

Requirements for LGR Proposals from Generation Panels

REVISION 12 - 2014-04-08

1 Overview

This document describes the requirements that LGR Proposals from Generation Panels must meet to be accepted for review by the Integration Panel.

2 Files

Proposals are not single documents; they typically consist, at a minimum, of a main document giving background, discussion, and justification and a separate XML file with the normative definition of the LGR. Annotated code tables, such as were provided with the MSR, are not required; however, if variants or whole label evaluation rules are specified, the Integration Panel requests that test cases are provided in machine readable form.

3 Script(s) covered

In the main document, list the script name(s) and corresponding ISO15924 script code(s) that the panel is submitting a proposal for.

Attach one LGR-XML file per script. The script codes should match the <language> element in the XML format. Use a single script code; for Japanese use “und-Japn” to cover the mixture of Han, Hiragana and Katakana.

For the normative specification for representing the LGR in XML, see [XML-LGR] and for the specific conventions used for the Root Zone LGR see Section 11.

4 Languages using each script

In the main document, list the principal languages and, if any, principal minority languages using each script.

Include languages in active, modern use, with a substantial community that uses the script to conduct everyday activities and for which gTLD or ccTLD applications can reasonably be expected at some point.

For some scripts with very diverse user communities (like Arabic, Latin, and Cyrillic), an *exhaustive* list of languages may not be feasible. In such cases, the focus should be on listing of as many

languages or language groups that are known to actively use the script for everyday writing as feasible.

Provide a breakdown of which languages have been investigated and covered, and those that have not been investigated for lack of expertise or participation on the Generation Panel.

Where appropriate, list countries or territories where this script has significant user communities.

Indicate whether any language(s) had no community representation on the panel. Describe the panel's outreach efforts to these user communities.

5 Related scripts

Are there related scripts (either among the scripts that this panel covers, or across panels)? If so, describe in the main document:

1. What are they, and how are they related?
2. What specific issues did the panel consider?

What approach did the panel undertake to ensure consistent treatment of repertoires and variants across related scripts?

Did the panel subdivide itself into sub-panels for related scripts? If so, how were they structured and work together with each other?

Did the panel coordinate with other panels? If so, provide details.

6 Justification and References

The Process Goals and the Principles in the [Procedure] (§§ A.3.4 and A.3.6) are meant to guide the Generation Panels and the Integration Panel. In the main document, discuss the the following as they relate to the Process Goals and Principles:

- the process of selecting the repertoire;
- if variants were specified, their choice and disposition;
- if Whole Label Evaluation rules were specified, how they were define;
- and the overall result of the Generation Panel's process.

For each of these items, indicate the sources of reference used, including the use of outside experts (advisors). Include advisors reports where available.

7 Code point repertoire

In the main document, describe how the panel arrived at the code point repertoire.

In the Justification for the repertoire, cover every code point included. Summary justification for well known subsets are acceptable, such as “code points in the range XXXX..YYYY form the basic alphabet for language Zzzz”.

Separately discuss any included code point where the inclusion could be seen as potentially problematic for the root zone.

Where appropriate, discuss how specific languages investigated relate to specific code points (or rules) in the proposal.

Did any controversy / contention or points of disagreement exist regarding any of the code points and rules (including those that did not make it into the proposed repertoire and rule set) contemplated by the panel? If so, provide a summary of the issues and how the panel reached its conclusion. Where appropriate provide a link to any archived discussions.

In the XML document, list the repertoire, as discussed below.

8 Variant mappings and dispositions

Your proposed LGR might contains rules represented by action elements in the XML format (other than the default rules) that transform these disposition values to the final dispositions for variant labels. If so, explain what those rules intend to accomplish and why they are appropriate for the root zone; also explain their corresponding disposition.

Provide several example labels (test cases) along with expected results of running the variant rules.

If the proposal contains code point variant rules that lead to allocatable or blocked labels, include examples of such labels as well as of labels without variants.

