ICANN 60
ANNUAL GENERAL
ABU DHABI
28 October–3 November 2017
IDN Root Zone LGR Workshop

ICANN 60
1 November 2017
Agenda

- Overview of Root Zone LGR version 2 - Marc Blanchet
- How to Use RZ-LGR-2 - Michel Suignard
- Update on LGR Toolset - Audric Schiltknecht
- Community Updates
  - Chinese GP Update - Kenny HUANG, Wei WANG
  - Japanese GP Update - Hiro Hotta
  - Korean GP Update - KIM Kyongsok
  - Greek GP Update - Panagiotis Papaspiropoulos
- Q/A
Root Zone LGR version 2 (RZ-LGR2)

Marc Blanchet
Integration Panel
What is RZ-LGR-2?

- Set of normative XML files and informative documents for 6 scripts
- Governs the way the root zone is operated for a given set of scripts
- Determines which Unicode code points are permitted in U-labels
- Determines which variants are allocatable or blocked
- Output is used by other procedures determining whether a label is allocated, delegated
Script Based

- Each label in root zone belongs to a single script
- In the future, for cases like Japanese and Korean, the script concept will be extended to cover well-defined sets of primary scripts (as Kana and Kanji (i.e. Han) for Japanese)
- RZ-LGR is released in stages to allow some LGRs to be available sooner in the root zone
  - Version 1 was Arabic only
  - Version 2 adds five more scripts
  - Many more in the pipeline
# Scripts Covered

<table>
<thead>
<tr>
<th>Script</th>
<th>Name in script</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arabic</td>
<td>العربية</td>
</tr>
<tr>
<td>Ethiopic</td>
<td>የጉራቹንዱ ውስጥ</td>
</tr>
<tr>
<td>Georgian</td>
<td>ქართული ენა</td>
</tr>
<tr>
<td>Khmer</td>
<td>សាធារណៈ</td>
</tr>
<tr>
<td>Lao</td>
<td>ສາຫາລັດ</td>
</tr>
<tr>
<td>Thai</td>
<td>อักษรไทย</td>
</tr>
</tbody>
</table>
RZ-LGR-2 Content

- **Normative**
  - One XML per script (element XML)
  - A common XML file

- **Informative**
  - Overview document (PDF)
  - HTML representation of the 7 XML files
  - Repertoire table (PDF)
    - Example: Thai
List of XML Files

- All files in [https://www.icann.org/sites/default/files/lgr/](https://www.icann.org/sites/default/files/lgr/)

<table>
<thead>
<tr>
<th>Script</th>
<th>File name in URL</th>
</tr>
</thead>
</table>

- Also provided, a mechanically generated and non-normative HTML presentation using same file name with .html extension (example: [https://www.icann.org/sites/default/files/lgr/lgr-2-arabic-script-26jul17-en.html](https://www.icann.org/sites/default/files/lgr/lgr-2-arabic-script-26jul17-en.html))
How Was It Created?

- Script based XML files generated by the Generation Panels
- These files reviewed and integrated by the Integration Panel
- Common file created by IP is the cumulative set of all integrated LGRs for:
  - Repertoire
  - Variants all in ‘blocked’ type
  - Character classes (renamed to avoid collision)
  - WLE rules (also renamed)
  - Actions
How to Use It?

- Covered in another presentation

LGR-2 Is Here, What Can You Do With It?
Thank You and Questions

- Root Zone Label Generation Rules (RZ-LGR-1 and RZ-LGR-2)

- ICANN IDN Document Repository
  - [https://community.icann.org/display/croscomlgrprocedure/Document+Repository](https://community.icann.org/display/croscomlgrprocedure/Document+Repository)

- RFC 7940 Representing Label Generation Rule Sets Using XML
  - [https://www.rfc-editor.org/info/rfc7940](https://www.rfc-editor.org/info/rfc7940)

- RFC 8228 Guidance on Designing Label Generation Rule Sets (LGRs) Supporting Variant Labels
  - [https://www.rfc-editor.org/rfc/rfc8228.txt](https://www.rfc-editor.org/rfc/rfc8228.txt)
How to Use RZ-LGR-2

Michel Suignard
Integration Panel
RZ-LGR-2 Is Here, What Can You Do with It?

**Existing TLD:** `t1`

**Applied-for TLD:** `t1`

**LGR Tool**
- Ethioic ኢትዮጵያ
- Arabic العربية
- Thai ไทย
- ...28 scripts

**Secure and Stable Results:**
- **Invalid TLD Label**
- **Valid TLD Label**
- **All Variants:** `{ t1v1, t1v2, t1v3, t1v4 }`
- **Allocatable**
- **Blocked**
The Root Zone LGR is split into
- One **Element LGR** per script (N files)
- A single **Merged LGR** (1 file)
- Each of these files in XML (normative) and HTML (informative)
- Plus documentation files (Overview and Code Tables)

Three tasks in using the LGR in applying for a label
- Validate
- Generate allocatable variants
- Check for collisions with delegated labels and their variants

Walking through these tasks will explain why (N+1 files are used)
Applying for a Label

- Each label in the Root Zone is in a single script and the application defines which script a label is applied for

1. Select script based **Element LGR** corresponding to application

2. Use Element LGR to validate label
   - check code points in label against repertoire
   - check context constraints on code points
   - check whole label constraints

3. Generate allocatable variant labels (if any)

**Note:** Element LGRs are based on the submitted script LGRs
- Lightly edited for consistency and common conventions
- Guaranteed match results of Script LGR proposals
4. Check for collisions
   o Use **Merged** file (aka. Common)
     • Merged file contains all non-reflexive variant mappings
     • All variant mapping types are set to “blocked” as needed for collision checking.
     • Variants mappings are symmetric and transitive:
       • Any label or variant label is part of only one variant label set
       • In each set, all labels are variants of each other
   o Calculate index variant
     • Map each code point to its smallest variant code point
   o Compare to index variants for delegated labels
     • If index variants match, either the labels or their variants collide
Example

- Example: existing label for TLD "همراه"
- Script: Arab
- Process with Element-LGR: lgr-2-arabic-script-26jul17-en.xml
  - Details of steps in evaluation are described in RFC 7940
- This label has
  - 1 original label "همراه" (0647 0645 0631 0627 0647)
  - 1 allocatable variant "ہمراه" (06C1 0645 0631 0627 06C1)
  - 268 blocked variants
  - 50 invalid variants
    - For example mixed use of 0647 vs. 06C1:
Other Tasks

- Other tasks for Common LGR
  - Verification against script LGRs
    - merged file contains all variant mappings
    - merged file contains all context/WLE rules

Notes:

- Merged file is derived from script LGR files
- Differences from script LGR files
  - Certain items (tags, classes, rules) renamed to avoid collisions
  - Comments/descriptions edited for consistency
  - References point to Element LGRs not source documents

- N+1 format (scripts + merged) could be useful for any other zones that support multiple scripts in parallel
Other Files

- Overview describes
  - How LGR is laid out
  - Review of submissions that were integrated
  - How to use the LGR
  - The main features for each script LGR (summary)

- Code Tables give visual overview of repertoire in relation to MSR-2 and set of IDNA2008 PVALID code points

- The informative HTML versions of the Element and Merged LGR
  - Easier to read for human reviewers
  - Provide some calculated info (counts, etc.)
  - Show glyph shapes for code points, Unicode names, etc.
What Should GPs Note for Future RZ-LGRs?

