Guidelines for Developing Reference LGRs for the Second Level

Date: 2020-05-27

Table of Contents

1 INTRODUCTION

This document describes the process to be followed in developing a set of reference label generation rulesets (LGR) to be made available for selected languages and scripts on the second level. These reference LGRs are intended to be comprehensive enough that they do not require further additions to be useful. At the same time, they should be relatively conservative. This should enable registries to adopt these LGRs either as is, or to take them as the basis for further modifications. The details of how and under what conditions registries make use of these rulesets are outside the scope of this document.

As described below, the process of developing each LGR takes as its starting point a review of an existing LGR or proposed reference LGR where available. For script LGRs these would normally be the corresponding Root Zone LGRs, while for language LGRs, they also include, but are not limited to existing IDN Tables for the second level. For certain languages there exists a set of proposed reference LGRs released into the public domain by IIS [DotSE]. In some cases a language LGR may also be derived directly from an existing script LGR.

Any LGR used as starting point is either confirmed, or modified to better fit the need of for the target script or language in the context of the Second Level. This work is based on information available from other authoritative sources as well as expertise represented by the development team and a set of external reviewers. The resulting proposed Reference LGRs then undergo public review.

The following sections describe various aspects of the development process in more detail.

2 TARGET LANGUAGE, WRITING SYSTEM AND SCRIPT

For the purposes of developing a reference LGR for a particular language, the modern writing system for that language will be considered. If there are multiple writing systems, each using a different script, then each of them would be the target for a different LGR, and the language identifier for the LGR will contain both the language as well as the script tag. Examples are the Cyrillic and Latin writing systems for Bosnian with language identifiers bs-Cyrl and bs-Latn, respectively.

In case of national or regional differences in writing systems for a given language, but using the same script, the reference LGR will be designed to accommodate all of them and be identified with a language tag not containing a country code or regional identifier. For example, the Swiss and German writing systems for the German language would be accommodated by a single German LGR.

The writing systems for some languages use multiple scripts. A single LGR will be designed to cover each such writing system across all the scripts it employs. For example the writing system for Japanese uses the Kanji (Han), Katakana, Hiragana, and Romaji (Latin) scripts.

For script-based LGRs, all modern writing systems in everyday use have been considered in developing the respective Root Zone LGRs (for the methodology see [MSR] and [RZ-LR-Overview]). The Root Zone LGRs will serve as starting point for developing second level reference LGRs for these scripts, or in some cases, their principal language(s).

3 SOURCES

For each reference LGR, the source references used for developing the repertoire are stated. These sources help to identify which code points to include in the repertoire required for IDNs in a given script or language; which context and other constraints must be satisfied, if any; and which code points to consider variants

Sources differ in the degree to which they are officially recognized, their authoritativeness and the details and nature of the repertoire subset they document. While all sources to be considered will document how the language uses a particular writing system, not all will be equally relevant for the task of defining a repertoire for use with IDNs.

3.1 Root Zone LGRs as Source

The Root Zone LGRs encompass a set of LGRs for all scripts in widespread modern use, some having already been published while others are still in development [RZ-LGR-Project]. They are developed for top level domain names in a process that combines strong community involvement with expert input and review. They reflect considerable research on how to securely address the needs for a given scripts in IDNs. They specifically address security and stability concerns for use of the script in domain labels, while incorporating the needs of all languages with stable orthography using the script. They cover all code points with general purpose use for these languages, while excluding those that pose a risk to the security and stability of the DNS.

The development of the Root Zone LGRs is based on RFCs 6912, 7940 and 8228 which provide guidelines and directions applicable to all levels of labels in a domain name, including top-level and second level. To the degree possible, their development took into account existing practice for IDNs for that script.

For a more comprehensive discussion of the methodology see [MSR] and [RZ-LGR-Overview].

While the RZ LGRs do reflect some restrictions specific to the Root Zone, they make a solid foundation for the task of defining reference LGRs for other zones. Despite having been developed on a per-script basis, they serve as a significant source even for language based LGRs; especially for languages written in a complex script because they identify the umbrella set of variants and context constraints required for that script.

