Internet Engineering Task Force	M.E. Davis
Internet-Draft	Google
Intended status: Informational	A. Phillips
Expires: January 14, 2012	Lab126
	Y. Umaoka
	IBM
	C. Falk
	Infinite Automata
	July 13, 2011

BCP 47 Extension T - Transformed Content
draft-davis-t-langtag-ext-03

Abstract

This document specifies an Extension to BCP 47 which provides subtags for specifying the source language or script of transformed content, including content that has been transliterated, transcribed, or translated, or in some other way influenced by the source. It also provides for additional information used for identification.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.
Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet- Drafts is at http://datatracker.ietf.org/drafts/current/.
Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."
This Internet-Draft will expire on January 14, 2012.

Copyright Notice

Copyright (c) 2011 IETF Trust and the persons identified as the document authors. All rights reserved.
This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document.

Table of Contents

*1. Introduction

*1.1. Requirements Language

*2. BCP47 Required Information

*2.1. Introduction

*2.2. Structure

*2.3. Canonicalization

*2.4. BCP47 Registration Form

*2.5. Field Definitions

*2.6. Registration of Field Subtags

*2.7. Machine-Readable Data

*3. Acknowledgements

*4. IANA Considerations

*5. Security Considerations

*6. References

*6.1. Normative References

*6.2. Informative References

*Authors' Addresses

1. Introduction

[BCP47] permits the definition and registration of language tag extensions "that contain a language component and are compatible with applications that understand language tags". This document defines an extension for specifying the source of content that has been transformed, including text that has been transliterated, transcribed, or translated, or in some other way influenced by the source. It may be used in queries to request content that has been transformed. The "singleton" identifier for this extension is 't'.

Language tags, as defined by [BCP47], are useful for identifying the language of content. There are mechanisms for specifying variant subtags for special purposes. However, these variants are insufficient for specifying content that has undergone transformations, including content that has been transliterated, transcribed, or translated. That is, for fully specifying such content, it is important to specify the source language and/or script. In addition, it may also be important to identify a particular specification for the transformation.

For example, suppose that Italian or Russian cities on a map are transcribed for Japanese users. Each name needs to be transliterated into katakana using rules appropriate for the specific source and target language. When tagging such data, it is important to be able to indicate not only the resulting content language ("ja" in this case), but also the source language.

Transforms such as transliteration may vary depending not only on the basis of the source and target script, but also on language. Thus the Russian <U+041F U+0443 U+0442 U+0438 U+043D> (which corresponds to the Cyrillic <PE, U, TE, I, EN>) transliterates into "Putin" in English but "Poutine" in French. The identifier could be used to indicate a desired mechanical transformation in an API, or could be used to tag data that has been converted (mechanically or by hand) according to a transliteration method.

The usage of this extension is not limited to formal transformations, and may include other instances where the content is in some other way influenced by the source. For example, this extension could be used to designate a request for a speech recognizer that is tailored specifically for 2nd-language speakers who are 1st-language speakers of a particular language (e.g. a recognizer for "English spoken with a Chinese accent").

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119.

2. BCP47 Required Information

2.1. Introduction

Identification of transformed content can be done using the 't' extension defined in this document. This extension is formed by the 't' singleton followed by a sequence of subtags that would form a language tag as defined by [BCP47]. This allows for the source language or script to be specified to the degree of precision required. There are restrictions on the sequence of subtags. They MUST form a regular, valid, canonical language tag, and MUST neither include extensions nor private use sequences introduced by the singleton 'x'. Where only the script is relevant (such as identifying a script-script transliteration) then 'und' is used for the primary language subtag. For example:

Language Tag	Description
ja-t-it	The content is Japanese, transformed from Italian.
ja-Kana-t-it	The content is Japanese Katakana, transformed from Italian.

Language Tag	Description
und-Latn-t-und-cyrl	The content is in the Latin script, transformed from the Cyrillic script.

Note that the sequence of subtags governed by 't' cannot contain a singleton (a single-character subtag), because that would start a new extension. For example, the tag "ja-t-i-ami" does not indicate that the source is in "i-ami", because "i-ami" is not a regular language tag in [BCP47]. That tag would express an empty 't' extension followed by an 'i' extension.