- Script LGR proposals are archived
  - Sole documentation of design decision behind LGR
  - Make sure all decisions are documented and cite references

- Script LGRs are input to Element LGRs
  - Following common templates and conventions
    - Section numbering, comment convention for XML
    - Conventions for organizing <description>
      - Minimizes need to adjust these

- Provide required out-of-repertoire variants
  - Any cross-script homoglyphs
  - Any in-repertoire variants
Thank You and Questions

- Packaging the MSR and RZ-LGR

- RFC7940
  https://tools.ietf.org/rfc/rfc7940

- IDN TLD portal (for RZ-LGR-2 files)
LGR Toolset Update

Audric Schiltknecht, Julien Bernard, Marc Blanchet
Viagénie
Contents

- LGR Toolset summary in one slide
- New features in 2017
  - Sets of LGR files
  - Label validation
  - HTML export
  - Interface improvements
LGR Toolset Summary

- Toolset to
  - Create, update, use Label Generation Rules
  - Validate labels, generate variants, verify collisions

- Available(*)
  - Opensource
  - Online as a service

- As
  - Cmdline and libraries in python
  - Web interface

LGR Sets

- Implements the Root Zone LGR structure as a set of LGRs(*)
  - Repertoire: cumulative repertoire of all the Element LGRs
  - Variants: union of the variant mappings from all the Element LGRs, with « blocked » type
  - Classes: union of the character classes from all the Element LGRs. Name are prefixed by script of Element LGR
  - WLE: cumulative set of Whole Label Evaluation rules and actions for all Element LGRs. Name are prefixed by script of Element LGR

* As described in Root Zone Label Generation Rules — RZ-LGR-2
## LGR Sets - Interface

### Import existing LGR

To import LGR set, select the XML files composing the LGR set. The set will automatically be created.

- **Select file(s)**: Browse...
  - 2 files selected.
  - If you select more than one file, this will create a LGR set.

- **Validating repertoire**
  - Code points will be limited to the selected repertoire.

- **LGR set name**
  - Root-Zone
  - The name of the set

⚠️ Note that importing large LGR files may take significant time to load on your browser.

### LGR Tool / root-zone - Root Zone LGR

<table>
<thead>
<tr>
<th>Code point</th>
<th>Character Name</th>
<th>Comments</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>-10D0 (α) 0 Variant(s)</td>
<td>GEORGIAN LETTER AN</td>
<td>Georgian</td>
<td>See code</td>
</tr>
<tr>
<td>-10D1 (β) 0 Variant(s)</td>
<td>GEORGIAN LETTER BAN</td>
<td>Georgian</td>
<td>See code</td>
</tr>
<tr>
<td>-10D2 (γ) 0 Variant(s)</td>
<td>GEORGIAN LETTER GAN</td>
<td>Georgian</td>
<td>See code</td>
</tr>
<tr>
<td>-10D3 (δ) 0 Variant(s)</td>
<td>GEORGIAN LETTER DON</td>
<td>Georgian</td>
<td>See code</td>
</tr>
</tbody>
</table>
### List of LGRs in the set

- lgr-2-georgian-script-01jun17-en
- lgr-2-khmer-script-01jun17-en

### Code Points

<table>
<thead>
<tr>
<th>Code Point</th>
<th>Character Name</th>
<th>Comments</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>U+10D0 (α)</td>
<td>GEORGIAN LETTER AN</td>
<td>Georgian</td>
<td>See code point</td>
</tr>
<tr>
<td>U+10D1 (🌐)</td>
<td>GEORGIAN LETTER BAN</td>
<td>Georgian</td>
<td>See code point</td>
</tr>
<tr>
<td>U+10D2 (羝)</td>
<td>GEORGIAN LETTER GAN</td>
<td>Georgian</td>
<td>See code point</td>
</tr>
<tr>
<td>U+10D3 (羝)</td>
<td>GEORGIAN LETTER DON</td>
<td>Georgian</td>
<td>See code point</td>
</tr>
</tbody>
</table>
Label Validation on a LGR Set

Delegated labels

Scripts from Elements LGR
LGR Validation - Process

- Element LGR is retrieved from selected script
- Label is validated against this Element LGR
- Collisions are checked using the Common-Merged LGR between
  - Input label (and its variants)
  - List of delegated labels (and their variants)
- If no collision, then generates variants using Element LGR
LGR Validation – Delegatable Use Case

Label valid in Element LGR (Khmer)

No collisions with delegated labels
LGR Validation – Non-Delegatable Use Case

Label valid in Element LGR (Khmer)

Collide with one of the delegated labels
LGR Sets - Tools

- Tools have been updated to handle LGR Sets (where applicable)
- New « Allocated Set labels » parameter: (optional) list of labels allocated in the given set. Used e.g. to check collisions
- New parameter for label disposition: « Script » of the label(s)
- New tool: cross-script variants
HTML Export

- Create a human-readable static version of the LGR
- Support LGR Sets
- Accessible from web interface or as a standalone cmdline (but still need some configuration)
Root Zone Label Generation Ruleset for the Arabic Script

Overview

This file contains a set of Label Generation Rules (LGR) for Arabic as would be appropriate for the Root Zone. For more details on this LGR and its development see TF-AIDN, "Proposal for Arabic Script Root Zone LGR", Version 3.4, 2013 November 18 [Proposal].

Repertoire

The repertoire is based on Section 3.2 in [Proposal] and only includes code points used by languages that are actively written in the Arabic script. It excludes code points for which TF-AIDN was unable to find sufficient evidence of use (see Appendix F in [Proposal]). The repertoire is based on [MSR-2], which is a subset of Unicode 6.3 [Unicode 6.3].
# Repertoire

## Summary

<table>
<thead>
<tr>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of elements in repertoire</td>
<td>385</td>
</tr>
<tr>
<td>Number of ranges in repertoire</td>
<td>0</td>
</tr>
<tr>
<td>Number of code point sequences</td>
<td>7</td>
</tr>
</tbody>
</table>

## Repertoire by Code Point

The following table lists the repertoire by code point (or code point sequence). The data in the Script and Name column are extracted from the Unicode character database. Where the comment in the original LGR is equal to the character name, it has been suppressed.

For any code point or sequence for which a variant is defined, the link to the associated variant set, or if mapped to itself, the variant type of that mapping is provided in the variants column.

<table>
<thead>
<tr>
<th>#</th>
<th>Code Point</th>
<th>Glyph</th>
<th>Script</th>
<th>Name</th>
<th>Tags</th>
<th>Required Context</th>
<th>Variants</th>
<th>Comment</th>
<th>References</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>U+0061</td>
<td>a</td>
<td>Latin</td>
<td>LATIN SMALL LETTER A</td>
<td>Latin</td>
<td></td>
<td>culus</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>U+0062</td>
<td>b</td>
<td>Latin</td>
<td>LATIN SMALL LETTER B</td>
<td>Latin</td>
<td></td>
<td>culus</td>
<td></td>
<td>[0]</td>
</tr>
<tr>
<td>3</td>
<td>U+0063</td>
<td>c</td>
<td>Latin</td>
<td>LATIN SMALL LETTER C</td>
<td>Latin</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>U+17D0</td>
<td>ិ̇</td>
<td>Khmer</td>
<td>KHMER SIGN SAMYOK SANYA</td>
<td>Khmer, und-Khmr-sign, sc:Khmr</td>
<td></td>
<td>var</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>U+17D2</td>
<td>ិ</td>
<td>Khmer</td>
<td>KHMER SIGN COENG</td>
<td>Khmer, sc:Khmr, und-Khmr-coeng</td>
<td></td>
<td></td>
<td></td>
<td>[2], [207], [208], [209]</td>
</tr>
<tr>
<td>6</td>
<td>U+17D2 U+178A</td>
<td>ិ d</td>
<td>Khmer</td>
<td>KHMER SIGN COENG</td>
<td>Khmer, sc:Khmr, und-Khmr-coeng</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>U+17D2 U+178F</td>
<td>ិ a</td>
<td>Khmer</td>
<td>KHMER SIGN COENG</td>
<td>Khmer, sc:Khmr, und-Khmr-coeng</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Link to WLE: [Link to WLE](#)

Link to variant set: [Link to variant set](#)
HTML Export – Variant Sets

Variant Sets

Summary

<table>
<thead>
<tr>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of variant sets</td>
<td>26</td>
</tr>
<tr>
<td>Largest variant set</td>
<td>8</td>
</tr>
<tr>
<td>Ordinary Variants by Type</td>
<td>blocked (243)</td>
</tr>
</tbody>
</table>

The following tables list each pair of variant mappings on one row.