To the degree applicable, the relevant Root Zone LGRs and by incorporation the references therein serve as authoritative and sufficient documentation for the requirements for a given script community. For reference LGRs based on existing Root Zone LGRs, such sources may be incorporated by reference to the underlying LGR. This includes the data on language-specific usage compiled during the development of script LGRs. Such data may be taken as authoritative for this purpose, as it was collected and vetted by community experts specifically for the task of defining language-specific sub-repertoires in the context of IDN labels. While other source can be quite valuable, their focus may be different and therefore may at times require additional analysis, corroboration or validation before being used for the purpose of defining reference repertoire.

3.2 Other Sources

The repertoire of characters needed for certain languages may be described in other sources, including International, National and other Standards for Information Technology. Of these, the Unicode Locales project [CLDR] provides a set of full language repertoires created as part of a rigorous process involving local experts and its data are implemented widely in products so we can assume they have withstood end-user testing.

1. **core subset from CLDR**; the Common Locale Data Repository maintained by the Unicode Consortium contains a specification for a core set that more or less captures the essential set of code points needed for representing texts written in a given language.
2. **auxiliary subset from CLDR**; the Common Locale Data Repository maintained by the Unicode Consortium contains a specification for an auxiliary set that in most cases captures the maximal set of code points needed for representing texts written in a given language.

For the task of determining the repertoire suitable for identifiers in a given language, the work done by registries for ccTLDs is invaluable, particularly where it involves the languages native to the territory or country.

The following list enumerates various sources to be used for the references:

- Standards:
  - international, national, industry, and internet standards
- Institution:
  - official and unofficial institutions
- Language description:
  - dictionaries, educational materials, linguistic descriptions
- Existing community reviewed LGRs:
  - Primarily Root Zone LGRs, but also other community reviewed LGRs
- Other:
  - surveys, online databases, IDN tables for ccTLDs

Comparatively few of the languages considered for reference LGRs have an official entity empowered to give authoritative rulings about orthography and other aspects of the language's use. Even where such official entities exist, their scope may be limited to a particular nation, or they may not be applicable to the purpose at hand, which is the creation of label generation rulesets for IDNs. In many cases, language authorities document orthodox alphabets that are based on some linguistic criteria, but that do not equal the actual set of code points minimally required or essential in writing the language.

> NOTE: For some languages there exist official or regulatory institutions governing orthography and usage (examples include the L'Académie Française for French, the Rat für deutsche Rechtschreibung for the German-speaking countries, and the Norsk språkråd for Nynorsk and Bokmål Norwegian). Other languages have unofficial but respected institutions guiding orthography and usage (for example, Duden for German, and the Oxford University Press for English). For the majority of languages there exist no official institutions; their description can be found in dictionaries, educational materials, scholarly linguistic texts, online databases and surveys and other kinds of documents.

Most official entities appear primarily concerned with what is called the "core subset" above, or simply the alphabet. For the majority of languages, particularly the alphabetic ones there is scant disagreement on what constitutes the core alphabet, barring small differences in national usage (such as the Swiss not using the 'sharp s' in writing German). For establishing a minimal essential subset, it scarcely matters then which source is referenced. (The core set for non-alphabetic languages may be less well defined and present particular challenges of their own).

There are isolated exceptions to the general lack of formal sources for wider subsets. For example, the Scandinavian countries embarked on a project of defining several subsets for their various languages via a formal standard [Nordic]. While none of the subsets defined there precisely matches the subset most useful for IDNs, the information provided allows one to narrow down likely candidates for a reference LGR repertoire.

The Unicode Locales project [CLDR] collects data relevant to locale support in a formal process driven by local expertise and subject to quality controls. It collects repertoire information on two levels, a core set that is geared towards the minimal set required for writing the language and an auxiliary set which extends the core to include all code points likely to be encountered in texts in the given language, including foreign names or words that customarily retain their original spellings. For purposes of developing the repertoire for a second level LGR, the first subset may be too restrictive and the second one too permissive.

The various cross-language surveys often provide useful, if sometimes less controlled, information by focusing on some of the more common extensions to the strict repertoire, as opposed to providing a fully maximal superset. By correlating their information, the scope of common extensions to the essential or strict subset can be narrowed down with a reasonable degree of confidence. It is still necessary to review these for suitability for use with IDNs; that is, to make a judgment call whether their inclusion in an LGR for the second level is desirable and warranted.

