~~    ...~~
~~  </vcalendar>~~
~~</icalendar>~~

~~>>Response~~

~~HTTP/1.1 403 Forbidden~~
~~Content-Type: application/xml; charset="utf-8"~~
~~Content-Length: xxxx~~

~~<?xml version="1.0" encoding="utf-8"~~
~~      xmlns:D="DAV:"~~
~~      xmlns:CW=" http://docs.oasis-open.org/ws-calendar/CalWS" ?>~~
```

```
~~<CW:error>~~
~~  <CW:target-exists/>~~
~~  <CW:description>Target of update must exist</C:description>~~
~~</CW:error>~~
```

~~4.7 Deletion of resources~~

~~Delete is defined in **[RFC 2616]** Section 9.7. In addition to conditions defined in that specification, servers must remove any references from the deleted resource to other resources. Resources are deleted with the DELETE method targeted at the resource URL. After a successful completion of a deletion a GET on that URL must result in a 404 - Not Found status.~~

~~4.7.1 Delete for Collections~~

~~Delete for collections may or may not be supported by the server. Certain collections are considered undeletable. On a successful deletion of a collection all contained resources to any depth must also be deleted.~~

~~4.7.2 Responses:~~

- ~~200: OK~~
- ~~403: Forbidden - no access~~
- ~~404: Not Found~~

~~4.8 Querying calendar resources~~

~~Querying provides a mechanism by which information can be obtained from the service through possibly complex queries. A list of icalendar properties can be specified to limit the amount of information returned to the client. A query takes the parts~~

- ~~Limitations on the data returned~~
- ~~Selection of the data~~
- ~~Optional timezone id for floating time calculations.~~

~~The current specification uses CalDAV multiget and calendar-query XML bodies as specified in **[RFC 4791]** with certain limitations and differences.~~

1. ~~The POST method is used for all requests, the action being identified by the outer element.~~
2. ~~While CalDAV servers generally only support [RFC 5545] and assume that as the default, the delivery format for CalWS will, by default, be [draft-xcal].~~
3. ~~The CalDAV query allows the specification of a number of DAV properties. Specification of these properties, with the exception of DAV:getetag, is considered an error in CalWS.~~
4. ~~The CalDAV:propnames element is invalid~~

~~With those differences, the CalDAV specification is the normative reference for this operation.~~

~~4.8.1 Limiting data returned~~

~~This is achieved by specifying one of the following~~

- ~~CalDAV:allprop return all properties (some properties are specified as not being part of the allprop set so are not returned)~~
- ~~CalDAV:prop An element which contains a list of properties to be returned . May only contain DAV:getetag and CalDAV:calendar-data~~

~~Of particular interest, and complexity, is the calendar-data property which can contain a time range to limit the range of recurrences returned and/or a list of calendar properties to return.~~

~~4.8.2 Pre/postconditions for calendar queries~~

~~The preconditions as defined in in **[RFC 4791]** Section 7.8 apply here. CalDav errors may be reported by the service when preconditions or postconditions are violated.~~

~~4.8.3 Example: time range limited retrieval~~

~~This example shows the time-range limited retrieval from a calendar which results in 2 events, one a recurring event and one a simple non-recurring event.~~

```
~~>> Request <<~~

~~POST /user/fred/calendar/ HTTP/1.1~~
~~Host: calws.example.com~~
~~Depth: 1~~
~~Content-Type: application/xml; charset="utf-8"~~
~~Content-Length: xxxx~~

~~<?xml version="1.0" encoding="utf-8" ?>~~
~~<C:calendar-query xmlns:D="DAV:"~~
~~              xmlns:C="urn:ietf:params:xml:ns:caldav">~~
~~  <D:prop>~~
~~    <D:getetag/>~~
~~    <C:calendar-data content-type="application/xml+calendar" >~~
~~      <C:comp name="VCALENDAR">~~
~~        <C:prop name="VERSION"/>~~
~~        <C:comp name="VEVENT">~~
~~          <C:prop name="SUMMARY"/>~~
~~          <C:prop name="UID"/>~~
~~          <C:prop name="DTSTART"/>~~
~~          <C:prop name="DTEND"/>~~
~~          <C:prop name="DURATION"/>~~
~~          <C:prop name="RRULE"/>~~
~~          <C:prop name="RDATE"/>~~
~~          <C:prop name="EXRULE"/>~~
~~          <C:prop name="EXDATE"/>~~
~~          <C:prop name="RECURRENCE-ID"/>~~
~~        </C:comp>~~
~~      </C:comp>~~
~~    </C:calendar-data>~~
~~  </D:prop>~~
~~  <C:filter>~~
~~    <C:comp-filter name="VCALENDAR">~~
~~      <C:comp-filter name="VEVENT">~~
~~        <C:time-range start="20060104T000000Z"~~
~~                      end="20060105T000000Z"/>~~
~~      </C:comp-filter>~~
~~    </C:comp-filter>~~
~~  </C:filter>~~
~~</C:calendar-query>~~

~~>> Response <<~~

~~HTTP/1.1 207 Multi-Status~~
~~Date: Sat, 11 Nov 2006 09:32:12 GMT~~
~~Content-Type: application/xml; charset="utf-8"~~
~~Content-Length: xxxx~~

