

LGR Toolset (beta) User Guide



IDN Program
24 October 2017

Introduction to LGR Toolset (beta)

- ⦿ Label Generation Rulesets (LGRs) specify metadata, code point repertoire, variant rules and Whole Label Evaluation (WLE) rules to generate labels
- ⦿ [RFC 7940](#) describes how LGR can be specified using XML, a machine readable format
- ⦿ LGR can be used to generate domain name labels for use in the internet's root zone and other levels
- ⦿ LGR Toolset can be used to
 - Create an LGR
 - View LGR as an HTML webpage
 - Merge multiple LGRs into a single LGR
 - Validate single label or multiple labels using an LGR
 - Determine cross-script variants of labels using a merged LGR
 - Manage LGRs by comparing or combining them
 - Review possible impact of a new or a revised LGR on existing labels

Availability of LGR Toolset (beta)

- ⦿ LGR Toolset is available with the following disclaimer:

THIS SOFTWARE IS PROVIDED BY ICANN AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ICANN OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- ⦿ Online beta deployment
 - Visit <https://lgrtool.icann.org/>
 - If needed, username: lgr and password: 37zEfM2LyN3DmSzjLaYoA
- ⦿ Open source package(s) release with BSD license
 - Released at github: [lgr-core](#), [lgr-django](#), [munidata](#), [picu](#)
- ⦿ For queries or feedback
 - Email to IDNProgram@icann.org
- ⦿ For further details, visit the [LGR Toolset webpage](#) or www.icann.org/idn

Agenda

1

Importing an existing LGR in a toolset
(pp. 9-12)

2

Importing multiple LGRs in a toolset
(pp. 13-16)

3

Creating an LGR via toolset
(pp. 17-37)

4

Summarizing LGR
(pp. 38-40)

5

Viewing LGR as XML and HTML
(pp. 41-52)

6

Validating labels using an LGR
(pp. 53-58)

Agenda (cont.)

7

Reviewing impact
on existing labels
by revising an
existing LGR
(pp. 59-64)

8

Reviewing impact
on existing labels
by introducing a
new LGR
(pp. 65-70)

9

Validating multiple
labels using an LGR

(pp. 71-76)

10

Validating multiple
labels using a merged
LGR
(pp. 77-82)

11

Viewing cross-script
variants of labels

(pp. 83-88)

12

Comparing LGRs

(pp. 89-93)

Agenda Details

- ⦿ **Import or Load LGR**
 - Import LGR
 - Select LGR with validating repertoire
- ⦿ **Import Multiple LGRs**
 - Import LGR
 - Select multiple LGRs with validating repertoire
 - Specify name of the merged LGR
- ⦿ **Create LGR**
 - Create LGR
 - Enter LGR Details
 - Add Code Points
 - Expand Ranges
 - Add References
 - Define Meta Data
 - Define Classes, Rules & Actions
 - Define Code Point Properties
 - Download LGR
- ⦿ **Summarize LGR**

Agenda Details (cont.)

- ⦿ **View LGR as XML and HTML**
- ⦿ **Validate label**
 - Validate label using a single LGR
 - Validate label using a merged LGR
- ⦿ **Review Impact on Existing Labels by Revising an Existing LGR**
 - Select “Diff labels of two LGRs”
 - Enter Details
 - Email Notification
 - Download Results
- ⦿ **Review Impact on Existing Labels by Introducing a New LGR**
 - Select “Get collisions in a list of labels”
 - Enter Details
 - Email Notification
 - Download Results
- ⦿ **Validate Multiple Labels using an LGR**
 - Select “Generate disp. annotations”
 - Enter Details
 - Email Notification
 - Download Results

Agenda Details (cont.)

- ◉ **Validate Multiple Labels using a Merged LGR**
 - Select “Generate disp. annotations”
 - Enter Details
 - Email Notification
 - Download Results
- ◉ **View Cross-script Variants of Labels**
 - Select “Cross-script variants”
 - Enter Details
 - Email Notification
 - Download Results
- ◉ **Compare LGRs**
 - Compare LGRs
 - Select Union, Intersection or Difference
 - With merged LGRs, “Diff” chosen by default

Import or Load LGR

Agenda Item #1

Import or Load LGR



Welcome to the LGR (Label Generation Ruleset) Tool


This application provides a convenient interface for browsing and editing LGR's conforming to the [Representing Label Generation Rulesets using XML](#) specification.

To start by using an existing LGR file in XML format, click on the "Import" button

Previously loaded LGR file(s)

No LGR has been previously loaded.

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@viagenie.ca.

Select LGR with Validating Repertoire

1. To import or load an existing LGR in XML format, click on “Choose Files”

2. Choose the “Validating repertoire” from the two given options

3. Click on the “Import” button

Imported LGR

🏠 LGR Tool / proposal-lao-lgr-31jan17-en - Proposed LGR for Lao

📁 Import 📄 New ⚙️ Tools ▾ ✓ Validate label 📄 Summary 👁️ View XML 🌐 HTML Output 📄 Download

Code points References Meta data Rules

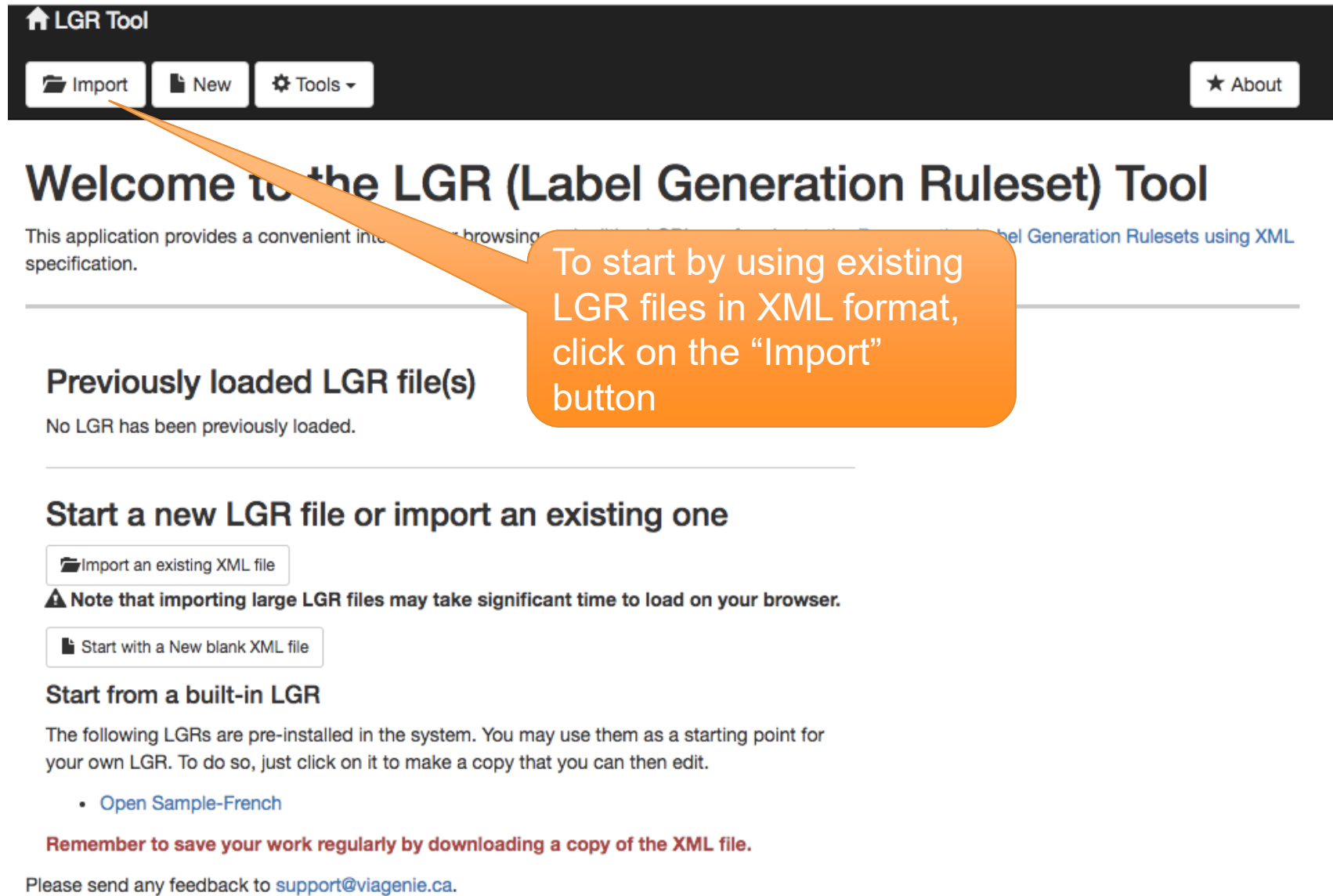
The screen looks like this after successful import of existing LGR file in XML format

	Name	Comments	Action
	LAO LETTER KHO		See code point
U+0E82 (๒) 0 Variant(s)	LAO LETTER KHO SUNG		See code point
U+0E84 (๔) 0 Variant(s)	LAO LETTER KHO TAM		See code point
U+0E87 (๗) 0 Variant(s)	LAO LETTER NGO		See code point
U+0E88 (๘) 0 Variant(s)	LAO LETTER CO		See code point
U+0E8A (๘) 0 Variant(s)	LAO LETTER SO TAM		See code point
U+0E8D (๕) 0 Variant(s)	LAO LETTER NYO		See code point
U+0E94 (๔) 0 Variant(s)	LAO LETTER DO		See code point
U+0E95 (๕) 0 Variant(s)	LAO LETTER TO		See code point
U+0E96 (๖) 0 Variant(s)	LAO LETTER THO SUNG		See code point

Import Multiple LGRs

Agenda Item #2

Import or Load LGRs



The screenshot shows the LGR Tool interface. At the top, there is a navigation bar with a home icon, the text 'LGR Tool', and three buttons: 'Import', 'New', and 'Tools'. An 'About' button with a star icon is in the top right. Below the navigation bar is a heading 'Welcome to the LGR (Label Generation Ruleset) Tool' followed by a paragraph: 'This application provides a convenient interface for browsing and managing Label Generation Rulesets using XML specification.' Below this is a section titled 'Previously loaded LGR file(s)' with the text 'No LGR has been previously loaded.' Underneath is a heading 'Start a new LGR file or import an existing one' with two buttons: 'Import an existing XML file' and 'Start with a New blank XML file'. A warning note states: 'Note that importing large LGR files may take significant time to load on your browser.' Below that is a section 'Start from a built-in LGR' with the text: '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.' A list item 'Open Sample-French' is shown. A reminder says: 'Remember to save your work regularly by downloading a copy of the XML file.' At the bottom, it says: 'Please send any feedback to support@viagenie.ca.'

