Tags for Identifying Languages

draft-phillips-langtags-10

Status of this Memo

This document is an Internet-Draft and is subject to all provisions of section 3 of RFC 3667. By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she become aware will be disclosed, in accordance with RFC 3668.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt.

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

This Internet-Draft will expire on August 15, 2005.

Copyright Notice

Copyright (C) The Internet Society (2005).

Abstract

This document 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. This document obsoletes RFC 3066 (which replaced RFC 1766).
### Table of Contents

1. Introduction ................................................. 3
2. The Language Tag ............................................ 4
   2.1 Syntax .................................................. 4
       2.1.1 Length Considerations ............................... 5
   2.2 Language Subtag Sources and Interpretation ............... 6
       2.2.1 Primary Language Subtag ............................ 7
       2.2.2 Extended Language Subtags .......................... 9
       2.2.3 Script Subtag .................................... 9
       2.2.4 Region Subtag ................................... 10
       2.2.5 Variant Subtags .................................. 11
       2.2.6 Extension Subtags ................................ 11
       2.2.7 Private Use Subtags ................................ 12
       2.2.8 Pre-Existing RFC 3066 Registrations ............... 13
       2.2.9 Possibilities for Registration ....................... 13
       2.2.10 Classes of Conformance ............................ 14
   2.3 Choice of Language Tag .................................... 15
   2.4 Meaning of the Language Tag ............................... 16
       2.4.1 Canonicalization of Language Tags .................... 17
   2.5 Considerations for Private Use Subtags .................... 18
3. IANA Considerations ........................................... 20
   3.1 Format of the IANA Language Subtag Registry ............... 20
   3.2 Stability of IANA Registry Entries ....................... 24
   3.3 Registration Procedure for Subtags ....................... 27
   3.4 Extensions and Extensions Namespace ....................... 29
4. Security Considerations ....................................... 32
5. Character Set Considerations .................................. 33
6. Changes from RFC 3066 ......................................... 34
7. References .................................................. 36
   Authors' Addresses ......................................... 38
A. Acknowledgements ............................................ 39
B. Examples of Language Tags (Informative) ....................... 40
C. Conversion of the RFC 3066 Language Tag Registry ............ 42
   Intellectual Property and Copyright Statements ............... 45
1. Introduction

Human beings on our planet have, past and present, used a number of languages. There are many reasons why one would want to identify the language used when presenting or requesting information.

Information about a user's language preferences commonly needs to be identified so that appropriate processing can be applied. For example, the user's language preferences in a browser can be used to select web pages appropriately. A choice of language preference can also be used to select among tools (such as dictionaries) to assist in the processing or understanding of content in different languages.

In addition, knowledge about the particular language used by some piece of information content may be useful or even required by some types of information processing; for example spell-checking, computer-synthesized speech, Braille transcription, or high-quality print renderings.

One means of indicating the language used is by labeling the information content with a language identifier. These identifiers can also be used to specify user preferences when selecting information content, or for labeling additional attributes of content and associated resources.

These identifiers can also be used to indicate additional attributes of content that are closely related to the language. In particular, it is often necessary to indicate specific information about the dialect, writing system, or orthography used in a document or resource, as these attributes may be important for the user to obtain information in a form that they can understand, or important in selecting appropriate processing resources for the given content.

This document specifies an identifier mechanism and a registration function for values to be used with that identifier mechanism. It also defines a mechanism for private use values and future extension.

This document replaces RFC 3066, which replaced RFC 1766. For a list of changes in this document, see Section 6.

The keywords "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] [11].
2. The Language Tag

2.1 Syntax

The language tag is composed of one or more parts: A primary language subtag and a (possibly empty) series of subsequent subtags. Subtags are distinguished by their length, position in the subtag sequence, and content, so that each type of subtag can be recognized solely by these features. This makes it possible to construct a parser that can extract and assign some semantic information to the subtags, even if specific subtag values are not recognized. Thus a parser need not have an up-to-date copy of the registered subtag values to perform most searching and matching operations.

The syntax of this tag in ABNF [RFC 2234] is:

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

lang            = 2*3ALPHA           ; shortest ISO 639 code
/ registered-lang
extlang         = 3ALPHA             ; reserved for future use
script          = 4ALPHA             ; ISO 15924 code
region          = 2ALPHA             ; ISO 3166 code
/ 3DIGIT             ; UN country number
variant         = ALPHA (4*7alphanum) ; registered variants
/ DIGIT (3*7alphanum)
extension       = singleton 1*"-" (2*8alphanum)) ; extension subtag(s)
privateuse      = ("x"/"X") 1*"-" (1*8alphanum)) ; private use subtag(s)
singleton       = ("a"-"w" / "y"-"z" / "A"-"W" / "Y"-"Z")
     ; Single letters: x/X is reserved for private use
registered-lang = 4*8ALPHA         ; registered language subtag
grandfathered   = 1*3ALPHA 1*2"-" (2*8alphanum)) ; grandfathered registration
     ; Note: i is the only singleton that starts
alphanum        = (ALPHA / DIGIT)   ; letters and numbers
```

Figure 1: Language Tag ABNF
The character "-" is HYPHEN-MINUS (ABNF: %x2D). Note that there is a subtlety in the ABNF for 'variant': variants may consist of sequences...
of up to eight characters.

Whitespace is not permitted in a language tag. For examples of language tags, see Appendix B.

Note that although [RFC 2234] [13] refers to octets, the language tags described in this document are sequences of characters from the US-ASCII repertoire. Language tags may be used in documents and applications that use other encodings, so long as these encompass the US-ASCII repertoire. An example of this would be an XML document that uses the Unicode UTF-16LE encoding.

The tags and their subtags, including private-use and extensions, are to be treated as case insensitive: there exist conventions for the capitalization of some of the subtags, but these should not be taken to carry meaning.

For example:
- [ISO 639] [1] recommends that language codes be written in lower case ('mn' Mongolian).
- [ISO 3166] [4] recommends that country codes be capitalized ('MN' Mongolia).
- [ISO 15924] [3] recommends that script codes use lower case with the initial letter capitalized ('Cyrl' Cyrillic).

However, in the tags defined by this document, the uppercase US-ASCII letters in the range 'A' (ABNF: %x41) through 'Z' (ABNF: %x5A) are considered equivalent and mapped directly to their US-ASCII lowercase equivalents in the range 'a' (ABNF: %x61) through 'z' (ABNF: %x7A). Thus the tag "mn-Cyrl-MN" is not distinct from "MN-cYRL-mn" or "mN-cYrL-Mn" (or any other combination) and each of these variations conveys the same meaning: Mongolian written in the Cyrillic script as used in Mongolia.

For informative examples of language tags, see Appendix B at the end of this document.

### 2.1.1 Length Considerations

Although neither the ABNF nor other guidelines in this document provide a fixed upper limit on the number of size of subtags in a Language Tag and it is possible to envision quite long and complex subtag sequences, in practice these are rare because additional granularity in tags seldom adds useful distinguishing information and because longer, more granular tags interfere with the meaning, understanding, and processing of language tags.

In particular, variant subtags SHOULD be used only with their recommended prefix. This limits most tags to a sequence of four
subtags (excluding any extensions or private use sequences). See Section 2.3 for more information on selecting the most appropriate Language Tag.

A conformant implementation need not support the storage of language tags which exceed a specified length. For an example, see [RFC 2231][12]. Any such a limitation MUST be clearly documented, and such documentation SHOULD include the disposition of any longer tags (for example, whether an error value is generated or the language tag is truncated). If truncation is permitted it SHOULD NOT permit a subtag to be divided.

### 2.2 Language Subtag Sources and Interpretation

The namespace of language tags and their subtags is administered by the Internet Assigned Numbers Authority (IANA)[17] according to the rules in Section 3 of this document. The registry maintained by IANA is the source for valid subtags: other standards referenced in this section provide the source material for that registry.

Terminology in this section:

- **Tag or tags** refers to a complete language tag, such as "fr-Latn-CA". Examples of tags in this document are enclosed in double-quotes ("en-US").
- **Subtag** refers to a specific section of a tag, separated by hyphen, such as the subtag 'Latn' in "fr-Latn-CA". Examples of subtags in this document are enclosed in single quotes ('Latn').
- **Code or codes** refers to tags defined in external standards (and which are used as subtags in this document). For example, 'Latn' is an [ISO 15924][3] script code which was used to define the 'Latn' script subtag for use in a language tag. Examples of codes in this document are enclosed in single quotes ('en', 'Latn').

The definitions in this section apply to the various subtags within the language tags defined by this document, excepting those "grandfathered" tags defined in Section 2.2.8.

Language tags are designed so that each subtag has unique length and content restrictions. These make identification of the subtag's type possible, even if the content of the subtag itself is unrecognized. This allows tags to be parsed and processed without reference to the latest version of the underlying standards or the IANA registry and makes the associated exception handling when parsing tags simpler.

Subtags in the IANA registry that do not come from an underlying standard can only appear in specific positions in a tag. Specifically, they can only occur as primary language subtags or as
variant subtags.

Note that sequences of private-use and extension subtags MUST occur at the end of the sequence of subtags and MUST NOT be interspersed with subtags defined elsewhere in this document.

Single letter and digit subtags are reserved for current or future use. These include the following current uses:

- The single letter subtag 'x' is reserved to introduce a sequence of private-use subtags. The interpretation of any private-use subtags is defined solely by private agreement and is not defined by the rules in this section or in any standard or registry defined in this document.
- All other single letter subtags are reserved to introduce standardized extension subtag sequences as described in Section 3.4.

The single letter subtag 'i' is used by some grandfathered tags, such as "i-enochian", where it always appears in the first position and cannot be confused with an extension.

### 2.2.1 Primary Language Subtag

The primary subtag is the first subtag in a language tag and cannot be empty. Except as noted, the primary subtag is the language subtag. The following rules apply to the assignment and interpretation of the primary subtag:

- All 2-character language subtags were defined in the IANA registry according to the assignments found in the standard ISO 639 Part 1, "ISO 639-1:2002, Codes for the representation of names of languages -- Part 1: Alpha-2 code" [ISO 639-1] [1], or using assignments subsequently made by the ISO 639 Part 1 maintenance agency or governing standardization bodies.
- All 3-character language subtags were defined in the IANA registry according to the assignments found in ISO 639 Part 2, "ISO 639-2:1998 - Codes for the representation of names of languages -- Part 2: Alpha-3 code - edition 1" [ISO 639-2] [2], or assignments subsequently made by the ISO 639 Part 2 maintenance agency or governing standardization bodies.
- The subtags in the range 'qaa' through 'qtz' are reserved for private use in language tags. These subtags correspond to codes reserved by ISO 639-2 for private use. These codes MAY be used for non-registered primary-language subtags (instead of using private-use subtags following 'x-'). Please refer to Section 2.5 for more information on private use subtags.
All language subtags of 4 to 8 characters in length in the IANA registry were defined via the registration process in Section 3.3 and MAY be used to form the primary language subtag. At the time this document was created, there were no examples of this kind of subtag and future registrations of this type will be discouraged: primary languages are STRONGLY RECOMMENDED for registration with ISO 639 and subtags rejected by ISO 639 will be closely scrutinized before they are registered with IANA.

The single character subtag 'x' as the primary subtag indicates that the language tag consists solely of subtags whose meaning is defined by private agreement. For example, in the tag "x-fr-CH", the subtags 'fr' and 'CH' should not be taken to represent the French language or the country of Switzerland (or any other value in the IANA registry) unless there is a private agreement in place to do so. See Section 2.5.

Other values MUST NOT be assigned to the primary subtag except by revision or update of this document.

Note: For languages that have both an ISO 639-1 2-character code and an ISO 639-2 3-character code, only the ISO 639-1 2-character code is defined in the IANA registry.

Note: For languages that have no ISO 639-1 2-character code and for which the ISO 639-2/T (Terminology) code and the ISO 639-2/B (Bibliographic) codes differ, only the Terminology code is defined in the IANA registry. At the time this document was created, all languages that had both kinds of 3-character code were also assigned a 2-character code; it is not expected that future assignments of this nature will occur.

Note: To avoid problems with versioning and subtag choice as experienced during the transition between RFC 1766 and RFC 3066, as well as the canonical nature of subtags defined by this document, the ISO 639 Registration Authority Joint Advisory Committee (ISO 639/RA-JAC) has included the following statement in [6]:

"A language code already in ISO 639-2 at the point of freezing ISO 639-1 shall not later be added to ISO 639-1. This is to ensure consistency in usage over time, since users are directed in Internet applications to employ the alpha-3 code when an alpha-2 code for that language is not available."

In order to avoid instability of the canonical form of tags, if a 2-character code is added to ISO 639-1 for a language for which a 3-character code was already included in ISO 639-2, the 2-character code will not be added as a subtag in the registry. See Section 3.2.

For example, if some content were tagged with 'haw' (Hawaiian), which
currently has no 2-character code, the tag would not be invalidated if ISO 639-1 were to assign a 2-character code to the Hawaiian language at a later date.

For example, one of the grandfathered IANA registrations is "i-enochian". The subtag 'enochian' could be registered in the IANA registry as a primary language subtag (assuming that ISO 639 does not register this language first), making tags such as "enochian-AQ" and "enochian-Latn" valid.

### 2.2.2 Extended Language Subtags

The following rules apply to the extended language subtags:

- Three letter subtags immediately following the primary subtag are reserved for future standardization, anticipating work that is currently under way on ISO 639.
- Extended language subtags MUST follow the primary subtag and precede any other subtags.
- There MAY be any additional number of extended language subtags.
- Extended language subtags will not be registered except by revision of this document.
- Extended language subtags MUST NOT be used to form language tags except by revision of this document.

Example: In a future revision or update of this document, the tag "zh-gan" (registered under RFC 3066) might become a valid non-grandfathered tag in which the subtag 'gan' might represent the Chinese dialect 'Gan'.

