LGR Toolset (beta) User Guide
IDN Program | October 2016
Label Generation Rulesets (LGRs) specify code point repertoire, variant rules and Whole Label Evaluation (WLE) rules, in addition to meta-data, to generated 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 allows for the following:
- Create a LGR
- Use a LGR to validate a label and determine its variants
- Manage LGRs, by comparing or combining them
- Review possible impact of a new or revised LGR on existing labels

For further details, visit the [LGR Toolset webpage](http://icann.org/idn) or [www.icann.org/idn](http://www.icann.org/idn)
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

○ For queries or feedback
  ○ Email to IDNProgram@icann.org
Agenda

1. Importing or loading an existing LGR in a toolset
2. Creating an LGR via toolset
3. Validating labels using an LGR
4. Summarizing LGR
5. Viewing LGR as XML
6. Comparing LGRs
Agenda

7  Review of impact on existing labels by revision of existing LGR

8  Review of impact on existing labels by introduction of new LGR

9  Validating multiple labels
Agenda Details

- **Import or Load LGR**
  - Import LGR
  - Select LGR with Validating Repertoire
  - Imported 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

- **Validate Label**

- **Summarize LGR**
Agenda Details

- View LGR as XML

- Compare LGRs
  - Compare LGRs
  - Union, Intersection or Diff(ERENCE)

- Review of Impact on Existing Labels by Revision of Existing LGR
  - Select “Diff”
  - Enter Details
  - Email Notification
  - Download Results

- Review of Impact on Existing Labels by Introduction of New LGR
  - Select “Collisions”
  - Enter Details
  - Email Notification
  - Download Results
Agenda Details

- **Validate Multiple Labels**
  - Select “Annotate”
  - Enter Details
  - Email Notification
  - Download Results
Import or Load LGR
Agenda Item #1
Import or Load LGR

To start by using an existing LGR file in XML format, click on the “Import” button.
Select LGR with Validating Repertoire

1. To import or load an existing LGR in XML file click on “Choose File”

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

3. Click on the “Import” button
The screen looks like this after successful import of existing LGR file in XML format:

<table>
<thead>
<tr>
<th>Code point</th>
<th>Character Name</th>
<th>Comments</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>U+0E81 (ฅ)</td>
<td>LAO LETTER KO</td>
<td></td>
<td>See code point</td>
</tr>
<tr>
<td>U+0E82 (囱)</td>
<td>LAO LETTER KHO SUNG</td>
<td></td>
<td>See code point</td>
</tr>
<tr>
<td>U+0E84 (ฅ)</td>
<td>LAO LETTER KHO TAM</td>
<td></td>
<td>See code point</td>
</tr>
<tr>
<td>U+0E87 (ฅ)</td>
<td>LAO LETTER NGO</td>
<td></td>
<td>See code point</td>
</tr>
<tr>
<td>U+0E88 (ฅ)</td>
<td>LAO LETTER CO</td>
<td></td>
<td>See code point</td>
</tr>
<tr>
<td>U+0E8A (ฅ)</td>
<td>LAO LETTER SO TAM</td>
<td></td>
<td>See code point</td>
</tr>
</tbody>
</table>
Create LGR
Agenda Item #2
To start by creating an LGR file in XML format, click on the “New” button.
Enter LGR Details

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

1. Write the code point to be added

2. Click on the “Add Code Point” button
Add Code Points

1. Add the first code point of the range
2. Add the last code point of the range
3. Click on “Next” button
Add Code Points

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

<table>
<thead>
<tr>
<th>Code point</th>
<th>Character Name</th>
<th>Comments</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>U+1780 () 0 Variant(s)</td>
<td>KHMER LETTER KAI</td>
<td>See code point</td>
<td></td>
</tr>
<tr>
<td>U+1781 () … U+179A (𝑖)</td>
<td>KHMER LETTER KHA … KHMER LETTER RO</td>
<td>See code point</td>
<td>Expand range</td>
</tr>
<tr>
<td>U+17B0 () 0 Variant(s)</td>
<td>KHMER INDEPENDENT VOWEL QAI</td>
<td>See code point</td>
<td></td>
</tr>
<tr>
<td>U+17B1 () 0 Variant(s)</td>
<td>KHMER INDEPENDENT VOWEL QOO TYPE ONE</td>
<td>See code point</td>
<td></td>
</tr>
<tr>
<td>U+17B3 () 0 Variant(s)</td>
<td>KHMER INDEPENDENT VOWEL QAU</td>
<td>See code point</td>
<td></td>
</tr>
</tbody>
</table>

To expand all the code points in this range, click on “Expand range”.