In identifying and qualifying sources for the development and verification of the draft repertoire it is worth bearing in mind that formal status may not always correspond with how relevant the provided information is for the task of selecting a repertoire for purposes of a reference LGR. Further, the formal status of a document (for example as an official International Standard) unfortunately also does not necessarily correlate with its accuracy.

In conclusion, for the purpose of developing a reference LGR for the second level, the repertoire for any given language or script is unlikely to precisely match the information in *any single* source in all cases, even an official or other authoritative source, unless such source explicitly considered the question of IDNs[1]. The reason has much to do with the special nature of mnemonic identifiers as compared to regular text, and the attendant stability and security requirements. This limitation needs to be taken into account in defining the general development process.

[1] As was done for Root Zone LGRs, for example.

4 TARGET REPERTOIRE

The following sections describe in more detail the considerations in deciding the subset of code points that will form the target repertoire for a given LGR. For languages that use an ideographic writing system these considerations differ somewhat from the general case.

4.1 Subsets of Code Points Used in Writing a Language

There are a number of possible ways to subset the collection of code points from a given script that are used in connection with a particular language:

1. **Strict, or core subset**; for alphabetic writing systems this would usually correspond to the standard alphabet for the language plus any additional PVALID code points that are essential to writing the language in all supported writing systems.
2. **Common subset**; this extends the strict subset to include code points commonly used to write words in the language, but not strictly essential. For example, this subset would include letters needed to write common loan words, where they conventionally retain all or part of their original spelling. For ideographic writing systems, there is no well-defined cutoff between "essential" and "common use", although some countries have created minimal lists for educational purposes.
3. **Extended subset including names**; this would further add code points that, given prevailing practice, are commonly used for writing names, including names of foreign origin. For ideographic and alphabetic writing systems the practices around names differ; for alphabetic languages it is mostly a question of certain names of foreign origin conventionally retaining their original spelling.
4. **Full set including rarely used code points**; this would include all code points that are encountered in writing the language, however rarely they are used. This set would include less commonly or only rarely retained diacritics on letters in foreign words or names, as well as historic and other specialized forms. For ideographic writing systems, the set of rare characters is rather open ended. For these writing systems, many code points used exclusively for names may be considered specialized or even idiosyncratic and would thus fall into this subset.

For the purposes of developing a reference LGR, the chosen subset should be geared towards a set most useful for expressing mnemonic identifiers, whether they are based on words, names, or artificial monikers. The natural choice for a target repertoire would then fall somewhere between the Core and the Extended subset.

Because each of the Root Zone LGRs was developed as superset of the requirements for all modern languages with stable orthographies commonly using a given script, they, and the data on language-specific usage they reference can directly provide an authoritative, yet conservative starting repertoire for language-based LGRs. While some restrictions specific to the Root Zone (such as lack of digits) will need to be relaxed, their development process already reflects the due diligence and community consensus on what is both appropriate and required for secure mnemonic identifiers. Where available, they obviate the need to repeat the full analysis from scratch.

4.2 Subsets of Code Points Used in Writing a Script

For script-based reference LGRs, the target repertoire is given by the underlying Root Zone LGR for the script, augmented by these code points: digits, hyphen plus any code points excluded from the Root Zone LGR repertoire solely because of the fact that set Root Zone is more restrictive than other zones. [RFC6912].

4.3 Consideration applicable to both language and script based LGRs

In the context of IDN labels, the repertoire will need to satisfy additional constraints such as being limited to PVALID or CONTEXTO/CONTEXTJ code points as defined for IDNA 2008. For the purpose of this work, the code points in a given repertoire normally belong to a single script (except as indicated earlier), augmented with the Hyphen (U+002D) and the ASCII digits U+0030..U+0039. The IIS Guidelines for IDN Reference Tables [SE-Guidelines] used for the creation of the initial set of reference LGRs [DotSE] follows a similar model and forms part of the basis for this work. For scripts that use native digits, these may be supported by themselves, or as variants of ASCII digits, depending on local usage.