### 2.2.3 Script Subtag

The following rules apply to the script subtags:

- All 4-character subtags were defined according to ISO 15924 [3]--"Codes for the representation of the names of scripts": alpha-4 script codes, or subsequently assigned by the ISO 15924 maintenance agency or governing standardization bodies, denoting the script or writing system used in conjunction with this language.
- Script subtags MUST immediately follow the primary language subtag and all extended language subtags and MUST occur before any other type of subtag described below.
- The subtags 'Qaaa' through 'Qabx' are reserved for private use in language tags. These subtags correspond to codes reserved by ISO 15924 for private use. These codes MAY be used for non-registered script values. Please refer to Section 2.5 for more information on private-use subtags.
Script subtags cannot be registered using the process in Section 3.3 of this document. Variant subtags may be considered for registration for that purpose.

Example: "de-Latn" represents German written using the Latin script.

### 2.2.4 Region Subtag

The following rules apply to the region subtags:

- The region subtag defines language variations used in a specific region, geographic, or political area. Region subtags MUST follow any language, extended language, or script subtags and MUST precede all other subtags.
- All 2-character subtags following the primary subtag were defined in the IANA registry according to the assignments found in ISO 3166 [4] - "Codes for the representation of names of countries and their subdivisions - Part 1: Country codes"--alpha-2 country codes or assignments subsequently made by the ISO 3166 maintenance agency or governing standardization bodies.
- All 3-character codes consisting of digit (numeric) characters were defined in the IANA registry according to the assignments found in UN Standard Country or Area Codes for Statistical Use [5] or assignments subsequently made by the governing standards body. Note that not all of the UN M.49 codes are defined in the IANA registry:
  * UN numeric codes assigned to 'macro-geographical (continental)' or sub-regions not associated with an assigned ISO 3166 alpha-2 code *are* defined.
  * UN numeric codes for 'economic groupings' or 'other groupings' are *not* defined in the IANA registry and MUST NOT be used to form language tags.
  * Countries with ambiguous ISO 3166 alpha-2 codes as defined in Section 3.2 are defined in the registry and are canonical for the given country or region defined.
  * The alphanumeric codes in Appendix X of the UN document are *not* defined and MUST NOT be used to form language tags. (At the time this document was created these values match the ISO 3166 alpha-2 codes.)
- There may be at most one region subtag in a language tag.
- The subtags 'AA', 'QM'- 'QZ', 'XA'- 'XZ', and 'ZZ' are reserved for private use in language tags. These subtags correspond to codes reserved by ISO 3166 for private use. These codes MAY be used for private use region subtags (instead of using a private-use subtag sequence). Please refer to Section 2.5 for more information on private use subtags.

"de-Latn-CH" represents German ('de') written using the Latin script.
('Latn') as used in Switzerland ('CH').

"sr-Latn-CS" represents Serbian ('sr') written using Latin script ('Latn') as used in Serbia and Montenegro ('CS').

"es-419" represents Spanish ('es') as used in the UN-defined Latin America and Caribbean region ('419').

### 2.2.5 Variant Subtags

The following rules apply to the variant subtags:

- Variant subtags, as a collection in the IANA registry, are not associated with any external standard. Variant subtags and their meanings are defined by the registration process defined in Section 3.3.
- Variant subtags MUST follow all of the other defined subtags, but precede any extension or private-use subtag sequences.
- More than one variant MAY be used to form the language tag.
- Variant subtags MUST be registered with IANA according to the rules in Section 3.3 of this document before being used to form language tags. In order to distinguish variants from other types of subtags, registrations must meet the following length and content restrictions:
  * Variant subtags that begin with a letter (a-z, A-Z) MUST be at least five characters long.
  * Variant subtags that begin with a digit (0-9) MUST be at least four characters long.
  * The maximum length of a variant subtag is eight characters long.

"en-boont" represents the Boontling dialect of English.

"de-CH-1996" represents German as used in Switzerland and as written using the spelling reform beginning in the year 1996 C.E.

### 2.2.6 Extension Subtags

The following rules apply to extensions:

- Extension subtags are separated from the other subtags defined in this document by a single-letter subtag ("singleton"). The singleton MUST be one allocated to a registration authority via the mechanism described in Section 3.4 and cannot be the letter 'x', which is reserved for private-use subtag sequences.
- Note: Private-use subtag sequences starting with the singleton subtag 'x' are described below.
An extension MUST follow at least a primary language subtag. That is, a language tag cannot begin with an extension. Extensions extend language tags, they do not override or replace them. For example, "a-value" is not a well-formed language tag, while "de-a-value" is.

Each singleton subtag MUST appear at most one time in each tag (other than as a private-use subtag). That is, singleton subtags MUST NOT be repeated. For example, the tag "en-a-bbb-a-ccc" is invalid because the subtag 'a' appears twice.

Extension subtags MUST meet all of the requirements for the content and format of subtags defined in this document. Extension subtags MUST meet whatever requirements are set by the document that defines their singleton prefix and whatever requirements are provided by the maintaining authority.

Each extension subtag MUST be from two to eight characters long and consist solely of letters or digits, with each subtag separated by a single '-'.

Each singleton MUST be followed by at least one extension subtag. For example, the tag "tlh-a-b-foo" is invalid because the first singleton 'a' is followed immediately by another singleton 'b'.

Extension subtags MUST follow all language, extended language, script, region, and variant subtags in a tag.

All subtags following the singleton and before another singleton are part of the extension. Example: In the tag "fr-a-Latn", the subtag 'Latn' does not represent the script subtag 'Latn' defined in the IANA Language Subtag Registry. Its meaning is defined by the extension 'a'.

In the event that more than one extension appears in a single tag, the tag SHOULD be canonicalized as described in Section 2.4.1.

For example, if the prefix singleton 'r' and the shown subtags were defined, then the following tag would be a valid example: "en-Latn-GB-boont-r-extended-sequence-x-private"

2.2.7 Private Use Subtags

The following rules apply to private-use subtags:

- Private-use subtags are separated from the other subtags defined in this document by the reserved single-character subtag 'x'.
- Private-use subtags MUST follow all language, extended language, script, region, variant, and extension subtags in the tag. Another way of saying this is that all subtags following the singleton 'x' MUST be considered private use. Example: The subtag 'US' in the tag "en-x-US" is a private use subtag.
- Unlike Extensions, a tag MAY consist entirely of private-use subtags.
No source is defined for private use subtags. Use of private use subtags is by private agreement and SHOULD NOT be considered part of this document.

For example: Users who wished to utilize SIL Ethnologue for identification might agree to exchange tags such as "az-Arab-x-AZE-derbend". This example contains two private-use subtags. The first is 'AZE' and the second is 'derbend'.

2.2.8 Pre-Existing RFC 3066 Registrations

Existing IANA-registered language tags from RFC 1766 and/or RFC 3066 that are not defined by additions to this document maintain their validity. IANA will maintain these tags in the registry under either the "grandfathered" or "redundant" type. For more information see Appendix C.

It is important to note that all language tags formed under the guidelines in this document were either legal, well-formed tags or were valid for potential registration under RFC 3066.

2.2.9 Possibilities for Registration

Possibilities for registration of subtags include:

- Primary language subtags for languages not listed in ISO 639 that are not variants of any listed or registered language, can be registered. At the time this document was created there were no examples of this form of subtag. Before attempting to register a language subtag, there MUST be an attempt to register the language with ISO 639. No language subtags will be registered for codes that exist in ISO 639-1 or ISO 639-2, which are under consideration by the ISO 639 maintenance or registration authorities, or which have never been attempted for registration with those authorities. If ISO 639 has previously rejected a language for registration, it is reasonable to assume that there MUST be additional very compelling evidence of need before it will be registered in the IANA registry (to the extent that it is very unlikely that any subtags will be registered of this type).

- Dialect or other divisions or variations within a language, its orthography, writing system, regional variation, or historical usage may be registered as variant subtags. An example is the 'scouse' subtag (the Scouse dialect of English).

This document leaves the decision on what subtags are appropriate or not to the registration process described in Section 3.3.

ISO 639 defines a maintenance agency for additions to and changes in
the list of languages in ISO 639. This agency is:

International Information Centre for Terminology (Infoterm)
Aichholzgasse 6/12, AT-1120
Wien, Austria
Phone: +43 1 26 75 35 Ext. 312 Fax: +43 1 216 32 72

ISO 639-2 defines a maintenance agency for additions to and changes in the list of languages in ISO 639-2. This agency is:

Library of Congress
Network Development and MARC Standards Office
Washington, D.C. 20540 USA
Phone: +1 202 707 6237 Fax: +1 202 707 0115
URL: http://www.loc.gov/standards/iso639

The maintenance agency for ISO 3166 (country codes) is:

ISO 3166 Maintenance Agency
c/o International Organization for Standardization
Case postale 56
CH-1211 Geneva 20 Switzerland
Phone: +41 22 749 72 33 Fax: +41 22 749 73 49

The registration authority for ISO 15924 (script codes) is:

Unicode Consortium Box 391476
Mountain View, CA 94039-1476, USA
URL: http://www.unicode.org/iso15924

The Statistics Division of the United Nations Secretariat maintains the Standard Country or Area Codes for Statistical Use and can be reached at:

Statistical Services Branch
Statistics Division
United Nations, Room DC2-1620
New York, NY 10017, USA
Fax: +1-212-963-0623
E-mail: statistics@un.org
URL: http://unstats.un.org/unsd/methods/m49/m49alpha.htm

### 2.2.10 Classes of Conformance

Implementations may wish to express their level of conformance with the rules and practices described in this document. There are
generally two classes of conforming implementations: "well-formed" processors and "validating" processors. Claims of conformance SHOULDN'T explicitly reference one of these definitions.

An implementation that claims to check for well-formed language tags MUST:
1. Check that the tag and all of its subtags, including extension and private-use subtags, conform to the ABNF or that the tag is on the list of grandfathered tags.