In a properly specified LGR, all members of each variant set are variants of each other, a property called transitivity. Because of that, all variant sets are necessarily disjoint. In each set, shading is used to group mappings from the same source code point or sequence.

Variant Set 1 — 3 Members

<table>
<thead>
<tr>
<th>#</th>
<th>Source</th>
<th>Glyph</th>
<th>Target</th>
<th>Glyph</th>
<th>Type(s)</th>
<th>References</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>U+0067</td>
<td>g</td>
<td>U+0067</td>
<td>g</td>
<td>blocked</td>
<td></td>
<td>Identity</td>
</tr>
<tr>
<td>2</td>
<td>U+0067</td>
<td>g</td>
<td>U+0581</td>
<td>g</td>
<td>blocked</td>
<td></td>
<td>Homoglyph</td>
</tr>
<tr>
<td>3</td>
<td>U+0581</td>
<td>g</td>
<td>U+0067</td>
<td>g</td>
<td>blocked</td>
<td></td>
<td>Latin small letter G</td>
</tr>
</tbody>
</table>

Variant Set 2 — 8 Members

<table>
<thead>
<tr>
<th>#</th>
<th>Source</th>
<th>Glyph</th>
<th>Target</th>
<th>Glyph</th>
<th>Type(s)</th>
<th>References</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>U+0068</td>
<td>h</td>
<td>U+0068</td>
<td>h</td>
<td>blocked</td>
<td></td>
<td>Identity</td>
</tr>
<tr>
<td>2</td>
<td>U+0068</td>
<td>h</td>
<td>U+04BB</td>
<td>h</td>
<td>blocked</td>
<td></td>
<td>Homoglyph</td>
</tr>
<tr>
<td>3</td>
<td>U+0068</td>
<td>h</td>
<td>U+0570</td>
<td>h</td>
<td>blocked</td>
<td></td>
<td>Homoglyph</td>
</tr>
<tr>
<td>4</td>
<td>U+04BB</td>
<td>h</td>
<td>U+0068</td>
<td>h</td>
<td>blocked</td>
<td></td>
<td>Homoglyph</td>
</tr>
<tr>
<td>5</td>
<td>U+04BB</td>
<td>h</td>
<td>U+04BB</td>
<td>h</td>
<td>blocked</td>
<td></td>
<td>Identity</td>
</tr>
<tr>
<td>6</td>
<td>U+04BB</td>
<td>h</td>
<td>U+0067</td>
<td>h</td>
<td>blocked</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>U+04BB</td>
<td>h</td>
<td>U+0067</td>
<td>h</td>
<td>blocked</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>U+04BB</td>
<td>h</td>
<td>U+04BB</td>
<td>h</td>
<td>blocked</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## HTML Export – Classes

### Classes, Rules and Actions

#### Character Classes

The following table lists all top-level classes with their definition and the regular expression defining their members.

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition</th>
<th>Count</th>
<th>Members</th>
<th>References</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Latin</td>
<td></td>
<td>181</td>
<td>U+0067 U+0068 U+006E U+006F U+0071 U+0075 U+0269</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Greek</td>
<td></td>
<td>77</td>
<td>U+03B7 U+03B9 U+03BF</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cyrillic</td>
<td></td>
<td>80</td>
<td>U+043E U+0448 U+04BB</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Armenian</td>
<td></td>
<td>53</td>
<td>U+0561-U+0586</td>
<td></td>
<td></td>
</tr>
<tr>
<td>sc:Armn</td>
<td></td>
<td>53</td>
<td>U+0561-U+0586</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Arabic</td>
<td></td>
<td>1214</td>
<td>U+0620-U+063A U+0641-U+064A U+0672 U+0679-U+0681 U+0683-U+068F U+0691 U+0693 U+0695-U+0699 U+06A0 U+06A2 U+06B3-U+06BD U+06FC-U+06F5 U+0751 U+0752 U+0756-U+0760 U+0762 U+0763 U+0766-U+0768 U+076A U+076E-U+0771 U+08A0 U+08A2 U+08A9</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>U+0620-U+063A U+0641-U+064A U+0672 U+0679-U+0681 U+0683-U+068F U+0691 U+0693 U+0695-U+0699 U+06A0 U+06A2 U+06A4 U+06A6-U+06AB U+06AD-U+06B1 U+06B3</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### Class combination

Legend

- **Members or Ranges**: Lists the members of the class as code points (xxx) or as ranges of code points (xxx-yyy). Any class too numerous to list in full is elided with "...".
- **Tag=ttt**: An anonymous class implicitly defined based on tag value.
- ```nim`:: named character set`: Reference to a named character set `name`.
- ```nim`(n, u, \, \_\_\_\_\_\_\_\_\)_\_ set operators`: Sets may be combined by set operators (n = intersection, u = union, \ = difference, \_\_\_\_\_\_\_\_\_ = symmetric difference).
Table: Whole label evaluation and context rules