Code points that are not in common use are often not reliably recognized or entered by the user population; their inclusion in the repertoire may incur additional risk, for example in terms of confusability. On the other hand, a very strict subset would exclude many code points common in loan words or in names, including personal names of originally foreign origin, and which are ordinarily used with their spelling retained. A blanket prohibition on these, for purpose of the second level, seems not well motivated, particularly if the use of the affected code points is fairly common and they are often accessible even on traditional keyboards. Such code points, while they may be viewed as foreign and not part of the essential set are nevertheless easily recognized and identified by the users, and form part of the fixed spelling of the words in question.

They should be contrasted to code points used for historic, linguistic, poetic and other specialized purposes, including cases where there is no fixed spelling, or where the choice of a diacritic depends on the context, such as position of the word in a sentence (stress). Such code points provide little benefit for identifiers while increasing the attendant risks.

Some non-Latin writing systems make use of the Basic Latin subsets of the Latin script for a variety of purposes, such as corporate or product names. Extending the repertoire to include the Basic Latin subset would seem indicated in cases where this is common practice for the second level. However, there are also compelling reasons to exclude such script mixture. In the case of LGRs using the Cyrillic or Greek script for example, there would be a strong risk of confusion, due to shared letter shapes with Latin. In the case of the Hebrew or Arabic scripts, issues of bidirectional text layout would be introduced. In all of these cases, security and stability concerns would strongly argue against inclusion of the Basic Latin set in these LGRs. Even where these particular issues do not obtain, the inclusion of Basic Latin letters would require strong evidence in favor and would be subject to restrictions, such as requiring the

presence of at least one letter outside the ASCII range.[2] The ideographic scripts require some additional considerations described below in the Development Process section.

The alphabets for some Latin-based languages nominally do not contain some of the letters A-Z. As a matter of common practice these are always included in the repertoire.

4.4 Digits

For some scripts, there exist native digits. These may be used instead of, or in parallel to the European digits in the ASCII range. See [NUMERALS] for a discussion of the different preferences. Reference LGRs will add digits, unless script- or language-specific considerations demand otherwise.

For LGRs that combine multiple sets of digits (e.g. European and native) there is a desire to treat digits as semantic variants: the same numerical value will be treated as "same", no matter the digit set used. In addition, WLE rules will ensure that each label only contains digits from a single set. If multiple LGRs for different languages/scripts are used together in the same zone, all native digits sets from such LGRs would become variants of each other when transitivity of variants is enforced across the zone.

This poses a difficulty in those cases where digits are homoglyphs of other digits of different value (or homoglyph of letters). The variant system can handle one, but not both, and a choice would have to be made. As it turns out, the known cases occur for scripts that either do not use native digits, therefore obviating the need to define variants for them, or do not use European digits, allowing the reference LGR to base variants on homoglyph relations rather than digit value.

All-digit labels for ASCII digits, such as "123" are not IDNs and are out of scope[2]. Other than for Arabic, where this is required by the Bidi Rule of RFC 5893, no restriction is implemented on labels consisting only of native digits in these reference LGRs.

5 DEVELOPMENT PROCESS

5.1 Language-based LGRs

The process for the initial batch of language-based LGRs did proceed approximately as follows:

1. Start with the CLDR core set (excluding DISALLOWED code points)
2. Add European digits and HYPHEN-MINUS where applicable
3. Identify and qualify additional sources to verify and double check the set[3]
4. Review the set from any .SE IDN table in comparison to the set from step (2)
5. Compare the set to sets from available sources (except DISALLOWED)
6. Make adjustments based on available sources and/or expert input
7. Normally, the result should not exceed the CLDR auxiliary set
8. Enumerate the sources from step (4) for included code points

[2] A label consisting entirely of code points in the ASCII range, by definition, does not constitute an IDN and is therefore outside the remit for these reference LGRs. Note, however, that IDNs may be variants of such labels.
[3] Select the most authoritative and appropriate source as discussed in section "Sources" above.

