Issues of Common Locale Data Repository in Indian Languages

Swaran Lata, Director and Head, TDIL Programme, DIT
Parul Bajaj, Project Engineer, Cdac
Somnath Chandra , Joint Director, TDIL Programme, DIT

ABSTRACT

This paper reveals the fact about CLDR. The Common Locale Data Repository is a volunteer-developed and maintained resource coordinated and administered by the Unicode Consortium. CLDR is associated with languages and country names, date formats, time zone names, and so forth. It is based on BCP 47.

INTRODUCTION

Not long ago, computer systems were like separate worlds, isolated from one another. The internet and related events have changed all that. A single system can be built of many different components, hardware and software, all needing to work together. Many different technologies have been important in bridging the gaps; in the internationalization arena, Unicode has provided a lingua franca for communicating textual data. However, there remain differences in the locale data used by different systems. Its goal is to gather basic linguistic information for various "locales," essentially combinations of a language and a location.

LOCALE

In computing, locale is a set of parameters that defines the user's language, country and any special variant preferences that the user wants to see in their user interface. Usually a locale identifier consists of at least a language identifier and a region identifier. A locale is an identifier (id) that refers to a set of user preferences that tend to be shared across significant swaths of the world. Traditionally, the data associated with this id provides support for formatting and parsing of dates, times, numbers, and currencies; for measurement units, for sort-order (collation), plus translated names for time zones, languages, countries, and scripts. The data can also include support for text boundaries (character, word, line, and sentence), text transformations (including transliterations), and other services. There are many tools available on various technologies like java.util.Locale, .Net Culture, LANG (setlocale in C, C++), NLS_LANG in Oracle.

SCOPE AND NEED OF CLDR

There are many things that locale data can cover. It could cover industry specific topics. CLDR limits its scope to a few specific topics. It is depend upon user's choice which can be change over time due to some political or cultural reasons like Introduction of new currencies, like the Euro, Standard sorting of Old Marathi or Manipuri and many of these preferences have varying degrees of standardization like in United States most of the people uses 12 hour time, however some of the people uses 24 hour time format. So, CLDR is very broad topic in itself. Localization is about Linguistic and Cultural adaption of digital content to the requirements of foreign markets and the management of multilingualism across the global and digital information flow. Text processing depends on languages spelling, stemming, tokenization, word/line/sentence boundaries, thesauri, terminology, morphological analysis, font and stylistic traditions, collation and IT systems are depend on language negotiation localization, message selection, user interface, presentation, number/date/time/etc. formatting, list presentation.

CLDR FORMAT

The Unicode Locale Data Markup Language (LDML) is the XML format for the communication of locale data. This provides a common format for systems to interchange locale data so that they can get the same results in the services provided by internationalization libraries. It also provides a standard format that can allow users to customize the behavior of a system. Unicode LDML can used to let a user encapsulate specialized sorting behavior for a specific domain, or create a customized locale for a minority language. It is also used in the Unicode Common Locale Data Repository (CLDR). CLDR uses an open process for reconciling differences between the locale data used on different systems and validating the data, to produce with a useful, common, consistent base of locale data. There are two kinds of data that can be expressed in LDML: language-dependent data and supplementary data. In either case, data can be split across multiple files, which can be in multiple directory trees. Types of Locale Data are as follows: Dates/time formats: , Number/Currency formats, Measurement System, Translated names for language, territory, script, timezones, currencies, Script and characters used by a language and Collation Specification as Sorting, Searching and Matching. The CLDR is totally based on BCP 47. Structure of CLDR controlled by Locale Data Markup Language (LDML) specification http://unicode.org/reports/tr35.That element contains the following elements:

<identity>: The identity element contains information identifying the target locale for this data, and general information about the version of this data.

<version number="$Revision: 1.227 $">

The version element provides, in an attribute, the version of this file. The contents of the element can contain textual notes about the changes between this version and the last. For example:

<version number="1.1">Various notes and changes in version 1.1</version>

<fallback>: Sometimes, it happens that there may not be full data for a particular locale. It provides a mechanism to indicate what the best available fallback locales would be. For example, for a Marathi speaker, the best fallback if data is unavailable might be Marathi. That is, suppose we have found a Marathi bundle, but it does not contain translation for the key "IN" (for the country India). It is best to return "Indiae", rather than falling back to the value default language such as Russian and getting "Кітай" .