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

```
~~<D:multistatus xmlns:D="DAV:"~~
               ~~xmlns:C="urn:ietf:params:xml:ns:caldav">~~
  ~~<D:response>~~
    ~~<D:href>http://cal.example.com/bernard/work/abcd2.ics</D:href>~~
    ~~<D:propstat>~~
      ~~<D:prop>~~
        ~~<D:getetag>"fffff-abcd2"</D:getetag>~~
        ~~<C:calendar-data content-type="application/xml+calendar" >~~
          ~~<xc:icalendar~~
             ~~xmlns:xc="urn:ietf:params:xml:ns:icalendar-2.0">~~
   ~~<xc:vcalendar>~~
     ~~<xc:properties>~~
      ~~<xc:calscale><text>GREGORIAN</text></xc:calscale>~~
      ~~<xc:prodid>~~
       ~~<xc:text>-//Example Inc.//Example Calendar//EN</xc:text>~~
      ~~</xc:prodid>~~
      ~~<xc:version><xc:text>2.0</xc:text></xc:version>~~
     ~~</xc:properties>~~
     ~~<xc:components>~~
      ~~<xc:vevent>~~
       ~~<xc:properties>~~
        ~~<xc:dtstart>~~
          ~~<xc:parameters>~~
            ~~<xc:tzid>US/Eastern<xc:tzid>~~
          ~~<xc:parameters>~~
          ~~<xc:date-time>20060102T120000</xc:date-time>~~
        ~~</xc:dtstart>~~
        ~~<xc:duration><xc:duration>PT1H</xc:duration></xc:duration>~~
        ~~<xc:summary>~~
         ~~<xc:text>Event #2</xc:text>~~
        ~~</xc:summary>~~
        ~~<xc:uid>~~
         ~~<xc:text>00959BC664CA650E933C892C@example.com</xc:text>~~
        ~~</xc:uid>~~
        ~~<xc:rrule>~~
          ~~<xc:recur>~~
            ~~<xc:freq>DAILY</xc:freq>~~
            ~~<xc:count>5</xc:count>~~
          ~~</xc:recur>~~
        ~~</xc:rrule>~~
       ~~</xc:properties>~~
      ~~</xc:vevent>~~

      ~~<xc:vevent>~~
       ~~<xc:properties>~~
        ~~<xc:dtstart>~~
          ~~<xc:parameters>~~
            ~~<xc:tzid>US/Eastern<xc:tzid>~~
          ~~<xc:parameters>~~
          ~~<xc:date-time>20060104T140000</xc:date-time>~~
        ~~</xc:dtstart>~~
        ~~<xc:duration><xc:duration>PT1H</xc:duration></xc:duration>~~
        ~~<xc:summary>~~
         ~~<xc:text>Event #2 bis</xc:text>~~
        ~~</xc:summary>~~
        ~~<xc:uid>~~
         ~~<xc:text>00959BC664CA650E933C892C@example.com</xc:text>~~
        ~~</xc:uid>~~
        ~~<xc:recurrence-id>~~
          ~~<xc:parameters>~~
            ~~<xc:tzid>US/Eastern<xc:tzid>~~
          ~~<xc:parameters>~~
          ~~<xc:date-time>20060104T120000</xc:date-time>~~
        ~~</xc:recurrence-id>~~
        ~~<xc:rrule>~~
          ~~<xc:recur>~~
            ~~<xc:freq>DAILY</xc:freq>~~
            ~~<xc:count>5</xc:count>~~
          ~~</xc:recur>~~
        ~~</xc:rrule>~~
       ~~</xc:properties>~~
      ~~</xc:vevent>~~

      ~~<xc:vevent>~~
       ~~<xc:properties>~~
        ~~<xc:dtstart>~~
          ~~<xc:parameters>~~
            ~~<xc:tzid>US/Eastern<xc:tzid>~~
          ~~<xc:parameters>~~
          ~~<xc:date-time>20060106T140000</xc:date-time>~~
        ~~</xc:dtstart>~~
        ~~<xc:duration><xc:duration>PT1H</xc:duration></xc:duration>~~
        ~~<xc:summary>~~
         ~~<xc:text>Event #2 bis bis</xc:text>~~
        ~~</xc:summary>~~
        ~~<xc:uid>~~
         ~~<xc:text>00959BC664CA650E933C892C@example.com</xc:text>~~
        ~~</xc:uid>~~
        ~~<xc:recurrence-id>~~
          ~~<xc:parameters>~~
            ~~<xc:tzid>US/Eastern<xc:tzid>~~
          ~~<xc:parameters>~~
          ~~<xc:date-time>20060106T120000</xc:date-time>~~
        ~~</xc:recurrence-id>~~
        ~~<xc:rrule>~~
          ~~<xc:recur>~~
            ~~<xc:freq>DAILY</xc:freq>~~
            ~~<xc:count>5</xc:count>~~
          ~~</xc:recur>~~
        ~~</xc:rrule>~~
       ~~</xc:properties>~~
      ~~</xc:vevent>~~
     ~~</xc:components>~~
    ~~</xc:vcalendar>~~
   ~~</xc:icalendar>~~
           ~~</C:calendar-data>~~
         ~~</D:prop>~~
         ~~<D:status>HTTP/1.1 200 OK</D:status>~~
       ~~</D:propstat>~~
     ~~</D:response>~~
     ~~<D:response>~~
       ~~<D:href>http://cal.example.com/bernard/work/abcd3.ics</D:href>~~
       ~~<D:propstat>~~
         ~~<D:prop>~~
           ~~<D:getetag>"fffff-abcd3"</D:getetag>~~
           ~~<C:calendar-data content-type="application/xml+calendar" >~~
             ~~<xcal:icalendar~~
                ~~xmlns:xc="urn:ietf:params:xml:ns:icalendar-2.0">~~
    ~~<xc:vcalendar>~~
     ~~<xc:properties>~~
      ~~<xc:calscale><text>GREGORIAN</text></xc:calscale>~~
      ~~<xc:prodid>~~
       ~~<xc:text>-//Example Inc.//Example Calendar//EN</xc:text>~~
      ~~</xc:prodid>~~
      ~~<xc:version><xc:text>2.0</xc:text></xc:version>~~
     ~~</xc:properties>~~
     ~~<xc:components>~~
      ~~<xc:vevent>~~
       ~~<xc:properties>~~
        ~~<xc:dtstart>~~
          ~~<xc:parameters>~~
            ~~<xc:tzid>US/Eastern<xc:tzid>~~
          ~~<xc:parameters>~~
          ~~<xc:date-time>20060104T100000</xc:date-time>~~
        ~~</xc:dtstart>~~
        ~~<xc:duration><xc:duration>PT1H</xc:duration></xc:duration>~~
        ~~<xc:summary>~~
         ~~<xc:text>Event #3</xc:text>~~
        ~~</xc:summary>~~
        ~~<xc:uid>~~
         ~~<xc:text>DC6C50A017428C5216A2F1CD@example.com</xc:text>~~
        ~~</xc:uid>~~
        ~~<xc:rrule>~~
          ~~<xc:recur>~~
            ~~<xc:freq>DAILY</xc:freq>~~
            ~~<xc:count>5</xc:count>~~
          ~~</xc:recur>~~
        ~~</xc:rrule>~~
       ~~</xc:properties>~~
      ~~</xc:vevent>~~
     ~~</xc:components>~~
    ~~</xc:vcalendar>~~
   ~~</xc:icalendar>~~
        ~~</C:calendar-data>~~
      ~~</D:prop>~~
      ~~<D:status>HTTP/1.1 200 OK</D:status>~~
    ~~</D:propstat>~~
  ~~</D:response>~~
~~</D:multistatus>~~
```

~~4.9 Free-busy queries~~

~~Free-busy queries are used to obtain free-busy information for a calendar-collection or principals. The result contains information only for events to which the current principal has sufficient access.~~

~~When targeted at a calendar collection the result is based only on the calendaring entities contained in that collection. When targeted at a principal free-busy URL the result will be based on all information which affect the principals free-busy status, for example availability.~~

~~The possible targets are:~~

- ~~A calendar collection URL~~
- ~~The XRD link with relation CalWS/current-principal-freebusy~~
- ~~The XRD link with relation CalWS/principal-freebusy with a principal given in the request.~~

~~The query follows the specification defined in **[FreeBusy Read URL]** with certain limitations. As an authenticated user to the CalWS service scheduling read-freebusy privileges must have been granted. As an unauthenticated user equivalent access must have been granted to unauthenticated access.~~

~~Freebusy information is returned by default as xcalendar vfreebusy components, as defined by **[draft-xcal]**. Such a component is not meant to conform to the requirements of VFREEBUSY components in **[RFC 5546]**. The VFREEBUSY component SHOULD conform to section "4.6.4 Free/Busy Component" of **[RFC 5545]**. A client SHOULD ignore the ORGANIZER field..~~