Some language-based LGRs may instead be able to be derived from the existing work on Root Zone LGRs, as discussed above, bypassing this process. Instead, a streamlined process is used:

1. Start with the Root Zone LGR or proposed LGR for the script
2. Identify the subset of code points used for the language in question, using data compiled by community experts during the development of the Root Zone LGR
3. Add European digits and HYPHEN-MINUS where applicable
4. Add Native digits where applicable
5. Make adjustments where relevant to the second level and document
6. Remove variants and WLE rules no longer applicable (because of reduced repertoire)
7. Adjust any WLE or context rules to better fit the language (for example, where a script LGR had to be more permissive, or more restrictive than necessary for a language LGR)
8. Enumerate any external resources used for step (5)

5.2 Script-based LGRs

For script-based LGRs, the detailed analysis for the requirements of a public zone shared by multiple languages using the same script was already carried out during development of the Root Zone LGR. The development process for reference LGRs for the second level therefore simplifies to the task of relaxing restrictions that are particular to the Root Zone (letter principle), such as adding back the digits, hyphen and any such characters as were excluded solely because the Root Zone must be the most restrictive [RCF6912].

1. Start with the Root Zone LGR for the script
2. Add European digits and HYPHEN-MINUS, where applicable
3. Add native digits, where applicable
4. Make adjustments where relevant to the second level and document
5. Enumerate any external sources used for step (4)

Where a principal language for a script-based LGR uses most of the code points in the repertoire, an alternative process using the aforementioned steps may instead aim to identify the language subset and thus derive the language-based LGR from the script-based one. In many cases in the development of the Root Zone LGR, such language subsets were identified during development of the script LGR.

5.3 Languages using Ideographic Writing Systems

For ideographic languages, the process by necessity must orient itself on current best practice for the second level, while aiming for compatibility with the Root Zone practice to the degree possible, for consistency. As repertoire cut-offs are by necessity more arbitrary for ideographic writing systems, they tend to follow existing subsets, based on national standards, educational targets or international efforts at creating core sets [IICORE]. However, considerable development effort has been expended to arrive at workable repertoires and variants sets for these languages, most recently in the context of the RZ LGR. The purpose of the reference LGR cannot be to replace these efforts by a de-novo approach, but must rest on careful review, and perhaps a conservative selection based on or closely oriented on existing solutions.

The Japanese writing system mixes a fixed repertoire of Romaji (ASCII Latin subset) and Kana characters (Hiragana and Katakana) respectively, with an open-ended set of Kanji (ideographs). Due to the open-ended nature of the Kanji repertoire, to achieve stability it has been common practice to use the same subset for all usage pertaining to IDNs. This set is called JIS X 208-1990 [JISX], specified by the Information Technology Standards Commission of Japan (ITSCJ) and consists of 6356 ideographs which cover all basic needs for the Japanese language. Because of strong consensus on that set, it is not expected that any further enquiry would result in a different set; however any deviation from this set in the Root Zone LGR for Japanese should be taken into account.

The modern Korean writing system mixes common use of ASCII Latin with a large, but fixed set of Hangul set [Johab] consisting of 11,172 Hangul syllables. Korea has also made use of an ideographic system (Hanja) but that is not in established modern use for identifiers. Hanja, like Kanji is an open-ended set. Given the lack of track-record, it is not expected that Hanja will be included in the second-level Korean LGR.

The Chinese writing system also typically mixes the ASCII Latin and an open ended set of Hanzi Ideographs. There is no single authoritative source defining a Chinese set. There are several standards covering different Chinese communities. While some slight differences exist between current practice on the second level and the Root Zone LGR, most of these differences reflect attempts to increase security and reduce the multiplicity of allocatable variants. It is likely that a second- level Chinese LGR will benefit from the work performed at the root level.

5.4 Notes on the Intersection of Language and IDN Labels

From the [VIP-Cyrl] study: “The contents of the DNS are about mnemonics, not about ‘words’ or longer statements in particular languages. The fact that something can be written in a particular language, or even looked up in its dictionary, does not imply an entitlement to have that string appear in the DNS. Nevertheless, the aspiration is to implement an approach that approximates the natural language usage as nearly as possible.”

Why then a focus on language-based LGRs? Users are most familiar with the set of letters or other written symbols that are associated with their language, and labels that remain in that set (even if they do not spell out actual words) are more easily recognized and entered. They also naturally reflect the limitation of many input devices or input methods; this may affect some languages more than others.

The orthographies of some languages have features that do not lend themselves to the purpose of creating robust mnemonic labels; not supporting them may reduce the set of possible labels at the benefit of a more robust DNS. On the other hand, many languages routinely retain the elements of “foreign” orthographies in the spelling of loan words; in that case users are normally familiar with these additional letters and usually find them supported on their keyboards.