To start by using existing LGR files in XML format, click on the "Import" button

Select LGRs with Validating Repertoire

The screenshot shows the 'LGR Tool' interface. At the top, there is a navigation bar with 'Import', 'New', and 'Tools' buttons, and an 'About' button on the right. The main heading is 'Import existing LGR'. Below it, a sub-heading reads: 'To import LGR set, select the XML files composing the LGR set. The set will automatically be created.'

Step 1: A callout points to the 'Choose Files' button in the 'Select file(s)' section, which shows '4 files' selected. Below this, a note states: 'If you select more than one file, this will create a LGR set'.

Step 2: A callout points to the 'Validating repertoire' dropdown menu. It is currently open, showing two options: 'idna2008_6.3.0' (with a checkmark) and 'msr-2-wle-rules-13apr15-en' (highlighted in blue).

Step 3: A callout points to the 'LGR set name' input field, which contains the text 'multiple-lgrs'. Below the field, a note says: 'The name of the set'.

Step 4: A callout points to the blue 'Import' button located at the bottom right of the form.

At the bottom left, there is a language selector set to 'English (en)' and a 'Go' button. A warning note is partially visible: 'Note that importing large files may take...'.

Imported LGRs As a Merged LGR

LGR Tool / multiple-lgrs - LGR Proposed for Ethiopic Script|Proposed LGR for Khmer|P...

Import New Tools Validate label Summary View XML HTML Output Download

The screen looks like this after successful import of multiple LGR files in XML format

	Rules	Embedded LGRs	Comments	Action
				See code point
				See code point
U+0E03 (๓) 0 Variant(s)			THAI CHARACTER KHO KHUAT	See code point
U+0E04 (๔) 0 Variant(s)			THAI CHARACTER KHO KHWAI	See code point
U+0E05 (๕) 0 Variant(s)			THAI CHARACTER KHO KHON	See code point
U+0E06 (๖) 0 Variant(s)			THAI CHARACTER KHO RAKHANG	See code point
U+0E07 (๗) 0 Variant(s)			THAI CHARACTER NGO NGU	See code point
U+0E08 (๘) 0 Variant(s)			THAI CHARACTER CHO CHAN	See code point
U+0E09 (๙) 0 Variant(s)			THAI CHARACTER CHO CHING	See code point
U+0E0A (๑๐) 0 Variant(s)			THAI CHARACTER CHO CHANG	See code point
U+0E0B (๑๑) 0 Variant(s)			THAI CHARACTER SO SO	See code point

Create an LGR

Agenda Item #3

Create an LGR



The screenshot shows the top navigation bar of the LGR Tool. It includes a home icon, the text 'LGR Tool', and three buttons: 'Import', 'New', and 'Tools'. An orange callout bubble points to the 'New' button. In the top right corner, there is a '★ About' button.

Welcome to the LGR (Label Generation Ruleset) Tool

This application provides a convenient interface for browsing, creating, editing LGR files, and generating Rulesets using XML specification.

To start by creating an LGR file in XML format, click on the “New” button

Previously loaded LGR file(s)

No LGR has been previously loaded.

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@viagenie.ca.

Enter LGR Details

↑ LGR Tool

Import New Tools

★ About

Create a new LGR

Name

Validating repertoire

- ✓ idna2008_6.3.0
- msr-2-wle-rules-13apr15-en

Create

English (en) Go

1. Write name for the LGR being created

2. Select Validating repertoire from the given options for the "New" LGR

3. Click on the "Create" button

Add Code Points

The screenshot shows the LGR Tool interface for a new LGR. At the top, there is a navigation bar with a home icon and the text 'LGR Tool / new-lgr'. Below this is a toolbar with buttons for 'Import', 'New', 'Tools', 'Validate label', 'Summary', 'View XML', 'HTML Output', and 'Download'. The main content area has tabs for 'Code points', 'References', 'Meta data', and 'Rules'. On the right side of the main content area, there are two buttons: 'Expand range(s)' and 'Add code point(s)'. Below the buttons is a table with the following columns: 'Code point', 'Character Name', 'Comments', and 'Action'. At the bottom left, there is a language selector set to 'English (en)' and a 'Go' button. A red callout bubble points to the 'Add code point(s)' button.

Code point	Character Name	Comments	Action
------------	----------------	----------	--------

Click on “Add code points” to add code points to the newly created LGR

Add Code Points

The screenshot shows the LGR Tool interface with a modal dialog titled "Add code point(s)". The dialog has three tabs: "Code point", "Code point range", and "Import from file". The "Code point" tab is active, showing a text input field with "1780" and a "Add Code Point" button. The "Code point range" tab is also visible. The "Import from file" tab is also visible. A callout box points to the dialog title, stating "To add code points, there are three different ways". Three other callout boxes point to the tabs, explaining the methods: "1. Add code points one by one", "2. Add code points by giving a range", and "3. Add code points by importing from a file".

To add code points, there are three different ways

1. Add code points one by one
2. Add code points by giving a range
3. Add code points by importing from a file

Add Code Points

LGR Tool / new-lgr

Import New

Code points Referen

Code point

English (en) Go

Output Download

Add code point(s)

Action

Add code point(s)

Code point

Code point range Import from file

Code point

1780

Override repertoire

Add Code Point

1. Write the code point to be added

2. Click on the "Add Code Point" button

Add Code Points

The screenshot shows the 'Add code point(s)' dialog box in the LGR Tool. The dialog has three tabs: 'Code point', 'Code point range', and 'Import from file'. The 'Code point range' tab is selected. It contains two input fields: 'First code point' with the value '1781' and 'Last code point' with the value '179A'. A blue 'Next' button is located at the bottom right of the dialog. Three orange callout boxes provide instructions: '1. Add the first code point of the range' points to the 'First code point' field, '2. Add the last code point of the range' points to the 'Last code point' field, and '3. Click on "Next" button' points to the 'Next' button.

Add Code Points

The screenshot shows the 'Add code point(s)' dialog box in the LGR Tool. The dialog has three tabs: 'Code point', 'Code point range', and 'Import from file'. The 'Import from file' tab is active. It contains a 'Select a file' section with a 'Choose File' button and the text 'Code Points.txt'. Below that is a 'File type' dropdown menu with 'RFC3743' selected. There is also an unchecked checkbox for 'Manual import' and a blue 'Next' button.

1. Select the file containing code points
2. Select the file type from the given options
3. "Manual import" is optional – it lets you check each code point in the file before adding to the LGR
4. Click on the "Next" button

Expand Ranges

🏠 LGR Tool / new-lgr

📁 Import 📄 New ⚙️ Tools ✓ Validate label 📄 Summary 👁 View XML 🌐 HTML Output ⬇️ Download

Code points References Meta data Rules

Expand range(s) **Add code point(s)**

Code point		Comments	Action
U+1780 (ក) 0 Variant(s)	KHMER LETTER KA		See code point
U+1781 (ខ) ... U+1784 (គ)	KHMER LETTER KHA ... KH		See code point Expand range
U+1785 (ឃ) 0 Variant(s)	KHMER LETTER CA		See code point
U+1787 (ង) 0 Variant(s)	KHMER LETTER CO		See code point
U+1788 (ច) 0 Variant(s)	KHMER LETTER CHO		See code point
U+1789 (ឆ) 0 Variant(s)	KHMER LETTER NYO		See code point
U+178A (ចា) 0 Variant(s)	KHMER LETTER DA		See code point
U+178B (ត) 0 Variant(s)	KHMER LETTER TTHA		See code point

To expand all the code points in all the ranges of the LGR, click on "Expand range(s)"

To expand all the code points in this range, click on "Expand range"

Add References

LGR Tool / new-lgr

Import New Tools Validate label Summary View XML HTML Output Download

Code points **References** Meta data Rules

Existing references

Reference id	Description	URL	Action
--------------	-------------	-----	--------

Save

New reference

Description	URL
The Unicode Consortium. The Unicode Standard, Version 6.3.0,	Any code point cited w:

Add

English (en) Go

©

To add references to your LGR, click on "References" tab

1. Add detailed reference

2. Add comments to the added reference

3. Click on the "Add" button

Add References

🏠 LGR Tool / new-lgr

📁 Import 📄 New ⚙️ Tools ▾ ✓ Validate label 📄 Summary 👁️ View XML 🌐 HTML Output ⬇️ Download

New reference created

Code points **References** Meta data Rules

“Existing references”
tab shows the added
references

Existing references

Reference id	Description	URL	Action
0	The Unicode Consortium. The Unicode Standard, Version 6.3.0,	Any code point cited was originally encoded in Unicode	🗑️

Save

New reference

Description	URL
The Unicode Consortium. The Unicode Standard, Version 6.3.0,	Any code point cited w:

Add

English (en) ▾ Go

©

Define Meta Data

🏠 LGR Tool / new-lgr

📁 Import 📄 New ⚙️ Tools ▾ ✓ Validate label 📄 Summary 👁 View XML 🌐 HTML Output ⬇️ Download

Code points References **Meta data** Rules

Version	<input type="text" value="1"/>
Version comment	<input type="text"/>
Date	<input type="text" value="2017-10-10"/>
Language	<input type="text" value="und-Khmer"/>
Scope	<input type="text" value="."/>
Scope type	<input type="text" value="domain"/>
Validity start	<input type="text"/>
Validity end	<input type="text"/>
Unicode version	<input type="text" value="6.3.0"/>
Description	<input type="text"/>

Click on "Meta data" tab to add meta information about the LGR

Define Meta Data

Second half of the “Meta data” tab

LGR Tool / new-lgr

Import New Tools Validate label Summary View XML HTML Output Download

Validity start

Validity end

Unicode version 6.3.0

Description

```
<h1>Label Generation Rules for Khmer</h1>
<h2>Overview</h2>
<p>For more details on this proposal see "Proposal for a Khmer Script Root Zone LGR [Propo
<h2>Repertoire</h2>
<p>According to Section 5 "Repertoire", in "[Proposal]".</p>
<h2>Variants</h2>
<p>According to Section 6 "Variants", in "[Proposal]", two Khmer consonants are variants of each
in their
subscript form only. This is captured by a variant relation between the two subscript sequences, w
are listed
explicitly in the repertoire.</p>
<h2>Character Classes</h2>
```

Description type

- text/plain
- ✓ text/html

Validating repertoire

msr-2-wle-rules-13apr15-en

Cancel Save

Add any text for describing the LGR

Select “Description type” from the given options

“Validating repertoire” is the same as selected in the beginning

After filling every detail, click on the “Save” button

Define Classes, Rules & Actions

The screenshot shows the LGR Tool interface for a new LGR. At the top, there is a navigation bar with the title 'LGR Tool / new-lgr' and several buttons: 'Import', 'New', 'Tools', 'Validate label', 'View XML', 'HTML Output', and 'Download'. Below this is a tabbed interface with 'Code points', 'References', 'Meta data', and 'Rules'. The 'Rules' tab is selected and highlighted. Below the tabs are three sections: 'Classes', 'Rules', and 'Actions'. Each section has a 'New class', 'New rule', and 'New action' button respectively. Three orange callout boxes provide instructions: one pointing to the 'Rules' tab, and two others pointing to the 'New class', 'New rule', and 'New action' buttons. At the bottom left, there is a language selector set to 'English (en)' and a 'Go' button. A small copyright symbol is visible at the bottom right of the interface.