2. Check that singleton subtags that identify extensions do not repeat. For example, the tag "en-a-xx-b-yy-a-zz" is not well-formed.

Well-formed processors are strongly encouraged to implement the canonicalization rules contained in Section 2.4.1.

An implementation that claims to be validating MUST:
1. Check that the tag is well-formed.
2. Specify the particular registry date for which the implementation performs validation of subtags.
3. Check that either the tag is a grandfathered tag, or that all language, script, region, and variant subtags consist of valid codes for use in language tags according to the IANA registry as of the particular date specified by the implementation.
4. Specify which, if any, extension RFCs as defined in Section 3.4 are supported, including version, revision, and date.
5. For any such extensions supported, check that all subtags used in that extension are valid.
6. If the processor generates tags, it MUST do so in canonical form, including any supported extensions, as defined in Section 2.4.1.

### 2.3 Choice of Language Tag

One may occasionally be faced with several possible tags for the same body of text.

Interoperability is best served when all users use the same language tag in order to represent the same language. If an application has requirements that make the rules here inapplicable, then that application risks damaging interoperability. It is STRONGLY RECOMMENDED that users not define their own rules for language tag choice.

Standards, protocols and applications that reference this document normatively but apply different rules to the ones given in this section MUST specify how the procedure varies from the one given here.
1. Use as precise a tag as possible, but no more specific than is justified. For example, 'de' might suffice for tagging an email written in German, while "de-CH-1996" is probably unnecessarily precise for such a task.

2. Avoid using subtags that are not important for distinguishing content in an application. For example, including the script subtag in "en-Latn-US" is generally unnecessary, since nearly all English texts are written in the Latin script and it is generally not important to filter out those few that are not.

3. Use the canonical subtag from the IANA registry in preference to any of its aliases. For example, you should use 'he' for Hebrew in preference to 'iw'.

4. You SHOULD NOT use the 'UND' (Undetermined) language subtag to label content, even if the language is unknown. Omitting the tag is preferred. Some protocols may force you to give a value for the language tag and the 'UND' subtag may be useful when matching language tags in certain situations.

5. You SHOULD NOT use the 'MUL' (Multiple) subtag if the protocol allows you to use multiple languages, as is the case for the Content-Language header in HTTP.

6. You SHOULD NOT use the same variant subtag more than once within a language tag. For example, you should not use "en-US-boont-boont".

To ensure consistent backward compatibility, this document contains several provisions to account for potential instability in the standards used to define the subtags that make up language tags. These provisions mean that no language tag created under the rules in this document will become obsolete. In addition, tags that are in canonical form will always be in canonical form.

2.4 Meaning of the Language Tag

The language tag always defines a language as spoken (or written, signed or otherwise signaled) by human beings for communication of information to other human beings. Computer languages such as programming languages are explicitly excluded.

If a language tag B contains language tag A as a prefix, then B is typically "narrower" or "more specific" than A. For example, "zh-Hant-TW" is more specific than "zh-Hant".

This relationship is not guaranteed in all cases: specifically, languages that begin with the same sequence of subtags are NOT guaranteed to be mutually intelligible, although they may be. For example, the tag "az" shares a prefix with both "az-Latn" (Azerbaijani written using the Latin script) and "az-Cyrl" (Azerbaijani written using the Cyrillic script). A person fluent in
one script may not be able to read the other, even though the text might be identical. Content tagged as "az" most probably is written in just one script and thus might not be intelligible to a reader familiar with the other script.

The relationship between the tag and the information it relates to is defined by the standard describing the context in which it appears. Accordingly, this section can only give possible examples of its usage.

- For a single information object, the associated language tags might be interpreted as the set of languages that is required for a complete comprehension of the complete object. Example: Plain text documents.
- For an aggregation of information objects, the associated language tags could be taken as the set of languages used inside components of that aggregation. Examples: Document stores and libraries.
- For information objects whose purpose is to provide alternatives, the associated language tags could be regarded as a hint that the content is provided in several languages, and that one has to inspect each of the alternatives in order to find its language or languages. In this case, the presence of multiple tags might not mean that one needs to be multi-lingual to get complete understanding of the document. Example: MIME multipart/alternative.
- In markup languages, such as HTML and XML, language information can be added to each part of the document identified by the markup structure (including the whole document itself). For example, one could write <span lang="FR">C'est la vie.</span> inside a Norwegian document; the Norwegian-speaking user could then access a French-Norwegian dictionary to find out what the marked section meant. If the user were listening to that document through a speech synthesis interface, this formation could be used to signal the synthesizer to appropriately apply French text-to-speech pronunciation rules to that span of text, instead of misapplying the Norwegian rules.

### 2.4.1 Canonicalization of Language Tags

Since a particular language tag may be used in many processes, language tags SHOULD always be created or generated in a canonical form.

A language tag is in canonical form when:

1. The tag is well-formed according the rules in Section 2.1 and Section 2.2.
2. None of the subtags in the language tag has a canonical_value mapping in the IANA registry (see Section 3.1). Subtags with a canonical_value mapping MUST be replaced with their mapping in
order to canonicalize the tag.

3. If more than one extension subtag sequence exists, the extension
   sequences are ordered into case-insensitive ASCII order by
   singleton subtag.

Example: The language tag "en-A-aaa-B-ccc-bbb-x-xyz" is in canonical
form, while "en-B-ccc-bbb-A-aaa-X-xyz" is well-formed but not in
canonical form.

Example: The language tag "en-NH" (English as used in the New
Hebrides) is not canonical because the 'NH' subtag has a canonical
mapping to 'VU' (Vanuatu).

Note: Canonicalization of language tags does not imply anything about
the use of upper or lowercase letter in subtags as described in
Section 2.1. All comparisons MUST be performed in a case-insensitive
manner.

An extension MUST define any relationships that may exist between the
various subtags in the extension and thus MAY define an alternate
canonicalization scheme for the extension's subtags. Extensions MAY
define how the order of the extension's subtags are interpreted. For
example, an extension could define that its subtags are in canonical
order when the subtags are placed into ASCII order: that is,
"en-a-aaa-bbb-ccc" instead of "en-a-ccc-bbb-aaa". Another extension
might define that the order of the subtags influences their semantic
meaning (so that "en-b-ccc-bbb-aaa" has a different value from
"en-b-aaa-bbb-ccc"). However, extension specifications SHOULD be
designed so that they are tolerant of the typical processes described in
Section 3.4.

2.5 Considerations for Private Use Subtags

Private-use subtags require private agreement between the parties
that intend to use or exchange language tags that use them and great
care should be used in employing them in content or protocols
intended for general use. Private-use subtags are simply useless for
information exchange without prior arrangement.

The value and semantic meaning of private-use tags and of the subtags
used within such a language tag are not defined by this document.

The use of subtags defined in the IANA registry as having a specific
private use meaning convey more information that a purely private use
tag prefixed by the singleton subtag 'x'. For applications this
additional information may be useful.

For example, the region subtags 'AA', 'ZZ' and in the ranges
'QM'-'QZ' and 'XA'-'XZ' (derived from ISO 3166 private use codes) may be used to form a language tag. A tag such as "zh-Hans-XQ" conveys a great deal of public, interchangeable information about the language material (that it is Chinese in the simplified Chinese script and is suitable for some geographic region 'XQ'). While the precise geographic region is not known outside of private agreement, the tag conveys far more information than an opaque tag such as "x-someLang", which contains no information about the language subtag or script subtag outside of the private agreement.

However, in some cases content tagged with private use subtags may interact with other systems in a different and possibly unsuitable manner compared to tags that use opaque, privately defined subtags, so the choice of the best approach may depend on the particular domain in question.
3. IANA Considerations

This section deals with the processes and requirements necessary to maintain the registry of subtags and extensions for use in language tags as defined by this document and in accordance with the requirements of RFC 2434 [15].

The language subtag registry will be maintained so that, except for extension subtags, it is possible to validate all of the subtags that appear in a language tag under the provisions of this document or its revisions or successors. In addition, the meaning of the various subtags will be unambiguous and stable over time. (The meaning of private-use subtags, of course, is not defined by the IANA registry.)

The registry defined under this document contains a comprehensive list of all the subtags valid in language tags. This allows implementers a straightforward and reliable way to validate language tags.

3.1 Format of the IANA Language Subtag Registry

The IANA Language Subtag Registry will consist of a text file that is machine readable in the format described in this section, plus copies of the registration forms approved by the Language Subtag Reviewer in accordance with the process described in Section 3.3. With the exception of the registration forms for grandfathered and redundant tags, no registration records will be maintained for the initial set of subtags.

Each record in the subtag registry will consist of a series of fields separated by the symbol "|" (%x7D) and terminated by a newline. Text appearing after the symbol "#" (%x23) contains comments. Whitespace surrounding fields in the file is ignored. If a field contains more than one value, the values are separated by semicolons (%x3B).

There is a single date record at the start of the file which indicates the most recent modification date of the file. It has two fields: the type field is "date", and the second field is the modification date, in the "full-date" format specified in RFC 3339 [19]. For example: 2004-06-28 represents June 28, 2004 in the Gregorian calendar:

```
| date | 2004-06-28 |
```

The fields in each subtag record, in order, are:

```
| type| subtag| description| date| canonical_value| recommended_prefix # comments |
```

The character "vertical line" ("|", %x7D) delimits each of the fields.

Empty fields (and their separators) at the end of the record may be omitted.

Leading or trailing whitespace in each field is not part of the content.

When the type is "grandfathered" or "redundant", then the subtag field is actually a whole tag.

The "recommended_prefix" field is empty, except where the type is "variant".

The "comments" field is optional and appears only at the end of a record, following a "number sign" ("#", %x23).

The sequence '..' denotes a range of values. Such a range represents all subtags of the same length that are alphabetically within that range, including the values explicitly mentioned. For example 'a..c' denotes the values 'a', 'b', and 'c'.

The field 'type' MUST consist of one of the following strings: "language", "extlang", "script", "region", "variant", "grandfathered", and "redundant" and denotes the type of subtag (or tag, in the case of "grandfathered" and "redundant").

The field 'subtag' contains the subtag being defined.

The field 'description' contains a description of the subtag transcribed into ASCII.

Note: Descriptions in registry entries that correspond to ISO 639, ISO 15924, ISO 3166 or UN M.49 codes are intended only to indicate the meaning of that identifier as defined in the source standard at the time it was added to the registry. The description does not replace the content of the source standard itself. The descriptions are not intended to be the English localized names for the subtags and localization or translation of language tag and subtag descriptions is out of scope of this document.

The field 'date' contains the date the record was added to the registry in the "full-date" format specified in RFC 3339 [20]. For example: 2004-06-28 represents June 28, 2004, in the Gregorian calendar.

The field 'canonical value' represents a canonical mapping of this record to a subtag record of the same 'type'. Note that this field SHALL NOT be modified (except for records of type "grandfathered") therefore a subtag whose record contains no canonical mapping when the record is created is a canonical form and will remain so.

The field 'recommended prefix' is for use with registered variants
and contains a semicolon separated list of language-ranges considered most appropriate for use with this subtag. Additional values can be added to this field for variants only via additional registration. Other modification of this field (such as removing or changing values) is not permitted.

The field 'comments' may contain additional information about the subtag, as deemed appropriate for understanding the registry and implementing language tags using the various subtags. These values can be changed via the registration process and no guarantee of stability is provided.

# IANA Language Subtag Registry
# This registry lists all valid subtags for language tags
# created under RFC XXXX.
date| 2004-08-07

# language codes: ISO 639 and registered codes

# ISO 639-1 (alpha-2) codes
language| aa| Afar| 2004-07-06| |
language| ab| Abkhazian| 2004-07-06| |
language| ae| Avestan| 2004-07-06| |
language| he| hebrew| 2004-06-28| |
language| iw| hebrew| 2004-06-28| he | #note mapping
language| qaa..qtz| PRIVATE USE| 2004-07-06| |
language| raj| Rajasthani| 2004-07-06| |
language| seuss| Hypothetical Language| 2005-04-01 ||# registered language

# script codes: ISO 15924

script| Arab| Arabic| 2004-07-06| |
script| Armn| Armenian| 2004-07-06| |
script| Bali| Balinese| 2004-07-06| |

# region codes: ISO 3166 and UN codes

# ISO 3166-1 alpha-2 codes
region| AA| PRIVATE USE| 2004-08-01| |
region| AD| Andorra| 2004-07-06| |
region| AE| United Arab Emirates| 2004-07-06| |
region| AF| Afghanistan| 2004-07-06| |
region| CS| Serbia and Montenegro| 2003-07-23| |
region| YU| Yugoslavia| 2004-06-28| |

# United Nations M.49 numeric codes
region| 001| World| 2004-07-06| |
region| 002| Africa| 2004-07-06| |
region| 003| North America| 2004-07-06| |
region| 005| South America| 2004-07-06| |
region| 200| Czechoslovakia| 2004-07-06| | #formerly used code CS

## registered variants

variant| boont| Boontling| 2003-02-14| | en
variant| gaulish| Gaulish| 2001-05-25| | cel
variant| guoyu| Mandarin or Standard Chinese| 1999-12-18| | zh

# grandfathered from RFC 3066

grandfathered| en-GB-oed| English, Oxford English Dictionary spelling| 2003-07-09| |
grandfathered| i-ami| Amis| 1999-05-25| |
grandfathered| i-bnn| Bunun| 1999-05-25| |

# redundant

# The following codes were registered as complete tags, but can now be
# composed of registered subtags and do not require registration.

redundant| art-lojban| Lojban| 2001-11-11| | # use language art + variant lojban
redundant| az-Arab| Azerbaijani in Arabic script| 2003-05-30| | # use language az + script Arab
redundant| az-Cyrl| Azerbaijani in Cyrillic script| 2003-05-30| | # use language az + script Cyril
redundant| en-boont| Boontling| 2003-02-14| | # use language en + variant boont

Figure 2: Example of the Registry Format

Maintenance of the registry requires that as new codes are assigned
by ISO 639, ISO 15924, and ISO 3166, the Language Subtag Reviewer
will evaluate each assignment, determine whether it conflicts with
existing registry entries, and submit the information to IANA for
inclusion in the registry.

Note: The redundant and grandfathered entries together are the
complete list of tags registered under RFC 3066 [18]. The redundant
tags are those that can now be formed using the subtags defined in
Section 2.2. The grandfathered entries are those that can never be
legal under those same provisions. The items in both lists are
permanent and stable, although grandfathered items may be deprecated
over time. Refer to Appendix C for more information.

The Language Subtag Reviewer MUST ensure that new subtags meet the
requirements in Section 2.3 or submit an appropriate alternate subtag
as described in that section. She or he will use the following form to submit this information:
The field 'record text' contains the exact record that IANA is to insert into the Language Subtag Registry. The contents of the remaining fields must exactly match those in this field.

Whenever an entry is created or modified in the registry, the 'date' record at the start of the registry is updated to reflect the most recent modification date in the RFC 3339 [20] "full-date" format.

3.2 Stability of IANA Registry Entries

The stability of entries and their meaning in the registry is critical to the long term stability of language tags. The rules in this section guarantee that a specific language tag's meaning is stable over time and will not change and that the choice of language tag for specific content is also stable over time.

These rules specifically deal with how changes to codes (including withdrawal and deprecation of codes) maintained by ISO 639, ISO 15924, ISO 3166, and UN M.49 are reflected in the IANA Language Subtag Registry. Assignments to the IANA Language Subtag Registry MUST follow the following stability rules:

- Values in the fields 'type', 'subtag', 'date' and 'canonical value' MUST NOT be changed and are guaranteed to be stable over time.

- Values in the 'description' field MUST NOT be changed in a way that would invalidate previously-existing tags. They may be broadened somewhat in scope, changed to add information, or adapted to the most common modern usage. For example, countries occasionally change their official names: an historical example of this would be "Upper Volta" changing to "Burkina Faso".

- Values in the field 'recommended prefix' MAY be added via the registration process.

- Values in the field 'recommended prefix' MAY be modified, so long as the modifications broaden the set of recommended prefixes. That is, a recommended prefix MAY be replaced by one of its own prefixes. For example, the prefix "en-US" could be replaced by
"en", but not by the ranges "en-Latn", "fr", or "en-US-boont".
- Values in the field 'recommended prefix' MUST NOT be removed.
- The field 'comments' MAY be added, changed, modified, or removed via the registration process or any of the processes or considerations described in this section.
- Codes assigned by ISO 639, ISO 15924, and ISO 3166 that do not conflict with existing subtags of the associated type and whose meaning is not the same as an existing subtag of the same type are entered into the IANA registry as new records and their value is canonical for the meaning assigned to them.
- Codes assigned by ISO 639, ISO 15924, or ISO 3166 that are withdrawn by their respective maintenance or registration authority remain valid in language tags. The registration process MAY be used to add a note indicating the withdrawal of the code by the respective standard.
- Codes assigned by ISO 639, ISO 15924, or ISO 3166 that do not conflict with existing subtags of the associated type but which represent the same meaning as an existing subtag of that type are entered into the IANA registry as new records. The field 'canonical value' for that record MUST contain the existing subtag of the same meaning
  Example If ISO 3166 were to assign the code 'IM' to represent the value "Isle of Man" (represented in the IANA registry by the UN M.49 code '833'), '833' remains the canonical subtag and 'IM' would be assigned '833' as a canonical value. This prevents tags that are in canonical form from becoming non-canonical.
  Example If the tag 'enochian' were registered as a primary language subtag and ISO 639 subsequently assigned an alpha-3 code to the same language, the new ISO 639 code would be entered into the IANA registry as a subtag with a canonical mapping to 'enochian'. The new ISO code can be used, but it is not canonical.
- Codes assigned by ISO 639, ISO 15924, or ISO 3166 that conflict with existing subtags of the associated type MUST NOT be entered into the registry. The following additional considerations apply:
  * For ISO 639 codes, if the newly assigned code's meaning is not represented by a subtag in the IANA registry, the Language Subtag Reviewer, as described in Section 3.3, shall prepare a proposal for entering in the IANA registry as soon as practical a registered language subtag as an alternate value for the new code. The form of the registered language subtag will be at the discretion of the Language Subtag Reviewer and must conform to other restrictions on language subtags in this document.
