ICANN 82 COMMUNITY FORUM

String Similarity Evaluation in the New gTLD Program: Next Round

IDN Program

Sarmad Hussain, IDN-UA Program Senior Director

ICANN 82
12 March 2025

Agenda

- Overview
- Update on the String Similarity Data Collection
- Update on the String Similarity Evaluation Tool
- Overall Timeline
- Q&A

Overview

- Based on the Sub Pro
  - Affirmation 24.1 states that strings must not be confusingly similar to an existing top-level domain or a Reserved Name.
  - Affirmation 24.2 affirms the visual standard for determining similarity.

- Based on the IDN EPDP Phase 1
  - Final Recommendation 4.1 identifies the scope of string comparisons (details in the next slide).
  - Final Recommendation 4.2 states that decision by the String Similarity Review Panel.
  - Final Recommendation 4.3 states that the scope of omission by String Similarity Review Panel based on guidelines.

Overview

- IDN EPDP Phase 1 Final Recommendation 4.1: an applied-for primary ASCII or IDN gTLD string and all of its allocatable variant label(s) must be compared against the scope shown in the table.

Categories for Comparison:		The applied-for gTLD string		
		Primary gTLD string	All allocatable variant string(s)	All blocked variant string(s)
• Existing gTLD • The gTLD string applied-for in the previous round(s) still in the process • Existing ccTLD • Requested IDN ccTLD • Other Applied-for gTLD • Blocked Name • Any two-Character ASCII	Primary String	Yes	Yes	Yes*
	All allocatable variant string(s)	Yes	Yes	Yes*
	All blocked variant string(s)	Yes*	Yes*	No

Overview

- Initial guidelines for conducting a String Similarity Review published for public comment.
- Due to the enhanced scope of String Similarity Evaluation to cover variant strings, a two-stage process is being considered.
  - Data-driven pre-screening.
    - Similarity data collection from script experts
    - String Similarity Evaluation Tool
  - Manual review conducted by an expert panel, using the input from the pre-screening.

Update on the String Similarity Data Collection

Michael Bauland, Knipp Medien und Kommunikation GmbH

Similarity Data Collection

- Overview
  - RZ-LGR Version 5 covers 25 scripts (soon 26 with Thaana script to be added)
  - All repertoires of all scripts need to get analysed
  - Delegate tasks for each script to experts
- Scope of Work
  - ASCII, compare to ASCII characters, small and capital (a-z, A-Z)
  - In-Script, compare all elements to each other
  - Cross-Script, compare to elements in repertoire of other scripts within the same family
  - Miscellaneous, Capitalised elements within script, Conjuncts or compounded characters, Simple shapes (e.g. circles “o”, lines “l”, curves “c”, “s”, “Ո“), Underlined versions

Similarity Categorization

- Categories

1. Identical or Near Identical (homograph) and already defined variant
2. Highly Confusable (strongly similar, or near homograph)
3. Similar
4. Distantly Similar (weakly similar)
5. Distinct (not similar)

Similarity Categorization

Category 1: Identical or Near Identical + Variants

- Variants:
  All existing (already defined) variants automatically belong to this category, independent of their visual appearance

- Identical:
  Cases of identical characters

- Near Identical:
  Easy to confuse even side-by-side

Similarity Categorization

Category 1: Identical or Near Identical + Variants

Examples

- Hebrew:
  - מוסך vs. מוסר
- Latin:
  - testi vs. testi̲
- Bengali:
  - রাজ vs. ৰাজ

Similarity Categorization

Category 2: Highly Confusable

- Confused by a large majority of attentive users, when not looking at cases side-by-side

Examples:

- Ethiopic:
  - ሀገሪ vs. ሀገሪ
- Latin:
  - for vs. for

Similarity Categorization

Category 3: Similar

- Confused by a simple majority of attentive users, when not looking at cases side-by-side
- Fallback Replacements (mainly Latin), e.g., instead of è (00E8) writing e (0065)

Examples:

- Latin:
  - québec vs. quebec
- Bengali:
  - ভাশুর vs. ভাণ্ডার

Similarity Categorization

Category 4: Distantly Similar

- Confused by a minority of attentive users, when not looking at cases side-by-side; or similar, but not really confusable
- Experts are not actively looking for all of these cases, but note them on corner cases

Examples:

- Hebrew:
  - מדגם vs. מרגם
- Ethiopic:
  - ዐይነ vs. ዐይነ

Similarity Categorization

Category 5: Distinct

- These will be the vast majority of all pairs. There is no need to look at them.

