Reference Label Generation Rules (LGR) for the Second Level — Overview and Summary

REVISION – October 7, 2016

Table of Contents

1 Overview 1

1.1 Reference Label Generation Rules (LGR) Files 2

2 Notes 2

2.1 Repertoire 2

2.1.1 Sources for Repertoire 2

2.2 Extended Code Points 3

2.3 Excluded Code Points 3

2.4 Sequences 3

2.5 Variants 3

2.6 Whole Label Evaluation (WLE) Rules 3

2.6.1 Protocol-defined Rules 4

2.6.2 Reference LGR-specific rules 4

2.7 Metadata 5

3 Use of Reference LGRs 5

3.1 Subset Repertoires 5

3.2 Overlapping Repertoires 5

3.3 Repertoire Extensions 6

3.4 Variants and Rules 6

4 Expert Review 6

5 Contributors 6

6 References 7

1 Overview

This document describes a set of proposed Reference Label Generation Rules (LGRs) for the Second Level. These language-based LGRs were developed according to the "Guidelines for Developing Reference LGRs for the Second Level" [Guidelines]. The guidelines define a process that builds on the results of a previous project [IIS] but provides additional review and development, documentation and translation to XML [RFC7940]. In some cases, these LGRs extend the repertoire compared to [IIS].

This document provides some general background related to the design and design process for these LGRs, as well as general considerations relevant to anyone wishing to adopt or adapt these LGRs.

The LGRs are specific to a given language (and in some cases the combination of language and script) but not necessarily specific to a geographically compact user community. Each file has been reviewed by one or more linguistic experts, as well as reviewed by a separate expert for DNS stability and security issues. The reader of this document is assumed to be familiar with the [Guidelines].

1.1 Reference Label Generation Rules (LGR) Files

The normative definition of each reference LGR is provided as an XML file, as given at the Second Level LGR References webpage for Belarusian, Bosnian (Cyrillic), Bosnian (Latin), Bulgarian, Chinese, Danish, English, Finnish, French, German, Hebrew, Hungarian, Icelandic, Italian, Korean, Latvian, Lithuanian, Macedonian, Montenegrin, Norwegian, Polish, Portuguese, Russian, Serbian, Spanish, Swedish and Ukrainian, which are released on 7 October 2016.

The LGRs are expressed using a standard format defined in "Representing Label Generation Rulesets in XML" [RFC7940].

Each of these files contains all the LGRs applicable to the labels from that language, and only those rules. Each file contains a complete description, a repertoire with optional variants, and WLE Rules, as well as detailed references that link each included code point to a reference providing data for justifying its inclusion.

From each XML file, a non-normative HTML presentation is generated mechanically, also available at the Second Level LGR References webpage. These are provided for the convenience of the reader. The HTML presentation is augmented by summary data as well as data extracted from the Unicode Character Database [UCD].

2 Notes

The development and review process followed the [Guidelines]. The following notes provide some additional highlights as well as information that is not specific to an individual file.

2.1 Repertoire

The repertoire for each LGR is based on a consensus repertoire derived from the sources consulted. In many cases, this caters to more than the code points needed to write the native vocabulary of the language by including code points that are in common use for loan words and the like. Where a language has multiple user communities with some variation of usage, a single, combined LGR is produced. The details are described in each of the LGRs.

2.1.1 Sources for Repertoire

In determining the repertoire, a large number of sources was investigated, from spelling dictionaries released by language authorities, RCFs and national or international standards, to other sources such as ordinary dictionaries, the Common Locale Data Repository (a project of the Unicode Consortium) [CLDR] and finally existing IDN practice for ccTLDs aimed at users of a native language. The sources and their contribution to the development of the repertoire are documented in detail in each of the LGRs.

2.2 Extended Code Points

Many, though not all, of the languages are written by compact communities that are in contact with other languages in the same region or in the same country. In those cases, native users may have familiarity with or need for access to an extended set of code points, for example for names of people or places. The Reference LGRs provide for those code points by listing them as “extended-cp” if they are not already catered for in the core repertoire. As written, the LGRs treat these extended code points as ineligible for a label, but users could easily remove the restriction to tailor the LGR to their needs. (See also Section 2.6).

This is in contrast to script-based LGRs that typically provide for the full repertoire needed for all languages sharing a common script, or those country-based LGRs, that provide for the needs of users from the same country or territory, irrespective of whether they write a majority or minority language prevalent in the country.