  * For all subtags whose meaning is derived from an external standard (i.e. ISO 639, ISO 15924, ISO 3166, or UN M.49), if a new meaning is assigned to an existing code and the new meaning broadens the meaning of that code, then the meaning for the associated subtag MAY be changed to match. The meaning of a
subtag MUST NOT be narrowed, however, as this can result in an
unknown proportion of the existing uses of a subtag becoming
invalid. Note: ISO 639 MA/RA has adopted a similar stability
policy.

* For ISO 15924 codes, if the newly assigned code's meaning is
not represented by a subtag in the IANA registry, the Language
Subtag Reviewer, as described in Section 3.3, shall prepare a
proposal for entering in the IANA registry as soon as practical
a registered variant subtag as an alternate value for the new
code. The form of the registered variant subtag will be at the
discretion of the Language Subtag Reviewer and must conform to
other restrictions on variant subtags in this document.

* For ISO 3166 codes, if the newly assigned code's meaning is
associated with the same UN M.49 code as another 'region'
subtag, then the existing region subtag remains as the
canonical entry for that region and no new entry is created. A
note MAY be added to the existing region subtag indicating the
relationship to the new ISO 3166 code.

* For ISO 3166 codes, if the newly assigned code's meaning is
associated with a UN M.49 code that is not represented by an
existing region subtag, then the Language Subtag Reviewer,
as described in Section 3.3, shall prepare a proposal for
entering the appropriate numeric UN country code as an entry in
the IANA registry.

* For ISO 3166 codes, if there is no associated UN numeric code,
then the Language Subtag Reviewer SHALL petition the UN to
create one. If there is no response from the UN within ninety
days of the request being sent, the Language Subtag Reviewer
shall prepare a proposal for entering in the IANA registry as
soon as practical a registered variant subtag as an alternate
code. The form of the registered variant
subtag will be at the discretion of the Language Subtag
Reviewer and must conform to other restrictions on variant
subtags in this document. This situation is very unlikely to
ever occur.

o Stability provisions apply to grandfathered tags with this
exception: should all of the subtags in a grandfathered tag become
valid subtags in the IANA registry, then the grandfathered tag
MUST be marked as redundant. Note that this will not affect
language tags that match the grandfathered tag, since these tags
will now match valid generative subtag sequences. For example, if
the subtag 'gan' in the language tag "zh-gan" were to be
registered as an extended language subtag, then the grandfathered
tag "zh-gan" would be deprecated (but existing content or
implementations that use "zh-gan" would remain valid).

Language tags formed under RFC 3066 that use the region subtag 'CS'
were ambiguous, since tags produced before 2003 used that code for
the (now dissolved) country Czechoslovakia. ISO 3166 assigned this
code to the country Serbia and Montenegro in 2003 and this draft
makes that the canonical value for this subtag. To form a language
tag for the region Czechoslovakia, the UN M.49 code '200' is included
in the registry. As a practical matter, applications that encounter
the RFC 3066 tag "cs-CS" or "sk-CS" MAY wish to convert that to
"cs-200" or "sk-200" (or use one of the successor region subtags,
such as 'CZ' or 'SK'), since that is the most likely interpretation.

3.3 Registration Procedure for Subtags

The procedure given here MUST be used by anyone who wants to use a
subtag not currently in the IANA Language Subtag Registry.

Only primary language and variant subtags will be considered for
independent registration. (Subtags required for stability and
subtags required to keep the registry synchronized with ISO 639, ISO
15924, ISO 3166, and UN M.49 within the limits defined by this
document are the only exceptions to this. See Section 3.2.)

This procedure MAY also be used to register or alter the information
for the "description", "note", or "recommended prefix" fields in a
subtag's record as described in Figure 2. Changes to all other
fields in the IANA registry are NOT permitted.

If registering a new language subtag, the process starts by filling
out the registration form reproduced below. Note that each response
is not limited in size and should take the room necessary to
adequately describe the registration.

LANGUAGE SUBTAG REGISTRATION FORM
1. Name of requester:
2. E-mail address of requester:
3. Subtag to be registered:
4. Type of Registration:
   [ ] language
   [ ] variant
5. Description of subtag (in English or transcribed into ASCII):
6. Intended meaning of the subtag:
7. Recommended prefix(es) of subtag (for variants):
8. Native name of the language or variation (transcribed into ASCII):
9. Reference to published description of the language (book or article):
10. Any other relevant information:

Figure 4

The subtag registration form MUST be sent to
<ietf-languages@iana.org> for a two week review period before it can
be submitted to IANA. (This is an open list. Requests to be added should be sent to <ietf-languages-request@iana.org>.)

Variant subtags are generally registered for use with a particular range of language tags. For example, the subtag 'boont' is intended for use with language tags that start with the primary language subtag "en", since Boontling is a dialect of English. Thus the subtag 'boont' could be included in tags such as "en-Latn-boont" or "en-US-boont". This information is stored in the "recommended prefix" field in the registry and MUST be provided in the registration form.

Any subtag MAY be incorporated into a variety of language tags, according to the rules of Section 2.1, including tags that do not match any of the recommended prefixes of the registered subtag. (Note that this is probably a poor choice.) This makes validation simpler and thus more uniform across implementations, and does not require the registration of a separate subtag for the same purpose and meaning but a different recommended prefix.

The recommended prefixes for a given registered subtag will be maintained in the IANA registry as a guide to usage. If it is necessary to add an additional prefix to that list for an existing language tag, that can be done by filing an additional registration form. In that form, the "Any other relevant information:" field should indicate that it is the addition of an additional recommended prefix.

Requests to add a recommended prefix to a subtag that imply a different semantic meaning will probably be rejected. For example, a request to add the prefix "de" to the subtag 'nedis' so that the tag "de-nedis" represented some German dialect would be rejected. The 'nedis' subtag represents a particular Slovenian dialect and the additional registration would change the semantic meaning assigned to the subtag. A separate subtag should be proposed instead.

The Language Subtag Reviewer is responsible for responding to requests for the registration of subtags through the registration process and is appointed by the IESG.

When the two week period has passed the Language Subtag Reviewer either forwards the request toiana@iana.org, or rejects it because of significant objections raised on the list or due to problems with constraints in this document (which should be explicitly cited). The reviewer may also extend the review period in two week increments to permit further discussion. The reviewer must indicate on the list whether the registration has been accepted, rejected, or extended following each two week period.
Note that the reviewer can raise objections on the list if he or she so desires. The important thing is that the objection must be made publicly.

The applicant is free to modify a rejected application with additional information and submit it again; this restarts the two week comment period.

Decisions made by the reviewer may be appealed to the IESG [RFC 2028][10] under the same rules as other IETF decisions [RFC 2026][21].

All approved registration forms are available online in the directory http://www.iana.org/numbers.html under "languages".

Updates of registrations follow the same procedure as registrations. The subtag reviewer decides whether to allow a new registrant to update a registration made by someone else; normally objections by the original registrant would carry extra weight in such a decision.

Registrations are permanent and stable. Once registered, subtags will not be removed from the registry and will remain the canonical method of referring to a specific language or variant. This provision does not apply to grandfathered tags, which may become deprecated due to registration of subtags. For example, the tag "i-navajo" is deprecated in favor of the ISO 639-1 based subtag 'nv'.

Note: The purpose of the "published description" in the registration form is intended as an aid to people trying to verify whether a language is registered or what language or language variation a particular subtag refers to. In most cases, reference to an authoritative grammar or dictionary of that language will be useful; in cases where no such work exists, other well known works describing that language or in that language may be appropriate. The subtag reviewer decides what constitutes "good enough" reference material. This requirement is not intended to exclude particular languages or dialects due to the size of the speaker population or lack of a standardized orthography. Minority languages will be considered equally on their own merits.

3.4 Extensions and Extensions Namespace

Extension subtags are those introduced by single-letter subtags other than 'x-'. They are reserved for the generation of identifiers which contain a language component, and are compatible with applications that process language tags according to this specification. For example, they might be used to define locale identifiers, which are generally based on language.
The structure and form of extensions are defined by this document so that implementations can be created that are forward compatible with applications that may be created using single-letter subtags in the future. In addition, defining a mechanism for maintaining single-letter subtags will lend to the stability of this document by reducing the likely need for future revisions or updates.