<table>
<thead>
<tr>
<th>Name</th>
<th>Regular Expression</th>
<th>Used as Trigger</th>
<th>Used as Context</th>
<th>Anchor</th>
<th>References</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Common-leading-combining-mark</td>
<td>(start) [:class property:gc=Mn,U[:class property:gc=Mc:]]</td>
<td>True</td>
<td>False</td>
<td>False</td>
<td></td>
<td>default WLE rule matching labels with leading combining marks ✔</td>
</tr>
<tr>
<td>und-Arab-no-mix-kaf-keheh</td>
<td>(U+0643 (any){0,} U+06A9</td>
<td>U+06A9 (any){0,} U+0643)</td>
<td>True</td>
<td>False</td>
<td>False</td>
<td>[100] WLE Rule 1: do not mix Arabic letters KAF and KEHEH in the same label</td>
</tr>
<tr>
<td>und-Arab-no-mix-kaf-swash</td>
<td>(U+0643 (any){0,} U+06AA</td>
<td>U+06AA (any){0,} U+0643)</td>
<td>True</td>
<td>False</td>
<td>False</td>
<td>[100] WLE Rule 2: do not mix Arabic letters KAF and SWASH KAF in the same label</td>
</tr>
<tr>
<td>und-Arab-no-mix-alef-maksura-farsi-yeh</td>
<td>(U+0649 (any){0,} U+06CC</td>
<td>U+06CC (any){0,} U+0649)</td>
<td>True</td>
<td>False</td>
<td>False</td>
<td>[100] WLE Rule 3: do not mix Arabic letters ALEF MAKSURA and FARSI YEH in the same label</td>
</tr>
<tr>
<td>und-Arab-no-mix-heh-goal</td>
<td>(U+0647 (any){0,} U+06C1</td>
<td>U+06C1 (any){0,} U+0647)</td>
<td>True</td>
<td>False</td>
<td>False</td>
<td>[100] WLE Rule 4: do not mix Arabic letters HEH and HEH GOAL in the same label</td>
</tr>
<tr>
<td>und-Arab-no-mix-heh-goal-ae</td>
<td>(U+06C1 (any){0,} U+06D5</td>
<td>U+06D5 (any){0,} U+06C1)</td>
<td>True</td>
<td>False</td>
<td>False</td>
<td>[100] WLE Rule 5: do not mix Arabic letters HEH GOAL and AE in the same label</td>
</tr>
<tr>
<td>und-Arab-no-mix-heh-ae</td>
<td>(U+0647 (any){0,} U+06D5</td>
<td>U+06D5 (any){0,} U+0647)</td>
<td>True</td>
<td>False</td>
<td>False</td>
<td>[100] WLE Rule 6: do not mix Arabic letters HEH and AE in the same label</td>
</tr>
<tr>
<td>und-Arab-no-mix-heh-doachashmee</td>
<td>(U+0647 (any){0,} U+06BE</td>
<td>U+06BE (any){0,} U+0647)</td>
<td>True</td>
<td>False</td>
<td>False</td>
<td>[100] WLE Rule 7: do not mix Arabic letters HEH and HEH DOACHASHMEE in the same label</td>
</tr>
<tr>
<td>und-Arab-no-mix-teh-marbuta-goal</td>
<td>(U+0629 (any){0,} U+06C3</td>
<td>U+06C3 (any){0,} U+0629)</td>
<td>True</td>
<td>False</td>
<td>False</td>
<td>[100] WLE Rule 8: do not mix Arabic letters TEH MARBUTA and FEH WITH DOT MOVED BELOW in the same label</td>
</tr>
<tr>
<td>und-Arab-no-mix-noon-with-three-dots-above-yeh-with-three-dots-below</td>
<td>(U+06BD (any){0,} U+06D1</td>
<td>U+06D1 (any){0,} U+06BD)</td>
<td>True</td>
<td>False</td>
<td>False</td>
<td>[100] WLE Rule 9: do not mix Arabic letters NOON WITH THREE DOTS ABOVE and YEH WITH THREE DOTS BELOW in the same label</td>
</tr>
</tbody>
</table>
Interface Improvements

Welcome to the LGR (Label Generation Ruleset) Tool

This application provides a convenient interface for browsing and editing LGRs conforming to the Representing Label Generation Rulesets using XML specification.

Previously loaded LGR file(s)
Previously, you edited the following LGR file(s). Click on its title to resume your editing session.

- [View lgr-2-arabic-script-2017-02-23-en]

LGR sets
- [View rootzone]
- [View embedded LGRs]

Start a new LGR file or import an existing one

- [Import an existing XML file]
- [Note that importing large LGR files may take significant time to load on your browser.]
- [Start with a New blank XML file]

Start from a built-in LGR
The following LGRs are pre-installed in the system. You may use them as a starting point for your own LGR. To do so, just click on it to make a copy that you can then edit.

- [Open Sample-French]

Remember to save your work regularly by downloading a copy of the XML file.

Please send any feedback to support@viagenio.ca.

List of loaded LGR and LGR Sets

Your saved results
The following files contain your tools computation results.

- Note that these files could be cleaned up regularly.
  - [Download 20170104_202015_annotation_sample-french.txt.gz]
  - [Download 20170104_202010_annotation_sample-french.txt.gz]

Regroup create/import buttons

Tool outputs (hidden if none)
Thank You and Questions

- RFC 7940 Representing Label Generation Rulesets Using XML
  - [https://www.rfc-editor.org/info/rfc7940](https://www.rfc-editor.org/info/rfc7940)

- LGR Toolset
  - [https://lgrtool.icann.org](https://lgrtool.icann.org)

- LGR Toolset User Manual
Chinese GP Update

Kenny HUANG
Wei WANG
Agenda

1. CGP Work Review
2. CGP Proposal Draft 201707
3. CJK Coordination
4. Repertoire Size
5. Allocatable Label Number
6. Next Step & Suggestions
CGP Work Review

Next Steps

- CJK Meeting@Beijing
  - CJ Meeting@Dallas
  - CJK Meeting@Seoul
- CK Meeting@Taipei
- CK Meeting@Seoul

**Timeline**

- Sep 2014: CGP Formed
- Mar 2015: CGP Repertoire & Variant Extension Review
- May 2015: CGP Proposal Draft v1 IP Feedback v1
- Oct 2015: CJK Meeting@Seoul
- Mar 2016: CGP Proposal Draft v2 IP Feedback v2
- Jun 2016: CJK Meeting@Dallas
- Aug 2016: CGP Proposal Draft v3
- Sep 2016: CGP Proposal Draft v4
- Nov 2016: CJK Meeting@Beijing
- Jan 2017
- Jul 2017
Script and Languages Covered

<table>
<thead>
<tr>
<th>Language</th>
<th>ISO 15924 Code</th>
<th>Countries</th>
<th>Local Names of the Script</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chinese</td>
<td>cdo, cji, cmn, cpx, czh, czo, gan, hak, hsn, lzh, mnp, nan, wuu, yue, zho</td>
<td>China</td>
<td>汉字 Hanzi</td>
</tr>
<tr>
<td>Japanese</td>
<td>jpn</td>
<td>Japan</td>
<td>漢字 Kanji</td>
</tr>
<tr>
<td>Korean</td>
<td>kor</td>
<td>Korea</td>
<td>한자 Hanja</td>
</tr>
</tbody>
</table>

To Korea 2nd century BC to 5th century AD
To Japan 5th century AD

Hanzi (Hans & Hant)

Kanji
+ Hiragana
+ Katakana

Hanja
+ Hangul

Kore

Jpan
Team and Work Process

Members, 23 experts from 10 countries/regions
China mainland, Taiwan, Hong Kong, Macau, Singapore, Malaysia, as well as members from Europe and North America.

Advisor, Edmon CHUNG
CEO of dotAsia and Co-Chair of the Universal Acceptance Steering Group

CJK coordination working group
- Code Point Repertoire
- 19,746 Characters/Code Points

Note: two chars not included in current MSR
Code Point Variants

Definition
- "characters with different visual forms but with the same pronunciations and with the same meanings as the corresponding official forms in the given language contexts."

Simplified and Traditional
- Every code point in the CGP repertoire has its preferred simplified variant(s), preferred traditional variant(s), and reserved variant(s)
- A code point might have a reflexive preferred S/T variant
- A code point might have no reserved variant

<table>
<thead>
<tr>
<th>Sub-Type</th>
<th>Type</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;simp&quot;</td>
<td>Allocatable</td>
<td>preferred simplified variant char;</td>
</tr>
<tr>
<td>&quot;r-simp&quot;</td>
<td>Allocatable</td>
<td>reflexive preferred simplified variant char;</td>
</tr>
<tr>
<td>&quot;trad&quot;</td>
<td>Allocatable</td>
<td>preferred traditional variant char</td>
</tr>
<tr>
<td>&quot;r-trad&quot;</td>
<td>Allocatable</td>
<td>reflexive preferred traditional variant char</td>
</tr>
<tr>
<td>&quot;both&quot;</td>
<td>Allocatable</td>
<td>preferred simplified and traditional variant chars are the same</td>
</tr>
<tr>
<td>&quot;r-both&quot;</td>
<td>Allocatable</td>
<td>reflexive preferred simp and trad variant chars are the same</td>
</tr>
<tr>
<td>&quot;r-neither&quot;</td>
<td>Blocked</td>
<td>Non-allocatable reflexive/original char</td>
</tr>
<tr>
<td>&quot;blocked&quot;</td>
<td>Blocked</td>
<td>Non-allocatable variant char</td>
</tr>
</tbody>
</table>
Whole Label Generation Rules

```xml
<rules>
<!--Action elements - order defines precedence--> 
[action disp="invalid" match="leading-combining-mark" comment="labels with leading combining marks are invalid"/>
[action disp="blocked" any-variant="blocked" comment="default action for blocked variant"/>

[action disp="allocatable" only-variants="simp r-simp both r-both" comment="simplified label"/>
[action disp="allocatable" only-variants="trad r-trad both r-both" comment="traditional label"/>
[action disp="allocatable" only-variants="r-simp r-trad r-both r-neither" comment="original label"/>

[action disp="blocked" any-variant="simp trad both r-simp r-trad r-both r-neither" comment="block any other mixed labels"/>

[action disp="allocatable" comment="catch-all"/>
</rules>
```
CJK Coordination