<localeDisplayNames>: It provides translated display names for languages, territories, scripts, variants and keywords used in CLDR and most of this information is at the language level, since it typically does not vary by territory, only language. An example

```
<localeDisplayNames>
<language type="aa">अफ़ार</language>
<language type="aa" alt="proposed-x1001"
 draft="unconfirmed">अफ़र</language>
<language type="ab">अब्खाज़ियन्</language>
<language type="ach">अकोली</language>
<language type="ae"
 draft="contributed">अवेस्तन</language>
<language type="af">अफ्रीकी</language>
<language type="afa">अफ्रो-एशियाई भाषाएँ</language>
<language type="afa" alt="proposed-x1001"
draft="unconfirmed">अफ्रो-एशियाई भाषाऐ</language>
<language type="afh">अफ्रीलीयाई</language>
```

<layout>: This top-level element specifies general layout features. It currently only has one possible element (other than <special>, which is always permitted).
<orientation lines="top-to-bottom" characters="left-to-right" />
The lines and characters attributes specify the default general ordering of lines within a page, and characters within a line. The values are:

Orientation Attributes

Vertical	top-to-bottom
	bottom-to-top
Horizontal	left-to-right
	right-to-left

If the lines value is one of the vertical attributes, then the characters value must be one of the horizontal attributes, and vice versa.

<characters>: Characters allows for creation of exemplar character sets. An exemplar set specifies the set of characters that must be present in order to properly render the language. Auxiliary exemplar set defines additional characters that may appear in foreign words or phrases but in Lower case only.

```
- <characters>
<exemplarCharacters>[ॐ ०-९ अ-ऍ ए-ऑ ओ-न प-र ल ळ व-ह
़ ँ-ः ऽ ् ा-ॅ े-ॉ ो ौ]</exemplarCharacters>
<exemplarCharacters type="auxiliary">
[\u200C\u200D]</exemplarCharacters>
</characters>
```

<delimiters>: The delimiters supply common delimiters for bracketing quotations.

```
<quotationStart>“</quotationStart>
<quotationEnd>”</quotationEnd>
<alternateQuotationStart>‘</alternateQuotationStart>
<alternateQuotationEnd>’</alternateQuotationEnd>
```

<measurement>: The measurement element is deprecated in the main LDML files, because the data is more appropriately organized as connected to territories, not to linguistic data. Instead, the similar element in the supplemental data file should be used.

<dates>: Date defines representation of calendars using various calendaring systems (Gregorian, Buddhist, Islamic, Japanese, etc.). It also defines formatting for dates, times, eras and time zones either wide, abbreviated, or narrow and Date and time formats use patterns of letters to define proper formatting. It provides information of Weeks and Relative day/time translations for example, yesterday, tomorrow, etc.

```
<calendar type="gregorian">
<months>
<monthContext type="format">
<monthWidth type="abbreviated">
<month type="1">जनवरी</month>
<month type="2">फरवरी</month>
<month type="2"  alt="proposed-x1001”
draft="unconfirmed" >फ़रवरी</month>
<month type="3">मार्च</month>
<month type="4">अप्रैल</month>
<month type="5">मई</month>
<month type="6">जून</month>
<month type="7">जुलाई</month>
<month type="8">अगस्त</month>
<month type="9">सितम्बर</month>
<month type="10">अक्तूबर</month>
<month type="10" alt="proposed-x1001"
 draft="unconfirmed">अक्टूबर</month>
```

```
<month type="11">नवम्बर</month>

<month type="11" alt="proposed-x1001"

 draft="unconfirmed">नवंबर</month>

<month type="12">दिसम्बर</month>

<month type="12" alt="proposed-x1001"

 draft="unconfirmed">दिसंबर</month>
```

<numbers>: This tag specifies proper localized formatting of numeric quantities in Decimal, Scientific, Currency and Percentages. It also Includes localized decimal, thousands separators, currency symbols, etc.

```
<symbols>
    <decimal>.</decimal>
    <group>,</group>
    <list>;</list>
    <percentSign>%</percentSign>
    <nativeZeroDigit>0</nativeZeroDigit>
    <patternDigit>#</patternDigit>
    <plusSign>+</plusSign>
    <minusSign>-</minusSign>
    <exponential>E</exponential>
    <perMille>‰</perMille>
    <infinity>∞</infinity>
    <nan>☹</nan>
</symbols>
```