The goal of a reference LGR must therefore be to strive to provide the most complete, yet secure, minimal set of code points from which users of the language can construct mnemonic labels, without unduly limiting these mnemonics to match actual words. At the same time, and to be useful as a

reference, the LGR should indicate a suggested outer limit, beyond which letters tend to become unfamiliar to most users, which increases the risk that they are confused.

The orthography of languages changes (slowly) over time. While letters that were in historical use and are now obsolete should be excluded when newly developing an LGR, nothing in this document, or in the LGRs to be developed under these guidelines, shall be construed to apply to already delegated labels.

6 TARGET VARIANT SET

This section describes the considerations to be used in developing LGRs that have variants.

The process of developing the reference LGRs for the second level builds on existing work such as the Variants Issue Project [VIP] and the Root Zone LGR Project [RZ-LGR-Project] which provide a definition of variant as well as suggest which variants are appropriate for various scripts (See also [RZ-LGR-Documentation]). The results of these projects can be seen as authoritative for script-based LGRs. For many reference LGRs that are based on a language it may be natural to consider only the variants specific to the language in question under the implicit assumption that the zone will be limited to the given language. This may not be true in all cases, and it may be appropriate to include out-of-repertoire variants as well.

In contrast, for script reference LGRs and language LGRs derived from them it will be assumed by default that the zone may be shared among scripts. Where appropriate, cross-script variants are defined to allow the mutual exclusions of labels from different scripts that could otherwise be seen as equivalent substitute by the users. This is in addition to any in-script variants defined. In order to actually process cross-script variants during label allocation it is necessary to use a merged (common) LGR that embodies a superset of variants and repertoire (while ignoring certain script-specific features). See Section 5.2 in [RZ-LGR-Overview]. A single common file applies to any subset of covered LGRs.

In general, the variant problem is specific to the issue of internationalized identifiers and IDNs in particular. Therefore, it is not expected that the existing general sources will have much detail available that can be cited or applied directly.

However, existing practice on the second level and particularly the Root Zone should usefully be considered. Script-specific reference LGRs in particular can be expected to follow the Root Zone LGRs' treatment of variants by default, deviating only where necessary for reasons of additional repertoire (e.g. digits).

Multiple sets of digits are possible in some scripts. The general policy is to make all digits variants of all digits in the same LGR with the same value. As described below, whole label evaluation rules will ensure that all digits in any label are always from the same set. In this context, ideographic numerals (i.e. ideographs that can also represent decimal digits) are not considered variants of the European digits. Some digits in certain scripts may share a shape with a letter, requiring a variant relation.

If the LGR includes allocatable variants it will also include any necessary steps to strictly minimize the number of computed allocatable variant labels. These steps could include the use of special variant subtypes or WLE rules enforcing consistent choice of variant across a label (such as disallowing a mixture of original and variant code point in the same label).

Because many zones will support more than one language or more than one script, variants are defined in ways that balances in-script concerns with cross-script variants.

7 TARGET SET OF WHOLE LABEL EVALUATION RULES

LGRs are intended for mechanical evaluation of applied for labels. It is therefore proper to include some of the protocol limitations[4], such as the allowed occurrence of the hyphen, as well as other context rules among the Whole Label Evaluation (WLE) rules. This allows a one pass evaluation of applied-for labels for validity and variants.

Other restrictions, such as the requirements for Normalization and limits on the overall length of labels are best handled outside the LGR, as they are the same for all LGRs.

Some code points for certain languages may have limits on the context in which they can appear. These would be represented as WLE rules (or directly as context rules on the code points). It is expected that the majority of these rules will have been already documented in the relevant RFCs or in Root Zone LGRs.

Among others, context rules will be defined for

- HYPHEN---MINUS (U+002D) as specified in RFC5891.
- CONTEXTO and CONTEXTJ code points as specified in RFC5891 (e.g. U+ 0660).
- RTL labels specified in RFC5893 (e.g. U+0030).

To reduce the number of permutations of variants, some LGRs will have rules that limit all code points from a given label to be in the same subset, for instance all digits to be in the same set of digits.