~~Since a Freebusy query can only refer to a single user, a client will already know how to match the result component to a user. A server MUST only return a single vfreebusy component.~~

~~4.9.1 ACCEPT header~~

~~The Accept header is used to specify the format for the returned data. In the absence of a header the data should be returned as specified in **[draft-xcal]**, that is, as if the following had been specified~~

```
~~ACCEPT: application/xml+calendar~~
```

~~4.9.2 URL Query Parameters~~

~~None of these parameters are required except for the conditions noted below. Appropriate defaults will be supplied by the server.~~

~~4.9.2.1 start~~

~~**Default**: The default value is left up to the server. It may be the current day, start of the current month, etc.~~

~~**Description:**Specifies the start date for the Freebusy data. The server is free to ignore this value and return data in any time range. The client must check the data for the returned time range.~~

~~**Format**:A profile of an **[RFC3339]** Date/Time. Fractional time is not supported. The server MUST support the expanded version e.g.~~

```
~~2007-01-02T13:00:00-08:00~~
```

~~It is up to the server to interpret local date/times.~~

~~**Example**:~~

```
~~2007-02-03T15:30:00-0800~~
~~2007-12-01T10:15:00Z~~
```

~~**Notes**: Specifying only a start date/time without specifying an end-date/time or period should be interpreted as in **[RFC 5545**]. The effective period should cover the remainder of that day.~~

~~Date-only values are disallowed as the server cannot determine the correct start of the day. Only UTC or date/time with offset values are permitted.~~

~~4.9.2.2 end~~

~~**Default**: Same as start~~

~~**Description**: Specifies the end date for the Freebusy data. The server is free to ignore this value.~~

~~**Format**: Same as start~~

~~**Example**: Same as start~~

~~4.9.2.3 period~~

~~**Default**: The default value is left up to the server. The recommended value is "P42D".~~

~~**Description**: Specifies the amount of Freebusy data to return. A client cannot specify both a period and an end date. Period is relative to the start parameter.~~

~~**Format**: A duration as defined in section 4.3.6 of **[RFC 5545]**~~

~~**Example**:~~

```
~~P42D~~
```

~~4.9.2.4 account~~

~~**Default**: none~~

~~**Description**: Specifies the principal when the request is targeted at the XRD CalWS/principal-freebusy. Specification of this parameter is an error otherwise.~~

~~**Format**: Server specific~~

~~**Example**:~~

```
~~fred~~
~~/principals/users/jim~~
~~user1@example.com~~
```

~~4.9.3 URL parameters - notes~~

~~The server is free to ignore the start, end and period parameters. It is recommended that the server return at least 6 weeks of data from the current day.~~

~~A client MUST check the time range in the VFREEBUSY response as a server may return a different time range than the requested range.~~

~~4.9.4 HTTP Operations~~

~~The server SHOULD return an Etag response header for a successful GET request targeting a Freebusy read URL. Clients MAY use the Etag response header value to do subsequent "conditional" GET requests that will avoid re-sending the Freebusy data again if it has not changed.~~

~~4.9.5 Response Codes~~

~~Below are the typical status codes returned by a GET request targeting a Free-busy URL. Note that other HTTP status codes not listed here might also be returned by a server.~~

- ~~200 OK~~
- ~~302 Found~~
- ~~400 Start parameter could not be understood / End parameter could not be understood / Period parameter could not be understood~~
- ~~401 Unauthorized~~
- ~~403 Forbidden~~
- ~~404 The data for the requested principal is not currently available, but may be available later.~~
- ~~406 The requested format in the accept header is not supported.~~
- ~~410 The data for the requested principal is no longer available~~
- ~~500 General server error~~

~~4.9.6 Examples~~

~~The following are examples of URLs used to retrieve Free-busy data for a user:~~

```
~~http://www.example.com/freebusy/user1@example.com?~~
~~start=2007-09-01T00:00:00-08:00&end=2007-09-31T00:00:00-08:00~~

~~http://www.example.com/freebusy/user1@example.com?~~
~~start=2007-09-01T00:00:00-08:00&end=2007-09-31T00:00:00-08:00~~

~~http://www.example.com/freebusy/user1@example.com~~

~~http://www.example.com/freebusy?user=user%201@example.com&~~
~~start=2008-01-01T00:00:00Z&end=2008-12-31T00:00:00Z~~
```

~~Some Request/Response Examples:~~

~~A URL with no query parameters:~~

```
~~>> Request <<~~
~~GET /freebusy/bernard/ HTTP/1.1~~
~~Host: www.example.com~~
```

```
~~>> Response <<~~
~~HTTP/1.1 200 OK~~
~~Content-Type: application/xml+calendar; charset="utf-8"~~
~~Content-Length: xxxx~~

~~<xc:icalendar xmlns:xc="urn:ietf:params:xml:ns:icalendar-2.0">~~
  ~~<xc:vcalendar>~~
    ~~<xc:properties>~~
      ~~<xc:calscale><text>GREGORIAN</text></xc:calscale>~~
      ~~<xc:prodid>~~
        ~~<xc:text>-//Example Inc.//Example Calendar//EN</xc:text>~~
      ~~</xc:prodid>~~
      ~~<xc:version><xc:text>2.0</xc:text></xc:version>~~
    ~~</xc:properties>~~
    ~~<xc:components>~~
      ~~<xc:vfreebusy>~~
        ~~<xc:properties>~~
          ~~<xc:uid>~~
            ~~<xc:text>76ef34-54a3d2@example.com</xc:text>~~
          ~~</xc:uid>~~
          ~~<xc:dtstart>~~
            ~~<xc:date-time>20060101T000000Z</xc:date-time>~~
          ~~</xc:dtstart>~~
          ~~<xc:dtend>~~
            ~~<xc:date-time>20060108T000000Z</xc:date-time>~~
          ~~</xc:dtend>~~
          ~~<xc:dtstamp>~~
            ~~<xc:date-time>20050530T123421Z</xc:date-time>~~
          ~~</xc:dtstamp>~~
          ~~<xc:freebusy>~~
            ~~<xc:parameters>~~
              ~~<xc:fbtype>BUSYTENTATIVE<xc:fbtype>~~
            ~~<xc:parameters>~~
            ~~<xc:period>20060102T100000Z/20060102T120000Z</xc:period>~~
          ~~</xc:freebusy>~~
          ~~<xc:freebusy>~~
            ~~<xc:period>20060103T100000Z/20060103T120000Z</xc:period>~~
          ~~</xc:freebusy>~~
          ~~<xc:freebusy>~~
            ~~<xc:period>20060104T100000Z/20060104T120000Z</xc:period>~~
          ~~</xc:freebusy>~~
          ~~<xc:freebusy>~~
            ~~<xc:parameters>~~
              ~~<xc:fbtype>BUSYUNAVAILABLE<xc:fbtype>~~
            ~~<xc:parameters>~~
            ~~<xc:period>20060105T100000Z/20060105T120000Z</xc:period>~~
          ~~</xc:freebusy>~~
          ~~<xc:freebusy>~~
            ~~<xc:period>20060106T100000Z/20060106T120000Z</xc:period>~~
          ~~</xc:freebusy>~~
      ~~</xc:vfreebusy>~~
    ~~</xc:components>~~
  ~~</xc:vcalendar>~~
~~<xc:icalendar>~~
```