<units>: These elements specify the localized way of formatting quantities of units such as years, months, days, hours, minutes and seconds— for example, in English, "1 day" or "3 days". The English rules that produce this example are as follows ({0} indicates the position of the formatted numeric value):

```
<unit type="day">
<unitPattern count="one">{0} day</unitName>
<unitPattern count="other">{0} days</unitName>
</unit>
```

<collations>: It contains one or more collation elements, distinguished by type. It contains Information in collation directory, not main. It is the XML version of Java/ICU collation syntax. Its Unicode collation algorithm has base http://unicode.org/reports/tr10

<posix>:- A colon-separated list of strings that would normally be recognized as "yes" and "no" responses. For cased languages, this shall include only the lower case version. POSIX locale generation tools must generate the upper case equivalents, and the abbreviated versions, and add the English words wherever they do not conflict.

```
<posix>

<messages>

<yesstr draft="contributed">हाँ</yesstr>
```

```
<nostr draft="contributed">नहीं</nostr>

</messages>

</posix>
```

<references>: The references section supplies a central location for specifying references and standards. The uri should be supplied if at all possible. If not online, then a ISBN number should be supplied, such as in the following example:

```
<reference type="R2"

uri="http://www.ur.se/nyhetsjournalistik/3lan.html">Lands
koder på Internet</reference>

<reference type="R3" uri="URN:ISBN:91-47-04974-

 X">Svenska skrivregler</reference>
```

BCP 47

A Best Current Practice (BCP) means that a certain manner of proceeding is in general the most logical choice a de facto standard of sorts. BCP is the name published by the Internet Engineering Task Force. A Best Current Practice is only a noncommittal suggestion, as is to be proceeded in a certain case. It is thus more flexible than a standard: As requirements and conditions change, from time to time, another manner of proceeding can be more promising, and subsequently implemented. Should the requirements permanently change, a BCP should be revised. It describes the structure, content, construction, and semantics of language tags for use in cases where it is desirable to indicate the language used in an information object. It also describes how to register values for use in language tags and the creation of user-defined extensions for private interchange. The language tag is composed of one or more parts, known as "subtags". Each subtag consists of a sequence of alphanumeric characters. BCP 47 can be used in no. of modern standards such as http, html, xml, png. Each language tag is composed of one or more “subtags” separated by hyphens. With the exception of private use language tags and grandfathered language tags, the subtags occur in the following order: a language subtag (potentially followed by up to three extended language subtags)- an optional script subtag, an optional region subtag, optional variant subtags, optional extension subtags, optional private use subtags. Language subtags are mainly derived from ISO 639-1 and ISO 639-2, script subtags from ISO 15924, and region subtags from ISO 3166-1 alpha-2 and UN M.49. Variant subtags are not derived from any standard. No extension subtags have yet been defined. The Language Subtag Registry, maintained by IANA, lists the current valid public subtags. Language tags consist of only language subtag, or a language subtag and a region subtag. For example, bn represents Bengali, and consists of a single language subtag (from ISO 639-1), while bn-IN represents Indian Bengali, and consists of the language subtag bn followed by the region subtag IN (from ISO 3166-1).Subtags are not case sensitive, but the specification recommends using the same case as in the Language Subtag Registry, where region subtags are uppercase, script subtags are titlecase and all other subtags are lowercase. Syntax will be:

```
Language-Tag  = langtag
              / privateuse          ; private use tag
              / grandfathered       ; grandfathered registrations
 langtag = (language ["-" script]  ["-" region]   *("-" variant)
*("-" extension)   ["-" privateuse])
```

For Example, bn language code is used for Bengali language; Beng Script code is used for Bengali Script and IN is used as Region Code for India. So, to represent the whole Language tag we can declare it as bn-Beng-IN (Bengali language uses Bengali Script in India.)