Click on "Rules" tab to add certain rules to the LGR

Click on "New class" button to add classes to the LGR

Click on "New rule" button to add rules to the LGR

Click on "New action" button to add actions to the LGR

Define Classes

🏠 LGR Tool / new-lgr

📁 Import 📄 New ⚙️ Tools ▾ ✓ Validate label 📄 Summary 👁️ View XML 🌐 HTML Output ⬇️ Download

Code points References Meta data **Rules**

Classes

```
<class name="consonant" from-tag="consonant" comment="Any consonant" />
```

💾 Save ✕ Cancel

Rules

Actions

English (en) ▾ Go

©

1. Add classes in the relevant box

2. Click on the "Save" button

Define Rules

🏠 LGR Tool / new-lgr

📁 Import 📄 New ⚙️ Tools ▾ ✓ Validate label 📄 Summary 👁️ View XML 🌐 HTML Output 📄 Download

Rules

```
<rule name="follows-consonant" comment="checks if sign code point consonant follows a consonant">
  <look-behind>
    <class by-ref="consonant" />
  </look-behind>
  <anchor />
</rule>
```

💾 Save ✕ Cancel

1. Add the rule in the relevant box

2. Click on the "Save" button

Actions

Define Actions

🏠 LGR Tool / new-lgr

📁 Import 📄 New ⚙️ Tools ✓ Validate label 📄 Summary 👁️ View XML 🌐 HTML Output 📄 Download

```
<rule name="follows-consonant" comment="checks if sign code point consonant follows a consonant">
  <look-behind>
    <class by-ref="consonant" />
  </look-behind>
  <anchor/>
</rule>
```

Actions

📄 Save ✕ Cancel

```
<action disp="blocked" any-variant="blocked" />
```

English (en) Go ©

Define Code Point Properties

LGR Tool / new-lgr

Import New Tools Validate label Summary View XML HTML Output Download

Code points References Meta data Rules

Expand range(s) Add code point(s)

Code point		Comments	Action
U+1780 (័) 0 Variant(s)			See code point
U+1781 (េ) ... U+1784 (ឺ)	KHMER LETTER KHA ... KHMER LETTER NGO		See code point Expand range
U+1785 (ឺ) 0 Variant(s)	KHMER LETTER CA		See code point
U+1787 (ឺ) 0 Variant(s)	KHMER LETTER CO		See code point
U+1788 (ឺ) 0 Variant(s)	KHMER LETTER CHO		See code point
U+1789 (ឺ) 0 Variant(s)	KHMER LETTER NYO		See code point
U+178A (ឺ) 0 Variant(s)	KHMER LETTER DA		See code point
U+178B (ឺ) 0 Variant(s)	KHMER LETTER TTHA		See code point
U+178C (ឺ) 0 Variant(s)	KHMER LETTER DO		See code point

Click on the "See code point" button to add code point details/properties

Define Code Point Properties - 1

1. Add variant for the code point

2. Click on the "Add variant button"

3. Add details for the added variant – type, comment, when-rule, not-when rule and some action

4. Add tags for the code point

5. Add when-rule/not-when rule from the list of rules it provides

6. Add any description for the code point

Code point	Type	Comments	When	Not When	References	Action
U+1785 (្ក) KHMER LETTER CA Age: 3.0.0.0	blocked					Delete variant Edit references

Tags:

When:

Not-When:

Comment:

Define Code Point Properties - 2

LGR Tool / new-lgr / U+1785 (្ក) KHMER LETTER CA

Import New Tools Validate label Summary View XML HTML Output Download

When

follows-consonant

Not-When

Comment

Any text description for the code point

Save variants, tags, context rules and comment

7. Click on “Save variants, tags, context rules and comment” button

References

No references associated with code point.

Edit

8. Click on the “Edit” button to add references to the code point

Delete code point

To delete code point & its details, click on the “Delete code point” button

English (en) Go

Download LGR

LGR Tool / new-lgr

Import New Tools Validate label Summary View XML HTML Output Download

Code points References M

Expand range(s) Add code point(s)

Code point		Action
U+1781 (𑄁) 0 Variant(s)		See code point
U+1782 (𑄂) 0 Variant(s)		See code point
U+1783 (𑄃) 0 Variant(s)	KHMER LETTER KHO	See code point
U+1784 (𑄄) 0 Variant(s)	KHMER LETTER NGO	See code point
U+1785 (𑄅) 0 Variant(s)	KHMER LETTER CA	See code point
U+1786 (𑄆) 0 Variant(s)	KHMER LETTER CHA	See code point
U+1787 (𑄇) 0 Variant(s)	KHMER LETTER CO	See code point
U+1788 (𑄈) 0 Variant(s)	KHMER LETTER CHO	See code point
U+1789 (𑄉) 0 Variant(s)	KHMER LETTER NYO	See code point

Click on "Download" button to download the created LGR. Always download before closing the browser as the LGRs are not saved on the server

Summarize LGR

Agenda Item #4

Summarize LGR

LGR Tool / proposal-lao-lgr-31jan17-en - Proposed LGR for Lao

Import New Tools Validate label Summary View XML HTML Output Download

Code points References Meta data Rules

Expand range(s) Add code point(s)

Code point	Action
U+0E81 (๑) 0 Variant(s)	See code point
U+0E82 (๒) 0 Variant(s)	See code point
U+0E84 (๔) 0 Variant(s)	See code point
U+0E87 (๗) 0 Variant(s)	See code point
U+0E88 (๘) 0 Variant(s)	See code point
U+0E8A (๙) 0 Variant(s)	LAO LETTER SO TAM See code point
U+0E8D (๓) 0 Variant(s)	LAO LETTER NYO See code point
U+0E94 (๔) 0 Variant(s)	LAO LETTER DO See code point
U+0E95 (๕) 0 Variant(s)	LAO LETTER TO See code point
U+0E96 (๖) 0 Variant(s)	LAO LETTER THO SUNG See code point

Click on “Summary” button to get summary of the entire LGR. It is important to note that when summary is generated, LGR check is also performed. Therefore, this function should also be used for checking an LGR before use, e.g. when it is created or imported

Summarize LGR

The screenshot shows the 'LGR Summary' window of the LGR Tool. The window title is 'LGR Tool / proposal-lao-lgr-31jan17-en - Proposed LGR for Lao'. The main content area lists several checks and their results:

- Testing XML validity
- RNG validation OK
- Testing XML validity done
- Testing symmetry
- Symmetry test done
- Testing transitivity
- Transitivity test done
- Testing conditional variants
- Conditional variants test done
- Rebuilding LGR 'proposal-lao-lgr-31jan17-en' with U+0E95 (c) 0 Variant(s)
- Rebuilding LGR 'proposal-lao-lgr-31jan17-en' done

Below the list, a 'General summary' section provides the following statistics:

- Number of code points: 52.
- Number of ranges: 0.
- Largest range: (length: 0).
- Number of sequences: 1.
- Largest sequence: U+0EB2 U+0EB0 (length: 2).
- Variants:
- Total number of variants: 0.
- Average number of variants per code point: 0.

At the bottom right of the window, there are two buttons: 'Save summary' and 'Close'.

Two orange callout boxes are overlaid on the image:

- The first callout box, pointing to the 'LGR Summary' title, contains the text 'Summarized LGR'.
- The second callout box, pointing to the list of checks, contains the text 'Summary of checks performed, including symmetry and transitivity'.

View LGR as XML and HTML

Agenda Item #5

View LGR As XML

🏠 LGR Tool / proposal-thai-lgr-25may17-en-copy - Thai Script Root Zone LGR Version 6.9

📁 Import 📄 New ⚙️ Tools ▾ ✓ Validate label 📄 Summary 👁️ View XML 🌐 HTML Output 📄 Download

Code points References Meta data Rules

Expand range(s) Add code point(s)

Code point	Code	Comments	Action
U+0E01 (๑) 0 Variant(s)	THAI CHARACTER KO THAI		See code point
U+0E02 (๒) 0 Variant(s)	THAI CHARACTER KHO KHAI		See code point
U+0E03 (๓) 0 Variant(s)	THAI CHARACTER KHO KHUAT		See code point
U+0E04 (๔) 0 Variant(s)	THAI CHARACTER KHO KHWAI		See code point
U+0E05 (๕) 0 Variant(s)	THAI CHARACTER KHO KHON		See code point
U+0E06 (๖) 0 Variant(s)	THAI CHARACTER KHO RAKHANG		See code point
U+0E07 (๗) 0 Variant(s)	THAI CHARACTER NGO NGU		See code point
U+0E08 (๘) 0 Variant(s)	THAI CHARACTER CHO CHAN		See code point
U+0E09 (๙) 0 Variant(s)	THAI CHARACTER CHO CHING		See code point
U+0E0A (๐) 0 Variant(s)	THAI CHARACTER CHO CHANG		See code point