Status

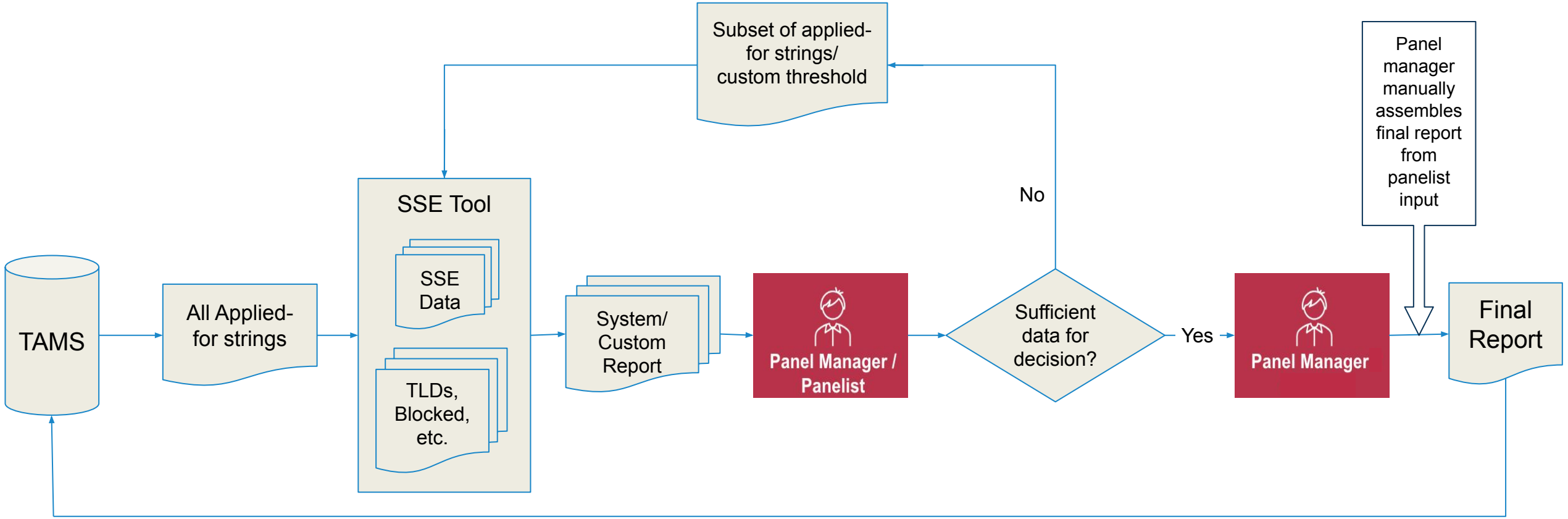
Status	# Scripts
Preparation	1
Expert working on initial data	1
Internal review in progress	4
Preparing for Public Comment	20

The 26 scripts includes: Arabic, Armenian, Bangla, Chinese (Han), Cyrillic, Devanagari, Ethiopic, Georgian, Greek, Gujarati, Gurmukhi, Hebrew, Japanese (Hiragana, Katakana, and Kanji [Han]), Kannada, Khmer, Korean (Hangul and Hanja [Han]), Lao, Latin, Malayalam, Myanmar, Oriya, Sinhala, Tamil, Thaana, Telugu, and Thai.

Update on the String Similarity Evaluation Tool

Julien Bernard, Cofomo Quebec Inc.