8 SPECIFICATION AND DOCUMENTATION OF EACH LGR

A label generation ruleset (LGR) consists of four elements: a descriptive preamble, a code point repertoire, an optional set of variant code point definitions, together with a specification of which variants lead to valid allocatable or blocked variant labels, and an optional set of whole label evaluation (WLE) rules that further restrict the set of valid labels. The LGR will be specified in an XML file using the schema developed for specifying LGRs in XML [RFC7940]. Any variants defined will follow the guidelines in [RFC8228]. See also Sections 6.3, 6.5 and 6.6 in [RZ-LGR-Overview].

[4] The IDNA protocol only applies to IDNs, that is, those labels that contain at least one code point outside the ASCII range. LGRs that contain code points in the ASCII range would be expected to enforce this restriction.

A second document will contain a human readable summary of the LGR, mechanically generated from the XML, including notes on the LGR, its sources and its development. In certain cases, additional reports were created as part of an external review by linguistic and DNS security and stability experts, in addition to public review conducted for all LGRs. The ICANN website will maintain an archive of all reviews and public comments.

9 REVIEW PROCESS

For selected LGRs that are not closely based on existing LGRs it may be appropriate to institute a separate process of expert review, in addition to the normal community review. If such a review has been decided on as part of the project parameters, it would proceed as follows.

9.1 Linguistic Review

Where applicable, individual LGRs will be reviewed by expert reviewers addressing linguistic issues relevant to the specification of label generation rules for IDNs in the given language. Among other considerations, this review will be guided by the following questions:

1. Does the set of code points and label generation rules satisfactorily characterize the repertoire required for use of this language and script to define second-level labels?
   Specifically can all of the following be answered in the negative:
   a. Does the set of code points omit any required characters?
   b. Does the set of code points omit any desirable characters?
   c. Does the set of code points include any unnecessary characters?
   d. Does the set of code points include any undesired characters?

   e. Does the LGR omit any required variant rules?
   f. Does the LGR omit any desirable variant rules?
   g. Does the LGR include any unnecessary variant rules?
   h. Does the LGR include any undesired variant rules?

   i. Does the LGR omit any required WLE rules?
   j. Does the LGR omit any desirable WLE rules?
   k. Does the LGR include any unnecessary WLE rules?
   l. Does the LGR include any undesired WLE rules?
2. Are the authorities cited by the LGR among the best available in relation to the relevant issues? Could use of other authorities have led to better choices in the set of CPs and rules?
3. Has adequate provision been made for labels (e.g. for familiar but alien names or loan words) which exceed the bounds of repertoire of code points essential for the language?
4. Will extended code points (and variant or WLE rules) have undesired consequences for the repertoire as a whole?
5. Does the XML file accurately characterize the desired set of code points and rules for the language and script, and so match the descriptive document?

9.2 DNS Security and Stability Review

Where applicable, individual LGRs will be separately reviewed for stability and security issues that are pertinent to a label generation ruleset for IDNs on the second level. Among other considerations the review will be guided by the following questions:

1. Does the repertoire allow undesirable script mixing?
2. Does the LGR include only PVALID, CONTEXTJ or CONTEXTO code points?
3. If the LGR contains CONTEXTJ/CONTEXTO code points, is sufficient justification given for their inclusion in the LGR?
4. If the LGR includes combining marks:
    a. Are they limited to specific code point sequences?
    b. If not, does the LGR use other means (rules, variant relations) to restrict undesirable sequences using these combining marks?
5. If the LGR contains code points or variants that may present a security or stability concern, does it include rules to mitigate the risks?
6. Are there any security or stability concerns with regards to variants in the LGR?
    a. Does the LGR omit any variant mappings that are necessary to mitigate security risks?
    b. Does the LGR include any variant mappings that may cause security concerns (e.g. overly complex, over-produce allocatable variants, non-symmetrical or non-transitive?)
    c. Does the LGR include any variant mappings that cause the LGR to be not well behaved (e.g. overlapped or null variants that are not properly constrained)?
    d. If the LGR includes allocatable variants, does it take any necessary steps to minimize allocatable variant labels (whether via special variant subtypes or WLE rules)?
7. Are there any security or stability concerns with regards to WLE rules in the LGR?
    a. Does the LGR omit any WLE rules that are necessary to mitigate security risks?
    b. Does the LGR define WLE rules that may cause security concerns (e.g. overly complex, or impermissibly broad)?
8. Does the LGR satisfy, or otherwise discuss and adequately address any tension among, the principles laid out in Sections 3 and 4 of RFC 6912?