In the XML document, specify the variant mappings, dispositions (and optionally actions) as discussed below.

9 Whole label evaluation rules (WLE)

If there are WLEs in your proposed LGR, provide an explanation in the main document of what they are intended to accomplish and why they are appropriate for the Root Zone.

Provide examples of labels that pass these rules and labels that fail (test cases).

In the XML document, list any WLE rules and associated actions as discussed below.

10 Format for test cases

For test cases, use one or more plain text files with test cases in UTF-8, with one label per line. Lines starting with # may be used for comments. Use comments to indicate, for example, which labels should pass or fail the WLE rules.

11 XML Format Features Used or not Used for LGR Proposals

This section describes features permitted or required for use in the XML files that accompany LGR Proposals, and in some cases, specifies required settings.

11.1 File format and lgr element

For the formal specification of the XML format see [XML-LGR].

The file must contain one <lgr> element, containing a <meta>, <data> and <rules> element in that order. The file begins:

```
<?xml version="1.0"?>
<lgr xmlns="http://www.iana.org/lgr/0.1">
```

The xml name space in the “xmlns” attribute given here is temporary, so check for updates before submitting the file.

The use of xml comments in the file is discouraged, because they are not preserved by the tools used in the integration process. The use of the “comment” attribute in the XML format is preferred. If an element does not allow a comment attribute, its containing element usually does, or the information can be provided in the <description> element or as part of the main documentation for the proposal.

ICANN or the Integration Panel can assist with creating and validating the XML prior to submission.

11.2 Meta element

- A meta element must be present in each XML file with these elements:

```
<version>
<date>
<language>
<domain>
<unicode-version>
<description>
```

- The version element is set to “1” for the first submission, and increased by 1 for each re-submission, any comment attribute is ignored.
- The date element gives the date of submission.
- The single <language> element must contain the script code for the LGR. Since the root zone LGR operates on a script level, the language subtag should be set to “und-”, e.g. “und-Cyrl” or “und-Japn”.
- The single domain element is set to “.” to indicate the root zone.
- The unicode version element is set to “6.3.0”.
- The description element should give some summary information, but is not intended to contain the full rationale or justification text for the LGR proposal. It should contain any information relevant to understanding the file itself.

- A <references> element may be provided. References may be used to give information about characters or rules, for example by the source of code point (either by Unicode version, or by some other character collection). References may also cite sections of the main proposal document. Note that the “ref” element on a code point is multi-valued, allowing several ids that are separated by space.
- A <validity-start> or <validity-end> element must not be used.

11.3 Data element

The <data> element contains all information on repertoire, and, where present, variants.

All features defined for <char> and <range> elements are permissible. Any intended use of “when” and “not-when” attributes may pose potential problems in integration and should be discussed with the Integration Panel.

“Tag” attributes are required only if a whole label evaluation rule is added that makes use of them. The script tag attributes from the MSR (e.g. “sc:Latn”) may be retained. (Note that the “tag” attribute is multi-valued: it consists of a list of space-separated tags.)

The mapping rules for variants, if any are specified, must form a set closed under symmetry and transitivity. Hence, if

```
<char cp="XXXX"> <var cp="YYYY" /></char>
```

is specified, symmetry requires that

```
<char cp="YYYY"> <var cp="XXXX" /></char>
```

is specified also. If

```
<char cp="XXXX"> <var cp="YYYY" /></char>
<char cp="YYYY"> <var cp="ZZZZ" /></char>
```

are specified, transitivity requires that

```
<char cp="ZZZZ"> <var cp="XXXX" /></char>
```

is specified also.

For each <var> element specifying a mapping, a “disp” attribute (not shown in the simplified example above) must be supplied, to set the disposition value for that variant code point mapping. These “disp” attributes do not have to be symmetric or transitive. For example:

```
<char cp="ZZZZ"> <var cp="XXXX" disp="blocked" /></char>
```

Typically, the disposition value is one of “blocked” or “allocatable”. The default rules are sufficient to evaluate these into the corresponding variant label dispositions.