CLDR WITH ISO 8601, 15924, 3166, 639, 4217

ISO 639 (Language Codes): ISO 639 is the set of international standards that lists short codes for language names. It provides two set of three-letter alphabetic codes for the representation of names of languages. ISO 639 consists of different parts, of which two parts have been approved and a third part that is in the final approval (FDIS) stage. The other parts are works in progress. The ISO 639 code is divided into two parts i.e. Two-character code (639-1) Example: hi for Hindi, kn for kannada and Three character code (639-2, 639-3,639-5) Example: mar for Marathi, san for sanskrit (ISO 639-1)

- ISO 639-1 (alpha2 codes^{676}) (136 codes)
- ISO 639-2 (alpha3 codes^{17576}) (about 500)
- ISO 639-3 (alpha3 codes) (about 7000)
- ISO 639-4 (principles for encoding)
- ISO 639-5 (language families)
- ISO 639-6 (alpha4 codes) (under development)

There's no definitive list associating scripts with languages (because most languages could be written with most scripts, and this actually happens when people are transliteratingng foreign languages).This is a N-to-N relation, which only includes the standardized orthographies, and even lots of standard languages are missing the most basic information about the well-known scripts they are written with. For example: * language "br" (Breton)->script "Latn" (Latin) is missing, despite it is the only standardized script for the language. There are many indic scripts are not associated with the most language using it.

ISO 15924 (Script Code): There is a list in ISO 15924 used in Unicode 4.0 in which some Indian Scripts are included. Some more Scripts are added in Unicode 5.0 as well.

ISO 3166 (Region Codes): to denote the area in which a term, phrase, or language is used and provides two and three-character codes for representing names of countries. These two standards were developed independently, and there was no attempt to use the same code for a language as that for the country in which it is spoken. One should use codes from each list independently. However, the language code and country code may be used together to indicate a language variation spoken in a particular country as defined by BCP 47.For Example, Panjabi as spoken in the India may be indicated with the following: **pa-IN**.

ISO 4217 (Currency Codes): ISO 4217 is the international standard describing three-letter codes (also known as the currency code) to define the names of currencies established by the International Organization for Standardization (ISO). The ISO 4217 code list is the established norm in banking and business all over the world for defining different currencies, and in many countries the codes for the more common currencies are so well known publicly, that exchange rates published in newspapers or posted in banks use only these to define the different currencies, instead of translated currency names or ambiguous currency symbols. ISO 4217 codes are used on airline tickets and international train tickets to remove any ambiguity about the price.

ISO 8601(Date and Time): ISO 8601 is an international standard for date and time representations issued by the International Organization for Standardization (ISO). Specifically, the standard is titled "Data elements and interchange formats — Information interchange — Representation of dates and times." The signature feature of ISO 8601 date and time representations is the ordering of date and time values from the most to the least significant or, in plain terms, from the largest (the year) to the smallest (the second).

CLDR AND UNICODE

These are the Draft CLDR data available in UNICODE CLDR initiative for Indian languages.

S.NO.	**LANGUAGE**	**COUNTRY**
1.	Assamese	India
2.	Bengali	Bangladesh, India
3.	Gujarati	India
4.	Hindi	India
5.	Kannada	India
6.	Konkani	India
7.	Malayalam	India
8.	Marathi	India
9.	Oriya	India
10.	Punjabi	India, Pakistan
11.	Punjabi (Gurmukhi)	India
12.	Sanskrit	India
13.	Tamil	India
14.	Telugu	India
15.	Urdu	India, Pakistan

CLDR AND LOCALIZATION

Localization is the process of adapting software for a specific region or language by adding locale-specific components and translating text. It is usually abbreviated as l10n (where 10 stand for the number of letters between the l and the n in Localization . Internationalization is the process of designing a software application so that it can be adapted to various languages and regions without engineering changes. It is usually abbreviated as i18n where 18 stand for the number of letters between the i and the n in internationalization. CLDR is by far the largest and most extensive standard repository of locale data. This data is used by a wide spectrum of companies for their software internationalization and localization.

CLDR - ***A strategic tool for the development of***

E-Governance.

The Indian government is using IT to facilitate governance. The last couple of years have seen e-governance drop roots in India. IT enables the delivery of government services as it caters to a large base of people across different segments and geographical locations. It also gives citizens easy access to tangible benefits, be it through simple applications such as online form filling, bill sourcing and payments, or complex applications like distance education and tele-medicine.