IANA will maintain a registry of allocated single-letter subtags. This registry contains the following information: letter identifier; name; purpose; RFC defining the subtag namespace and its use; and the name, URL, and email address of the maintaining authority.

Allocation of a single-letter subtag shall take the form of an RFC defining the name, purpose, processes, and procedures for maintaining the subtags. The maintaining or registering authority, including name, contact email, discussion list email, and URL location of the registry must be indicated clearly in the RFC. The RFC MUST specify each of the following:

1. The specification MUST reference the specific version or revision of this document that govern its creation and MUST reference this section of this document.
2. The specification and all subtags defined by the specification MUST follow the ABNF and other rules for the formation of tags and subtags as defined in this document. In particular it MUST specify that case is not significant.
3. The specification MUST specify a canonical representation.
4. The specification of valid subtags MUST be available over the Internet and at no cost.
5. The specification MUST be in the public domain or available via a royalty-free license acceptable to the IETF and specified in the RFC.
6. The specification MUST be versioned and each version of the specification MUST be numbered, dated, and stable.
7. The specification MUST be stable. That is, extension subtags, once defined by a specification, MUST NOT be retracted or change in meaning in any substantial way.
8. IANA MUST be informed of changes to the contact information and URL for the specification.

The determination of whether an Internet-Draft meets the above conditions and the decision to grant or withhold such authority rests solely with the IESG, and is subject to the normal review and appeals process associated with the RFC process.

Extension authors are strongly cautioned that many (including most well-formed) processors will be unaware of any special relationships or meaning inherent in the order of extension subtags. Extension authors SHOULD avoid subtag relationships or canonicalization
mechanisms that interfere with matching or with length restrictions that may exist in common protocols where the extension is used. In particular, applications may truncate the subtags in doing matching or in fitting into limited lengths, so it is RECOMMENDED that the most significant information be in the most significant (left-most) subtags, and that the specification gracefully handle truncated subtags.

When a language tag is to be used in a specific, known, protocol, it is RECOMMENDED that that the language tag not contain extensions not supported by that protocol. In addition, it should be noted that some protocols may impose upper limits on the length of the strings used to store or transport the language tag.
4. Security Considerations

The only security issue that has been raised with language tags since the publication of RFC 1766, which stated that "Security issues are believed to be irrelevant to this memo", is a concern with language identifiers used in content negotiation - that they may be used to infer the nationality of the sender, and thus identify potential targets for surveillance.

This is a special case of the general problem that anything you send is visible to the receiving party. It is useful to be aware that such concerns can exist in some cases.

The evaluation of the exact magnitude of the threat, and any possible countermeasures, is left to each application protocol.

Although the specification of valid subtags for an extension MUST be available over the Internet, implementations SHOULD NOT mechanically depend on it being always accessible, to prevent denial-of-service attacks.
5. Character Set Considerations

The syntax in this document requires that language tags use only the characters A-Z, a-z, 0-9, and HYPHEN-MINUS, which are present in most character sets, so presentation of language tags should not have any character set issues.

Rendering of characters based on the content of a language tag is not addressed in this memo. Historically, some languages have relied on the use of specific character sets or other information in order to infer how a specific character should be rendered (notably this applies to language and culture specific variations of Han ideographs as used in Japanese, Chinese, and Korean). When language tags are applied to spans of text, rendering engines may use that information in deciding which font to use in the absence of other information, particularly where languages with distinct writing traditions use the same characters.
6. Changes from RFC 3066

The main goals for this revision of language tags were the following:

'Compatibility.' All valid RFC 3066 language tags (including those in the IANA registry) remain valid in this specification. Thus there is complete backward compatibility of this specification with existing content. In addition, this document defines language tags in such a way as to ensure future compatibility, and processors based solely on the RFC 3066 ABNF (such as those described in XML Schema version 1.0) will be able to process tags described by this document.

'Stability.' Because of the changes in underlying ISO standards, a valid RFC 3066 language tag may become invalid (or have its meaning change) at a later date. With so much of the world's computing infrastructure dependent on language tags, this is simply unacceptable: it invalidates content that may have an extensive shelf-life. In this specification, once a language tag is valid, it remains valid forever. Previously, there was no way to determine when two tags were equivalent. This specification provides a stable mechanism for doing so, through the use of canonical forms. These are also stable, so that implementations can depend on the use of canonical forms to assess equivalency.

'Validity.' The structure of language tags defined by this document makes it possible to determine if a particular tag is well-formed without regard for the actual content or "meaning" of the tag as a whole. This is important because the registry and underlying standards change over time. In addition, it must be possible to determine if a tag is valid (or not) for a given point in time in order to provide reproducible, testable results. This process must not be error-prone; otherwise even intelligent people will generate implementations that give different results. This specification provides for that by having a single data file, with specific versioning information, so that the validity of language tags at any point in time can be precisely determined (instead of interpolating values from many separate sources).

'Extensibility.' It is important to be able to differentiate between written forms of language -- for many implementations this is more important than distinguishing between spoken variants of a language. Languages are written in a wide variety of different scripts, so this document provides for the generative use of ISO 15924 script codes. Like the generative use of ISO language and country codes in RFC 3066, this allows combinations to be produced without resorting to the registration process. The addition of UN codes provides for the generation of language tags with regional scope, which is also
required for information technology.

The recast of the registry from containing whole language tags to subtags is a key part of this. An important feature of RFC 3066 was that it allowed generative use of subtags. This allows people to meaningfully use generated tags, without the delays in registering whole tags, and the burden on the registry of having to supply all of the combinations that people may find useful.

Because of the widespread use of language tags, it is potentially disruptive to have periodic revisions of the core specification, despite demonstrated need. The extension mechanism provides for a way for independent RFCs to define extensions to language tags. These extensions have a very constrained, well-defined structure to prevent extensions from interfering with implementations of language tags defined in this document. The document also anticipates features of ISO 639-3 with the addition of the extlang subtags. The use and definition of private use tags has also been modified, to allow people to move as much information as possible out of private use tags, and into the regular structure. The goal is to dramatically reduce the need to produce a revision of this document in the future.

The specific changes in this document to meet these goals are:
  o Defines the ABNF and rules for subtags so that the category of all subtags can be determined without reference to the registry.
  o Adds the concept of well-formed vs. validating processors, defining the rules by which an implementation can claim to be one or the other.
  o Changes the IANA language tag registry to a language subtag registry that provides a complete list of valid subtags in the IANA registry. This allows for robust implementation and ease of maintenance. The language subtag registry becomes the canonical source for forming language tags.
  o Provides a process that guarantees stability of language tags, by handling reuse of values by ISO 639, ISO 15924, and ISO 3166 in the event that they register a previously used value for a new purpose.
  o Allows ISO 15924 script code subtags and allows them to be used generatively. Adds the concept of a variant subtag and allows variants to be used generatively. Adds the ability to use a class of UN tags as regions.
  o Defines the private-use tags in ISO 639, ISO 15924, and ISO 3166 as the mechanism for creating private-use language, script, and region subtags respectively.
  o Adds a well-defined extension mechanism.
  o Defines an extended language subtag, possibly for use with certain anticipated features of ISO 639-3.
Ed Note: The following items are provided for the convenience of reviewers and will be removed from the final document.

Changes between draft-09 and this version are:

- Split the document into two drafts: this document and draft-phillips-langmatching-00. (A.Phillips, M.Davis)
- Removed all references to the matching and language ranges sections from this document. (A.Phillips, M.Davis)