- Coordination within CGP
  - CDNC variants
  - TGSCC and IICORE variants review (172)
  - dotAsia variant coordination (69)

- Coordination between C, J and K
  - 445 variant mappings (146 variant groups)
The Issue of Repertoire Size

Dictionary

- 16th-11th century BC 商甲骨文 3,500 ~ 4,500
- 202 BC-220 AD 汉朝 说文解字 9,353
- 960-1279 AD 宋朝 类篇 31,319
- 1710 清朝 康熙字典 47,035
- 1959 Japanese 大汉和辞典 49,964
- 1994 China Mainland 中华字海 87,019
- 2004 Taiwan 異體字字典 106,230
The Issue of Repertoire Size

Paper
• 2007, A Survey on the Usage of Chinese Characters and Phrases in the Newspapers, Radio, TV, and Web, 8128
• 2010, survey on Chinese Weblog Wording, 20923

Standard
• BIG5 13,053
• CNS11643 76,067
• GBK 21,886
• GB18030 70,244
• Unicode 80,388

Legislation & Regulation
• China’s Ministry of Civil Affairs issued Notification 2016[33], requiring government departments to update the naming related information system in public service and administration areas, to cover the characters in national standard GB13000 (20902 chars) or GB18030 (70244 chars)
In the CGP variant mapping table (Appendix I), for all 19746 characters, there are 3 characters with 2 PSVs, 127 with 2 PTVs, 5 with 3 PTVs and 1 with 4 PTVs, which means, all together 136 characters have multiple preferred variant characters. These 136 characters will generate multiple all-simplified labels or all-traditional labels, which will lead to an over-production of allocatable labels issue at the root zone level.

<table>
<thead>
<tr>
<th>Original Char</th>
<th>Allocatable Simplified Variant</th>
<th>Allocatable Traditional Variant</th>
<th>Blocked Variant</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>B</td>
<td>C</td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>E</td>
<td>F</td>
<td>G</td>
</tr>
<tr>
<td>H</td>
<td>I, J</td>
<td>K</td>
<td></td>
</tr>
<tr>
<td>L</td>
<td>M, N</td>
<td>O</td>
<td>P</td>
</tr>
</tbody>
</table>

Input: AD  Allocatable: BE; CF  Blocked: BF, CE
Input: AH  Allocatable: BI, CJ; CK  Blocked: BK, CI, CJ
Input: HL  Allocatable: IM, IN, JM, JN; KO  Blocked: … …
CGP tend to keep the **multiple variant mappings** under the current LGR framework. Instead, CGP would propose to introduce new **types and rules** to **reduce the number of allocatable labels**.

### Original Char | Allocatable Simplified Variant | Allocatable Traditional Variant | Blocked Variant
---|---|---|---
A | B | C |
D | E | F | G
H | I, J | K |
L | M, N | O | P
## Next Step and Suggestions

Variants imported from J & K: 43 characters

<table>
<thead>
<tr>
<th>Unicode</th>
<th>Char</th>
<th>All Variant Chars</th>
</tr>
</thead>
<tbody>
<tr>
<td>967A</td>
<td>険</td>
<td>J 碇(7877)礦(78B1)礭(7906)险(9669)陥(967A)陥(96AA)鹼(9E7C)</td>
</tr>
<tr>
<td>7E4A</td>
<td>繍</td>
<td>J 她(5B45)繍(7E34)繕(7E4A)繡(7E8E)繽(7E96)紡(7EA4)</td>
</tr>
<tr>
<td>9421</td>
<td>鉄</td>
<td>J 鉄(9244)銅(9295)鉄(9421)鉄(9435)鉄(94C1)</td>
</tr>
<tr>
<td>9D8F</td>
<td>雞</td>
<td>J 雞(96DE)雞(9CEE)鷄(9D8F)鴨(9DC4)鵝(9E21)</td>
</tr>
<tr>
<td>4FAD</td>
<td>侭</td>
<td>J 侭(4FAD)侭(5118)侭(5C3D)侭(76E1)</td>
</tr>
<tr>
<td>6442</td>
<td>掸</td>
<td>J 掼(6315)搎(6442)撉(6444)撿(651D)</td>
</tr>
<tr>
<td>685F</td>
<td>桧</td>
<td>J 桧(6808)枱(685F)枱(68E7)轄(8F4F)</td>
</tr>
<tr>
<td>7E4B</td>
<td>繫</td>
<td>J 繫(7E4B)繫(7E6B)</td>
</tr>
<tr>
<td>81D3</td>
<td>臓</td>
<td>J 臓(810F)臏(81D3)臔(81DF)臤(9AD2)</td>
</tr>
<tr>
<td>8217</td>
<td>舸</td>
<td>J 舸(8216)舃(8217)舮(92EA)舯(94FA)</td>
</tr>
<tr>
<td>9039</td>
<td>達</td>
<td>J 達(8FBE)達(8FD6)達(9039)達(9054)</td>
</tr>
<tr>
<td>9271</td>
<td>銿</td>
<td>J 銿(77FF)礦(7926)銘(9271)銘(945B)</td>
</tr>
<tr>
<td>3960</td>
<td>慎</td>
<td>K 慎(3960)謹(8ADD)謹(8C1E)</td>
</tr>
<tr>
<td>784F</td>
<td>研</td>
<td>K 研(63C5)研(7814)研(784F)</td>
</tr>
<tr>
<td>663B</td>
<td>昴</td>
<td>K 昴(6602)昴(663B)</td>
</tr>
<tr>
<td>7A36</td>
<td>毎</td>
<td>K 毎(7A22)毎(7A36)</td>
</tr>
</tbody>
</table>
To ICANN

PHASE 1 (2011)
Case Studies:
- Arabic
- Chinese
- Cyrillic
- Devanagari
- Greek
- Latin

Integrated Issues Report

PHASE 2 (2011-12)

Projects:
- P1 LGR XML Specification
- P2.1 LGR Process for the Root Zone
- P6 User Experience Study for TLD Variants

PHASE 3 (2012-13)

Projects:
- P2.2 LGR Development
- P1 LGR Specification and Toolset
- P7 LGR Implementation

PHASE 4 (Since 2013)

There should be more interaction between P2.2, P1 and P7. The generation rule, algorithm and system design will never be able to replace rational thought and case analysis from Applicant Evaluation Panel. P2.2 might not be able to fix all issues related to IDN variant. P7 are wanted to join the discussion together with GPs and IP.
Thank You All

- Integration Panel
- Edmon Chung
- Sarmad Hussain
- ICANN Singapore

Q&A
Japanese GP Update

Hiro Hotta
JapanseGP (JGP) update

1 November, 2017
Hiro Hotta <hotta@jprs.co.jp>

Development from March 2017 is written in red
What JGP (Japanese Generation Panel) does