To expand all the code points in all the ranges of the LGR, click on “Expand range(s)”.
Add References

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

“Existing references” tab shows the added references

<table>
<thead>
<tr>
<th>Reference id</th>
<th>Description</th>
<th>URL</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>PRIMARY SCHOOL GRADE 1, MOEYS, ISBN 9-789-995-001-674, Publication</td>
<td>Any code point cited are for diacritics</td>
<td></td>
</tr>
</tbody>
</table>

Save

New reference

<table>
<thead>
<tr>
<th>Description</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>PRIMARY SCHOOL GRADE 1, MOEYS, ISBN 9-789-995-001-674, Publication</td>
<td>Any code point cited are for diacritics</td>
</tr>
</tbody>
</table>

Add
Define Meta Data

Click on “Meta data” tab to add meta information about the LGR.

The compulsory items have been filled in this picture whereas optional items have been left empty.

<table>
<thead>
<tr>
<th>Code points</th>
<th>References</th>
<th>Meta data</th>
<th>Rules</th>
</tr>
</thead>
<tbody>
<tr>
<td>Version</td>
<td></td>
<td>Version</td>
<td></td>
</tr>
<tr>
<td>Version comment</td>
<td></td>
<td>comment</td>
<td></td>
</tr>
<tr>
<td>Date</td>
<td></td>
<td>Date</td>
<td></td>
</tr>
<tr>
<td>Language</td>
<td></td>
<td>Language</td>
<td></td>
</tr>
<tr>
<td>Scope</td>
<td></td>
<td>Scope</td>
<td></td>
</tr>
<tr>
<td>Scope type</td>
<td></td>
<td>Scope type</td>
<td></td>
</tr>
<tr>
<td>Validity start</td>
<td></td>
<td>Validity</td>
<td></td>
</tr>
<tr>
<td>Validity end</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Version: 1
Date: 2016-08-16
Language: und-Khmr
Scope type: domain
Define Meta Data

Second half of the “Meta data” tab

- 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 Link]”</p>`
  - `<h2>Repertoire</h2>`
  - `<p>According to Section 5 "Repertoire", in "Proposal for a Khmer Script Root Zone LGR".</p>`
  - `<h2>Variant</h2>`
  - `<p>According to Section 6 "Variants", in "Proposal for Khmer Script Root Zone LGR", two Khmer consonants are variants of each other in their subscript form only. This is captured by a variant relation between the two subscript sequences, which are listed explicitly in the repertoire.</p>`
  - `<h2>Character Classes</h2>`
  - `<p>TBD: Some consonants are "base only", these do not take subscript form. The character U+17C9 KHMER SIGN MUUSIKATOAN is used</p>`

- Description type: `text/html`
- Validating repertoire: `msr-2-wle-rules-13apr15-en`

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

- 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

1. Add classes in the relevant box

2. Click on the “Save” button
Define Rules

1. Add the rule in the relevant box

2. Click on the “Save” button
Define Actions

1. Add action in the relevant box

2. Click on the “Save” button

<anchor/>

</rule>

Actions

<action disp="blocked" any-variant="blocked" /></actions>
**Define Code Point Properties**

Click on the “See code point” button to add code point details/properties.