Nowhere, can the benefit of CLDR and public-private partnerships be seen more clearly than in e-Governance initiatives. E-Governance, by nature, is citizen-centric rather than officer-centric. Therefore, it is necessary to be interactive, rather than just presenting information. For e-Governance to be truly effective and sustainable, it will have to go to the very doorsteps of the people - and CLDR can help a lot in making this possible.

CLDR in W3C

The World Wide Web Consortium (W3C) develops interoperable technologies (specifications, guidelines, software, and tools) to lead the Web to its full potential.

Goals of W3C

- Web for Everyone
- Web on Everything
- Knowledge Base
- Trust and Confidence

CLDR data can be used for many W3C activities like internationalization, Language Tags and Locale Identifiers for the World Wide Web.

To support users in different languages, software systems must not only use translated text, but must also be adapted to local conventions. An identifier for such a set of default conventions for a given user community (identified by language, script, region, variant, etc.) is commonly referred to as a locale.

Locale Specific data is the name used for denoting default conventions of a language, script or a region community. Those data which are specific to a locale is called locale specific data.

A collection of common, necessary locale data for all world languages is provided by Common Locale Data Repository (CLDR) and this locale data can contribute very much to error free and fast translation and avoid confusion.

CLDR AND INDIAN LANGUAGE TAGS

Identification of Language and its locale is a challenge by itself. IETF has developed BCP 47 for both Language Tags and Locale. Since Indian languages have varieties. Indian Language hierarchy is much unmanaged. It has dialect, sub dialect, regional and stylistic differences and patois, also different communities have different requirements in terms of terminology, librarians, computer systems, translators. Language needs to identify in written form and locale provides the bases how to write data in particular language correctly. To make data accessible on internet through locale there should be harmony between operating environment, Application server, Web servers and acceptance of CLDR/LDML of the particular language. A variety of problems were associated with language tags, despite their success. The main problem associated with Language tags and CLDR is script variation. Most languages are customarily written in a single script. They may be transcribed in another script, but most native speakers and most content in that language use a single script. A few languages are written equally—or at least "commonly"—in more than one script. Some of the languages are undergoing transitions (Devanagari script written in more than one script. For example, Kashmiri can be written in either Arabic or Devanagari script. Both traditions are historical to the language, not artificially imposed. The most notable problem was the relative ease of registration for language tags compared to the action of the various ISO maintenance and registration bodies. Many of the registered tags were later deprecated due to standards action and another problem is instability in ISO 3166 (the region codes). Codes in ISO 3166 are changing all the time, which is not a surprise, given that countries are changing name, boundaries, and organization with some regularity. ISO 3166 not only remove old codes but they sometimes introduce new country or region.

SPECIFIC ISSUES RELATED TO INDIAN LANGUAGES

<u>INDIAN (Missing Date format)</u>

S.No.	Code	English	Proposed	others
1.	**Indian/date/ full/standard / standard**	EEEE, MMMM d, y	EEEE,d MMMM, y	EEEE, y MMMM dd
	Example	*Tuesday, Bhadra 23, 1921*	मंगलवार २३ भद्र १९२१	मंगलवार, १९२१ भद्र २३
2.	**Indian/date/ long/standard /standard**	MMMM d, y	*d MMMM y*	*yyyy MMMM d*
	Example	*Bhadra 23, 1921*	२३ भद्र १९२१	१९२१ भद्र २३
3.	**Indian/date/ medium/ standard/ standard**	MMM d, y	*dd-MM-yyyy*	*y MMM d*
	Example	*Bhadra 23, 1921*	२३-०६-१९२१	१९२१ भद्र २३
4.	**Indian/date/ short/ standard /standard**	M/d/yy	*d-M-yy*	*yyyy-MM-dd*
	Example	*6/23/21*	२३-६-२१	१९२१-०६-२३

** भद्र भाद्र

<u>Gregorian (Missing Date Interval Format)</u>

S.No.	Code	English	Proposed	Others
1.	**DateTime/ interval/ intervalitems [Id="MEEd’ [id=@’d’]**	EEE, M/d – EEE, M/d	EEE, MM-dd – EEE, MM-dd	
	Example	*Wed, 2/13 – Thu, 2/14*	बुध, ०२-१३ – गुरु, ०२-१४	
2.	**DateTime/ interval/ intervalitems [id="MEEd’ [id=@’M’]**	EEE, M/d – EEE, M/d	*EEE, MM-dd – EEE, MM-dd*	
	Example	*Wed, 2/13 – Fri, 3/14*	बुध, ०२-१३ – शुक्र, ०३-१४	