~~A URL with start and end parameters:~~

```
~~>> Request <<~~
~~GET /freebusy/user1@example.com?start=2007-09-01T00:00:00-08:00&end=2007-09-~~
~~31T00:00:00-08:00~~
~~HTTP/1.1~~
~~Host: www.example.com~~

```

```
~~>> Response <<~~
~~HTTP/1.1 200 OK~~
~~Content-Type: application/xml+calendar; charset="utf-8"~~
~~Content-Length: xxxx~~

~~<xc:icalendar xmlns:xc="urn:ietf:params:xml:ns:icalendar-2.0">~~
  ~~<xc:vcalendar>~~
    ~~<xc:properties>~~
      ~~<xc:calscale><text>GREGORIAN</text></xc:calscale>~~
      ~~<xc:prodid>~~
        ~~<xc:text>-//Example Inc.//Example Calendar//EN</xc:text>~~
      ~~</xc:prodid>~~
      ~~<xc:version><xc:text>2.0</xc:text></xc:version>~~
    ~~</xc:properties>~~
    ~~<xc:components>~~
      ~~<xc:vfreebusy>~~
        ~~<xc:properties>~~
          ~~<xc:uid>~~
            ~~<xc:text>76ef34-54a3d2@example.com</xc:text>~~
          ~~</xc:uid>~~
          ~~<xc:dtstart>~~
            ~~<xc:date-time>20070901T000000Z</xc:date-time>~~
          ~~</xc:dtstart>~~
          ~~<xc:dtend>~~
            ~~<xc:date-time>20070931T000000Z</xc:date-time>~~
          ~~</xc:dtend>~~
          ~~<xc:dtstamp>~~
            ~~<xc:date-time>20050530T123421Z</xc:date-time>~~
          ~~</xc:dtstamp>~~
          ~~<xc:freebusy>~~
            ~~<xc:period>20070915T230000Z/20070916T010000Z</xc:period>~~
          ~~</xc:freebusy>~~
      ~~</xc:vfreebusy>~~
    ~~</xc:components>~~
  ~~</xc:vcalendar>~~
~~<xc:icalendar>~~
```

~~A URL for which the server does not have any data for that user:~~

```
~~>> Request <<~~
~~GET /freebusy/user1@example.com?start=2012-12-01T00:00:00-08:00&end=2012-12-31T00:00:00-08:00~~
~~HTTP/1.1~~
~~Host: www.example.com~~

~~>> Response <<~~
~~HTTP/1.1 404 No data~~
```

~~5~~4Conformance and Rules for WS-Calendar and Referencing Specifications

~~5.1~~4.1 Introduction

This section specifies conformance related to the ~~semantic model and RESTful Services. While the semantic model applies to all WS-Calendar implementations; the other conformance statements are relevant only to those using those services.~~

~~If~~information modelIf the implementer is merely using WS-Calendar as part of a larger business or service communication, they SHALL follow not only the semantic rules herein, but SHALL also conform to the rules for specifying inheritance in referencing standards.

~~5.2~~4.2 ~~Semantic~~ Conformance Rules for WS-Calendar

There are five kinds of conformance that must be addressed for WS-Calendar and specifications that reference WS-Calendar.

- Conformance to the ***inheritance rules*** in WS-Calendar, including the direction of inheritance
- ***Specific attributes*** for each type that MUST or MUST NOT be inherited~~.~~
- ***Conformance rules*** that Referencing Specifications MUST follow
- Description of ***Covarying attributes*** with respect to the Reference Specification
- ***Semantic Conformance*** for the information within the artifacts exchanged~~.~~

We address each of these in the following sections~~.~~

~~5.2.1~~4.2.1 Inheritance in WS-Calendar

In this section we define rules that define inheritance including direction.

I1: Proximity Rule Within a given lineage, inheritance is evaluated though each Parent to the Child before what the Child bequeaths is evaluated.

I2: Direction Rule Intervals MAY inherit attributes from the nearest gluon subject to the Proximity Rule and Override Rule, provided those attributes are defined as Inheritable.

I3: Override Rule If and only if there is no value for a given attribute of a Gluon or Interval, that Gluon or Interval SHALL inherit the value for that attribute from its nearest Ancestor in conformance to the Proximity Rule.

I4: Comparison Rule Two Sequences are equivalent if a comparison of the respective Intervals succeeds as if each Sequence were fully Bound and redundant Gluons are removed.

I5: Designated Interval Inheritance [To facilitate composition of Sequences] the Designated Interval in the ultimate Ancestor of a Gluon is the Designated Interval of the composed Sequence.~~[8]~~ Special conformance rules for Designated Intervals apply only to the Interval linked from the Designator Gluon.

I6: Start Time Inheritance When a start time is specified through inheritance, that start time is inherited only by the Designated Interval; the start time of all other Intervals are computed through the durations and temporal; relationships within the Sequence. The ~~designated~~Designated Interval is the Interval whose parent is at the end of the lineage.

~~5.2.2~~4.2.2 Specific Attribute Inheritance ~~in WS-Calendar~~

In WS-Calendar the following attributes MUST be inherited in conformance to the Rules (same for Gluons and Intervals):

- dtStart
- dtEnd
- ~~duration~~
- ~~designatedInterval~~Duration
- Designated Interval (Gluon, special upward inheritance rule)
- ~~performance~~
- ~~performanceInterval~~
- Tolerance

In WS-Calendar the following attributes MUST NOT be inherited

- UID (Gluons and Intervals)
- Temporal Relationships (between Intervals)

4.2.3 General Conformance Issues

This specification is general purpose. Standards that claim conformance to this specification may need to restrict the variability inherent in the expressions of Date and Time to improve interoperation within their own interactions. Aspects of Date and Time that may reward attention and conformance statements include:

- **Precision** – Does the conforming specification express time in Hours or in milliseconds. Consider a standard format recommendation.
- **Time Zones and UTC** – Business interactions have a "natural" choice of local, time zone, or UTC based expression of time. Intents may be local, as they tie to the business processes that drive them. Tenders may be Time-zone based, as they are driven by the local business process, but may require future action across changes in time and in time zone. Transaction recording may demand UTC, for complete unambiguity. The specification cannot require one or another, but particular business processes may require appropriate conformance statements.

~~[8] We are assuming here that Sequences can be composed to form new Sequences. This needs detailed discussion as the rules for Designated Intervals cannot easily be applied to a Sequence of Sequences.~~

- **Business Purpose** – Because WS-Calendar is general purpose, it does not distinguish between different exchanges that may have different purposes. For example, a general indication of capability and/or timeliness may be appropriate for a market tender, and an unanchored Sequence may be appropriate. In the same specification, performance execution could require merely the Gluon to Anchor the Interval. If the distinction between Unanchored and Anchored Interval is critical for a set of interactions, the referencing specification SHALL indicate the proper form for a given exchange.