Click on "View XML" to get an XML view of the LGR

View LGR As XML

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼<lgr xmlns="urn:ietf:params:xml:ns:lgr-1.0">
  ▼<meta>
    <version comment="Thai Script Root Zone LGR Version 6.9">2</version>
    <date>2017-05-25</date>
    <unicode-version>6.3.0</unicode-version>
    <language>und-Thai</language>
    <scope type="domain">.</scope>
  ▼<description type="text/html">
    ▼<![CDATA[
      <h1>Label Generation Rules for the Thai Script</h1> <h2>Overview</h2> <p>This file contains Label
      Generation Rules (LGR) for the Thai script as would be appropriate for the Root zone. For more details on
      this LGR see "Proposal for a Thai Script Root Zone LGR [Proposal]" </p> <h2>Repertoire</h
    ]]>
    ▼<![CDATA[
      2> <p> In addition to the 68 code points according to Section 5 "Repertoire" in [Proposal], three sequences
      have been defined. The sequence U+0E4D U+0E32 was defined to replace the disallowed U+0E33 (THAI CHARACTER
      SARA AM) and to facilitate implementation of WLE rule <b>follows-consonant-tone</b> as a context rule. The
      other two sequences were defined to restrict U+0E45 (THAI CHARACTER LAKKHANGYAO) from appearing in any
      context other than these sequences. Accordingly, while U+0E45 is not listed by itself it brings the total
      of distinct code points to 69.</p> <h2>Variants</h2> <p>According to Section 6 "Variants", in "[Proposal]",
      this LGR defines no variants.</p> <h2>Character Classes</h2> <p>The Thai Script is an abugida in which
      consonant-vowel sequences are written as a unit: each unit is based on a consonant letter, and vowel, tone
      mark or diacritic notation are secondary. It is written with the combining marks stacked above or below the
      base consonant, like diacritics in European languages. However, although the concepts are quite similar,
      the implementations are significantly different.</p> <p>There are 44 characters that are classified as
      consonants, code points from this subset have been given the tag "cons". </p> <p>The 18 vowel symbols
      pronounced after a consonant are non-sequential: they can be located before (lv) , after (fv), above (av)
      or below (bv) the consonant, or in a combination of these positions, code points from this subset have been
      given the tag "fv1", "fv2", "fv3", "av", "bv", "lv". There are three code point sequences defined that include
      vowels. (Code point sequences do not carry tag values; instead, for code point sequences the subset values
      are identified in comments).</p> <p>There are 5 phonemic tones: mid, low, falling, high, and rising. These
      5 tones are represented by 4 tone marks plus the absence of a mark. Code points from this subset have been
      given the tag "tone"</p> <p>There are 3 diacritic symbols that have been included here and given the tag
      "ad". They differ in their frequency and purpose of usage. See also the discussion in section 5.4 in
      [Proposal].</p> <ul> <li>U+0E47 (MAITAIKHU) and U+0E4C (THANTHAKHAT) are commonly used in everyday
      communicating words</li> <li>U+0E4D (NIKHAHIT) is included because of its use to decompose U+0E33 (SARA AM,
      SARA AM) which is in common use, but U+0E33 can also be used by itself </li> </ul> </p> </h2> </h1>
    ]]>
  </description>
</lgr>
```

XML view of the LGR

View LGR As HTML

🏠 LGR Tool / merged-igr-1 - LGR Proposed for Ethiopic Script|Proposed LGR for Lao|Tha...

📁 Import 📄 New ⚙️ Tools ▾ ✓ Validate label 📄 Summary 👁 View XML 🌐 HTML Output 📄 Download

Code points References Data Rules Embedded LGRs

Code point Comments Action

U+0E01 (๑) 0 Variant(s)		See code point
U+0E02 (๒) 0 Variant(s)		See code point
U+0E03 (๓) 0 Variant(s)	THAI CHARACTER KHO KHUAT	See code point
U+0E04 (๔) 0 Variant(s)	THAI CHARACTER KHO KHWAI	See code point
U+0E05 (๕) 0 Variant(s)	THAI CHARACTER KHO KHON	See code point
U+0E06 (๖) 0 Variant(s)	THAI CHARACTER KHO RAKHANG	See code point
U+0E07 (๗) 0 Variant(s)	THAI CHARACTER NGO NGU	See code point
U+0E08 (๘) 0 Variant(s)	THAI CHARACTER CHO CHAN	See code point
U+0E09 (๙) 0 Variant(s)	THAI CHARACTER CHO CHING	See code point
U+0E0A (๑๐) 0 Variant(s)	THAI CHARACTER CHO CHANG	See code point

1. Click on "Import" to load a single or multiple existing LGR file(s). See "Import" for details

2. Click on "HTML Output" button to view the HTML output of the LGR

Meta Data - 1

Meta data in LGR

Merged-Lgr-1

LGR Version	2
Date	2017-10-21
Language(s)	und-Ethi und-Lao und-Thai
Scope(s)	domain: .
Unicode Version	6.3.0

This document is mechanically formatted from the XML file for the LGR. It provides additional summary data and explanatory text. The XML file remains the sole normative specification of the LGR.

Table of Contents

- 1 Description
- 2 Repertoire
- 3 Variant Sets
- 4 Classes, Rules and Actions
 - 4.1 Character Classes
 - 4.2 Whole label evaluation and context rules
 - 4.3 Actions
- 5 Table of References

Description

Meta Data - 2

Label Generation Rules for the Thai Script

Meta data in LGR

Overview

This file contains Label Generation Rules (LGR) for the Thai script as would be appropriate for the Root zone. For more details on this LGR see "Proposal for a Thai Script Root Zone LGR [Proposal]"

Repertoire

In addition to the 68 code points according to Section 5 "Repertoire" in [Proposal], three sequences have been defined. The sequence U+0E4D U+0E32 was defined to replace the disallowed U+0E33 (THAI CHARACTER SARA AM) and to facilitate implementation of WLE rule **follows-consonant-tone** as a context rule. The other two sequences were defined to restrict U+0E45 (THAI CHARACTER LAKKHANGYAO) from appearing in any context other than these sequences. Accordingly, while U+0E45 is not listed by itself it brings the total of distinct code points to 69.

Variants

According to Section 6 "Variants", in "[Proposal]", this LGR defines no variants.

Character Classes

The Thai Script is an abugida in which consonant–vowel sequences are written as a unit: each unit is based on a consonant letter, and vowel, tone mark or diacritic notation are secondary. It is written with the combining marks stacked above or below the base consonant, like diacritics in European languages. However, although the concepts are quite similar, the implementations are significantly different.

There are 44 characters that are classified as consonants, code points from this subset have been given the tag "cons".

The 18 vowel symbols pronounced after a consonant are non-sequential: they can be located before (lv) , after (fv) , above (av) or below (bv) the consonant, or in a combination of these positions, code points from this subset have been given the tag "fv1","fv2","fv3","av","bv","lv". There are three code point sequences defined that include vowels. (Code point sequences do not carry tag values; instead, for code point sequences the subset values are identified in comments).

There are 5 phonemic tones: mid, low, falling, high, and rising. These 5 tones are represented by 4 tone marks plus the absence of a mark. Code

Repertoire

Repertoire

Summary

Number of elements in repertoire	434
Number of ranges in repertoire	0
Number of code point sequences	4

Repertoire section
in LGR

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.

#	Code Point	Glyph	Script	Name	Tags	Required Context	Variants	Comment	References
1	U+0E01	ก	Thai	THAI CHARACTER KO KAI	Thai,und-Thai-cons				[5] , [100] , [101]
2	U+0E02	ข	Thai	THAI CHARACTER KHO KHAI	Thai,und-Thai-cons				[5] , [100] , [101]
3	U+0E03	ฃ	Thai	THAI CHARACTER KHO KHUAT	Thai,und-Thai-cons				[5] , [100] , [101]
4	U+0E04	ค	Thai	THAI CHARACTER KHO KHWAI	Thai,und-Thai-cons				[5] , [100] , [101]
				THAI					[5] , [100]

Variant Sets

Variant Sets

Summary

Number of variant sets	30
Largest variant set	4
Ordinary Variants by Type	blocked (98)

Variant Sets section in LGR

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 – 6 Members

#	Source	Glyph	Target	Glyph	Type(s)	References	Comment
1	U+1200	ᵀ	U+1210	ᵀ	blocked	[4]	
2	U+1200	ᵀ	U+1280	ᵀ	blocked	[4]	
3	U+1210	ᵀ	U+1200	ᵀ	blocked	[4]	
4	U+1210	ᵀ	U+1280	ᵀ	blocked	[4]	
5	U+1280	ᵀ	U+1200	ᵀ	blocked	[4]	
6	U+1280	ᵀ	U+1210	ᵀ	blocked	[4]	

Variant Set 2 – 6 Members

#	Source	Glyph	Target	Glyph	Type(s)	References	Comment
1	U+1201	ᵀ	U+1211	ᵀ	blocked	[4]	
2	U+1201	ᵀ	U+1281	ᵀ	blocked	[4]	
3	U+1211	ᵀ	U+1201	ᵀ	blocked	[4]	
4	U+1211	ᵀ	U+1281	ᵀ	blocked	[4]	

Classes

Classes, Rules and Actions

Character Classes

Character classes
in LGR

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

Name	Definition	Count	Members	References	Comment
und-Laoo-Cf	Tag= und-Laoo-Cf	25	<code>[[:class tag=und-Laoo-Cf:]]</code>		
und-Laoo-consonant	Tag= und-Laoo-consonant	32	<code>[[:class tag=und-Laoo-consonant:]]</code>		
und-Laoo-semi-consonant	Tag= und-Laoo-semi-consonant	37	<code>[[:class tag=und-Laoo-semi-consonant:]]</code>		
und-Laoo-tone-mark	Tag= und-Laoo-tone-mark	32	<code>[[:class tag=und-Laoo-tone-mark:]]</code>		
und-Laoo-vowel-above	Tag= und-Laoo-vowel-above	34	<code>[[:class tag=und-Laoo-vowel-above:]]</code>		
und-Laoo-vowel-below	Tag= und-Laoo-vowel-below	34	<code>[[:class tag=und-Laoo-vowel-below:]]</code>		
und-Thai-above-vowel	Tag= und-Thai-av	25	<code>[[:class tag=und-Thai-av:]]</code>		Any above vowel
und-Thai-below-vowel	Tag= und-Thai-bv	25	<code>[[:class tag=und-Thai-bv:]]</code>		Any below vowel
und-Thai-consonant	Tag= und-Thai-cons	27	<code>[[:class tag=und-Thai-cons:]]</code>		Any Consonant
und-Thai-sara-aa	Tag= und-Thai-sara-aa	30	<code>[[:class tag=und-Thai-sara-aa:]]</code>		SARA AA
und-Thai-tone	Tag= und-Thai-tone	27	<code>[[:class tag=und-Thai-tone:]]</code>		Any tone mark
und-Thai-c-av-bv		198	<code>([:und-Thai-consonant:]∪[:und-Thai-above-vowel:]∪[:und-Thai-below-vowel:])</code>		Any consonant, vowel-above or vowel-below
und-Thai-ct		117	<code>([:und-Thai-consonant:]∪[:und-Thai-tone:])</code>		Any consonant or tone

Rules

Whole Label evaluation and context rules in LGR

Whole label evaluation and context rules

The following table lists all the top-level, or named rules defined in the LGR and indicates whether they are used as trigger in an action or as context (when or not-when) for a code point. (Any use of context rules for variants is not indicated).

Name	Regular Expression	Used as Trigger	Used as Context	Anchor	References	Comment
Common-leading-combining-mark	(start) ([:class property:gc=Mn:] [:class property:gc=Mc:])	True	False	False		None
und-Laoo-follows-consonant	([:und-Laoo-consonant:])← 🚫	False	True	True		WLE Rule No. 1; semi-consonant must follow a consonant
und-Laoo-precedes-consonant	🚫 → ([:und-Laoo-consonant:])	False	True	True		WLE Rule No. 2; vowel-before precedes a main consonant cluster
und-Laoo-follows-main-consonant	(([:und-Laoo-consonant:] [:und-Laoo-semi-consonant:]))← 🚫	False	True	True		WLE Rule No. 3; vowel-above, and vowel-below follow a main consonant C
und-Laoo-follows-C-tonemark-vabove	(([:und-Laoo-consonant:] [:und-Laoo-semi-consonant:] [:und-Laoo-tone-mark:] [:und-Laoo-vowel-above:]))← 🚫	False	True	True		WLE Rule No. 4; vowel-after follows a main consonant, tone-mark or vowel-above
und-Laoo-consonant-cluster	([:und-Laoo-consonant:]){1,2} ([:und-Laoo-semi-consonant:]){0,1}	False	False	False		Defining consonant cluster for Rule No. 5
und-Laoo-follows-vbefore-consonant-cluster	(U+0EC0[:und-Laoo-consonant-cluster:])← 🚫	False	True	True		WLE Rule No. 5; The sequence (0EB2 0EB0) follows a vowel before, and a consonant cluster
und-Laoo-follows-C-vabove-vbelow	(([:und-Laoo-consonant:] [:und-Laoo-semi-consonant:] [:und-Laoo-vowel-above:] [:und-Laoo-vowel-below:]))← 🚫	False	True	True		WLE Rule No. 6; A tone-mark follows a main consonant, vowel-above or vowel-below
und-Laoo-follows-Cf	([:und-Laoo-Cf:])← 🚫	False	True	True		WLE Rule No. 7; The sign 0ECC can only occur after final