2.3 Excluded Code Points

For most languages, some sources include a number of very rarely used code points or some that are historic or limited to special purposes, like poetry and religious works. Such uses are rarely germane to IDNs. Consequently, such code points have been excluded from these LGRs and documented as such.

2.4 Sequences

In a small number of cases, code points occur only in fixed combinations. Where that is the case, the repertoire contains these code points only as part of an explicitly specified code point sequence. This prevents unneeded combinations. This consideration primarily applies to combining marks such as diacritics. Rendering systems may fail to provide a predictable presentation of combining marks if they are present outside of expected contexts, whether applied to unexpected base characters or, for example, repeatedly applied. In the latter case, in particular, there is a danger of “overprinting”, which would mask the presence of the extraneous diacritic. Finally, some diacritics are easily confused with one another. Allowing unrestricted combinations would allow these diacritics to be applied to base characters that normally take different diacritics, greatly adding to the risk of creating confusable labels.

2.5 Variants

The majority of language based LGRs does not include the definition of any variant. Where variants are included, their selection is informed by existing registry practice, as well as by the work performed at ICANN on the script LGRs for the Root Zone.

2.6 Whole Label Evaluation (WLE) Rules

WLE rules implement a further constraint on labels, for example, by limiting certain code points from occurring at the beginning of a label, repeatedly, or simultaneously with other code points in the same label. Context rules are a form of a WLE rule that defines a constraint on the surrounding context for a given code point (see [RFC7940]).

2.6.1 Protocol-defined Rules

Because the XML format for the LGR supports machine-evaluation of labels for validity, these reference LGRs include all relevant constraints on labels defined in the IDNA protocol itself. In this way, the LGRs can be used to validate all constraints on the label in one pass.

Common rules:

- Hyphen Restrictions — restricts the allowable placement of U+002D (-) HYPHEN (no leading/ending hyphen and no hyphen in 3-4 position). These constraints are described in section 4.2.3.1 of [RFC5891].
- Leading Combining Marks — restricts the allowable placement for combining marks (no leading combining mark). This constraint is described in section 4.2.3.2 of [RFC5891].

Rules for Right-to-Left labels:

- Leading Digit — restricts the allowable placement of digits for right-to-left labels (no leading digit in RTL label). This constraint is described in section 2.1 of [RFC5893].
- Mixed Digits — prevents the mixing of European and Arabic (Indic) digits. This constraint is described in appendix A.8 and A.9 of [RFC58932].

Context rules:

- Japanese in Label — restricts the occurrence of KATAKANA MIDDLE DOT to labels containing at least one code point from any of these scripts: Han, Hiragana, or Katakana; This rule is described in Appendix A.7 of [RFC5892].

For these reference LGRs, the protocol-derived rules, other than the common rules, have only been included if they are needed for labels in the given script. The description section of each LGR file lists the rules and their associated references.

If a label to be validated has already been tested against protocol-derived constraints by the time the LGR is applied, these rules would be redundant and could be removed.

2.6.2 Reference LGR-specific rules

A small number of the LGRs contain additional LGR-specific WLE rules, reflecting a further constraints on possible labels based on the nature of the language or script. These are documented in detail in the description section of the respective LGRs.

Finally, the 2nd level reference LGRs use special a context rule to support adapting a reference LGR to a specific zone. (For details, see Section 2.2). By default, this rule is present in all tables, whether actually associated with any code points or not.

Special rule:

- Extended-cp —this context rule always fails. That means, as published, the LGRs do not allow the code points identified as extended by having been given a context of “extended-cp”. Simply

changing that rule so it always matches would enable the entire set of extended code points without the need to edit the list of characters. Alternatively, the context condition could be removed from individual code points, thus enabling them one by one. Likewise, to create a subset, a code point can be disabled by adding the “extended-cp” context condition. Doing so would mark the code point as deliberately not included instead of merely omitted.

2.7 Metadata

The XML file format defines a number of elements for metadata. Several elements are not relevant to reference LGRs, but would be relevant to actual, deployed LGRs. These elements include <scope>, <validity-start>, and <validity-end>. For more details see [RFC7940]. In adopting a reference LGR as the LGR for a specific zone, values for these elements should be supplied.