4.2.4 Covarying Elements

Some elements of WS-Calendar objects may be **covarying**, meaning that they change together. Such elements are treated as a single element for inheritance, they are either inherited together or the child keeps its current values intact. This becomes important if one or more of a covarying set have default values. In that case, if any are present, then inheritance should deem they are all present, albeit some perhaps in their default values.

~~5.2.3~~4.2.5 Conformance of Intervals~~ in WS-Calendar~~

~~5.2.3.1~~4.2.5.1 Intervals

WS-Calendar Intervals SHALL have a Duration.

Intervals MAY have a ~~StartTime.~~Start Time.

Intervals SHALL ~~NOT include an END time.~~have a Duration AND a dtStart OR a dtEnd. If a non-compliant Interval is received with ~~an END time, it may~~both a dtStart and a dtEnd, then the dtEnd SHALL be ignored.

Within a Sequence, a maximum of a single Interval MAY have a dtStart or a dtEnd.

~~5.2.3.2~~4.2.5.2 Other Elements

A ~~performance~~Tolerance Property component SHALL ~~not~~NOT include Start, Stop, and Duration elements. Two out of the three elements is acceptable, but not three.

In Partitions, the Description, Summary and Priority of each Interval SHALL be excluded.

A~~ Calendar~~ Gluon may have either a dtStart or a dtEnd, but may not have both.

~~5.2.4~~4.2.6 Conformance of Bound Intervals and Sequences~~ in WS-Calendar~~

Actionable services require Bound Intervals as part of a Bound Sequence. Services may include Intervals that ~~arenot~~are not bound for informational or negotiation purposes. Some of these are modeled and described as constraints in the UML models that have been produced separately.

- Intervals SHALL have values assigned for dtStart and duration, either explicitly or through inheritance

- Intervals SHALL have no value assigned for dtEnd[9]
- Within a Sequence at most the Designated Interval may have dtStart and duration with a value specified or inherited.~~[10]~~
- If Sequences are composed to create other Sequences, then the Designated Intervals within the composing Sequence are ignored.
- Any specification claiming conformance to WS-Calendar MUST satisfy all of the following conditions:
  - Follow the same style of inheritance (per the Rules)
  - Specify attribute inheritability in the specification claiming conformance
  - Specify whether certain sets of elements must be inherited as a group or specify that all elements can be inherited or not on an individual basis

~~5.3 Conformance Rules for RESTful Services~~

~~Still to come~~

~~5.4~~4.3 Conformance Rules for Specifications Claiming Conformance to WS-Calendar

Specifications that claim conformance to WS-Calendar SHALL specify inheritance rules for use within their specification. These rules SHALL NOT violate override the Proximity, Direction, or Override Rules. If the specification includes covariant elements, those elements ~~SHAL~~SHALL be clearly designated in the specification.

~~Specifications that normatively reference and claim conformance with WS-Calendar SHALL define the business meaning of zero duration Intervals.~~

Specifications that normatively reference and claim conformance with WS-Calendar SHALL define the business meaning of zero duration Intervals.

4.4 Security Considerations

Part 1 of WS-Calendar describes an informational model. Specifications claiming conformance with WS-Calendar may use the schedule and interval communication as but a small part of their overall communications.

[9] ~~While VTODO objects allow for all three of dtStart, dtEnd, and duration, the scheduling use for automation is simpler if only dtStart and duration are used.~~

~~[10] Note that composition of Sequences to create other Sequences raises issues both of inheritance direction and the meaning of subSequences. We suggest an approach of ignoring Designated Intervals with respect to the composed Sequence as simpler than having the new subSequences change form and not be reusable.~~

Communications that claim conformance to this specification should select the communication and the well-known methods to secure that communication appropriate to the information exchanged and paying heed to the costs of both communication failure and of inappropriate disclosure. To the extent that normal schedule servers are used, the capabilities of security of those systems should be considered as well. Those concerns are out of scope for this specification.

Specifications which do not use the REST or SOAP interactions face similar concerns in designing the authentication, authorization, interactions, and storage of the information artifacts produced. Such concerns are out of scope within this general model.

Acknowledgements

The following individuals have participated in the creation of this specification and are gratefully acknowledged:

Participants:

Bruce Bartell, Southern California Edison
Brad Benson, Trane
Edward Cazalet, Individual
Toby Considine, University of North Carolina at Chapel Hill
William Cox, Individual
Sharon Dinges, Trane
Mike~~ Douglas~~, Douglass, Rensselaer Polytechnic Institute
Craig Gemmill, Tridium, Inc.
Girish Ghatikar, Lawrence Berkeley National Laboratory
Gerald Gray, Southern California Edison
David Hardin, ENERNOC
Gale Horst, Electric Power Research Institute (EPRI)
Gershon Janssen, Individual
Ed Koch, Akuacom Inc.
Benoit Lepeuple, LonMark International*
Carl Mattocks, CheckMi*
Robert Old, Siemens AG
Alexander Papaspyrou, Technische Universitat Dortmund
Joshua Phillips, ISO/RTO Council (IRC)
Jeremy J. Roberts, LonMark International
David Thewlis, CalConnect

The Calendaring and Scheduling Consortium (CalConnect) TC-XML committee worked closely with WS-Calendar Technical Committee, bridging to developing IETF standards and contributing the services definitions that make up Services in Section 4. The Technical Committee gratefully acknowledges their assistance and cooperation as well. Contributors to TC XML include:

Cyrus Daboo, Apple
Mike ~~Douglas~~Douglass, Rensselaer Polytechnic Institute
Steven Lees, Microsoft
Tong Li, IBM

An Introduction to Internet Calendaring

The WS-Calendar Technical Committee thanks CalConnect for contributing this overview of iCalendar and its use.

icalendar

History

The iCalendar specification was first produced by the IETF in 1998 as RFC 2445 [1]. Since then it has become the dominant standard for calendar data interchange on the internet and between devices (desktop computers, mobile phones etc.). The specification was revised in 2009 as RFC 5545 [4].

Alongside iCalendar is the iTIP specification (RFC 2446 [2] and revised as RFC 5546[5]) that defines how iCalendar is used to carry out scheduling operations (for example, how an organizer can invite attendees to a meeting and receive their replies). This forms the basis for email-based scheduling using iMIP (the specification that describes how to use iTIP with email - RFC ~~2447~~6047 [3]).

iCalendar itself is a text-based data format. However, an XML format is also available, providing a one-to-one mapping to the text format (draft [7]).

iCalendar data files typically have a .ics file name extension. Most desktop calendar clients can import or export iCalendar data, or directly access such data over the Internet using a variety of protocols.