Actions

Actions in LGR

Actions

The following table lists the actions that are used to assign dispositions to labels and variant labels, based on the specified conditions. The order of actions defines their precedence: the first action triggered by a label is the one defining its disposition.

#	Condition	Rule / Variant Set	Disposition	References	Comment
1	if label match	Common-leading-combining-mark	→ invalid		
2	if at least one variant is in	{out-of-repertoire-var}	→ invalid		any variant label with a code point out of repertoire is invalid
3	if label match	Common-leading-combining-mark	→ invalid		labels must not commence with a combining mark ☺
4	if at least one variant is in	{out-of-repertoire-var}	→ invalid		any variant label with a code point out of repertoire is invalid ☺
5	if label match	Common-leading-combining-mark	→ invalid		
6	if at least one variant is in	{out-of-repertoire-var}	→ invalid		any variant label with a code point out of repertoire is invalid

Legend

{...} - variant type set: In the "Rule/Variant Set" column the notation {...} means a set of variant types.

Table of References

- [0] The Unicode Standard 1.1, The Unicode Consortium, Mountain View, CA. 1993
- [1] The Online Encyclopedia of Writing Systems & Languages, <http://www.omniglot.com/writing/amharic.htm>, <http://www.omniglot.com/writing/argobba.htm>, <http://www.omniglot.com/writing/awngi.htm>, <http://www.omniglot.com/writing/harari.htm>, <http://www.omniglot.com/writing/xamtanga.htm>, <http://www.omniglot.com/writing/silte.htm>, <http://www.omniglot.com/writing/tigre.htm>, <http://www.omniglot.com/writing/tigrinya.htm>
Ethiopic Script Versions for the Eight Languages
- [2] Corpus Analysis performed by crawling 598 html and 40 PDF files with Tigrigna Contents published online, August 2016
Cited as Auxiliary Evidence for Tigrigna Code Points
- [3] Corpus Analysis performed by crawling 14,850 html Amharic Contents of size 1.8 GB published online, August 2016
Cited as Auxiliary Evidence for Amharic Code Points

Table of References

Legend

{...} - variant type set: In the "Rule/Variant Set" column the notation {...} means a set of variant types.

Table of References
in LGR

Table of References

- [0] The Unicode Standard 1.1, The Unicode Consortium, Mountain View, CA. 1993
- [1] The Online Encyclopedia of Writing Systems & Languages, <http://www.omniglot.com/writing/amharic.htm>, <http://www.omniglot.com/writing/argobba.htm>, <http://www.omniglot.com/writing/awngi.htm>, <http://www.omniglot.com/writing/harari.htm>, <http://www.omniglot.com/writing/xamtanga.htm>, <http://www.omniglot.com/writing/silte.htm>, <http://www.omniglot.com/writing/tigre.htm>, <http://www.omniglot.com/writing/tigrinya.htm>
Ethiopic Script Versions for the Eight Languages
- [2] Corpus Analysis performed by crawling 598 html and 40 PDF files with Tigrigna Contents published online, August 2016
Cited as Auxiliary Evidence for Tigrigna Code Points
- [3] Corpus Analysis performed by crawling 14,850 html Amharic Contents of size 1.8 GB published online, August 2016
Cited as Auxiliary Evidence for Amharic Code Points
- [4] Daniel Yacob "Application of the Double Metaphone Algorithm to Amharic Orthography", International Conference of Ethiopian Studies XV, <https://pdfs.semanticscholar.org/2f71/033d74d2f17a9502867e4a43dc4374500726.pdf>
Cited for Amharic-Driven Variants in Ethiopic Script
- [5] The Unicode Standard 1.1
- [100] Thai Industrial Standard (TIS) 1566-2541(1988) (<http://www.ratchakitcha.soc.go.th/DATA/PDF/2542/E/088/9.PDF>)
- [101] Computers and the Thai Language (http://lexitron.nectec.or.th/KM_HL5001/file_HL5001/Paper/Inter%20Journal/krrn_52085.pdf)
- [201] Lao grammar book published by the Ministry of Education in 1967, see Appendix B, Figure 1
- [202] Lao grammar book published by the Ministry of Education in 1967, see Appendix B, Figure 2
- [203] Lao grammar book published by the Ministry of Education in 1967, see Appendix B, Figure 3
- [204] Lao grammar book published by the Ministry of Education in 2000, see Appendix B, Figure 4
- [205] Lao grammar book published by the Ministry of Education in 2000, see Appendix B, Figure 5
- [206] Lao grammar book published by the Ministry of Education in 2000, see Appendix B, Figure 6
- [207] Lao grammar 1935, see Appendix B, Figure 7

Validate Label

Agenda Item #6

Validate Label

The screenshot shows the LGR Tool interface for a proposed LGR for Lao. The top navigation bar includes buttons for Import, New, Tools, Validate label, Summary, View XML, HTML Output, and Download. Below the navigation bar, there are tabs for Code points, References, Meta data, and Rules. A table lists Lao letters with their Unicode code points and variant counts. Callout boxes provide instructions: 1. Click on "Import" to load a single or multiple existing LGR files. See "Import" for details. 2. Always view "Summary" as the tool checks the loaded LGR(s) during this process. 3. For validating labels, click on "Validate label" button.

Code point	Name	Comment	Action
U+0E87 (ɔ) 0 Variant(s)	LAO LETTER NGO		See code point
U+0E88 (ɔ) 0 Variant(s)	LAO LETTER CO		See code point
U+0E8A (ɔ) 0 Variant(s)	LAO LETTER SO TAM		See code point
U+0E8D (ɔ) 0 Variant(s)	LAO LETTER NYO		See code point
U+0E94 (ɔ) 0 Variant(s)	LAO LETTER DO		See code point
U+0E95 (ɔ) 0 Variant(s)	LAO LETTER TO		See code point

Validate Label with a Single LGR

LGR Tool / proposal-lao-lgr-31jan17-en - Proposed LGR for Lao

Validate label

Label ທະບືນ

maximum length: 63 code points

Validate

1. Enter the label to be validated

2. Click on the "Validate" button

U+0E96 (໗) 0 Variant(s) LAO LETTER THO SUNG See code point

Validate Label with a Single LGR

LGR Tool / proposal-lao-igr-31jan17-en - Proposed LGR for Lao

Validate label

Label:

Maximum length: 63 code points

✓ VALID

U-label	Disposition	Code point sequence	
ກະເບີງ xn-- p6c6c0cj8b2a	valid	U+0E81 (ກ) U+0EB0 (ະ) U+0EC0 (ຸ) U+0E9B (ຸ) U+0EBB (ຸ) U+0EB2 (ຸ)	<input type="button" value="Show / hide rule"/>

Variant labels (including original as last)

1 variant label(s) generated.
By disposition: Counter({'valid': 1})

ກະເບີງ xn-- p6c6c0cj8b2a	valid	U+0E81 (ກ) U+0EB0 (ະ) U+0EC0 (ຸ) U+0E9B (ຸ) U+0EBB (ຸ) U+0EB2 (ຸ)	<input type="button" value="Show / hide rule"/>
--------------------------------	-------	---	---

U+0E96 (ກ) 0 Variant(s) LAO LETTER THO SUNG See code point

Results given by “Validate label” show if the label is valid and also list its variants and their dispositions

Validate Label with a Merged LGR

The screenshot shows the 'Validate label' window in the LGR Tool. The window title is 'LGR Tool / merged-lgr - Proposed LGR for Armenian|Proposed LGR for Lao|Thai Scrip...'. The main content area has a 'Label' input field containing 'ကမ္ဘောဇ', a 'Validate' button, and a section for 'Allocated Set labels' with a 'Script' dropdown menu. The 'Script' dropdown is open, showing 'und-Armn', 'und-Lao' (selected), and 'und-Thai'. The 'Allocated Set labels' section includes a 'Choose File' button and the text 'No file chosen'. Below this, there is a description: 'Optional list of labels already allocated in the LGR that will be used to check for collisions when evaluating labels'. The 'Label' field has a note 'Maximum length: 63 code points'.

1. Enter the label to be validated

2. Select a script from the list of the scripts of different LGRs forming the merged LGR

Optional file of allocated set labels to check for collisions

3. Click on the "Validate" button

Validate Label with a Merged LGR

LGR Tool / merged-lgr - Proposed LGR for Armenian|Proposed LGR for Lao|Thai Scrip...

Validate label

Script: und-Lao
The script used to validate the label

✓ VALID

U-label	Disposition	Code point sequence	
ກຮຸ້ງ xn-- p6c6c0cj8b2a	valid	U+0E81 (ກ) U+0EB0 (ຮ) U+0EC0 (ງ) U+0E9B (ຸ) U+0EBB (ຸ) U+0EB2 (ຸ)	Show / hide rule

✓ No collision

Collision

No collision.

Variant labels (including original as last)

1 variant label(s) generated.
By disposition: Counter({'valid': 1})

ກຮຸ້ງ xn-- p6c6c0cj8b2a	valid	U+0E81 (ກ) U+0EB0 (ຮ) U+0EC0 (ງ) U+0E9B (ຸ) U+0EBB (ຸ) U+0EB2 (ຸ)	Show / hide rule
-------------------------------	-------	---	------------------

Result given by “Validate label” shows if the label is valid and if it has collisions with any of the labels given in the allocated set labels file. It also lists its variants and their dispositions

Review Impact on Existing Labels by Revising an Existing LGR

Agenda Item #7

Select Difference Labels of Two LGRs

🏠 LGR Tool / proposal-lao-lgr-31jan17-en - Proposed LGR for Lao

📁 Import 📄 New ⚙️ Tools ▾ ✓ Validate label 📄 Summary 👁️ View XML 🌐 HTML Output 📄 Download

Code points Referen

Code point

U+0E81 (๓) 0 Variant(s)

U+0E82 (๔) 0 Variant(s) LAO LETTER KHO SUNG See code point

U+0E84 (๕) 0 Variant(s) LAO LETTER KHO TAM See code point

U+0E87 (๖) 0 Variant(s) LAO LETTER NGO See code point

U+0E88 (๗) 0 Variant(s) LAO LETTER CO See code point

U+0E8A (๘) 0 Variant(s) LAO LETTER SO TAM See code point

U+0E8D (๙) 0 Variant(s) LAO LETTER NYO See code point

U+0E94 (๑๐) 0 Variant(s) LAO LETTER DO See code point

U+0E95 (๑๑) 0 Variant(s) LAO LETTER TO See code point

U+0E98 (๑๒) 0 Variant(s) LAO LETTER THO SUNG See code point

Tools ▾

- ↔️ Compare two LGRs
- 🔍 Diff labels of two LGRs
- ✂️ Get collisions in a list of labels
- ☰ Generate disp. annotations

Click on “Diff labels of two LGRs” button to determine label collisions caused by modifying an LGR

Enter Details

🏠 LGR Tool / proposal-lao-lgr-02aug17-en - Proposed LGR for Lao

📁 Import 📄 New ⚙️ Tools ▾ ✓ Validate label 📄 Summary 👁 View XML 🌐 HTML Output 📄 Download

First LGR **1. Select first LGR**
First LGR to use in diff

Second LGR **2. Select second LGR**
Second LGR to use in diff

Labels test-labels.txt **3. Select file containing labels**
List of labels to use in diff. File must be encoded in UNIX format.

As the computing may be very long, we will warn by e-mail once the result can be downloaded. Please provide a valid e-mail address:

E-mail **4. Enter your email address where you want to receive results**

5. Check "Check collisions" if you want to check label collisions as well Check collisions
Also check for collision of labels in both LGR

6. Check "Output rules" if you want to check output rules for each label Output rules
Show rules in output (this can be very verbose)

7. Click on "Get diff" button

⚠ Note that diff tool is not available for LGR sets

Email Notification

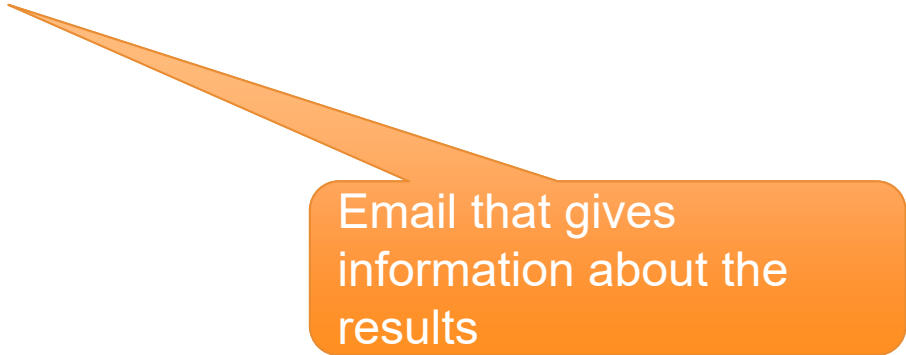
Hi,

The processing of diff from labels provided in the file 'test-labels.txt' between LGR 'proposal-lao-lgr-31jan17-en' and LGR 'proposal-lao-lgr-02aug17-en' has been successfully completed.

You should now be able to download it from your home screen under the name: '20171020_133848_diff_proposal-lao-lgr-31jan17-en_proposal-lao-lgr-02aug17-en.txt.gz'.

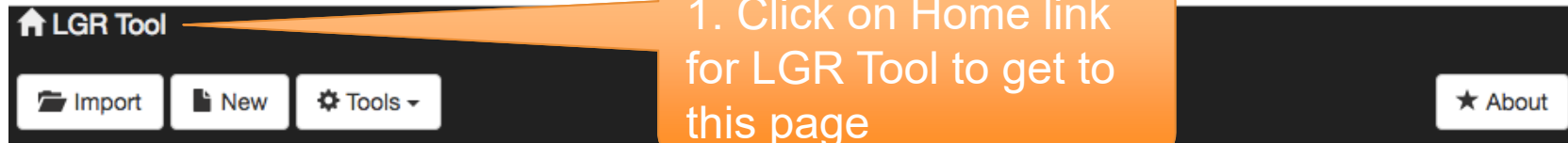
Please refresh the home page if you don't see the link.

Best regards



Email that gives information about the results

Download Results



1. Click on Home link for LGR Tool to get to this page



Welcome to the LGR (Label Generation Ruleset) Tool

This application provides a convenient interface for browsing and editing LGR's 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.

LGRs

- [View proposal-lao-lgr-02aug17-en](#) 
- [View proposal-lao-lgr-31jan17-en](#) 

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](#)

Your saved results

The following files contains your tools computation results.

 **Note that these files could be cleaned up regularly.**

- [Download 20171020_133848_diff_proposal-lao-lgr-31jan17-en_proposal-lao-lgr-02aug17-en.txt.gz](#) 

2. Click on the download link on the homepage to get the “Diff labels of two LGRs” result

Results