• Mandate
  – Proposing LGR for TLDs of Japanese language/scripts
  – that can co-exist with LGRs for other languages/scripts

• Steps
  Step1 : Populate JGP with diverse experts
  Step2 : Define the requirements and basic framework of Japanese LGR based on the expertise and experience of Japanese IDNs
  Step3 : Coordinate with other language Generation Panels especially of those languages interrelated with Japanese
  Step4 : Finalize LGR following necessary consultation with IP and Japanese community
What JGP should care about

• Consultation with the local community
  – Post JGP discussions/activities on JGP web-site
  – Presentation and discussion about LGR
    • with Japan Trademark Association
    • IGCJ (Internet Governance Conference Japan) events
  – Public comment forum

• Coordination among CJK communities

• Coordination with IP

• Coordination with global communities

• Alignment with the rules for second-level domain labels
  – Under .jp (Japanese TLD)
  – Under gTLDs (such as .asia and .com with second-level registration of IDNs)
Step1 : Manning JGP

- Members and their expertise
  - Hiro Hotta chair
    - Policy/business aspects of registry/registrar
  - Akinori Maemura vice chair
    - Internet governance and domain name in general
  - Shigeki Goto
    - Internet in general
  - Kazunori Konishi
    - Internet in general
  - Tsugizo Kubo
    - Trademarks and domain names
  - Yoshitaka Murakami
    - Trademarks and gTLD markets from registry/registrar perspective
  - Shuichi Tashiro
    - Character codes
  - Yoshiro Yoneya
    - Technical aspects of IDN, LGR
Step2 : First Version of Japanese LGR

• Scopes of the character codes
  – Kanji, Hiragana, Katakana
  – For Kanji
    • JIS (Japanese Industrial Standard) level-1 and level-2

• Variants
  – For Kanji
    • Japanese LGR will define no variants for itself
    • Final Japanese LGR will import (= passively adopt) variants of
      Chinese LGR and Korean LGR

• WLE (whole label evaluation)
  – Japanese LGR may have no or very limited number of tiny rules
    for the usage of characters even if defined
Step 3: CJK Coordination

Japanese LGR
- Hira (ひ)
- Katakana (ア)

Chinese LGR
- Han (漢)

Korean LGR
- Hangul (한)

Coordinated definition of variants has been completed

* "Han" is called "Kanji" in Japan, "Hanja" in Korea
Step 4: Consultation with IP and Japanese community

- Reduction of the number of allocatable labels
  - Any combination of characters is allowed in Japanese strings
  - It may make the number of variant strings very huge, as many variant groups are defined by importing Chinese variants
  - E.G., 慶應大学 has 3 variant strings – 慶應大学/慶応大學/慶應大學
  - JGP needs to reduce the number of allocatable labels
  - JGP is trying to solve it by limiting allowed strings by employing the notion that “allocatable labels basically consists of day-use Japanese characters”
  - However - it seems Japanese community is not comfortable with this solution because most gTLDs in Japanese scripts may not be general nouns but trade names or geo names that often encompass personal names or geo names, which sometimes contain characters that are not “day-use Japanese characters”
  - JGP is still seeking the way to reduce allocatable labels
Korean GP Status Update

KIM Kyongsook
Korean GP Chair
0. Agenda

- Introduction
- K-LGR v0.7 (2017.03.03.)
  - 11172 Hangul Syllables, 4758 Hanja characters, 152 Variant groups
- K-LGR proposal and XML sent to IP
- History of KGP activities
- Timeline of KGP activities
1. Introduction

- Characters included for “Kore” (Korean Label)
  - Both Hangeul (Hangul) syllables and Hanja chars are included in K-LGR

- K-LGR v0.7 (3 March 2017)
  - 11172 Hangeul syllables
  - 4758 Hanja chars, 152 variant groups

- In January 2017, the Name and Address Committee of Korea Internet Governance Alliance (KIGA, http://kiga.or.kr) formally created a WG for allowing Hanja at the second level under .KR/.한국
  - The WG started working
2. K-LGR v0.7 (3 March 2017)

- A list of Hangul Syllables for K-LGR v0.7 (3 March 2017)
  - 11172 Hangul Syllables (U+AC00 ~ U+D7A3) ← ISO/IEC 10646

- A list of Hanja characters for K-LGR v0.7 (3 March 2017)

<table>
<thead>
<tr>
<th>Source of Hanja Character Set</th>
<th># chars</th>
</tr>
</thead>
<tbody>
<tr>
<td>1) KS X 1001 (268 comptb. chars excluded)</td>
<td>4620</td>
</tr>
<tr>
<td>2) IICORE - K column marked</td>
<td>4743</td>
</tr>
<tr>
<td>K-LGR v0.7 (3 March 2017): Hanja List (Union of 1) and 2))</td>
<td>4758</td>
</tr>
</tbody>
</table>

- K-LGR v0.7 (3 March 2017): 4758 Hanja chars, 152 variant groups
  - 61 chars in KP0 (= KPS 9566) and/or Hanja Test deleted
- No conflict in variant groups between K-LGR v0.7 (3 March 2017) and C-LGR (31 March 2017)
3. K-LGR Proposal and XML Sent To IP

- 11 May 2015, repertoire and variant groups in K-LGR v0.1 sent to IP
  - 21 June 2015, IP comments received by KGP
- 20 August 2016, XML for K-LGR v0.4 sent to IP
  - 14 September 2016, IP comments received by KGP
- 21 December 2016, Proposal and XML for K-LGR v0.6 sent to IP
  - 2 February 2017, IP comments received by KGP
- 24 May 2017, Proposal and XML for K-LGR v0.7 sent to IP
  - 24 July 2017, IP comments received by KGP
- ? December 2017, Rev. Proposal and XML for K-LGR v0.7 will be sent to IP
### 4. History of KGP Activities (1)

<table>
<thead>
<tr>
<th>Year</th>
<th>Month</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>2013</td>
<td>Dec</td>
<td>Korean GP organized</td>
</tr>
<tr>
<td>2014</td>
<td>Mar</td>
<td>CJK coordination meeting during ICANN 49 @ Singapore</td>
</tr>
<tr>
<td></td>
<td>Jun</td>
<td>CJK coordination meeting during ICANN 50 @ London; KGP status update</td>
</tr>
<tr>
<td></td>
<td>Jun</td>
<td>1st KGP meeting</td>
</tr>
<tr>
<td></td>
<td>Aug</td>
<td>2nd KGP meeting</td>
</tr>
<tr>
<td></td>
<td>Oct</td>
<td>CJK coordination meeting during ICANN 51 @ LAX; KGP status update</td>
</tr>
<tr>
<td>2015</td>
<td>Jan</td>
<td>3rd KGP meeting; KGP reorganized</td>
</tr>
<tr>
<td></td>
<td>Feb</td>
<td>CJK coordination meeting during ICANN52 @ Singapore; KGP status update</td>
</tr>
<tr>
<td></td>
<td>Apr</td>
<td>4th and 5th KGP meetings; KGP reorganized</td>
</tr>
<tr>
<td></td>
<td>May</td>
<td>6th and 7th KGP meetings (K-LGR, v0.1); CJK coordination meeting in Seoul, Korea</td>
</tr>
<tr>
<td></td>
<td>Jun</td>
<td>8th KGP meeting (K-LGR, v0.2 CJK coordination meeting, ICANN53@Buenos Aires</td>
</tr>
<tr>
<td></td>
<td>Jul</td>
<td>9th KGP meeting and workshop; participated in APrIGF @ Macau</td>
</tr>
<tr>
<td></td>
<td>Aug</td>
<td>10th KGP meeting (K-LGR, v0.3)</td>
</tr>
<tr>
<td></td>
<td>Sep</td>
<td>11th KGP meeting</td>
</tr>
<tr>
<td></td>
<td>Oct</td>
<td>Call for formal Generation of KGP (Korean Script Generation Panel) to ICANN; CJK coordination meeting during ICANN54 @ Dublin</td>
</tr>
</tbody>
</table>
4. History of KGP Activities (2)