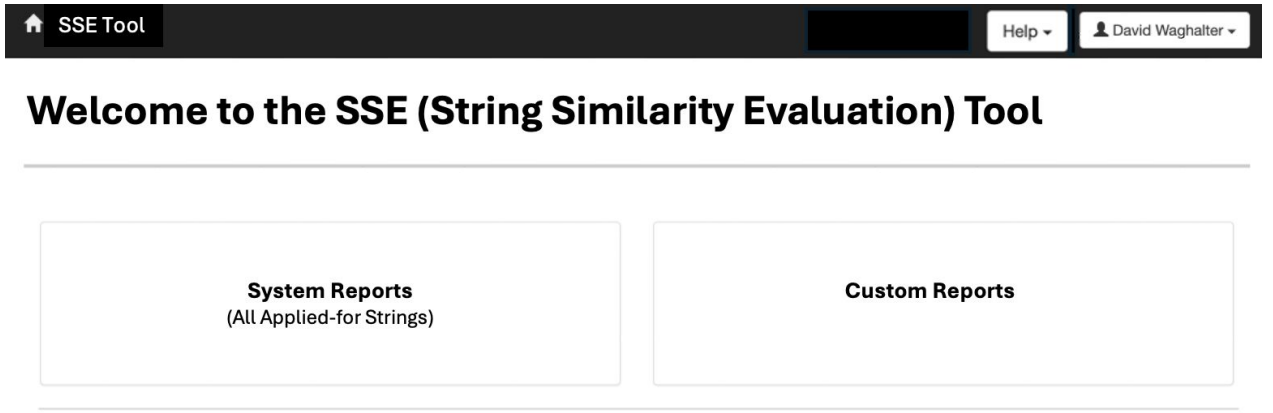
String Similarity Evaluation Process Flow



- Panelist, not tool, determines final decision on similarity
- Panelist can create custom report to evaluate subset of applied-for strings
- Panelist can manually edit outcome before providing to Panel manager to add to Final Report
- Once Final Report is uploaded to TLD Application Management System (TAMS), TAMS will handle determination (e.g., applied-for string similar to other applied-for string is in contention, similar to Blocked name is rejected, etc.)

Overview of the Tool

- Panelist with an ICANN account can access on ICANN.org
- Provides automated string similarity analysis intended to optimize manual review
- Incorporates Similarity Data comparison rules as described in previous slides



- Compares applied-for strings (including tool-generated variants) to:
  - All other applied-for strings
  - Existing gTLDs & ccTLDs
  - Requested IDN ccTLDs
  - Previous round, in-process applied-for gTLDs
  - Two-character ASCII
  - Blocked Names

- Tool functions
  - First pass: detect potentially confusable labels
  - Second pass: calculate a visual distance score
  - Labels with scores that exceed the threshold are included in the report for manual review.

Overview of the Tool (cont'd)

- Contention Threshold
  - Configurable value that determines comparison sensitivity
  - If similarity score between a pair of strings is below this value, the tool will report the variant set with the most similarity.
- Tool output for the panelists to review:
  - System Reports (all applied-for strings)
    - Generated using predetermined threshold values (details TBD)
  - Custom Reports (subset of applied-for strings)
    - Panelists can run on-demand Custom Reports on a subset of applied-for strings using custom thresholds
  - Reports will be available in CSV and HTML format

Mechanism

- Based on the similarity data files, calculate if any applied-for string (or its variants) is the same or similar to other in-process applied-for strings, existing TLDs, requested IDN ccTLDs, Two-character ASCII, or Blocked Names (or their variants).
  - Similar strings are grouped in potential contention sets
  - For each potential contention set, a score is computed between each string/variant pair
  - The report for the panelists to do the final review will contain the pair of potentially similar strings with a minimum score below a certain threshold
  - A metric to calculate the similarity score will be determined after data analysis

- Similarity Category: 5 levels of similarities between code points (as described in “String Similarity Data Collection” section, above)
    - Identical or Near Identical (homograph) and already defined variant
    - Highly Confusable (strongly similar, or near homograph)
    - Similar
    - Distantly Similar (weakly similar)
    - Distinct (not similar)