Data model

The iCalendar data format has a well defined data model. "iCalendar objects" encompass a set of "iCalendar ~~components~~Components" each of which contains a set of "iCalendar properties" and possibly other sub-~~components.~~Components. An iCalendar property consists of a name, a set of optional parameters (specified as "key-value" pairs) and a value.

iCalendar ~~components~~Components include:

"VEVENT" which represents an event

"VTODO" which represents a task or to-do

"VJOURNAL" which represents a journal entry

"VFREEBUSY" which represents periods of free or busy time information

"VTIMEZONE" which represents a timezone definition (timezone offset and daylight saving rules)

"VALARM" is currently the only defined sub-~~component~~Component and is used to set alarms or reminders on events or tasks.

Properties include:

"DTSTART" which represents a start time for a ~~component~~Component

"DTEND" which represents an end time for a ~~component~~Component

"SUMMARY" which represents a title or summary for a ~~component~~Component

"RRULE" which can specify rules for repeating events or tasks (for example, every day, every week on Tuesdays, etc.)

"ORGANIZER" which represents the calendar user who is organizing an event or assigning a task

"ATTENDEE" which represents calendar users attending an event or assigned a task

In addition to this data model and the pre-defined properties, the specification defines how all those are used together to define the semantics of calendar objects and scheduling. The semantics are basically a set of rules stating how all the ~~components~~Components and properties are used together to ensure that all iCalendar products can work together to achieve good interoperability. For example, a rule requires that all events must have one and only one "DTSTART" property. The most important part of the iCalendar specification is the semantics of the calendaring model that it represents. The use of text or XML to encode those is secondary.

Scheduling

The iTIP specification defines how iCalendar objects are exchanged in order to accomplish the key task needed to schedule events or tasks. An example of a simple workflow is as follows:

1. To schedule an event, an organizer creates the iCalendar object representing the event and adds calendar users as attendees.
2. The organizer then sends an iTIP "REQUEST" message to all the attendees.
3. Upon receipt of the scheduling message, each attendee can decide whether they want to attend the meeting or not.
4. Each attendee can then respond back to the organizer using an iTIP "REPLY" message indicating their own attendance status.

iTIP supports other types of scheduling messages, for example, to cancel meetings, add new instances to a repeating meeting, etc.

Extensibility

iCalendar was designed to be extensible, allowing for new ~~components~~Components, properties and parameters to be defined as needed. A registry exists to maintain the list of standard extensions with references to their definitions to ensure anyone can use them and work well with others.

Calendar data access and exchange protocols

Internet Calendar Subscriptions

An Internet calendar subscription is simply an iCalendar data file made available on a web server. Users can use this data in two ways:

The data can be downloaded from the web server and then imported directly into an iCalendar aware client. This solution works well for calendar data that is not likely to change over time (for example the list of national holidays for the next year).

Calendar clients that support "direct" subscriptions can use the URL to the calendar data on the web server to download the calendar data themselves. Additionally, the clients can check the web server on a regular basis for updates to the calendar data, and then update their own cached copy of it. This allows calendar data that changes over time to be kept synchronized.

CalDAV

CalDAV is a calendar access protocol and is defined in RFC 4791 [6]. The protocol is based on WebDAV which is an extension to HTTP that provides enhanced capabilities for document management on web servers.

CalDAV is used in a variety of different environments, ranging from very large internet service providers, to large and small corporations or institutions, and to small businesses and individuals.

CalDAV clients include desktop applications, mobile devices and browser-based solutions. It can also be used by "applets", for example, a web page panel that displays a user's upcoming events.

One of the key aspects of CalDAV is its data model. Simply put, it defines a "calendar home" for each calendar user, within which any number of "calendars" can be created. Each "calendar" can contain any number of iCalendar objects representing individual events, tasks or journal entries. This data model ensures that clients and servers can interoperate well.

In addition to providing simple operations to read, write and delete calendar data, CalDAV provides a querying mechanism to allow clients to fetch calendar data matching specific criteria. This is commonly used by clients to do "time-range" queries, i.e., find the set of events that occur within a given start/end time period.

CalDAV also supports access control allowing for features such as delegated calendars and calendar sharing.

CalDAV also specifies how scheduling operations can be done using the protocol. Whilst it uses the semantics of the iTIP protocol, it simplifies the process by allowing simple calendar data write operations to trigger the sending of scheduling messages, and it has the server automatically process the receipt of scheduling messages. Scheduling can be done with other users on the CalDAV server or with calendar users on other systems (via some form of "gateway").

ActiveSync/SyncML

ActiveSync and SyncML are technologies that allow multiple devices to synchronize data with a server, with calendar data being one of the classes of data supported. These have typically been used for low-end and high-end mobile devices.

CalWS

CalWS is a web services calendar access API developed by The Calendaring and Scheduling Consortium and the OASIS organization, to be used as part of the Oasis WS-Calendar standard. It provides an API to access and manipulate calendar data stored on a server. It follows a similar data model to CalDAV and has been designed to co-exist with a CalDAV service offering the same data.

iSchedule

iSchedule is a protocol to allow scheduling between users on different calendaring systems and across different internet domains. It transports iTIP scheduling messages using HTTP between servers. Servers use DNS and various security mechanisms to determine the authenticity of messages received.

It has been specifically designed to be independent of any calendar system in use at the endpoints, so that it is compatible with many different systems. This allows organizations with different calendar systems to exchange scheduling messages with each other, and also allows a single organization with multiple calendar systems (for example due to mergers, or different departmental requirements) to exchange scheduling messages between users of each system.

References

[1] https://datatracker.ietf.org/doc/rfc2445/ : ‘Internet Calendaring and Scheduling Core Object Specification’

[2] https://datatracker.ietf.org/doc/rfc2446/ :‘iCalendar Transport-Independent Interoperability Protocol’

[3] https://datatracker.ietf.org/doc/~~rfc2447~~rfc6047/ : ‘iCalendar Message-Based Interoperability Protocol’

[4] https://datatracker.ietf.org/doc/rfc5545/ :‘Internet Calendaring and Scheduling Core Object Specification’

[5] https://datatracker.ietf.org/doc/rfc5546/ : ‘iCalendar Transport-Independent Interoperability Protocol’

[6] https://datatracker.ietf.org/doc/rfc4791/ : ‘Calendaring Extensions to WebDAV’

[7] https://datatracker.ietf.org/doc/draft-daboo-et-al-icalendar-in-xml/ : ‘xCal: The XML format for iCalendar’

Overview of WS-Calendar, its Antecedents and its Use

iCalendar has long been the predominant message format for an Internet user to send meeting requests and tasks to other Internet users by email. The recipient can respond to the sender easily or counter propose another meeting date/time. iCalendar support is built into all major email systems and email clients. While SMTP is the predominant means to transport iCalendar messages, protocols including WebDAV and SyncML are used to transport collections of iCalendar information. No similar standard for service interactions has achieved similar widespread use.