```
## Comparison on label 'ꠘꠗ' [U+0EC0 U+0EA5 U+0EBB U+0EC8 U+0EB2]
```

```
### Test dispositions: ###
```

```
***
```

```
No changes in disposition.
```

```
***
```

```
### Test number of variants: ###
```

```
***
```

```
New Primary in LGR2:
```

```
'ꠘꠗ' [U+0EC0 U+0EA7 U+0EBB U+0EC9 U+0EB2]
```

```
Rules for LGR2:
```

```
***
```

```
***
```

```
New Variant in LGR2:
```

```
'ꠘꠗ' [U+0EC0 U+0EA5 U+0EBB U+0EC9 U+0EB2]
```

```
Rules for LGR2:
```

```
***
```

```
***
```

```
New Variant in LGR2:
```

```
'ꠘꠗ' [U+0EC0 U+0EA7 U+0EBB U+0EC8 U+0EB2]
```

```
Rules for LGR2:
```

```
***
```

Specifies changes in disposition of allocated set labels by revising an LGR

Specifies new variants of allocated set labels formed by revising an LGR

Review Impact on Existing Labels by Introducing a New LGR

Agenda Item #8

Select Get Collisions in a List of Labels

🏠 LGR Tool / proposal-lao-lgr-31jan17-en - Proposed LGR for Lao

📁 Import 📄 New ⚙️ Tools ▾ ✓ Validate label 📄 Summary 👁 View XML 🌐 HTML Output 📄 Download

Code points Referen

↔ Compare two LGRs
↔ Diff labels of two LGRs
✂ Get collisions in a list of labels
☰ Generate disp. annotations

Expand range(s) Add code point(s)

Code point	Comments	Action
U+0E81 (᧑) 0 Variant(s)		See code point
U+0E82 (᧒) 0 Variant(s)	LAO LETTER KHO SUNG	See code point
U+0E84 (᧔) 0 Variant(s)	LAO LETTER KHO TAM	See code point
U+0E87 (᧗) 0 Variant(s)	LAO LETTER NGO	See code point
U+0E88 (᧘) 0 Variant(s)	LAO LETTER CO	See code point
U+0E8A (᧚) 0 Variant(s)	LAO LETTER SO TAM	See code point
U+0E8D (᧝) 0 Variant(s)	LAO LETTER NYO	See code point
U+0E94 (᧼) 0 Variant(s)	LAO LETTER DO	See code point
U+0E95 (᧽) 0 Variant(s)	LAO LETTER TO	See code point

Click on “Get collisions in a list of labels” button to determine label collisions from an existing file when a new LGR is introduced – for example, two unique labels become variants of each other

Enter Details

LGR Tool

Import New Tools About

LGR proposal-lao-lgr-31jan17-en
LGR to use for collisions

Labels Choose File test-labels.txt
List of labels to use in diff. File must be encoded in UNIX format.

As the computing may be very long, we will warn by e-mail once the result can be downloaded. Please provide a valid e-mail address:

E-mail hifza.khalid@icann.org
Provide your e-mail address

Full dump
Print a full dump

Output rules
Show rules in output (this can be very verbose)

sets.

Get collisions

English (en) Go

©

1. Select LGR

2. Select Labels file

3. Enter your email address where you want result link notification

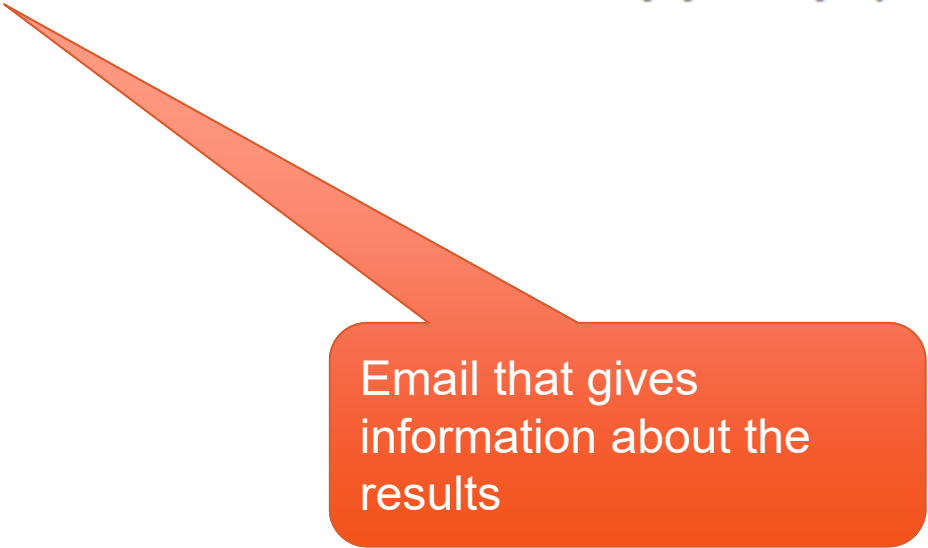
4. Check "Full Dump" to get summary of each operation done on the labels

5. Check "Output rules" to get rules that the label has gone through

6. Click on "Get collisions"

Email Notification

Hi,
The processing of collisions from labels provided in the file 'test-labels.txt' in LGR 'proposal-lao-lgr-31jan17-en' has been successfully completed. You should now be able to download it from your home screen under the name: '20171020_141758_collisions_proposal-lao-lgr-31jan17-en.txt.gz'. Please refresh the home page if you don't see the link.
Best regards



Email that gives information about the results

Download Results

🏠 LGR Tool

📁 Import 📄 New ⚙️ Tools ▾

★ About

Click on Home link for LGR Tool to get to this page

Welcome to the LGR (Label Generation Ruleset) Tool

This application provides a convenient interface for browsing and editing LGR's 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.

LGRs

- [View proposal-lao-lgr-31jan17-en](#) 🗑️

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](#)

Your saved results

The following files contains your tools computation results.

⚠️ Note that these files could be cleaned up regularly.

- [Download 20171020_141758_collisions_proposal-lao-lgr-31jan17-en.txt.gz](#) 🗑️

Click on the download link to get the "Collisions" results

Results

```
# Collisions #

## Collision ##
...
Label:          'ငါ့' | 'ငါ့'
Code points:    [U+0EC0 U+0EA5 U+0EBB U+0EC8 U+0EB2] | [U+0EC0 U+0EA7 U+0EBB U+0EC9 U+0EB2]
Category:      Primary | Primary
...