3 Use of Reference LGRs

The information in these reference LGRs represents the best available knowledge of the code points suitable for IDNs for users of a given language. As [RFC 6912] makes clear, IDNs are intended to be reasonable mnemonics, not text that is in a given language. However, what is a reasonable mnemonic is informed by the language of the user community. Letters or diacritics that are unfamiliar in appearance do not make good mnemonics. In that sense, the fact that these LGRs have been developed for a given language can also be understood as meaning that they were developed with users of a particular language in mind.

3.1 Subset Repertoires

Creating a subset of one of these LGRs would generally represent a more conservative choice (see [RFC6912]). However, the final choice will always have to be made in tension between the two goals of usability and conservatism. There are several issues to consider when contemplating the creation of a subset. The first affects usability. For example, consider the case of reducing any Latin-based LGR to the letters "a-z". This is undoubtedly a conservative choice. But, it also eliminates any gain in usability compared to non-IDN labels. A subset should always be a carefully designed consistent whole. (See also Section 2.2 and Section 2.6)

The next concern applies to LGRs that contain variants. For those LGRs, the effect of subsetting on the variant sets must be reviewed thoroughly. For each code point to be removed, all variant mappings related to that code point must also be removed. Once these are removed, it may not be possible to add the code point back again in a future version of the LGR due to the risk of stability issues. Finally, any rule that depends on the definition of a given code point must be updated if that code point is removed.

3.2 Overlapping Repertoires

Additional policies, variants and rules may be needed if any of these reference LGRs is adopted along with other LGRs that have an overlapping repertoire. This is especially relevant in the case of LGRs defining variants for LGR-specific rules.

3.3 Repertoire Extensions

There are two ways of extending these LGRs. The first is by allowing additional code points that are considered widely used in the context of a given language. The reference LGRs are constructed to provide the best available information on a suitable set of such extended code points (see Section 2.2).

The second is the use of these language-based reference LGRs as "building blocks" in assembling local, regional or script-based LGRs. When used in that fashion, care must be taken so that the resulting LGR provides for a consistent treatment of variants and WLE rules.

Any combined LGRs should be from a common script. The issue of mixed script labels is addressed in [RFC5890]. In combining LGRs into a single LGR it is recommended to first combine their core repertoires and, after eliminating duplicates, to consider possible additions from the extended sets separately. A combined LGR would have multiple "language" elements to indicate the range of languages covered, or a single language element indicating the script (see [RFC7940]).

3.4 Variants and Rules

When merging LGRs, or when using LGRs with overlapping repertoires in the same zone, the "rules" element in the XML must be given special scrutiny. Some of the "rule" and "class" elements may be merged safely. Others may have to be renamed to keep them distinct. "action" elements must be present in the order required for proper precedence in the merged XML.

That said, most of the LGRs presented here have a generic, default "rules" element. Any two LGRs with only the default rules can be merged and a single copy of the default rules appended.

4 Expert Review

The LGRs were reviewed by independent reviewers with expertise in Unicode and linguistics, as well as IDNA and DNS security. The LGRs were updated to reflect the input from the review. The expert reports are available at the Second Level LGR References webpage.

5 Contributors

The reference LGRs were developed by the Staff and Contractors of Sheypa LLC.

1. Developers

Asmus Freytag
Michel Suignard

2. Expert Reviewers

Michael Everson
Nicholas Ostler
Lu Qin
Wil Tan

6 References

[CLDR] CLDR - Unicode Common Locale Data Repository: http://cldr.unicode.org

[Guidelines] Internet Corporation for Assigned Names and Numbers, “Guidelines for Developing Reference LGRs for the Second Level”. (Los Angeles, California: ICANN, October 2015) https://www.icann.org/en/system/files/files/lgr-guidelines-second-level-30oct15-en.pdf.

[IIS] IIS, IDN Reference Tables, https://github.com/dotse/IDN-ref-tables

[RFC5890] Klensin, J., "Internationalized Domain Names for Applications (IDNA): Definitions and Document Framework", RFC 5890, August 2010, https://tools.ietf.org/html/rfc5890.

[RFC6912] Sullivan, A., Thaler, D., Klensin, J., and O. Kolkman, "Principles for Unicode Code Point Inclusion in Labels in the DNS", RFC 6912, April 2013, https://tools.ietf.org/html/rfc6912.

[RFC7940] Davies, K and Asmus Freytag: “Representing Label Generation Rulesets using XML”, August 2016 https://tools.ietf.org/html/rfc7940.