The Calendar and Scheduling Consortium (CalConnect), working within the IETF, updated the iCalendar standard in the summer of 2009 to support extension ([RFC5545]). In 2010, the same group defined [XCAL], a canonical XML serialization for iCalendar, currently (08/21/2008) on the recommended standards track within the IETF. This specification supports extensions, including handling non-standard, i.e., non-iCalendar, data during message storage and retrieval.

WS-Calendar builds on this work, and consists of extensions to the vocabulary of iCalendar, along with standard services to extend calendaring and scheduling into service interactions. iCalendar consists of a number of fields that support the delivery, update, and synchronization of if calendar messages and a list of ~~components.~~Components. The ~~components~~Components can specify defined relationships between each other.

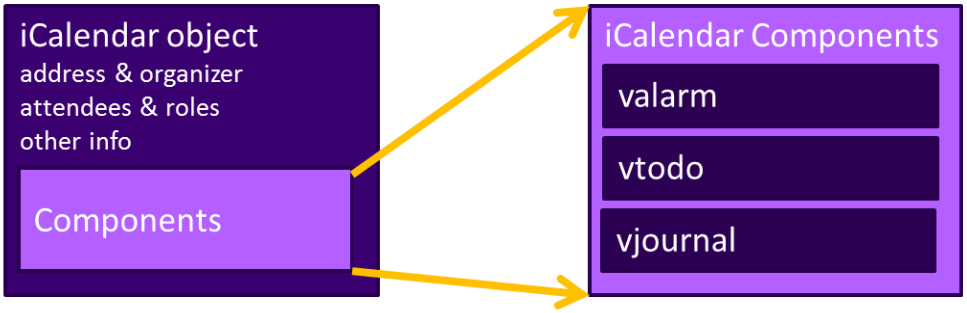
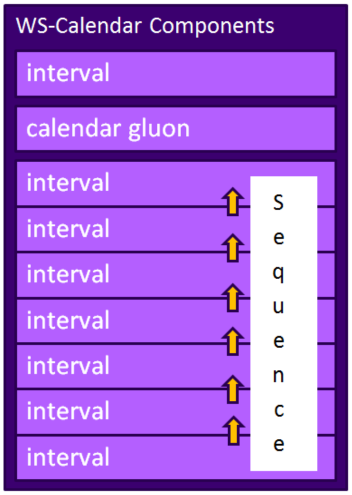


Figure 1: iCalendar overview

WS-Calendar defines the Interval, a profile of the ~~vtodo component~~VTODO Component requiring only a duration and an artifact to define service delivery and performance. WS-Calendar also defines the CalendarGluon ~~component~~Component, a container for holding only a service delivery and performance artifact, to associate with a ~~component~~Component or group of ~~components~~Components.



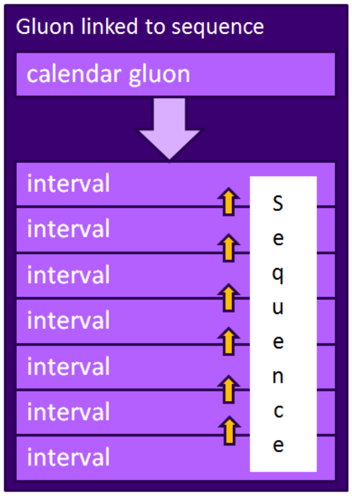


Figure 2: WS-Calendar and EMIX

A set of Intervals that have defined temporal relationships is a Sequence. Temporal relationships express how the occurrence of one Interval is related to another. For example, Interval B may begin 10 minutes after Interval A completes, or Interval D may start 5 minutes after Interval C starts. ~~An Calendar~~A Gluon linked to a Sequence defines service performance for all Intervals in the Sequence. Because each Interval has its own service performance contract, specifications built on WS-Calendar can define rules for inheritance and over-rides with a Sequence.

The Partition is a sub-class of a Sequence in which all Intervals follow consecutively with no lag time. Intervals in a Partition normally have the same Duration, but WS-Calendar does support overriding the duration on an individual basis.

Scheduling Sequences

A Sequence is a general pattern of behaviors and results that does not require a specific schedule. A publishing service may advertise a Sequence with no schedule, i.e., no specific time for performance. When the Sequence is invoked or contracted, a specific performance time is added. In the original iCalendar ~~components~~Components, this would add the starting date and time (dtStart) to the ~~component.~~Component. In WS-Calendar, we add the starting date and time only to the first Interval of a Sequence; the performance times for all other Intervals in the Sequence are derived from that one start time.

Academic Scheduling example

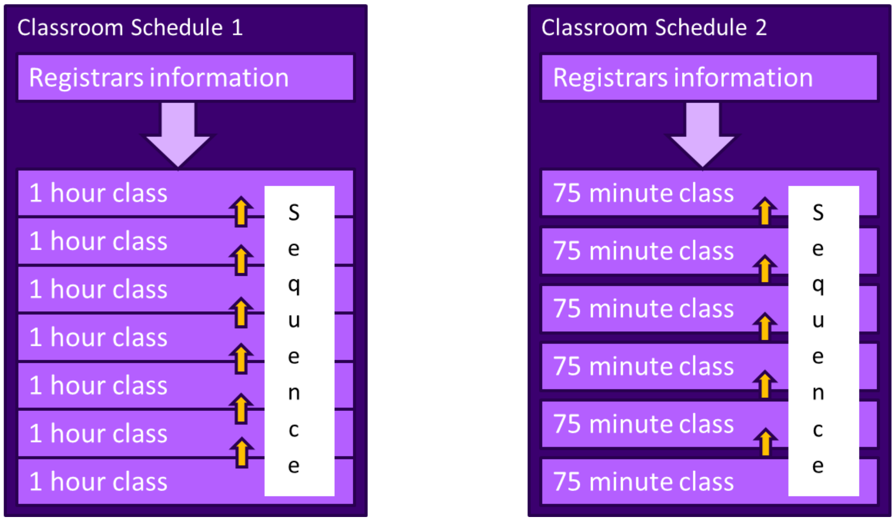


Figure 3: Classroom Scheduling Example

A college campus uses two schedules to schedule its buildings. In Schedule 1, classes start on the hour, and follow one after another; each class starts on the hour. In the second schedule, each class lasts an hour and a quarter, and there is a fifteen minute gap between classes; classes start on the half hour. On many campuses, the Sequence in Schedule 1 may describe classes taught on Monday, Wednesday, and Friday. Schedule 2 may describe classes taught on Tuesday and Thursday.

The registrar's office knows some key facts about each classroom, including whether it hosts a class during a particular period, and the number of students that will be in that class. The college wishes to optimize the provision of building services for each class. Such services may include adequate ventilation and comfortable temperatures to assure alert students. Other services may ensure that the classroom projection systems and A/V support services are warmed up in advance of a class, or powered off when a classroom is vacant.

Although most classes meet over typical schedule for the week (M-W-F or Tu-Th), some classes may not meet on Friday, or may have a tutorial section one day a week. The registrar's system, ever mindful of student privacy, shares only minimal information with the building systems such as how many students will be supported.

