CalConnect TC

Timezone Registry and Service Recommendations

Published Standard

Warning for drafts

This document is not a CalConnect Standard. It is distributed for review and comment, and is subject to change without notice and may not be referred to as a Standard. Recipients of this draft are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

Recipients of this draft are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

The Calendaring and Scheduling Consortium, Inc. 2006

The Calendaring and Scheduling Consortium, Inc.

4390 Chaffin Lane
McKinleyville
California 95519
United States of America

copyright@calconnect.org
www.calconnect.org

Contents

Introduction

This document contains technical recommendations on implementing a time zone registry and a time zone service.

Why are standardized time zones needed?

- For improved interoperability: Calendar applications need to have a reliable list of time zones and their associated rules in order to avoid the following common problems:
  - Consuming unknown time zones.
  - Consuming known time zones with identical `TZID`s but different rule.

- Calendar applications need to have means of updating time zones and all affected data (i.e.: Previously created recurring meetings.) in an efficient and correct manner.
- Having standardized time zones would open the door to using time zones by reference rather than by value (Sending only the time zone id rather than the whole time zone with its rules). This could potentially help applications where bandwidth usage is important such as mobile devices.
- Any other non-iCalendar products having to deal with time zones could also benefit from it (Operating systems, java…).

Timezone Registry and Service Recommendations

1. Time zone registry

1.1. Implementation recommendations for a time zone registry

Recommendation 1
The registry should contain both the name and the rules of the time zones. Storing only the names of the time zones or the names of the rule would mean having some other sort of registry to map these names to meaningful values.

Recommendation 2
Leveraging what already exists in terms of time zone database should be seriously considered in order to speed up the process of establishing the time zone registry standard. As of now, the Olson Database seems to be the most widely used "source of truth" in the software world.

Recommendation 3
Adding a version per time zone is not seen as a requirement because time zone changes usually occur in the future which means that past events will remain the same even if a new rule is added to an existing time zone. A time stamp should be sufficient for most use cases.

Recommendation 4
Final implementation should be done using a standardized process, the new time zone registry should be coordinated by IANA.

Recommendation 5
Updates of the registry could be done through: RFC, informational RFC, appointed/elected committee/individual who approves updates to the list.

2. Time zone service

2.1. Implementation recommendations for a time zone service

Recommendation 6
The time zone should be in a `VTIMEZONE` format as defined in RFC 2445.

Recommendation 7
The time zone service should be build on top of a known platform such as: HTTP, CalDAV, DNS, or ITIP.

Recommendation 8
The time zone service should be able to return only modified time zones; given the last time the time zone were fetched from the service and/or a list of `TZID`-TimeStamp, return the time zones that have changed.

Recommendation 9
The time zone service should be able to return a time zone based on a supplied alias; given an OS based time zone name, return the corresponding `VTIMEZONE` object. For example: The windows time zone name:"(GMT -05:00) Eastern Time (US Canada)" would return the `VTIMEZONE` Object with the `TZID: "/IANA/America/New_York"`.

Recommendation 10
The time zone service should be able to return a time zone based on a supplied `TZID`.

Recommendation 11

:2006

The time zone service should be able to return a time zone based on a VTIMEZONE object (Closest match). The “closest match” feature would be useful for consuming iCalendar objects originating from applications that don’t have any time zone service knowledge.

3. Incremental implementation proposal

Implementing a standardized time zone registry and service is no easy task. Here is a tentative proposal on how it could be done incrementally by leveraging what already exists. This is just describing how it could be done and is by no means set in stone.

3.1. Milestone 1: Publish the registry file

Recommendation 12
A group (Let’s call it the TZ Group) agrees to generate a VTIMEZONE object file every time a new TZDB (Olson database) file is released we’ll cal that file the “TZ registry file”.

Recommendation 13
The TZ Group registers a domain name to host the TZ Group web site. This site is to be used for the distribution of the newest “TZ registry file”.

3.2. Milestone 2: Define and setup the time zone service

Recommendation 14
A new RFC is written using CalDAV as a base protocol, that draft would describe how to search based on time zone ids, how to retrieve all time zones, how to return only time zones whose time stamp has changed…

Recommendation 15
Current implementers of CalDAV servers and clients implement their first iteration of the new draft.

Recommendation 16
The RFC draft becomes an official standard.

Recommendation 17
Implementers release their applications.

3.3. Milestone 3: Formalizing the registry

Recommendation 18
The TZ Group takes steps to standardize the way the “TZ registry file” is generated and distributed.

Recommendation 19
Another RFC is created to define and setup the IANA time zone registry.

Bibliography

[1] Time zone registry draft http://ietfreport.isoc.org/idref/draft-royer-timezone-registry/

[2] Time zone database http://www.twinsun.com/tz/tz-link.htm

[3] Windows time zones to TZID mapping http://unicode.org/cldr/data/diff/supplemental/supplemental.html — windows___tzid

[4] CalDAV Draft http://ietfreport.isoc.org/idref/draft-dusseault-caldav/ RFC 2445 http://www.ietf.org/rfc/rfc2445.txt

[5] Time zone problems and recommendations document http://www.calconnect.org/publications/icalendartimezoneproblemsandrecommendationsv1.0.pdf