

# Table of Contents

<b>NotesCalConnectTZProposal.....</b>	<b>1</b>
I18n Review Notes on CalConnect Timezone Proposal.....	1
Review Notes Summary.....	1
1. Localized Timezone Names Available in Unicode CLDR.....	1
2. Locale Type Definition to Point to RFC 4646.....	1
3. Character Encoding Dictated By Environment, TSA Cannot Specify It.....	2
4. Client Timezone Identification Being Drafted, Coordination Desirable.....	2
5. Timezones API Java Bindings: Java Datetime Being Revamped by JSR-310.....	2
6. XML Example Not To Indicate Character Encoding.....	2
7. Localized Author Name Could Be Provided in Various Languages.....	3

# NotesCalConnectTZProposal

## I18n Review Notes on CalConnect Timezone Proposal

This note provides feedback for the CalConnect timezone proposal from internationalization perspectives.

The CalConnect timezone proposal documents included in this review are as follows:

- Timezones Service and API draft ("TSA draft" draft-douglass-timezones-api-and-service-00)
- Timezones Author Registry ("TAR draft" draft-doubleass-timezones-author-registry-00)

The comments are in no particular order. The document's name and its page number in the Internet draft is indicated after the title of each item using an abbreviated form. (e.g. "TSA-page1" for page 1 in Timezones Service and API. "TAR-pagexx" for Timezones Author Registry) The text in a draft is quoted. The review is based on revision 00. For items that have no particular page, they are page x.

The status of this revision is completed and to be submitted to the CalConnect community.

Related discussions and documents may be found in the official CalConnect website at [calconnect.org](http://calconnect.org).

---

Revision: 1/28/09, by Dan Chiba, Oracle Server Globalization Technology

---

## Review Notes Summary

### 1. Localized Timezone Names Available in Unicode CLDR

*TSA-pagex TAR-pagex*

Unicode CLDR(Common Locale Data Repository) has localized timezone names. It would be desirable to clarify the relationship between the names in the timezone author registry and those in CLDR.

- Unicode CLDR Project: Common Locale Data Repository
- Unicode CLDR Survey Tool

### 2. Locale Type Definition to Point to RFC 4646

*TSA-page19 TAR-page4*

Locale is loosely defined as "a type which identifies a locale by language and country". It would be adequate to reference RFC 4646, Tags for Identifying Languages, which defines the standard language tags for identifying a language or locale.

RFC 2277, IETF Policy on Character Sets and Languages, suggests that protocols should use RFC 1766 language tags. RFC 4646 is the current successor of RFC 1766. RFC 4646 and 4647 are the current IETF best practice for language/locale identification (BCP 47).

*Proposed change: Add reference to point to RFC 4646 for the definition of locale.*

- RFC 4646, Tags for Identifying Languages
- IETF Policy on Character Sets and Languages
- RFC 3066, Tags for the Identification of Languages
- RFC 1766, Tags for the Identification of Languages

### 3. Character Encoding Dictated By Environment, TSA Cannot Specify It

*TSA-page19*

For the definition of a string, the TSA draft specifies that "String is a UTF-8 character sequence." It would be adequate to specify only the code character set, to be Unicode. The character encoding scheme cannot be specified, because it would depend on the implementing environment.

*Current:*

String: A UTF-8 character sequence.

*Proposed change:*

String: A Unicode character sequence.

- Wikipedia "Character Encoding" for coded character set vs. character encoding scheme
- The Unicode Standard

### 4. Client Timezone Identification Being Drafted, Coordination Desirable

*TSA-pagex*

TAR and TSA draft define a mechanism to define timezone information and deliver it to applications. It would be desirable to coordinate the standardization effort for them with the "Timezone Information in HTTP" draft, which defines an HTTP header to provide web client's timezone information to the web server.

*Proposal: Ensure consistency between TAR/TSA and Timezone Information in HTTP*

- Timezone Information in HTTP

### 5. Timezones API Java Bindings: Java Datetime Being Revamped by JSR-310

*TSA-page20*

TSA draft states that "\_The current Java api has definitions for timezone information and various methods defined to carry out operations on that information. It is unclear at this stage how this specification will interact with that api but at least it should deliver a subclass of the current Timezone class.\_".

JSR 310 is under development to provide a new Java SE API with better date, time and calendar classes. It is desirable to coordinate with JSR 310 so that the Timezone Service API and JSR 310 will be consistent.

*Proposal: Ensure consistency between TAR/TSA and JSR 310*

- JSR-310 Date and Time API

### 6. XML Example Not To Indicate Character Encoding

*TSA-page8,9,11&more*

XML examples should not have encoding="UTF-8" in the XML prologue because it misleads readers into thinking as if it was a significant source of encoding information. Generally, the encoding of an XML entity is to be identified without using the encoding declaration. Best practice is encoding the XML in UTF-8 and put no encoding declaration.

*Proposal: remove encoding="UTF-8" in the XML headers*

- Timezone Information in HTTP

## **7. Localized Author Name Could Be Provided in Various Languages**

*TSA-page19*

It would be desirable to have a standard representation form for a multilingual localized string. A timezone author's name could be provided in different languages that match the preferred languages of the audience.

*Proposal: make a provision for adopting a standard representation scheme for localizable strings.*

- "InternationalizedString" from W3C OWL Working Group, an example of standard localized string