Other values for the “disp” attribute on var elements are only permitted if corresponding action elements are defined in the rules section that evaluate these into either “blocked” or “allocatable”

dispositions for variant labels (see [VariantRules] for more information). The specification of reflexive variants

```
<char cp="XXXX"> <var cp="XXXX" /></char>
```

is permitted. The intent to use reflexive mappings or special “disp” values on <var> elements should be discussed with the Integration panel prior to submission.

In case the target of a variant mapping is outside the repertoire, symmetry requires adding the target to the repertoire. If such a code point is given a reflexive mapping as shown, the default rules will treat it correctly as an out-of-repertoire code point.

```
<char cp="XXXX">
  <var cp="XXXX" disp="out-of-repertoire" />
</char>
```

ICANN or the Integration Panel can assist with validating these requirements before submission.

Note: It may be easier to edit the repertoire and variant information as a plain text file, for example in an existing IDN table format, and then use a simple script to convert that to the XML format for the <data> section. This is how the Integration Panel created the XML for the MSR.

Alternatively, for a script not using variants or WLE rules, the data section could simply be created by copying only the desired entries from the MSR.

11.4 Rules element

All script LGRs require a rules element. The entire contents of the default <rules> element from the MSR MUST be included in all LGRs. There’s no requirement for an LGR to add to the default rules, and in fact it is likely that few Generation Panels will need to add additional rules.

As any intended additions to the rules may pose potential issues in integration, they should be discussed with the Integration Panel prior to submission.

Permitted disposition values for labels as result of any added actions are limited to “invalid”, “blocked”, and “allocatable”.

11.5 Example XML file for LGR Proposal

```
<?xml version="1.0"?>
<lgr xmlns="http://www.iana.org/lgr/0.1">
  <meta>
    <version>1</version>
    <date>2014-03-01</date>
    <language>und-Arab</language>
    <domain>.</domain>
    <description>
```

```

        Example specification for an Arabic script
        LGR Proposal. For Justification and Notes
        see the main document.
        This example contains no variants or added
        whole label evaluation rules. A dummy code
        point is provided so this example can be
        validated.
        The rules element contains the default
        rules from MSR-1.
    </description>
    <unicode-version>6.3.0</unicode-version>

    <references>
        <reference id="0">See description</reference>
    </references>
</meta>
<data>
    <char cp="0000" ref="0"/>
</data>
<rules>
    <!--Character class definitions (if any) go here-->

    <!--Whole label evaluation and context rules go here-->
    <rule name="leading-combining-mark">
        <start />
        <union>
            <class property="gc:Mn" />
            <class property="gc:Mc" />
        </union>
    </rule>

    <!--Action elements go here - order defines precedence-->
    <action disp="invalid" match="leading-combining-mark" />
    <action disp="invalid" any-variant="out-of-repertoire-var"
comment="any variant label with a code point out of repertoire is invalid" />
    <action disp="blocked" any-variant="blocked" />
    <action disp="allocatable" any-variant="allocatable" />
    <action disp="valid" comment="catch all" />
</rules>
</lgr>

```

While additional rules and actions may be added, the <rules> element must always contain a copy of the default rules supplied in the MSR.

12 References

[Procedure] Internet Corporation for Assigned Names and Numbers, "Procedure to Develop and Maintain the Label Generation Rules for the Root Zone in Respect of IDNA Labels." (Los Angeles, California: ICANN, March, 2013) <http://www.icann.org/en/resources/idn/variant-tlds/draft-lgr-procedure-20mar13-en.pdf>

[VariantRules] Integration Panel, "Variant Rules"

[XML-LGR] Davies, K. and A. Freytag, "Representing Label Generation Rulesets using XML",
<http://tools.ietf.org/html/draft-davies-idntables-07/>. Visited 2014-03-18.