Matching Language Identifiers
draft-phillips-langmatching-00

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 different mechanisms for comparing and matching the language identifiers defined by RFC3066bis. Possible algorithms for language negotiation and content selection are described. Portions of this document obsolete RFC 3066. [1]
Table of Contents

1. Introduction .................................................. 3

2. The Language Range .......................................... 4
   2.1 Extended Language Range ................................. 4
   2.2 Meaning of the Language Tag ............................. 5
      2.2.1 Default Matching Scheme .......................... 6
   2.3 Other Matching Schemes .................................. 7
   2.4 Considerations for Private Use Subtags ............... 8

3. IANA Considerations ......................................... 9

4. Security Considerations ..................................... 10

5. Character Set Considerations ............................... 11

6. References .................................................. 11
   Authors' Addresses .......................................... 13

A. Acknowledgements ........................................... 14
   Intellectual Property and Copyright Statements .......... 15
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.

Given a set of language identifiers, such as those defined in RFC3066bis, various mechanisms can be envisioned for performing language negotiation and tag matching. The suitability of a particular mechanism to a particular application depends on the needs of that application.

This document defines language ranges and syntax for specifying user preferences in a request for language content. It also specifies a default algorithm for matching language ranges to content (language tags), as well as alternate mechanisms suitable for certain applications.

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] [12].
2. The Language Range

A Language Range is a set of languages whose tags all begin with the same sequence of subtags. A Language Range can be represented by a 'language-range' tag, by using the definition from HTTP/1.1 [17]:

```
language-range = language-tag / "*"
```

That is, a language-range has the same syntax as a language-tag or is the single character "". This definition of language-range implies that there is a semantic relationship between tags that share the same subtag prefixes.

A language-range matches a language-tag if it exactly equals the tag, or if it exactly equals a prefix of the tag such that the first character following the prefix is ".". (That is, the language-range "en-de" matches the language tag "en-DE-boont", but not the language tag "en-Deva").

The special range "" matches any tag. A protocol which uses language ranges may specify additional rules about the semantics of ";"; for instance, HTTP/1.1 specifies that the range "" matches only languages not matched by any other range within an "Accept-Language:" header.

As noted above, not all languages or content denoted by a specific language-range may be mutually intelligible and this use of a prefix matching rule does not imply that language tags are assigned to languages in such a way that it is always true that if a user understands a language with a certain tag, then this user will also understand all languages with tags for which this tag is a prefix. The prefix rule simply allows the use of prefix tags if this is the case.

2.1 Extended Language Range

The simple matching described above is not always the most appropriate use of the information contained in language tags. Some applications may wish to define a more granular matching scheme based on extended language ranges:

```
extended-language-range = (subtag / "") *("-" (subtag / ""))
```

```
subtag = (1*8alphanum)
alphanum = ALPHA / DIGIT
```

In this language range scheme, a language range takes the form of a series of subtags or the special subrange of ";". For example, the language range "en-*-US" specifies a primary language of 'en', followed by any script subtag, followed by the region subtag 'US'.
Ed.Note> A more exact ABNF is possible to construct. It would need 
to capture the structure of langtag and show that each field is 
optional. My first stab at it was something like:

(\texttt{primary\_lang / "\*"})

[[ [ "-" (\texttt{script / "\*"})]
  [[ "-" (\texttt{region / "\*"})]
  [ ("-.*" / *"-" variant)])]]

### 2.2 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.

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.

### 2.2.1 Default Matching Scheme

Implementations that are searching for content or otherwise matching language tags to a language-range [Section 2] may choose to assume that there is a semantic relationship between two tags that share common prefixes. This is called 'language tag fallback'. The most common implementation follows this pattern:

When searching for content using language tag fallback, the language tag is progressively truncated from the end until a match is located. For example, starting with the tag "en-US-boont", searches or matches would first be performed with the whole tag, then with "en-US", and finally with "en". This allows some flexibility in finding content. It also typically provides better results when data is not available at a specific level of tag granularity or is sparsely populated (than if the default language for the system or content were used).

Tag to match: en-US-boont
1. en-US-boont
2. en-US
3. en

**Figure 4: Default Fallback Pattern Example**

When working with tags and ranges you should also note the following:

1. Private-use and Extension subtags are normally orthogonal to language tag fallback. Implementations should ignore unrecognized private-use and extension subtags when performing language tag fallback. Since these subtags are always at the end of the sequence of subtags, they naturally fall out of the default fallback pattern (above). Thus a request to match the tag "en-US-boont-x-1943" would produce exactly the same information content as the example above.
2. Implementations that choose not to interpret one or more private-use or extension subtags should not remove or modify
these extensions in content that they are processing. When a
language tag instance is to be used in a specific, known
protocol, and is not being passed through to other protocols,
language tags may be filtered to remove subtags and extensions
that are not supported by that protocol. This should be done
with caution, since it it is removing information that may be
relevant if services on the other end of the protocol would make
use of that information.

3. Some applications of language tags may want or need to consider
extensions and private-use subtags when matching tags. If
extensions and private-use subtags are included in a matching
process that utilizes the default fallback mechanism, then the
implementation should canonicalize the language tags and/or
ranges before performing the matching. Note that language tag
processors that claim to be "well-formed" processors as defined
in [1] generally fall into this category.

2.3 Other Matching Schemes

Implementations MAY choose to implement different styles of matching
for different kinds of processing. For example, an implementation
could treat an absent script subtag as a "wildcard" field; thus
"az-AZ" would match "az-AZ", "az-Cyrl-AZ", "az-Latn-AZ", etc. but
not "az". If one item is to be chosen, the implementation could pick
among those matches based on other information, such as the most
likely script used in the language/region in question.

Because the primary language subtag cannot be absent, the 'UND'
subtag might sometimes be used as a 'wildcard' for this style of
matching. For example, in a query where you want to select all
language tags that contain 'Latn' as the script code and 'AZ' as the
region code, you could use "und-Latn-AZ".

Extended language ranges are designed around this idea. An extended
language range matches a tag if:

- Each subtag in the extended language range that is not "*" exactly
  matches the subtag in the language tag in that position. For
  example, the range "en-*-US" matches "en-Latn-US".
- Each subtag in the extended language range that is "*" has a
  corresponding subtag in the language tag or that subtag is empty.
  For example, the range "en-*-US" matches "en-Latn-US" and also
  "en-US".
- Each subtag type that is not specified in the extended language
  range may contain additional values. For example, the range
  "en-*-US" matches the tag "en-Latn-US-boont".

Implementations may also wish to use semantic information external to
the language tags when performing fallback. For example, the primary
language subtags 'nn' (Nynorsk Norwegian) and 'nb' (Bokmal Norwegian) might both be usefully matched to the more general subtag 'no' (Norwegian). Or an application might infer that content labeled "zh-CN" is more likely to match the range "zh-Hans" than equivalent content labeled "zh-TW".

2.4 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 caution 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. Matching private use tags using language ranges or extended language ranges may result in unpredictable content being returned.
3. IANA Considerations

This document presents no new or existing considerations for IANA.
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 ranges 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 ranges use only the characters A-Z, a-z, 0-9, and HYPHEN-MINUS legal in language tags. These characters 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 References


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.
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.