COMMITTEE PROCESS

The Committee process is converted into five processes. These process are as follows:

1. When gathering data for a region and language, it is important to have multiple sources for that data to produce the most widely acceptable data.
2. Initial versions of data were based on the best available sources, but CLDR data will be modified and improved, in successive versions, by more input from the contributors inside and outside of the Unicode Consortium.
3. CLDR is a Repository, not a Registration. it will be vetted against the best available information.
4. Inputs are open, and gathered via the CLDR Survey Tool or recorded in a bug/feature request database (CLDR Bug Reports).
5. Changes in response to requests in the database may be entered into the repository snapshot over time by the maintainers of the repository, but the final approval of the release of any version of CLDR is up to the decision of the CLDR Technical Committee.

VETTING PROCESS FOR DATA

Once data for a country and language has been received, the data from the different sources will be compared to show agreements and differences. Initial data contributions are normally marked as "draft=provisional"; this may be changed once the data is vetted.

Note that there are two types of data in the repository: **Comparison data:** The contributor can be an individual or an organization. Data is normally gathered by calling public APIs, to ensure that the data matches what is actually in use. The data is only for comparison, and will not be changed except where necessary to update the data to match the external source. The only requirement is that all changed data be versioned, and the Version Numbering Scheme be used.

Common Data: The contents are decided upon by the CLDR Technical Committee, following its procedures and this process.

Contributors are encouraged to use local language and country contacts, inside and outside their organization, to help vet current common data and any new proposals for addition or amendment of common data. In particular, national standards organizations are encouraged to be involved in the data vetting process.

PROBLEMS FACED DURING PROCESS OF HINDI CLDR

CLDR has very Complex Formats. So it requires Experts knowledgeable both in technology and a specific language, Collation and Exemplar characters. It also requires close interaction of CLDR experts with language experts. Some major issues related to implementation of the Hindi CLDR process , the Languages which are included in the 8th Schedule need to be implemented first. The other aspiring languages (with regional dialects) such as Sourashtiya, Bhojpuri etc need to be taken up for deliberations in next phase, the major data fields required for Indian Language CLDR need to finalized first, to capture the variations in the Hindi Speaking states , the State Information Untis (SIUs) of NIC may be consulted for authentic information since the SIUs are implementing the E-governance applications. To capture the variations of date and time units, measurement units, the Land Record applications being implemented by SIUs of NIC may investigate. There is a need for deliberations for inclusion of special tags in XML CLDR data to capture the variations of Hindi, to obtain a holistic view of variations of CLDR data, at data may be captured from different places of a particular states, the Open Source Groups may be pooled in the CLDR collection of data, training of Language Groups /Linguists in the states need to be organized to sensitize the importance of the Locale Data and the Spelling Corrections required for the present CLDR 1.6 data need to be undertaken to make the data error free. In Gregorian Calendar Format some problems face by Indian languages as here in example of Hindi.

Gregorian Date Format (Not Approved but no alternative)

S.No.	Code	English	Proposed	Others
1.	**available date formats: [@id= ”MMMdd”]**	n/a	MMM dd	
	Example	-	सितम्बर १४	
2.	**available date formats: [@id= ”MMMMd”]**	MMMM d	d MMMM	
	Example	*September 14*	१४ सितम्बर	सितम्बर १४

SUGGESTIONS ON HINDI CLDR

Date Formats: We need to include (i) Saka शक (National Calendar of Govt. of India) (ii) Vikrama Sambat (शक सम) and (iii) Gregorian Calendar in the CLDR. In both the विक्रमी सवंत (present year + 57) – पंचाग / वि. सं. (2055) the counting of month starts from March. The Number of days in the month is either 28, 31 or 32. For this we need to define an extra tag "Calendar Type in the XML". We need to include variations of the names of the days as given below:

Thursday - गुरूवार, बृहस्पतिवार, वीरवार
Saturday - शनिश्चर, शनिच्चर
Sunday - रविवार, इतवार

There is need to correct the spellings of month as given below:

अक्टूबर (Preferred) – अक्तूबर
सितंबर (Preferred)
नवंबर (Preferred)
दिसंबर (Preferred)

**The abbreviated forms of the months in an year needs to be Included as given below:

जन	January
फर	February
मा	March
अ	April
म	May
जू	June
जु	July
अग	August
स्ति	September
अक्टू	October
नव	November
दिस	December

The quarters of the year may be represented as

First Quarter	पहली तिमाही
Second Quarter	दूसरी तिमाही
Third Quarter	तीसरी तिमाही
Fourth Quarter	चौथी तिमाही

Various Time-periods may be included as:

BC	ईसापूर्व / ईसा पूर्व
AD	सन – सन्

Various Indian Months of year also need to be included as भारतीय कैलेंडर

विक्रम संवत	**शक संवत**	**ग्रिगोरियन**
महीना/मास	**महीना/मास**	**महीना/मास**
चैत्र	चैत्र	जनवरी
वैशाख	वैशाख	फरवरी
ज्येष्ठ	ज्येष्ठ	मार्च
आषाढ़	आषाढ़	अप्रैल
श्रावण	श्रावण	मई
भाद्र	भाद्र	जून
आश्विन	आश्विन	जुलाई
कार्तिक	कार्तिक	अगस्त
अग्रहायन	अग्रहायन	सितम्बर
पौष	पौष	अक्टूबर
माघ	माघ	नवम्बर
फाल्गुन	फाल्गुन	दिसम्बर

The Fortnightly time period (15 days) period in Indian Calendar system along with specific days of transition of moon (Full moon and New Moon) may also be included as

पक्ष / पखवारा

Fortnight (Full moon)	कृष्ण - अमावस्या
Fortnight (New Moon)	शुक्ल - पूर्णिमा

तिथि	**सप्ताह / हफ़्ता**	**समय अवधि**	**युग**
प्रतिपदा	रविवार / इतवार	आज	सत युग
द्वितीया	सोमवार	कल	त्रेता युग
तृतीया	मंगलवार	परसों	द्वापर युग
चतुर्थी	बुधवार	नरसों	कलियुग
पंचमी	गुरुवार / बृहस्पतिवार	दिन	
षष्ठी	शुक्रवार	प्रातः / सुबह / प्रभात	
सप्तमी	शनिवार / शनीश्चर	पूर्वाह्न	
अष्टमी		अपराह्न	

नवमी		मध्याह्न / दोपहर	
दशमी		संध्या / शाम / सांझ	
एकादशी		रात / रात्रि	
द्वादशी		तिमाही	
त्रयोदशी		दूसरी तिमाही / छमाही	
चतुर्दशी		तीसरी तिमाही	
अमावस्या / पूर्णिमा		वर्ष / साल	
		दशाब्दी	
		शताब्दी	
		सहस्त्राब्दी	

Feedback from various sources

भारतीय संख्या में अंश की गणना

चौथाई	**0.25**
आधा	**0.50**
पौन	**0.75**
एक	**1.00**
सवा	**1.25**
डेढ़	**1.50**
पौने दो	**1.75**
दो	**2.00**
सवा दो	**2.25**
ढाई	**2.50**
पौने तीन	**2.75**
तीन	**3.00**
सवा तीन	**3.25**
साढ़े तीन	**3.50**
पौने चार	**3.75**
चार	**4.00**

Time Formats: In the time format we should follow the Indian Standard Time. The Full and abbreviated forms of units of time may be represented as

घंटा –	मिनट	– सेकैंड
मि.	से.	
मिन.	सेकं.	

समय मापांक तालिका

उन्मेष	पलक को उठाने मे लगा समय
निमेष	पलक को झुकाने मे लगा समय
ह्रस्व	चिड़ियाँ की आवाज चीं चीं
प्लुत	कौवे का काँ - काँ
दीर्घ	
अक्षर	24/60 / 2/5 सेकेंड
पल	60 अक्षर – 24 सेकेंड
घड़ी	60 पल – 24 मिनट
पहर	7.5 घड़ी – 3 घंटे
दिन + **रात**	60 / 24 घंटे / 8 पहर

Measurement System:

The Indian traditional measuring systems for distance, weight etc. also may be included in the Hindi CLDR