## Collision ##
...
Label:          'ငါ့' | 'ငါ့'
Code points:    [U+0EC0 U+0EA5 U+0EBB U+0EC8 U+0EB2] | [U+0EC0 U+0EA5 U+0EBB U+0EC9 U+0EB2]
Category:      Primary | Variant
...

### Details for label 'ငါ့' [U+0EC0 U+0EA5 U+0EBB U+0EC8 U+0EB2] ###
...
Variant 'ငါ့' [U+0EC0 U+0EA5 U+0EBB U+0EC9 U+0EB2]:
  Disposition: blocked
  Rules:
...

### Details for label 'ငါ့' [U+0EC0 U+0EA7 U+0EBB U+0EC9 U+0EB2] ###
...
Variant 'ငါ့' [U+0EC0 U+0EA5 U+0EBB U+0EC9 U+0EB2]:
  Disposition: blocked
  Rules:
...

```

Some of the allocated set labels have become variants of each other by introducing a new LGR

Details of a label and its corresponding variant(s)

Validate Multiple Labels using an LGR

Agenda Item #9

Select Generate Disposition Annotations

🏠 LGR Tool / proposal-lao-lgr-31jan17-en - Proposed LGR for Lao

Import New Tools Validate label Summary View XML HTML Output Download

Code points Refer

1. Click on "Import" to load an existing LGR file. See "Import" for details

2. Click on "Generate disp. annotations" button to validate labels given in a text file

Code point	Comments	Action
U+0E81 (᧑) 0 Variant(s)		See code point
U+0E82 (᧒) 0 Variant(s)	LAO LETTER KHO SUNG	See code point
U+0E84 (᧔) 0 Variant(s)	LAO LETTER KHO TAM	See code point
U+0E87 (᧗) 0 Variant(s)	LAO LETTER NGO	See code point
U+0E88 (᧘) 0 Variant(s)	LAO LETTER CO	See code point
U+0E8A (᧚) 0 Variant(s)	LAO LETTER SO TAM	See code point
U+0E8D (᧝) 0 Variant(s)	LAO LETTER NYO	See code point
U+0E94 (᧜) 0 Variant(s)	LAO LETTER DO	See code point
U+0E95 (᧝) 0 Variant(s)	LAO LETTER TO	See code point

Enter Details

🏠 LGR Tool / proposal-lao-lgr-31jan17-en - Proposed LGR for Lao

Import New Tools Validate label Summary View XML HTML Output Download

LGR proposal-lao-lgr-31jan17-en
LGR to use for annotation

Labels Choose File test-labels.txt
List of labels to use in diff. File must be encoded in UNIX format.

As the computing may be very long, we will warn by e-mail once the result can be downloaded. Please provide a valid e-mail address:

E-mail hifza.khalid@icann.org
Provide your e-mail address

English (en) Go

Annotate

1. Select LGR

2. Select Labels file for validation

3. Enter your email address where you want to receive results

4. Click on "Annotate" button

Email Notification

Hi,

The processing of annotation from labels provided in the file 'test-labels.txt' in LGR 'proposal-lao-lgr-31jan17-en' has been successfully completed. You should now be able to download it from your home screen under the name: '20171020_154110_annotation_proposal-lao-lgr-31jan17-en.txt.gz'. Please refresh the home page if you don't see the link.

Best regards



Email that gives information about the results

Download Results

🏠 LGR Tool

📁 Import 📄 New ⚙️ Tools ▾

★ About

Welcome to the LGR (Label Generation Ruleset) Tool

This application provides a convenient interface for browsing and editing LGR's 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.

LGRs

- [View proposal-lao-lgr-31jan17-en](#) 🗑️

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](#)

Your saved results

The following files contains your tools computation results.

⚠️ **Note that these files could be cleaned up regularly.**

- [Download 20171020_154110_annotation_proposal-lao-lgr-31jan17-en.txt.gz](#) 🗑️

Click on the download link to get the “Generate disp. annotations” results

Results

```
# The following labels from the set labels are invalid  
# End of filtered set labels
```

```
ໃນາະ: invalid  
ລາະ: invalid  
ກວ່າວ່າ: invalid  
ຢ໌: invalid  
ຢາງງງງ: invalid  
ເນາະ: valid  
ຢາງງງ: valid  
ຫຼຸດລົງ: valid  
ເດືອນຕູລາ: valid  
ສະຕ້ອກໂຮມ: valid  
ມ້ອນໄກເມີຮີ: valid  
ຮ້ອຍດູດ: valid  
ກວ່າເກົ່າ: valid  
ກວມເອົາ: valid  
ກະເປົາ: valid  
ການຄາດເດີນ: valid  
ການຈັບເອົາ: valid  
ການນໍາເຂົ້າ: valid  
ການຢາກເວົ້າ: valid  
ການລວມເອົາ: valid  
ການເຂົ້າ: valid  
ການເຂົ້າສ່ວນ: valid  
ການເຂົ້າຖົງ: valid  
ການເປັນເຈົ້າຂອງ: valid  
ການເລົ່າ: valid  
ການເວົ້າ: valid  
ການເອົາ: valid  
ການເອົາຂໍ້ມູນ: valid  
ການເອົາໃຈໃສ່: valid  
ສ້າງພະເຈົ້າ: valid  
ຄໍາເຊົາ: valid
```

Validation result of each label is written next to it

Validate Multiple Labels using a Merged LGR

Agenda Item #10

Select Generate Disposition Annotations

The screenshot shows the LGR Tool interface. At the top, there is a navigation bar with the title "LGR Tool / merged-lgr - Proposed LGR for Armenian|Proposed LGR for Lao|Thai Scrip...". Below the title is a toolbar with buttons for "Import", "New", "Tools", "Validate label", "Summary", "View XML", "HTML Output", and "Download". The "Tools" menu is open, showing options: "Compare two...", "Cross-script variants", and "Generate disp. annotations". An orange callout bubble points to the "Import" button with the text: "1. Click on 'Import' to load multiple existing LGR files. See 'Import' for details". Another orange callout bubble points to the "Generate disp. annotations" option with the text: "2. Click on 'Generate disp. annotations' button to validate labels given in a text file". Below the toolbar is a table with columns for "Code point", "Label", "Action", and "Disposition". The table lists several code points and their corresponding labels and dispositions.

Code point	Label	Disposition	Action
U+0067 (g) 2 Variant(s)		glyph	See code point
U+0068 (h) 3 Variant(s)	LATIN SMALL LETTER H	Cross-script homoglyph	See code point
U+006E (n) 2 Variant(s)	LATIN SMALL LETTER N	glyph	See code point
U+006F (o) 4 Variant(s)	LATIN SMALL LETTER O	glyph	See code point
U+0071 (q) 2 Variant(s)	LATIN SMALL LETTER Q	glyph	See code point
U+0075 (u) 2 Variant(s)	LATIN SMALL LETTER U	glyph	See code point
U+0269 (i) 3 Variant(s)	LATIN SMALL LETTER IOTA	Cross-script homoglyph	See code point
U+03B7 (n) 2 Variant(s)	GREEK SMALL LETTER ETA	Cross-script homoglyph	See code point
U+03B9 (i) 3 Variant(s)	GREEK SMALL LETTER IOTA	Cross-script homoglyph	See code point

Enter Details

🏠 LGR Tool / merged-lgr - Proposed LGR for Armenian|Proposed LGR for Lao|Thai Scrip...

📁 Import 📄 New ⚙️ Tools ▾ ✓ Validate label 📄 Summary 👁️ View XML 🌐 HTML Output 📄 Download

LGR merged-lgr
LGR to use for annotation

1. Select LGR

Allocated Set labels

Choose File No file chosen
Optional list of labels already allocated in the LGR set, that will be used to check for collisions

Optional file of allocated set labels to check for collisions

Script

- und-Armn
- ✓ und-Lao
- und-Thai

2. Select a script from the list of the scripts of different LGRs forming the merged LGR

Labels

Choose File test-labels.txt
List of labels to use in diff. File must be encoded in UNIX format.

3. Select Labels file for validation

As the computing may be very long, we will warn by e-mail once the result can be downloaded. Please provide a valid e-mail address:

E-mail hifza.khalid@icann.org
Provide your e-mail address

4. Enter your email address where you want to receive results

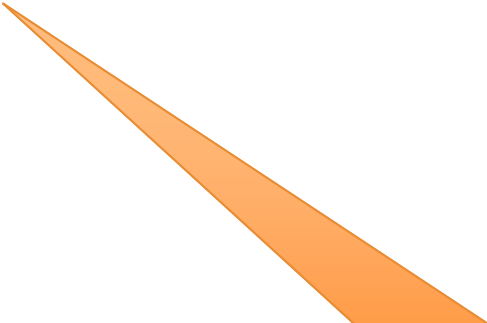
Annotate

5. Click on "Annotate" button

English (en) ▾ Go

Email Notification

Hi,
The processing of annotation from labels provided in the file 'test-labels.txt' in LGR set 'merged-lgr' with script 'proposal-lao-lgr-31jan17-en' has been successfully completed. You should now be able to download it from your home screen under the name: '20171020_160748_annotation_merged-lgr.txt.gz'. Please refresh the home page if you don't see the link.
Best regards



Email that gives information about the results

Download Results

🏠 LGR Tool

📁 Import

📄 New

⚙️ Tools ▾

★ About

Welcome to the LGR (Label Generation Ruleset) Tool

This application provides a convenient interface for browsing and editing LGR's 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.

LGR sets

- [View merged-lgr](#) ☰ 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.

Your saved results

The following files contains your tools computation results.

⚠️ **Note that these files could be cleaned up regularly.**

- [Download 20171020_160748_annotation_merged-lgr.txt.gz](#) 🗑️

Click on the download link to get the “Generate disp. annotations” results

Results

```
# The following labels from the set labels are invalid  
# End of filtered set labels
```