<table>
<thead>
<tr>
<th>Code point</th>
<th>Character Name</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>U+1780 (ី) 0 Variant(s)</td>
<td>KHMER LETTER KA</td>
<td>See code point</td>
</tr>
<tr>
<td>U+1781 (២) ... U+179A (៣)</td>
<td>KHMER LETTER KHA ... KHMER LETTER RO</td>
<td>See code point, Expand range</td>
</tr>
<tr>
<td>U+17B0 (៤) 0 Variant(s)</td>
<td>KHMER INDEPENDENT VOWEL QAI</td>
<td>See code point</td>
</tr>
<tr>
<td>U+17B1 (៥) 0 Variant(s)</td>
<td>KHMER INDEPENDENT VOWEL QOO TYPE ONE</td>
<td>See code point</td>
</tr>
<tr>
<td>U+17B3 (៦) 0 Variant(s)</td>
<td>KHMER INDEPENDENT VOWEL QAU</td>
<td>See code point</td>
</tr>
</tbody>
</table>
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
7. Click on “Save variants, tags, context rules and comment” button

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

To delete code point & its details, click on the “Delete code point” button
Click on “Download” button to download the created LGR. Always download before closing the browser as the LGRs are not saved on the server.
Validate Label
Agenda Item #3
Validate Label

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

2. Always view “Summary” as the tool checks the loaded LGR during this process.

3. For validating labels, click on “Validate label” button.
Validate Label

1. Enter the label to be validated

2. Click on the “Validate” button
Result given by “Validate label” shows whether the label is valid and also lists its variants and their dispositions. If the variant label file is too large, the results are not displayed by only available through download.

Results can also be downloaded.
Summarize LGR
Agenda Item #4
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.

<table>
<thead>
<tr>
<th>Code point</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>U+0E81 ( فلا ) 0 Variant(s)</td>
<td>See code point</td>
</tr>
<tr>
<td>U+0E82 ( فلا ) 0 Variant(s)</td>
<td>See code point</td>
</tr>
<tr>
<td>U+0E84 ( فلا ) 0 Variant(s)</td>
<td>See code point</td>
</tr>
<tr>
<td>U+0E87 ( فلا ) 0 Variant(s)</td>
<td>See code point</td>
</tr>
<tr>
<td>U+0E88 ( فلا ) 0 Variant(s)</td>
<td>See code point</td>
</tr>
<tr>
<td>U+0E8A ( فلا ) 0 Variant(s)</td>
<td>See code point</td>
</tr>
</tbody>
</table>
Summarize LGR

LGR Summary

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 'proposed-lgr-lao-script-20160921' with Unicode version 6.3.0 and Validating Repertoire 'msr-2-wle-rules-13apr15-en'
Rebuilding LGR 'proposed-lgr-lao-script-20160921' done

General summary:
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.
View LGR as XML
Agenda Item #5
View LGR as XML

Click on “View XML” to get an XML view of the LGR

<table>
<thead>
<tr>
<th>Code point</th>
<th>Character Name</th>
<th>Comments</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>U+0620 ()</td>
<td>ARABIC LETTER KASHMIRI YEH</td>
<td></td>
<td>See code point</td>
</tr>
<tr>
<td>U+0621 ( rampant mark)</td>
<td>ARABIC LETTER HAMZA</td>
<td></td>
<td>See code point</td>
</tr>
<tr>
<td>U+0622 ( ﾟ)</td>
<td>ARABIC LETTER ALEF WITH MADDA ABOVE</td>
<td></td>
<td>See code point</td>
</tr>
<tr>
<td>U+0623 ( ﾦ)</td>
<td>ARABIC LETTER ALEF WITH HAMZA ABOVE</td>
<td></td>
<td>See code point</td>
</tr>
<tr>
<td>U+0624 ( ﾨ)</td>
<td>ARABIC LETTER WAW WITH HAMZA ABOVE</td>
<td></td>
<td>See code point</td>
</tr>
<tr>
<td>U+0625 ( ﾤ)</td>
<td>ARABIC LETTER ALEF WITH HAMZA BELOW</td>
<td></td>
<td>See code point</td>
</tr>
<tr>
<td>U+0626 ( ﾪ)</td>
<td>ARABIC LETTER YEH WITH HAMZA ABOVE</td>
<td></td>
<td>See code point</td>
</tr>
<tr>
<td>U+0627 ( ﾭ)</td>
<td>ARABIC LETTER ALEF</td>
<td></td>
<td>See code point</td>
</tr>
</tbody>
</table>
View LGR as XML