It is sometimes necessary to indicate additional information about the transformation. This additional information is optionally supplied after the source in a series of one or more fields, where each field consists of a field separator subtag followed by one or more non-separator subtags. Each field separator subtag consists of a single letter followed by a single digit.

A transformation mechanism is an optional field that indicates the specification used for the transformation, such as "UNGEGN" for the the United Nations Group of Experts on Geographical Names transliterations and transcriptions. It uses the 'm0' field separator followed by certain subtags.

For example:

Language Tag	Description
und-Cyrl-t-und-latn-m0-ungegn-2007	the content is in Cyrillic, transformed from Latn, according to a UNGEGN specification dated 2007.

The field separator subtags such as 'm0' were chosen because they are short, visually distinctive, and cannot occur in a language subtag (outside of an extension and after 'x'), thus eliminating the potential for collision or confusion with the source language tag.

The field subtags are defined by Section 3 of Unicode Technical Standard #35: Unicode Locale Data Markup Language *[UTS35]*. As required by BCP 47, subtags follow the language tag ABNF and other rules for the formation of language tags and subtags, are restricted to the ASCII letters and digits, are not case sensitive, and do not exceed eight characters in length.

EDITORIAL NOTE: This new facility has been accepted by the Unicode CLDR committee for incorporation into the next version of Unicode CLDR, parallel with the structure of the 'u' extension [RFC6067], for which it is already the maintaining authority. The data and specification will be available by the time this internet draft has been approved. LDML is available over the Internet and at no cost, and is available via a royalty-free license at http://unicode.org/copyright.html. LDML is versioned, and each version of LDML is numbered, dated, and stable. Extension subtags, once defined by LDML, are never retracted or change in meaning in a substantial way.

The maintaining authority for the 't' extension is the Unicode Consortium:

Item	Value
Name	Unicode Consortium
Contact Email	cldr-contact@unicode.org
Discussion List Email	cldr-users@unicode.org
URL Location	cldr.unicode.org
Specification	Unicode Technical Standard #35 Unicode Locale Data Markup Language (LDML), http://unicode.org/reports/tr35/
Section	Section 3 Unicode Language and Locale Identifiers

2.2. Structure

The subtags in the 't' extension are of the following form:

```
t-ext=    "t"                       ; Extension
          (("-" lang *("-" field)) ; Source + optional field(s)
          / 1*("-" field))          ; Field(s) only (no source)

lang=     language                  ; BCP47, with restrictions
          ["-" script]
          ["-" region]
          *("-" variant)

field=    sep 1*("-" 3*8alphanum)   ; With restrictions

sep=      ALPHA DIGIT               ; Subtag separators
alphanum= ALPHA / DIGIT
```

where <language>, <script>, <region>, and <variant> rules are specified in [BCP47], <ALPHA> and <DIGIT> rules - in [RFC5234].
Description and restrictions:

a. The 't' extension MUST have at least one subtag.

b. The 't' extension normally starts with a source language tag, which MUST be a regular, canonical language tag as specified by [BCP47]. Tags described by the 'irregular' production in BCP 47 MUST NOT be used to form the language tag. The source language tag MAY be omitted: some field values do not require it.

c. There is optionally a sequence of fields, where each field has a separator followed by a sequence of one or more subtags. Two

identical field separators MUST NOT be present in the language tag.

d. The order of the subtags in a t extension is significant (see Section 2.3 Canonicalization).

e. The 't' subtag fields are defined by Section 3 of Unicode Technical Standard #35: Unicode Locale Data Markup Language *[UTS35]*.

2.3. Canonicalization

As required by [BCP47], the use of uppercase or lowercase letters is not significant in the subtags used in this extension. The canonical form for all subtags in the extension is lowercase, with the fields ordered by the separators, alphabetically.

2.4. BCP47 Registration Form

Per RFC 5646, Section 3.7 *[BCP47]*:

```
%%
Identifier: t
Description: Specifying Content Transforms
Comments: Subtags for the identification of content that has been
transformed, including but not limited to
transliteration, transcription, and translation.
Added: 2010-mm-dd
RFC: [TBD]
Authority: Unicode Consortium
Contact_Email: cldr-contact@unicode.org
Mailing_List: cldr-users@unicode.org
URL: http://www.unicode.org/Public/cldr/latest/core.zip
%%
```

2.5. Field Definitions

The structure of 't' field subtags is determined by the Unicode CLDR Technical Committee, in accordance with the policies and procedures in http://www.unicode.org/consortium/tc-procedures.html, and subject to the Unicode Consortium Policies on http://www.unicode.org/policies/policies.html.
Changes that can be made by successive versions of LDML *[UTS35]* by the Unicode Consortium without requiring a new RFC include:

*The allocation of new field separator subtags for use after the 't' extension.