Example: Output Report

- Each contention set consists of one or more rows of comparison pairs (Label 1 & Label 2). Each row contains the following columns:
  - Contention Set #
  - Similarity ID (unique row ID)
  - Label 1 & A-Label (Label 1 is always an applied-for string)
  - Label 1 App ID
  - Label 2 & A-Label
  - Label 2 App ID
  - Label 2 Source (ccTLD, gTLD, applied-for, blocked, etc.)
  - Label in Label 1's variant set with the most similarity
  - Label in Label 2's variant set with the most similarity
  - Level 3 Similar (Y/N)
  - Score
  - Rationale
  - Reviewer
  - Notes

Example: Output Report

Contention Set #	Similarity ID (unique row ID)	Primary Label 1	Primary A-Label 1	Label 1 App ID	Label 2	A-Label 2	Label 2 App ID	Label 2 Source	Label in the Label 1's variant set that cause the most similarity	Label in the Label 2's variant set that cause the most similarity	Level 3 Similar (Y/N)	Score	Rationale	Reviewer	Notes
1	1	exarnple (1)	-	210	example	-	-	Blocked name	exarnple	example					
1	2	exampie (2)	-	221	example	-	-	Blocked name	exampie	example					
1	3	exarnple	-	210	exampie	-	221	Applied-for	exarnple	exampie					
1	4	exampie		221	exampìe		222	Applied-for	exampíe (3)	exampìe					

Theoretical example with applied for strings: exarnple, exampie and exampìe.

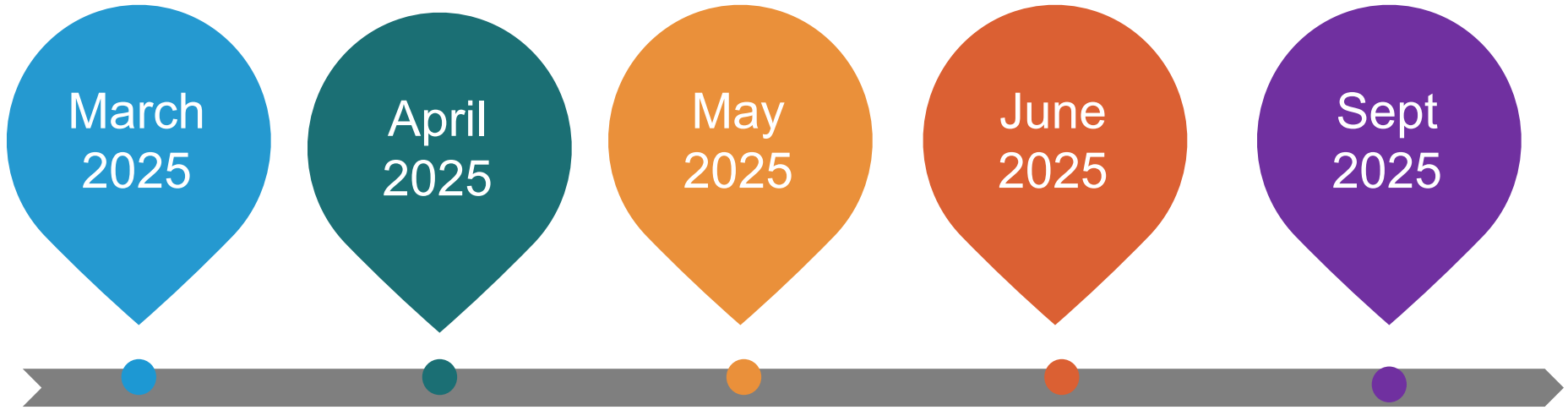
Similarities:

1. rn and m look similar
2. uppercase i (I) and l look similar
3. i has variant í, ì and í have a higher similarity score than ì and i. So exampie and exampìe are similar with the highest similarity score obtained with exampíe as a variant of exampie.

Timeline

David Waghalter, ICANN

Timeline of String Similarity Evaluation Data and Tool



Engage with ICANN – Thank You and Questions

One World, One Internet

Visit us at **icann.org**

@icann

facebook.com/icannorg

youtube.com/icannnews

flickr.com/icann

linkedin.com/company/icann

soundcloud.com/icann

instagram.com/icannorg