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

```xml
<lr xmlns="urn:ietf:params:xml:ns:lgr-1.0"> 
<meta> 
  <version comment="Root Zone LGR">1</version> 
  <date>2016-02-24</date> 
  <unicode-version>6.3.0</unicode-version> 
  <language>und-Arab</language> 
  <scope type="domain"></scope> 
  <description type="text/html"> 
    <![CDATA[ 
      <h1>Root Zone Label Generation Ruleset for the Arabic Script</h1> 
      <h2>Overview</h2> 
      <p>This document specifies an element LGR for a specific script that forms part of an integrated set of Label Generation Rules for the Root Zone. 
    ]]></p> 
  </description> 
</meta> 
<lr> 
  or more details on the Root Zone LGR and their development see "Root Zone Label Generation Rules - LGR-1: Overview and Summary", Integration Panel, 24 February 2016 [LGR-1].
  For more details on this element LGR and its development see TF-AI DN, "Proposal for Arabic Script Root Zone LGR", Version 3.4, 2015 November 18 [Proposal].
  The repertoire for this element LGR for the Arabic script is based on Section 3.2 in [Proposal] and only includes code points used
</lr> 
</lr>
```

- 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].

  - This LGR does not include combining marks or code point sequences. All combining marks have been excluded for these reasons:

    - First, they can significantly overproduce and would require additional rules to contain them effectively, complicating the design.

    - Second, even where they are required for some languages, they are optional for others.

    - Third, this also circumvents the issue raised by [IA8].

  - As part of the Root Zone, this LGR includes neither digits nor the HYPHEN-MINUS.

  - For further details, see Section 3.2 "Code point repertoire included", in [Proposal].

  - Each code point or range is tagged with the script or scripts that the code point is used with, and one or more references documenting sufficient justification for inclusion in the repertoire, see "References" below.

- Variants

  - This LGR includes "blocked" and "allocatable" variants, assigned according to Section 4 "Final recommendation of variants for Top Level Domains (TLDs)" in [Proposal]. These recommendations balance the desire to minimize the number...
Compare LGR
Agenda Item #6
Click on “Compare” under the “Tools” tab for comparing LGRs.
### Union, Intersection or Diff(erence)

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

<table>
<thead>
<tr>
<th>First LGR</th>
<th>proposed-lgr-laoscript-20160805</th>
</tr>
</thead>
<tbody>
<tr>
<td>Second LGR</td>
<td>proposed-lgr-laoscript-20160921</td>
</tr>
<tr>
<td>Action to perform on LGRs</td>
<td>Union</td>
</tr>
</tbody>
</table>

[Image of LGR Editor interface with annotations:](image-url)
Important Note: These operations only provide provisional results which must be manually reviewed and finalized.

Click on the “Download” button to get the result.
** Compare Metadata **

Compare Version
Same version value for both LGR: '2'.
Same version comment value for both LGR: 'Proposed LGR for Lao'.

Compare Description
Same description type value for both LGR: 'text/html'.
Same description value for both LGR: 'Label Generation Rules for LAO'

For more details on this proposal see "Proposal for a Lao Script Root Zone LGR [Proposal]."

The 51 code points are according to Section 5 “Repertoire” in [Proposal]. In addition, the sequence 0EB2 0EB0 has been defined.

This LGR defines no variants.

Some consonants have been given the tag of C2 and Cf. C2 is a set of those consonants that are used in main consonant clusters.

Whole Label Evaluation (WLE) rules

The LGR includes the set of required default WLE rules and actions applicable to the Root Zone.