- Added a note about the fact that all tags valid under this document would have been valid or valid to register under RFC 3066 to Section 2.2.8. (A.Phillips)
- Modified text as appropriate in several sections, notably about extensions, so that matching is not referred to or is referred to only in general terms (A.Phillips, M.Davis)
- Expunged the terminology "language range", since that section goes with matching (A.Phillips, M.Davis)

7 References


[8] Borenstein, N. and N. Freed, "MIME (Multipurpose Internet Mail
Internet-Draft langtags February 2005

Extensions) Part One: Mechanisms for Specifying and Describing
the Format of Internet Message Bodies", RFC 1521, September
1993.

1766, March 1995.

[10] Hovey, R. and S. Bradner, "The Organizations Involved in the

[11] Bradner, S., "Key words for use in RFCs to Indicate Requirement

Extensions: Character Sets, Languages, and Continuations", RFC
2231, November 1997.


Resource Identifiers (URI): Generic Syntax", RFC 2396, August
1998.

Considerations Section in RFCs", BCP 26, RFC 2434, October
1998.

[16] Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L.,
Leach, P. and T. Berners-Lee, "Hypertext Transfer Protocol --
HTTP/1.1", RFC 2616, June 1999.

Understanding Concerning the Technical Work of the Internet

[18] Alvestrand, H., "Tags for the Identification of Languages", BCP

[19] Yergeau, F., "UTF-8, a transformation format of ISO 10646", STD


Authors' Addresses

Addison Phillips (editor)
webMethods, Inc.
432 Lakeside Drive
Sunnyvale, CA  94088
US

EMail: aphillips@webmethods.com

Mark Davis
IBM

EMail: mark.davis@us.ibm.com
Appendix A. Acknowledgements

Any list of contributors is bound to be incomplete; please regard the following as only a selection from the group of people who have contributed to make this document what it is today.

The contributors to RFC 3066 and RFC 1766, the precursors of this document, made enormous contributions directly or indirectly to this document and are generally responsible for the success of language tags.

The following people (in alphabetical order) contributed to this document or to RFCs 1766 and 3066:


Very special thanks must go to Harald Tveit Alvestrand, who originated RFCs 1766 and 3066, and without whom this document would not have been possible. Special thanks must go to Michael Everson, who has served as language tag reviewer for almost the complete period since the publication of RFC 1766. Special thanks to Doug Ewell, for his production of the first complete subtag registry, and his work in producing a test parser for verifying language tags.
Appendix B. Examples of Language Tags (Informative)

Simple language subtag:
- de (German)
- fr (French)
- ja (Japanese)
- i-enochian (example of a grandfathered tag)

Language subtag plus Script subtag:
- zh-Hant (Traditional Chinese)
- en-Latn (English written in Latin script)
- sr-Cyrl (Serbian written with Cyrillic script)

Language-Script-Region:
- zh-Hans-CN (Simplified Chinese for the PRC)
- sr-Latn-CS (Serbian, Latin script, Serbia and Montenegro)

Language-Script-Region-Variant:
- en-Latn-US-boont (Boontling dialect of English)
- de-Latn-CH-1996 (German written in Latin script for Switzerland using the orthography of 1996)

Language-Region:
- de-DE (German for Germany)
- zh-SG (Chinese for Singapore)
- cs-200 (Czech for Czechoslovakia)
- sr-CS (Serbian for Serbia and Montenegro)
- es-419 (Spanish for Latin America and Caribbean region using the UN region code)

Other Mixtures:
- en-boont (Boontling dialect of English)

Private-use mechanism:
- de-CH-x-phonebk
- az-Arab-x-AZE-derbend

Extended language subtags (examples ONLY: extended languages must be defined by revision or update to this document):
- zh-min
- zh-min-nan-Hant-CN

Private-use subtags:
- x-whatever (private use using the singleton 'x')
- qaa-Qaaa-QM-x-southern (all private tags)
de-Qaaa (German, with a private script)
de-Latn-QM (German, Latin-script, private region)
de-Qaaa-DE (German, private script, for Germany)

Tags that use extensions (examples ONLY: extensions must be defined by revision or update to this document or by RFC):
en-US-u-islamCal
zh-CN-a-myExt-x-private
en-a-myExt-b-another

Some Invalid Tags:
de-419-DE (two region tags)
a-DE (use of a single character subtag in primary position; note that there are a few grandfathered tags that start with "i-" that are valid)
ar-a-aaa-b-bbb-a-ccc (two extensions with same single letter prefix)
Appendix C. Conversion of the RFC 3066 Language Tag Registry

Upon publication of this document as a BCP, the existing IANA language tag registry must be converted into the new subtag registry. This section defines the process for performing this conversion.

The impact on the IANA maintainers of the registry of this conversion will be a small increase in the frequency of new entries. The initial set of records represents no impact on IANA, since the work to create it will be performed externally.

When this document is published, an email request will be sent from the authors of this document to the list ietf-languages@iana.org requesting the conversion of the registry. In that request, the authors of this document will provide a URL whose referred content is the proposed IANA Language Subtag Registry following conversion. There will be a Last Call period of not less than four weeks for comments and corrections to be discussed on the ietf-languages@iana.org mail list. Changes as a result of comments will not restart the Last Call period. At the end of the period, the authors will forward the URL to IANA, which will post the new registry on-line.

Tags that are currently deprecated will be maintained as grandfathered entries. The record for the grandfathered entry will contain a note indicating that the entry is 'deprecated' and reason for the deprecation.

Tags that consist entirely of subtags that are valid under this document and which have the correct form and format for tags defined by this document are superseded by this document. Such tags are placed in the 'redundant' section of the registry. For example, zh-Hant is now defined by this document.

Tags that contain subtags which are consistent with registration under the guidelines in this document will have a new subtag registration created for each eligible subtag. If all of the subtags in the original tag are fully defined by the resulting registrations or by this document, then the original tag is superseded by this document. Such tags are placed in the 'redundant' section of the registry. For example, en-boont will result in a new subtag "boont" and the RFC 3066 registered tag 'en-boont' placed in the redundant section of the registry.

Tags that contain one or more subtags that do not match the valid registration pattern and which are not otherwise defined by this document are marked as 'grandfathered' by this document.
There will be a reasonable period in which the community may comment on the proposed list entries, which SHALL be no less than four weeks in length. At the completion of this period, the Language Subtag Reviewer will notify iana@iana.org and the ietf-languages mail lists that the task is complete and forward the necessary materials to IANA for publication.

Registrations that are in process under the rules defined in RFC 3066 MAY be completed under the former rules, at the discretion of the language tag reviewer. Any new registrations submitted after the request for conversion of the registry MUST be rejected.

All existing RFC 3066 language tag registrations will be maintained in perpetuity.

The rules governing the conversion of RFC 1766 and RFC 3066 registered tags are:

- If the formerly registered tag would now be defined by this document, then the existing tag is superseded by this document and is placed in the 'redundant' section of the registry: no subtag will be registered as a result. For example, 'zh-Hans' is now defined by the addition of ISO 15924 script codes.

- If the registered tag contained one or more subtags that follow the guidelines for registered language or variant subtags, and all of the subtags are either now defined by this document or would be valid to register, then each subtag not already covered by this document will be registered automatically by IANA without further review. The RFC 3066 registered tag is placed in the 'redundant' section of the registry. For example: the tag 'en-boont' fits the pattern for a registered variant. The variant subtag "boont" will be registered automatically and 'en-boont' put into the 'redundant' section of the registry.

- If the registered tag contains any subtags that are not otherwise valid for registration according to the rules in this document, then the tag as a whole is maintained as an exceptional case (that is, it is "grandfathered"). This includes special cases of Sign Language tags. For example, the tag 'zh-min-nan' is not covered by any addition and is grandfathered, as is 'sgn-BE-fr' (Belgian French Sign Language).

Users of tags that are grandfathered should consider registering appropriate subtags in the IANA subtag registry (but are not required to).

Where two subtags have the same meaning, the priority of which to make canonical SHALL be the following:
o As of January 1, 2005, if a code exists in the associated ISO standard and it is not deprecated or withdrawn as of that date, then it has priority.
o Otherwise, the earlier-registered tag in the associated ISO standard has priority.

UN numeric codes assigned to 'macro-geographical (continental)' or sub-regions not associated with an assigned ISO 3166 alpha-2 code are defined in the IANA registry and are valid for use in language tags. These codes MUST be added to the initial version of the registry. The UN numeric codes for 'economic groupings' or 'other groupings', and the alphanumeric codes in Appendix X of the UN document MUST NOT be added to the registry.
Intellectual Property Statement

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in BCP 78 and BCP 79.

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at http://www.ietf.org/ipr.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

Disclaimer of Validity

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Copyright Statement

Copyright (C) The Internet Society (2005). This document is subject to the rights, licenses and restrictions contained in BCP 78, and except as set forth therein, the authors retain all their rights.

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.