10 REFERENCES

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

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

[IANA] Internet Assigned Numbers Authority (IANA): “Repository of IDN Practices” http://www.iana.org/domains/idn-tables

[JISX] JIS X 0208-1990 Japanese Standards Association. Jouhou koukan you kanji fugoukei (Code of the Japanese Graphic Character Set for Information Interchange).

[JOHAB] KSX 1001:2004 (formerly KS C 5601-1992), Annex 3: Johab, Korean Industrial Standards Association. Code for Information Interchange (Hangeul and Hanja) (Jeongbo gyohwanyong buhogye).

[MSR] Integration Panel, "Maximal Starting Repertoire: MSR-3 Overview and Rationale", (ICANN, Los Angeles, 28 March 2018) https://www.icann.org/en/system/files/files/msr-3-overview-28mar18-en.pdf

[NORDIC] Nordic Cultural Requirements on Information Technology, INSTA Technical Report, STRI TS3, 1992, ISBN 9979-9004-3-1

[NUMERALS] Wikimedia, “Numerals in Indic Languages & Indic language Wikipedias” (accessed 2020-03-10), https://meta.wikimedia.org/wiki/CISA2K/Indic_Languages/Numerals_in_Indic_Languages_%26_Indic_language_Wikipedias

[RFC5890] Klensin, J., "Internationalized Domain Names for Applications (IDNA): Definitions and Document Framework", RFC 5890, August 2010.

[RFC5891] Klensin, J., "Internationalized Domain Names in Applications (IDNA): Protocol", RFC 5891, August 2010.

[RFC5892] Faltstrom, P., "The Unicode Code Points and Internationalized Domain Names for Applications (IDNA)", RFC 5892, August 2010.

[RFC5893] Alvestrand, H. and C. Karp, "Right-to-Left Scripts for Internationalized Domain Names for Applications (IDNA)", RFC 5893, August 2010.

[RFC5894] Klensin, J., "Internationalized Domain Names for Applications (IDNA): Background, Explanation, and Rationale", RFC 5894, August 2010.

[RFC6912] Sullivan, A., et al. “Principles for Unicode Code Point Inclusion in Labels in the DNS”, RFC 6912, April 2013

[RFC7940] Davies, J and Asmus Freytag: “Representing Label Generation Rulesets using XML” RFC 7940, August 2016.

[RFC 8228] A. Freytag, "Guidance on Designing Label Generation Rulesets (LGRs) Supporting Variant Labels", RFC 8228, August 2017.

[RZ-LGR-Project] ICANN, “Root Zone Label Generation Rules”, Project web page with links to most current versions and project goals, guidelines and contributors. https://www.icann.org/resources/pages/root-zone-lgr-2015-06-21-en

[RZ-LGR-Overview] Integration Panel, “Root Zone Label Generation Rules — LGR-3 Overview and Summary”, ICANN, 2019-07-10 (PDF) https://www.icann.org/sites/default/files/lgr/lgr-3-overview-10jul19-en.pdf

[SEGuidelines] Guidelines for IDN References Tables, 2014-10-10, Version A https://github.com/dotse/IDN-ref-tables/blob/master/Guidelines%20for%20IDN%20Reference%20Tables.pdf

[RZ-LGR-Documentation] ICANN, “Root Zone LGR Project Documentation” https://www.icann.org/resources/pages/root-zone-lgr-documentation-2017-12-15-en

[VIP] The IDN Variant Issues Project, Internet Corporation for Assigned Names and Numbers, “A Study of Issues Related to the Management of IDN Variant TLDs (Integrated Issues Report)” (ICANN, Los Angeles, February 2012), https://www.icann.org/en/system/files/files/idn-vip-integrated-issues-final-clean-20feb12-en.pdf

[VIP-Cyrl] The IDN Variant Issues Project, Internet Corporation for Assigned Names and Numbers, “IDN Variant TLDs – Cyrillic Script Issues”, (ICANN, Los Angeles, October 2011) https://archive.icann.org/en/topics/new-gtlds/cyrillic-vip-issues-report-06oct11-en.pdf