Rules provided in the LGR as described in Section 7 of [Proposal] reasonably restrict labels so that they conform to Lao syllable structure. Where possible these constraints are presented as context rules.
Review of Impact on Existing Labels by Revision of Existing LGR
Agenda Item #7
Click on “Diff” button to determine label collisions caused by changing a LGR with a later version.
Enter Details

1. Select first LGR
2. Select second LGR
3. Select file containing labels
4. Enter your email address where you want to receive results
5. Check collisions if you want to check label collisions as well
6. Check “Output rules” if you want to check output rules for each label
7. Click on “Get diff” button
Email Notification

Hi,
The processing of diff from labels provided in the attached file 'Test File.txt' between LGR 'proposed-lgr-laoscript-20160927' and LGR 'proposed-lgr-laoscript-20160921' has been successfully completed.
You should now be able to download it from your home screen under the name: 'diff_proposed-lgr-laoscript-20160927_proposed-lgr-laoscript-20160921_20160928_075526.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 Editor to get to this page

2. Click on the download link on the homepage to get the “Diff” results
Review of Impact on Existing Labels by Introduction of New LGR

Agenda Item #8
Click on “Collisions” button to determine label collisions from an existing file if an LGR is introduced – for example, two unique labels become variants of each other.
Enter Details

1. Select LGR
2. Select label 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”
Hi,
The processing of collisions from labels provided in the attached file 'Test File.txt' in LGR 'proposed-lgr-laoscript-20160927' has been successfully completed.
You should now be able to download it from your home screen under the name: 'collisions_proposed-lgr-laoscript-20160927_20160928_082947.txt.gz'.
Please refresh the home page if you don't see the link.
Best regards
Download Results

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

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

- View proposed-lgr-laoscript-20160927

You may also use one of the following options:
- Import an existing XML file
- Start with a New blank XML file

Alternatively, you may select one of the built-in LGR’s below as a starting point.

Built-in LGRs

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.

Your saved results

The following files contains your tools computation results.

⚠ Note that these files could be cleaned up regularly.

- Download collisions_proposed-lgr-laoscript-20160927_20160928_082947.txt.gz
Validate Multiple Labels
Agenda Item #9
Select Annotate

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

2. Click on “Annotate” button to validate labels given in a text file
1. Select LGR
2. Select Label file for validation
3. Enter your email address where you want to receive results
4. Click on “Annotate” button
Hi,

The processing of annotation from labels provided in the attached file 'Test File.txt' in LGR 'proposed-lgr-laoscript-20160927' has been successfully completed.

You should now be able to download it from your home screen under the name: 'annotation_proposed-lgr-laoscript-20160927_20160928_083613.txt.gz'.

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

Best regards

Email that gives information about the results
Welcome to the LGR (Label Generation Ruleset) Editor

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

To begin using this application, you may use one of the following options:

- Import an existing XML file
- Start with a New blank XML file

Alternatively, you may select one of the built-in LGRs below as a starting point.

Built-in LGRs

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.

Your saved results

The following files contain your tools computation results.

⚠️ Note that these files could be cleaned up regularly.

- Download annotation_proposed-lgr-laoscript-20160927_20160928_083613.txt.gz

Click on the download link to get the “Annotate” results.
## Thank You

We welcome feedback and queries
Email: IDNProgram@icann.org
Website: icann.org/idn

### Engage with ICANN

<table>
<thead>
<tr>
<th>Platform</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>Twitter</td>
<td>twitter.com/icann</td>
</tr>
<tr>
<td>Facebook</td>
<td>facebook.com/icannorg</td>
</tr>
<tr>
<td>YouTube</td>
<td>youtube.com/user/icannnews</td>
</tr>
<tr>
<td>LinkedIn</td>
<td>linkedin.com/company/icann</td>
</tr>
<tr>
<td>SoundCloud</td>
<td>soundcloud.com/icann</td>
</tr>
<tr>
<td>Weibo</td>
<td>weibo.com/ICANNorg</td>
</tr>
<tr>
<td>Flickr</td>
<td>flickr.com/photos/icann</td>
</tr>
<tr>
<td>SlideShare</td>
<td>slideshare.net/icannpresentations</td>
</tr>
</tbody>
</table>