*The allocation of subtags valid after a field separator subtag.

A new RFC would be required for material changes to an existing 't' subtag, or an incompatible change to the overall syntactic structure of the 't' extension; however, such a change would be contrary to the policies of the Unicode Consortium, and thus is not anticipated.
One field is initially specified in [UTS35]: the transform mechanism. That field is summarized here:

a. The transform mechanism consists of a sequence of subtags starting with the 'm0' separator followed by one or more mechanism subtags. Each mechanism subtag has a length of 3 to 8 alphanumeric characters. The sequence as a whole provides an identification of the specification for the transform, such as the mechanism subtag 'UNGEGN' in "und-Cyrl-t-und-latn-m0-ungegn". In many cases, only one mechanism subtag is necessary, but multiple subtags MAY be defined in [UTS35] where necessary.

b. Any purely numeric subtag is a representation of a date in the Gregorian calendar. It MAY occur in any mechanism field. If it does occur:

   *it MUST occur as the final subtag in the field,

   *it MUST NOT be the only subtag in the field, and

   *it MUST consist of a sequence of digits of the form YYYY, YYYYMM, or YYYYMMDD.

   For example, 20110623 represents June 23th, 2011. A date subtag SHOULD only be used where necessary, and then SHOULD be as short as possible. For example, suppose that the BGN transliteration specification for Cyrillic to Latin had three versions, dated June 11th, 1999; Dec 30th, 1999; and May 1st, 2011. In that case, the corresponding first two DATE subtags would require months to be distinctive (199906 and 199912), but the last subtag would only require the year (2011).

c. Some mechanisms may use a versioning system that is not distinguished by date, or not by date alone. In the latter case, the version will be of a form specified by [UTS35] for that mechanism. For example, if the mechanism XXX uses versions of the form v21a, then a tag could look like "ja-t-it-m0-xxx-v21a". If there are multiple subversions distinguished by date, then a tag could look like "ja-t-it-m0-xxx-v21a-2007".

A language tag with the t extension MAY be used to request a specific transform of content. In such a case, the recipient SHOULD return content that corresponds as closely as feasible to the requested transform, including the specification of the mechanism. For example, if the request is ja-t-it-m0-xxx-v21a-2007, and the recipient has content corresponding to both ja-t-it-m0-xxx-v21a-2007 and ja-t-it-m0-

xxx-v21a-2009, then the 2007 version would be preferred. As is the case for language matching as discussed in [BCP47], different implementations MAY have different measures of "closeness".

2.6. Registration of Field Subtags

Registration of transform mechanisms is requested by filing a ticket at cldr.unicode.org. The proposal in the ticket MUST contain the following information:

Item	Description
Subtag	The proposed mechanism subtag (or subtag sequence).
Description	A description of the proposed mechanism; that description MUST be sufficient to distinguish it from other mechanisms in use.
Version	If versioning for the mechanism is not done according to date, then a description of the versioning conventions used for the mechanism.

Proposals for clarifications of descriptions or additional aliases may also be requested by filing a ticket.
The committee MAY define a template for submissions that requests more information, if it is found that such information would be useful in evaluating proposals.
The committee MUST post each proposal publicly within 2 weeks after reception, to allow for comments. The committee must respond publicly to each proposal within 4 weeks after reception.
The response MAY:

* request more information or clarification

* accept the proposal, optionally with modifications to the subtag or description

* reject the proposal, because of significant objections raised on the mailing list or due to problems with constraints in this document or in [UTS35]

Accepted tickets result an a new entry in the machine-readable CLDR BCP47 data, or in the case of a clarified description, modifications to the description attribute value for an existing entry.

2.7. Machine-Readable Data

EDITORIAL NOTE: The following parallels the structure used for the 'u' extension [RFC6067], for which the Unicode Consortium is the maintaining authority. The data and specification will be available by the time this internet draft has been approved. The description field is in the process of being added to CLDR.