```
ໃນາະ: invalid  
ລາະ: invalid  
ກວ່າວ່າ: invalid  
ຢ໌: invalid  
ຢາງງງງ: invalid  
ເນາະ: valid  
ຢາງງງ: valid  
ຫຼຸດລົງ: valid  
ເດືອນຕູລາ: valid  
ສະຕ້ອກໂຮມ: valid  
ມ້ອນໄກເມີຮີ: valid  
ຮ້ອຍດູດ: valid  
ກວ່າເກົ່າ: valid  
ກວມເອົາ: valid  
ກະເປົາ: valid  
ການຄາດເດີນ: valid  
ການຈັບເອົາ: valid  
ການນໍາເຂົ້າ: valid  
ການຢາກເວົ້າ: valid  
ການລວມເອົາ: valid  
ການເຂົ້າ: valid  
ການເຂົ້າສ່ວນ: valid  
ການເຂົ້າຖົງ: valid  
ການເປັນເຈົ້າຂອງ: valid  
ການເລົ່າ: valid  
ການເວົ້າ: valid  
ການເອົາ: valid  
ການເອົາຂໍ້ມູນ: valid  
ການເອົາໃຈໃສ່: valid  
ສ້າງພະເຈົ້າ: valid  
ຄໍາເຊົາ: valid
```

Validation result of each label is written next to it

View Cross-script Variants of Labels

Agenda Item #11

Select Cross-script Variants

LGR Tool / merged-lgr - Proposed LGR for Armenian|Proposed LGR for Lao|Thai Scrip...

Import New Tools Validate label Summary View XML HTML Output Download

Code points Referer Compare Cross-script variants Generate disp. annotations

1. Click on "Import" to load multiple existing LGR files. See "Import" for details

2. Click on "Cross-script variants" button to view cross-script variants of labels given in a text file

Code point			Action
U+0067 (g) 2 Variant(s)			See code point
U+0068 (h) 3 Variant(s)	LATIN SMALL LETTER H	Cross-script homoglyph	See code point
U+006E (n) 2 Variant(s)	LATIN SMALL LETTER N	Cross-script homoglyph	See code point
U+006F (o) 4 Variant(s)	LATIN SMALL LETTER O		See code point
U+0071 (q) 2 Variant(s)	LATIN SMALL LETTER Q		See code point
U+0075 (u) 2 Variant(s)	LATIN SMALL LETTER U		See code point
U+0269 (i) 3 Variant(s)	LATIN SMALL LETTER IOTA	Cross-script homoglyph	See code point
U+03B7 (n) 2 Variant(s)	GREEK SMALL LETTER ETA	Cross-script homoglyph	See code point
U+03B9 (i) 3 Variant(s)	GREEK SMALL LETTER IOTA	Cross-script homoglyph	See code point
U+03BF (o) 4 Variant(s)	GREEK SMALL LETTER OMICRON	Cross-script homoglyph	See code point

Enter Details

The screenshot shows the 'LGR Tool' interface. At the top, there is a navigation bar with 'Import', 'New', 'Tools', and 'About' buttons. Below this, the 'LGR' field is set to 'merged-igr' with a dropdown arrow. The 'Labels' field shows a file named 'test-labels.txt' selected, with a 'Choose File' button. Below the labels field, there is a text input for an email address, which contains 'hifza.khalid@icann.org'. A blue 'Launch' button is positioned to the right of the email field. At the bottom left, there is a language selector set to 'English (en)' and a 'Go' button. Four orange callout boxes with white text and arrows point to the LGR field, the Labels file field, the email input field, and the Launch button.

LGR merged-igr
LGR to use for collisions

Labels Choose File test-labels.txt
List of labels to use in diff. File must be encoded in UNIX format.

As the computing may be very long, we will warn by e-mail once the result can be downloaded. Please provide a valid e-mail address:

E-mail hifza.khalid@icann.org
Provide your e-mail address


English (en) Go

Launch

©

Email Notification

Hi,
The processing of cross-script variants from labels provided in the file 'test-labels.txt' in LGR 'merged-lgr' has been successfully completed.
You should now be able to download it from your home screen under the name: '20171021_001353_cross_script_variants_merged-lgr.txt.gz'.
Please refresh the home page if you don't see the link.
Best regards



Email that gives information about the results

Download Results

LGR Tool

Import New Tools

About

Welcome to the LGR (Label Generation Ruleset) Tool

This application provides a convenient interface for browsing and editing LGR's 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.

LGR sets

- View merged-lgr 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.

Your saved results

The following files contains your tools computation results.

Note that these files could be cleaned up regularly.

- Download 20171021_001353_cross_script_variants_merged-lgr.txt.gz

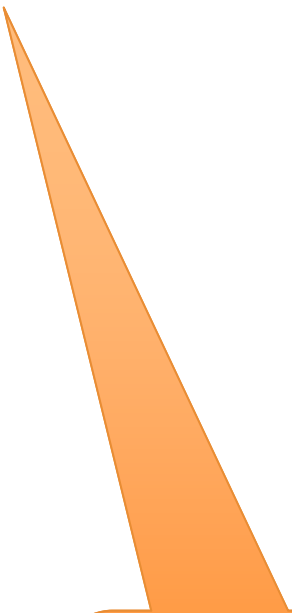
Click on the download link to get the "Cross-script variants" results

Results

```
Input label U+0574 U+0561 U+0575 U+0580 (dωjп) has cross-script variants:  
- Cross-variant U+0574 U+0448 U+0575 U+0580 (dωjп), disposition blocked:  
  + U+0448 (ш): Cyril  
Input label U+0578 U+0584 (np) has cross-script variants:  
- Cross-variant U+006E U+0584 (np), disposition blocked:  
  + U+006E (n): Latin
```



Cross-script variants of the label



Code points which are in other scripts than the one(s) defined in the LGR and the corresponding script



Disposition of cross-script variant

Compare LGRs

Agenda Item #12

Compare LGRs

🏠 LGR Tool / proposed-arabic-lgr-18092017-en - Proposed LGR for Arabic Script

📁 Import 📄 New ⚙️ Tools ▾ ✓ Validate label 📄 Summary 👁️ View XML 🌐 HTML Output 📄 Download

Code points Referen

↔️ Compare two LGRs
↔️ Diff labels of two LGRs
✂️ Get collisions in a list of lab
☰ Generate disp. annotations

Click on “Compare two LGRs” under the “Tools” tab for comparing LGRs

Expand range(s) Add code point(s)

Code point	Chara	Comments	Action
U+0620 () 0 Variant(s)	ARAB		See code point
U+0621 (◌) 0 Variant(s)	ARABIC LETTER HAMZA		See code point
U+0622 (◌) 4 Variant(s)	ARABIC LETTER ALEF WITH MADDA ABOVE		See code point
U+0623 (◌) 4 Variant(s)	ARABIC LETTER ALEF WITH HAMZA ABOVE		See code point
U+0624 (◌) 1 Variant(s)	ARABIC LETTER WAW WITH HAMZA ABOVE		See code point
U+0625 (◌) 4 Variant(s)	ARABIC LETTER ALEF WITH HAMZA BELOW		See code point
U+0626 (◌) 7 Variant(s)	ARABIC LETTER YEH WITH HAMZA ABOVE		See code point

Union, Intersection or Difference

The screenshot shows the LGR Tool interface for "proposed-arabic-lgr-18092017-en - Proposed LGR for Arabic Script". The interface includes a top navigation bar with buttons for "Import", "New", "Tools", "Validate label", "Summary", "View XML", "HTML Output", and "Download". Below this, there are three main input sections: "First LGR" (set to "proposed-arabic-lgr-18092016-en"), "Second LGR" (set to "proposed-arabic-lgr-18092017-en"), and "Action to perform on LGRs" (with a dropdown menu showing "Union", "Intersection", and "Diff" selected). A "Compare" button is located to the right of the dropdown. A note at the bottom left states: "Note that comparison will be performed only between two LGR sets." At the bottom left, there is a language selector set to "English (en)" and a "Go" button. Four orange callout boxes provide instructions: 1. Select first LGR, 2. Select second LGR, 3. Select "Union", "Intersection" or "Diff" to perform relevant function on the two LGRs, and 4. Click on the "Compare" button.

1. Select first LGR

2. Select second LGR

3. Select "Union", "Intersection" or "Diff" to perform relevant function on the two LGRs

4. Click on the "Compare" button

With Merged LGRs, Only Difference

The screenshot shows the LGR Tool interface with the following elements:

- Navigation Bar:** Home icon, "LGR Tool / merged-lgr-2 - LGR Proposed for Ethiopic Script|Proposed LGR for Lao|Tha...", and buttons for "Import", "New", "Tools", "Validate label", "Summary", "View XML", "HTML Output", and "Download".
- First LGR:** Input field containing "merged-lgr-1" with a callout box: "1. Select first LGR".
- Second LGR:** Input field containing "merged-lgr-2" with a callout box: "2. Select second LGR".
- Action to perform on LGRs:** Dropdown menu showing "Diff" with a callout box: "'Diff' selected by default".
- Compare Button:** A blue button labeled "Compare" with a callout box: "3. Click on the 'Compare' button".
- Note:** A warning icon and text: "Note that comparison will be performed only between two LGR sets or two simple LGRs, union and intersection are not available for LGR sets."
- Language Selection:** "English (en)" dropdown and "Go" button.

Result of Difference function

🏠 LGR Tool / proposed-arabic-lgr-18092017-en - Proposed LGR for Arabic Script

📁 Import

📄 New

⚙️ Tools ▾

✓ Validate label

📄 Summary

👁️ View XML

🌐 HTML Output

⬇️ Download

Result of diff of proposed-arabic-lgr-18092016-en with proposed-arabic-lgr-18092017-en

** Compare Metadata **

Compare Version

Same version value for both LGR: '1'.

Same version comment value for both LGR: 'Proposed LGR for Arabic Script'.

Compare Description

Same description type value for both LGR: 'text/html'.

Same description value for both LGR: '

<h1>Label Generation Rules for Arabic</h1>

<h2>Overview</h2>

<p>For more details on this proposal see

"https://www.icann.org/en/system/files/files/arabic-lgr-proposal_18nov15_en.pdf" - TE AFDN - 2015 - November 15

<h2>Repertoire and references</h2>

<p>According to Section 3.2 "Code point repertoire included",

<p>Reference values ("ref" attribute) from 0 to 14 refer to U...
corresponding code points were initially encoded. Reference v...
justifying the inclusion of the corresponding code points. Si...
multiple source reference values.</p>

<p>Reference values ("ref" attribute) from 100 and up refer...
corresponding code points in the "Proposal for Arabic Script R...

<p>The "tag" attribute for each code point or range indicates the script or scripts that the code
point is used with, using Unicode script identifiers preceded by "sc:".</p>

Differences of two LGRs

Important Note: These operations only provide provisional results which must be manually reviewed and finalized

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](https://plus.google.com/icann)



facebook.com/icannorg



weibo.com/ICANNorg



linkedin.com/company/icann



flickr.com/photos/icann



youtube.com/user/icannnews



slideshare.net/icannpresentations