2015
Nov: 12th KGP meeting

2016
Jan: 13th KGP meeting
Feb: The Korean Community “formally” Forms Generation Panel for Developing the Root Zone Label Generation Rules (LGR), 2016-02-01
Mar: CJK coordination meeting during ICANN55 @ Marrakesh; KGP status update
Mar: 14th KGP meeting (K-LGR, v0.4)
Mar: CJK coordination meeting @ Beijing, China
Apr: 15th KGP meeting
May: 16th KGP meeting
Jun: 17th KGP meeting
Jun: CJK coordination meeting during ICANN56 @ Helsinki; KGP status update
Jul: 18th KGP meeting
Aug: 19th KGP meeting
Sep: 20th KGP meeting; (K-LGR, v0.5); Public Hearing for K-LGR @KrIGF Workshop)
4. History of KGP Activities (3)

2016

- Sep: CJK coordination meeting @ Taipei, Taiwan
- Oct: 21\textsuperscript{st} and 22\textsuperscript{nd} KGP meetings
- Nov: CJK coordination meeting during ICANN 57 @ Hyderabad, India
- Nov: CJK Coordination Meeting @ Seoul during IETF #97 (K-LGR, v0.6)
- Dec: 23\textsuperscript{rd} KGP meeting

2017

- Feb: CK coordination meeting during CDNC meeting @ Beijing, China
- Mar: 24\textsuperscript{th} KGP meeting
  - K-LGR, v0.7 (3 March 2017): # Hanja chars 4819 -> 4758 (61 chars deleted)
- Mar: CJK coordination meeting during ICANN58 @ Copenhagen; KGP status update
- Apr: 25\textsuperscript{th} KGP meeting
- Sep: 26\textsuperscript{th} KGP meeting
- Sep: Public Hearing for K-LGR @KIGF Workshop
- Oct: KGP status update, ICANN60 @ Abu Dhabi, UAE
5. Timeline of KGP Activities

- **Dec. 2013**: Organization of KGP
- **May. 2015**: K-LGR v0.1
- **Jun. 2015**: K-LGR v0.2
- **Oct. 2015**: K-LGR v0.3
- **Feb. 2016**: KGP formally formed
- **Mar. 2016**: K-LGR v0.4
- **Sep. 2016**: K-LGR v0.5
- **Nov. 2016**: K-LGR v0.6
- **Mar. 2017**: K-LGR v0.7
Appendix. Hanja in K-LGR v0.7 (2017.03.03.):
K0 and IICORE/K (IK)

Venn Diagram of 2 sets showing number of Hanja chars:
K-LGR v0.7 (2017.03.03.): 4758 chars
K0 (KS X 1001): 4620 chars, IK (IICORE: K): 4743 chars
Greek GP Status Update

Panagiotis Papaspiliopoulos
Greek GP Chair
Script for Which The LGR Will Be Proposed

- Greek
- ISO 15924 Code: Grek
- ISO 15924 Key No: 200
- ISO 15924 English Name: Greek
- Property Value Alias: Greek
- Native Name of the script: Ελληνικά
- Maximal Starting Repertoire (MSR) version: MSR-2
- Unicode Standard 9.0: 0370-03FF
Composition of The Greek Generation Panel

- Individuals actively involved in policy development processes related to electronic communication, research and development related to the Greek language, standardization, computing and maintenance of the domain name system in Greece and Cyprus.

- The panelists come from governmental and regulatory authorities, academia, private sector, ccTLD registries of Greece and Cyprus.
The Greek Generation Panel was officially formed with the Decision 54020/1088/25 Nov 2015 of the Minister of Infrastructure, Transport and Networks of the Hellenic Republic.
Structure of The Proposal

- Background on script
  - Other languages using the Greek script (e.g. Pomak, Karamanlidika, Arvanitika etc)
  - The Greek language question
  - The Greek orthography

- Overall development process and methodology

- Repertoire

- Variants
  - Within Script Variants
  - Cross Script Variants

- Whole Label Evaluation rules
Useful Key Points

- The Pomak language
  - Oral, Rup dialects spoken in Thrace (~30,000 people)

- The Greek language question
  - Diglossia: Katharevousa vs. Dimotiki (e.g. Πειραιεύς - Πειραιάς)

- The Greek orthography
  - Within Script Variants
  - Cross Script Variants

- Whole Label Evaluation rules
  - Polytonic vs. Monotonic (e.g. the Lord’s Prayer)
    - Πάτερ ἡμῶν ὁ ἐν τοῖς οὐρανοῖς, ἁγιασθήτω τὸ ὄνομά σου, ἐλθέτω ἡ βασιλεία σου, γεννηθήτω τὸ θέλημά σου, ὡς ἐν οὐρανῷ καὶ ἐπὶ τῆς γῆς
    • Πάτερ ημῶν ο εν τοις ουρανοις, αγιασθητω το όνομα σου, ελθετω η βασιλεια σου, γεννηθητω το θελημα σου, ως εν ουρανω και επι της γης
Proposed Characters for Registrations

- Only Monotonic characters are to be allowed!
  - Monotonic characters are used today in the spelling of Greek words
  - Polytonic characters for TLD registration offer no significant advantage for the average user.
  - This recommendation stands only for the Greek TLDs; registries are encouraged to decide upon the use of the polytonic characters in their policy for the lower levels

- The Pomak language does not affect the formation of domain names with the Greek characters’ set
Within Script Variants: The Sigma and Final Sigma

- The sigma and final sigma
  - $\sigma$ (small, U+03C3) e.g. αστίδα (shield)
  - $\Sigma$ (capital, U+03A3) e.g. Σοφία (female name)
  - $\varsigma$ (small final, U+03C2) e.g. τέλος (end)

But..

- IDNA2003: $\sigma$ <-> $\Sigma$ <-> $\varsigma$ (e.g. θησαυρός – treasure)
  - θησαυρός → ΘΗΣΑΥΡΟΣ → θησαυροσ → not correct Greek!

- IDNA2008: $\sigma$ and $\varsigma$ are different accepted characters, treated separately – reverse mapping no possible!

- Conclusion: handle ‘$\sigma$’ and ‘$\varsigma$’ as within script variants!
### Within Script Variants: Vowels

<table>
<thead>
<tr>
<th>Non-accented vowel</th>
<th>Vowel with tonos</th>
<th>Vowel with diaeresis</th>
<th>Vowel with tonos and diaeresis</th>
</tr>
</thead>
<tbody>
<tr>
<td>α</td>
<td>á</td>
<td></td>
<td></td>
</tr>
<tr>
<td>U+03B1</td>
<td>U+03AC</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ε</td>
<td>é</td>
<td></td>
<td></td>
</tr>
<tr>
<td>U+03B5</td>
<td>U+03AD</td>
<td></td>
<td></td>
</tr>
<tr>
<td>η</td>
<td>η</td>
<td></td>
<td></td>
</tr>
<tr>
<td>U+03B7</td>
<td>U+03AE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ι</td>
<td>í</td>
<td></td>
<td></td>
</tr>
<tr>
<td>U+03B9</td>
<td>U+03AF</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ο</td>
<td>ó</td>
<td></td>
<td></td>
</tr>
<tr>
<td>U+03BF</td>
<td>U+03CC</td>
<td></td>
<td></td>
</tr>
<tr>
<td>υ</td>
<td>ύ</td>
<td></td>
<td></td>
</tr>
<tr>
<td>U+03C5</td>
<td>U+03CD</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ω</td>
<td>ώ</td>
<td></td>
<td></td>
</tr>
<tr>
<td>U+03C9</td>
<td>U+03CE</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Issues to be Concerned: Cross Script Variants