Beginning with CLDR version 1.7.2, machine-readable files are available listing the data defined for BCP47 extensions for each successive version of [UTS35]. These releases are listed on http://cldr.unicode.org/index/downloads. Each release has an associated data directory of the form "http://unicode.org/Public/cldr/<version>", where "<version>" is replaced by the release number. For example, for version 1.7.2, the "core.zip" file is located at http://unicode.org/Public/cldr/1.7.2/core.zip. The most recent version is always identified by the version "latest" and can be accessed by the URL in Section 2.4. Inside the "core.zip" file, the directory "common/bcp47" contains the data files listing the valid attributes, keys, and types for each successive version of [UTS35]. Each data file list the keys and types relevant to that topic. For example, mechanism.xml contains the subtags (types) for the t mechanisms.
The XML structure lists the keys, such as <key name="co" alias="collation">, with subelements for the types, such as <type name="phonebk" alias="phonebook"/>. The currently defined attributes for the mechanisms include:

Attribute	Description	Examples
name	The name of the mechanism, limited to 3-8 characters (or sequences of them).	UNGEGN, ALALC
description	A description of the name, with all and only that information necessary to distinguish one name from others with which it might be confused. Descriptions are not intended to provide general background information.	United Nations Group of Experts on Geographical Names; American Library Association-Library of Congress
since	Indicates the first version of CLDR where the name appears.	1.9, 2.0.1
alias	Alternative name of the key or type, not limited in number of characters. Aliases are intended for backwards compatibility, not to provide all possible alternate names or designations	

<type name="adp" since="1.9"/>

To get the version information in XML when working with the data files, the XML parser must be validating. When the 'core.zip' file is unzipped, the 'dtd' directory will be at the same level as the 'bcp47' directory; that is required for correct validation. For each release after CLDR 1.8, types introduced in that release are also marked in the

data files by the XML attribute "since", such as in the following example:
The data is also currently maintained in a source code repository, with each release tagged, for viewing directly without unzipping. For example, see:

*http://unicode.org/repos/cldr/tags/release-1-7-2/common/bcp47/

*http://unicode.org/repos/cldr/tags/release-1-8/common/bcp47/

For more information, see http://cldr.unicode.org/index/bcp47-extension.

3. Acknowledgements

Thanks to John Emmons and the rest of the Unicode CLDR Technical Committee for their work in developing the BCP 47 subtags for LDML.

4. IANA Considerations

This document will require IANA to insert the record in Section 2.4 into the Language Extensions Registry, according to Section 3.7. Extensions and the Extensions Registry of "Tags for Identifying Languages" in [BCP47]. Per Section 5.2 of [BCP47], there might be occasional (rare) requests by the Unicode Consortium (the "Authority" listed in the record) for maintenance of this record. Changes that can be submitted to IANA without the publication of a new RFC are limited to modification of the Comments, Contact_Email, Mailing_List, and URL fields. Any such requested changes MUST use the domain 'unicode.org' in any new addresses or URIs, MUST explicitly cite this document (so that IANA can reference these requirements), and MUST originate from the 'unicode.org' domain. The domain or authority can only be changed via a new RFC.
This document does not require IANA to create or maintain a new registry or otherwise impact IANA.

5. Security Considerations

The security considerations for this extension are the same as those for [BCP47]. See RFC 5646, Section 6, Security Considerations *[BCP47]*.

6. References

6.1. Normative References

[UTS35]	Davis, M, "Unicode Technical Standard #35: Locale Data Markup Language (LDML) ", December 2007.
[BCP47]	Davis, M.E. and A. Phillips, "Tags for the Identification of Language (BCP47)", September 2009.
[RFC6067]	

	Davis, M.E., Phillips, A. and Y. Yoshito, "BCP 47 Extension U", September 2010.
[RFC5234]	Crocker, , "Augmented BNF for Syntax Specifications: ABNF", 2008.

6.2. Informative References

, "

[ldml-registry]	Registry for Common Locale Data Repository tag elements", September 2009.

Authors' Addresses

Mark Davis Davis Google EMail: mark@macchiato.com

Addison Phillips Phillips Lab126 EMail: addison@lab126.com

Yoshito Umaoka Umaoka IBM EMail: yoshito_umaoka@us.ibm.com

Courtney Falk Falk Infinite Automata EMail: court@infiauto.com