8 खसखस	1 चावल
8 चावल	1 रत्ती
8 रत्ती	1 माशा
12 माशा	1 तोला
5 तोला	1 छटांक
4 छटांक	1 पाव
8 छटांक	आधा सेर
16 छटांक	1 सेर
40 सेर	1 मन
1 ग्राम	1 माशा / 15.48 ग्रेन
250 ग्राम	1 पाव
1000 ग्राम	1 किलोग्राम
5 सेर (किलोग्राम)	1 पसेरी
100 किलोग्राम	1 क्विंटल
1 क्विंटल	2.5 मन

दूरी मापांक तालिका

12 इंच	1 फुट
3 फुट	1 गज
220 गज	1 फर्लांग
8 फर्लांग	1 मील
1 बालिश्त	1 आधा हाथ
1 हाथ	आधा गज
1 गज	16 गिरह
10 डैकामीटर	1 डैसीमीटर
10 डैसीमीटर	1 मिलीमीटर
10 मिलीमीटर	1 सेंटीमीटर
100 सेंटीमीटर	1 मीटर
1000 मीटर	1 किलोमीटर

8 किलोमीटर	5 मील
1 मीटर	39.4 इंच

क्षेत्रफल मापांक तालिका

144 वर्ग	इंच1 वर्ग फुट
9 वर्ग फुट	1 वर्ग गज
4840 वर्ग	गज1 एकड़
20 बिसवांसी	1 बिसवा
20 बिसवा	1 बीघा
31/4 बीघा	1 एकड़
14,400 वर्ग फुट	1 बीघा
1 हेक्टेयर	2 एकड़
100 हेक्टेयर	1 वर्ग किलोमीटर
100 हेक्टेयर	1 ऐयर

आयतन मापांक तालिका

1 गेलन	4.5 लिटर
1 लिटर	1.75 पिंट
1000 लिटर	1 किलोलिटर

It has been suggested that there is no change required in the present CLDR 1.6 format on the following tags:

(1) Character Map (As it has been adopted from UNICODE standard)
(2) Variants
(3) Types
(4) Locale Display

Collation Sequence and Sorting Order

After deliberations It has been decided that, the sorting order notified by Central Hindi Directorate (CHD) would be adopted for the collation sequence and sorting order as given. A cluster tag (वर्ग) may be included as sub-tag. This may be done in the following way:

अ आ इ ई उ ऊ ऋ ए ऐ ओ औ
क ख ग घ ङ (कवर्ग)
च छ ज झ ञ (चवर्ग)
ट ठ ड ढ ण (टवर्ग)
त थ द ध न (तवर्ग)
प फ ब भ म (पवर्ग)
य र ल व
श ष स ह
ड़ ढ़
क्ष त्र ज्ञ श्र
◌ं ◌ः ◌ँ ◌ॅ ◌्
ख़ ज़ फ़

१ २ ३ ४ ५ ६ ७ ८ ९ ०

CONCLUSION

The standardization of Locale Data Repository is a very important task from the Localization of Indian Languages perspective, Locale data for Indian languages are available in the present version of UNICODE CLDR (Ver 1.6.1), and the data needs to be critically examined, corrected and vetted before the next version of CLDR comes up. Since, Hindi in National Language and being spoken in 11 States in India, we need to examine the Hindi Locale data first. Elaborating the clear distinction between the UNICODE encoding standardization activity and the objectives of Standardization of Locale Data in the standardization process of Locale Data, we need to accommodate variations of a particular language in the XML Locale repository, so as to enable effective localization.

REFERENCES

Localization/Internationalization Standards http://www.w3cindia.in/2006/08/Talks/Pat_Hall.pdf (Last Access on 06.01.09)

Internationalization and localization http://en.wikipedia.org/wiki/Internationalization_and_localization (Last Access on 06.01.09)

LDML http://www.unicode.org/reports/tr35/ (Last Access on 06.01.09)

CLDR http://www.unicode.org/cldr/ (Last Access on 06.01.09)

Localization Intelligence: Unicode's CLDR Project http://gilbane.com/globalization/2008/04/global_intelligence_for_free_u.html (Last Access on 06.01.09)

Language Tags and Locale Identifiers for the World Wide Web http://www.w3.org/TR/ltli/(Last Access on 06.01.09)