- Status: under process

- Cases under examination
  - Greek and Latin
  - Greek and Cyrillic
  - Greek and Armenian
  - Greek and other scripts

- Methodology:
  - Creation of tables per script (extensively)
  - Consideration of various fonts and sizes
  - Question: homoglyphs only or consider visual similarity cases too? To what extend?
  - Key factor: consideration of other GPs conclusions
# Cross Script Variants (Examples): Greek - Latin

<table>
<thead>
<tr>
<th>Greek Letter</th>
<th>Latin Letter</th>
</tr>
</thead>
<tbody>
<tr>
<td>U+03B1 α</td>
<td>α</td>
</tr>
<tr>
<td>U+03B2 β</td>
<td>β</td>
</tr>
<tr>
<td>U+03B7 η</td>
<td>h</td>
</tr>
<tr>
<td>U+03AE η</td>
<td>n</td>
</tr>
</tbody>
</table>

**Greek small letter ALPHA**

- Greek small letter ALPHA
- Latin small letter ALPHA

**Greek small letter BETA**

- Latin small letter SHARP S

**Greek small letter ETA**

- Latin small letter H
- Latin small letter N
- Latin small letter N WITH CEDILLA
- Latin small letter ENG
- Latin small letter H WITH HOOK
- Latin small letter N WITH DOT BELOW

**Greek small letter ETA WITH TONOS**

- Latin small letter N WITH ACUTE
- Latin small letter N WITH GRAVE
- Latin small letter N WITH DOT ABOVE
# Cross Script Variants (Examples): Greek - Cyrillic

<table>
<thead>
<tr>
<th>Greek Letter</th>
<th>Cyrillic Letter</th>
</tr>
</thead>
<tbody>
<tr>
<td>U+03B2 β</td>
<td>в U+0432</td>
</tr>
<tr>
<td>Greek small letter BETA</td>
<td>Cyrillic small letter VE</td>
</tr>
<tr>
<td>U+03B7 η</td>
<td>и U+0438</td>
</tr>
<tr>
<td>Greek small letter ETA</td>
<td>Cyrillic small letter I</td>
</tr>
<tr>
<td>U+03B8 θ</td>
<td>е U+04E9</td>
</tr>
<tr>
<td>Greek small letter THETA</td>
<td>Cyrillic small letter BARRED O</td>
</tr>
<tr>
<td>U+03BA κ</td>
<td>κ U+043A</td>
</tr>
<tr>
<td>Greek small letter KAPPA</td>
<td>Cyrillic small letter KA</td>
</tr>
<tr>
<td>U+03C4 τ</td>
<td>τ U+0442</td>
</tr>
<tr>
<td>Greek small letter TAU</td>
<td>Cyrillic small letter TE</td>
</tr>
<tr>
<td></td>
<td>τ U+04AD</td>
</tr>
<tr>
<td></td>
<td>Cyrillic small letter TE WITH DESCENDER</td>
</tr>
</tbody>
</table>
## Cross Script Variants (Example): Greek - Armenian

<table>
<thead>
<tr>
<th>Greek Letter</th>
<th>Armenian Letter</th>
</tr>
</thead>
<tbody>
<tr>
<td>U+03B7 η</td>
<td>η U+0572</td>
</tr>
<tr>
<td>Greek small letter ETA</td>
<td>Armenian small letter GHAD</td>
</tr>
<tr>
<td></td>
<td>η U+0578</td>
</tr>
<tr>
<td></td>
<td>Armenian small letter VO</td>
</tr>
<tr>
<td></td>
<td>η U+0564</td>
</tr>
<tr>
<td></td>
<td>Armenian small letter DA</td>
</tr>
<tr>
<td></td>
<td>η U+057C</td>
</tr>
<tr>
<td></td>
<td>Armenian small letter RA</td>
</tr>
<tr>
<td>Greek Letter</td>
<td>Georgian Letter</td>
</tr>
<tr>
<td>--------------</td>
<td>----------------</td>
</tr>
<tr>
<td>Χ (U+03BE)</td>
<td>ძ (U+10F0)</td>
</tr>
<tr>
<td>Greek small letter XI</td>
<td>Georgian letter HAE</td>
</tr>
<tr>
<td></td>
<td>წ (U+10F4)</td>
</tr>
<tr>
<td></td>
<td>Georgian letter HAR</td>
</tr>
<tr>
<td>ϕ (U+03C6)</td>
<td>ღ (U+10F6)</td>
</tr>
<tr>
<td>Greek small letter PHI</td>
<td>Georgian letter FI</td>
</tr>
<tr>
<td></td>
<td>ღ (U+10F6)</td>
</tr>
<tr>
<td></td>
<td>Georgian letter FI</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Ethiopic Syllable</strong> <strong>(Font: Ebrima)</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Greek Letter</th>
<th>Ethioopic Syllable</th>
</tr>
</thead>
<tbody>
<tr>
<td>Λ (U+03BB)</td>
<td>ኲ (U+1208)</td>
</tr>
<tr>
<td>Greek small letter LAMDA</td>
<td>Ethioopic syllable LA</td>
</tr>
<tr>
<td></td>
<td>ኲ (U+120B)</td>
</tr>
<tr>
<td></td>
<td>Ethioopic syllable LAA</td>
</tr>
<tr>
<td>ζ (U+03C2)</td>
<td>዆ (U+1293)</td>
</tr>
<tr>
<td>Greek small letter FINAL SIGMA</td>
<td>Ethioopic syllable NAA</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Myanmar Letter</strong> <strong>(Font size: 16)</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Greek Letter</th>
<th>Myanmar Letter</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ω (U+03C9)</td>
<td>ၗ (U+1003)</td>
</tr>
<tr>
<td>Greek small letter OMEGA</td>
<td>Myanmar letter GHA</td>
</tr>
<tr>
<td></td>
<td>ၗ (U+101A)</td>
</tr>
<tr>
<td></td>
<td>Myanmar letter YA</td>
</tr>
</tbody>
</table>
Issues to be Concerned: WLE Rules

- Status: under process
- Issues should been taken into consideration
  - Contextual information
  - Existing rules (since 2005)
  - User experience of the average Greek speaking Internet users
**Drawbacks - Reasons for (rather) Slow Progress**

- Heavy work load of the members of the Greek Generation Panel due to other demanding professional obligations

- Difficulty in settling face-to-face meetings (distance, little available time). Use of telephone and mailing lists only! → too much time to get all responses, conclude and move forward

- Careful steps due to problems in previous projects (i.e. discussions about confusing similarity cases during the IDN ccTLD Fast Track Process for .ελ and Greek .eu)

But,

- **We feel that we are moving to the final act !**
Thank You and Questions

○ Reach us at
  ○ GreekGP@icann.org
  ○ https://community.icann.org/display/croscomlgrprocedure/Greek+Script+GP
Engage with ICANN and IDN Program

Thank You and Questions
Reach us at: IDNProgram@icann.org
Website: icann.org/idn

twitter.com/icann
gplus.to/icann
facebook.com/icannorg
weibo.com/ICANNorg
linkedin.com/company/icann
flickr.com/photos/icann
youtube.com/user/icannnews
slideshare.net/icannpresentations