The Registrar's system schedule building systems using the ~~ Calendar~~ Gluon (registrar's information) and the student counts for each Interval, and schedules the Sequence in classroom schedule 1 three days a week for the next 10 weeks. The Registrar's system also schedules the Sequence in classroom schedule 2 two days a week, also for 10 weeks.

This example demonstrates a system (A) that offers services using either of two Sequences. Another business system (B) with minimal knowledge of how (A) works determines the performance requirements for (A). The business system (B) communicates what these expectations are by scheduling the Sequences offered by (A).

Market Performance schedule

A factory relies on an energy-intensive process which is performs twice a year for eight weeks. The factory has some flexibility about scheduling the process; it can perform the work in either the early morning or the early evening; it avoids the afternoon when energy costs are highest. The factory works up a detailed profile of when it will need energy to support this process.

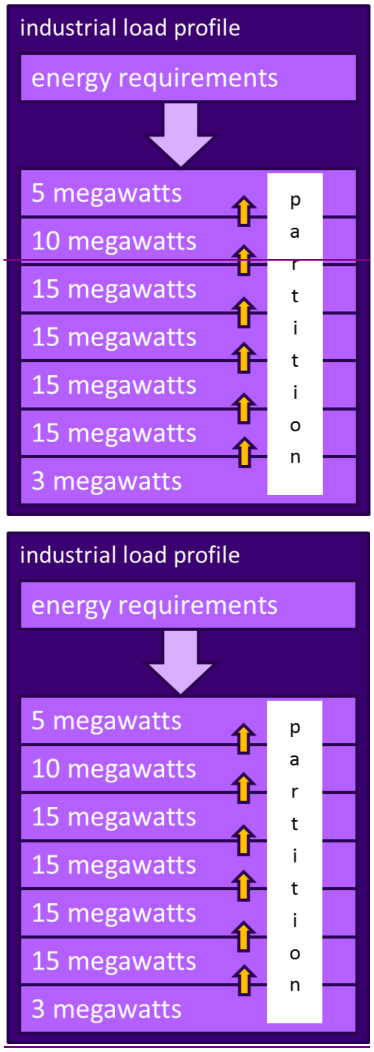


Figure 4: Daily Load Profile for Market Operations Example

Factory management has decided that they want to use only renewable energy products for this process. They approach two regional wind farms with the intent of making committed purchases of wind energy.

The wind farms consider their proposals taking into account the seasonal weather forecasts they use to project their weather capacity, and considering the costs that may be required to buy additional wind energy on the spot market to make up any shortfalls.

Each energy supplier submits of the same Sequence, a schedule, i.e. a daily starting time, and a price for the season's ~~prodUTCion.~~production. After considering the bids, and other internal costs of each proposal, the factory opts to accept a contract for the purchase of a fixed load profile (Partition), using the evening wind generation from one of the suppliers. This contract specifies Schedules of load purchases (starting data and time for the Sequence) for each day.

Revision History

Revision	Date	Editor	Changes Made
1.0 WD 01	2010-03-11	Toby Considine	Initial document, largely derived from Charter
1.0 WD 02	2010-03-30	Toby Considine	Straw-man assertion of elements, components to push conversation
1.0 WD 03	2010-04-27	Toby Considine	Cleaned up Elements, added [XPOINTER] use, xs:duration elements
1.0 WD 04	2010-05-09	Toby Considine	Aligned Chapter 4 with the vAlarm and vToDo objects.
1.0 WD 05	2010-05-18	Toby Considine	Responded to comments, added references, made references to [XCAL] more consistent,
1.0 WD 06	2010-05-10	Toby Considine	Responded to comments from CalConnect, mostly constancy of explanations
1.0 WD 07	2010-07-28	Toby Considine	Incorporated input from informal public review, esp. SGIP PAP04. Firmed up relationships between scheduled objects
1.0 WD 08	2010-08-07	Toby Considine	Aligned with Interval / Partition / Sequence language. Reduced performance characteristics to before / after durations.
1.0 WD 09	2010-08-15	Toby Considine	Formalized Attachment section and rolled Performance into the Attachment. Created RelatedComponent object. Added CalWS Outline to specification. Removed SOOP section
1.0 WD 10	2010-08-28	Toby Considine, Benoit Lepeuple	Updated Time Stamp section Added background Appendices Incorporated Association language to replace RelatedComponent Recast examples to show inheritance, remove inconsistencies
1.0 WD 11	2010-09-11	Toby Considine	Traceability Release in support of a re-shuffling of the document. Sections 3, 4 were re-shuffled to create: 3: Interval / Relationships / Time Stamps 4: Performance / Attachments 5: Associations & Inheritance Also, changed all associations to Gluons. No paragraphs have been changed, just shuffled, changes accepted, to create clean base for editing
1.0 WD 12	2010-09-14	Toby Considine Dave Thewlis	Edits for clarity and flow following changes in WD11, updated examples based upon XSD artifacts. Adding final contribution from CalConnect for Services.

1.0 WD 13		Toby Considine	Mechanistic processing of trivial comments for grammar, spelling, etc.
1.0 WD 14	2011-01-17	Toby Considine	Added Conformance rules, redefined inheritance, added terminology section in Section 1, added language on separability of information model, REST, and SOAP sections
1.0 WD 15	2011-01-27	Toby Considine	Pulled more definitions into Terminology Section, re-factored into multiple tables, Added Availability. Have not updated examples.
1.0 WD 15	2011-01-29	Toby Considine	Re-added footers to document (?!?) Added disclaimers on completeness prior to committee spec draft.
1.0 WD16	2011-02-07	Toby Considine	Minor changes to prepare for CSD as directed by TC
1.0 WD17	2011-03-01	Toby Considine	Reworked all examples, responded to numerous Jira editorial comments, eliminated “Mixed Inheritance of Schedule”, introduced Vavailability, eliminated UML chapter which confused more than enlightened.
1.0 WD18	2011-03-16	Toby Considine William Cox	Tightened language, spelling and grammar, consolidated chapters into “larger sections” Corrected to use CHILD link instead of PARENT in conformance with RFC5545. Replaced LINK language that was leftover from earlier schemas.
1.0 WD19	2011-03-19	Toby Considine	Changes to namespace to prepare for CSD, PR02, as directed by TC vote on 3/18/2011
1.0 WD20	2011-05-12	Toby Considine	Mechanical edits. Rebuilt document to remove cross-reference corruption (table and example lists), applied grammatical and punctuation changes from PR02, simple global replaces of terms. Reference checks. Refinement of logic of Duration/DtStart. Eliminated redefinition of VAVAILABILITY.
1.0 WD21	2011-05-16	Toby Considine	More Jira edits, especially unscrambling dtStart, dtEnd, and Duration, Vavailability, and many reference checks.
1.0 WD22	2011-05-20	William Cox	Eliminated Parts two and Three
1.0 WD23	2011-050-23	Toby Considine	Minor edits for clarity, final Jira issues
1.0 WD24	2011-05-26	Toby Considine	Examples updated
1.0 WD25	2011-05-26	William Cox	Eliminated remaining references to Parts Two

			and Three, corrected internal links