ICANN 78 ANNUAL GENERAL MEETING

ICANN 25 1998–2023

IANA Stewardship Transition 7 2016–2023

String Similarity Review Guidelines

Sarmad Hussain, Sr. Director IDN and UA Programs
Michel Suignard

ICANN79

5 March 2024

Contents

Contents (continued)

Overview

- Detailed guidelines for conducting a String Similarity Review
  - Scope: Top-Level Domains for which a risk of visual confusability could be established,
  - Background and criteria for the determination of various similarity factors,
  - Methodology for an efficient evaluation process,
  - Foundational input for a String Similarity Review Panel.
- Public Comment Review link:
  - https://www.icann.org/en/public-comment/proceeding/string-similarity-review-guidelines-07-02-2024
- String Similarity Review Guidelines
  - https://itp.cdn.icann.org/en/files/strategic-initiatives/string-similarity-review-guidelines-07-02-2024-en.pdf

Overview

- String similarity in the context of variants:
  - Assumption is that any blocked variant has been eliminated prior to this review,
    - Nevertheless evaluated against proposed labels.
  - String similarity includes visual similarities between ASCII strings (not part of RZ variants scope).
- Two-stage Process:
  - Data-driven pre-screening,
  - Review process undertaken by an expert panel.

Key Concepts

- Confusability
- Similarity
- Variants

Degrees of similarities

Category	Code Point	Script	Code Point	Script
Visually “same”	c (U+0063) c (U+0441)	Latin Cyrillic	ی (U+0649) ی (U+06CC)	Arabic Arabic
Visually similar	ç (U+00E7) ς (U+03C2)	Latin Greek	ঘ (U+0918) ঘ (U+0998)	Devanagari Bengali (Bangla)
Visually distinct	a (U+0061) f (U+0066)	Latin Latin	ペ (U+30DA) ホ (U+30DB)	Japanese Japanese

Confusability

- **Confusability:** the degree at which two labels can be substituted for each other without that fact being noticed. Users are confused about which label they are selecting.
- **Confusable:** said of two labels for which the degree of confusability exceeds a defined threshold.
  - This threshold is a matter of definition, no a-priori value,
  - Determination based on purpose (Root Zone domain names),
  - Labels can be short (one or two letters),
  - Include ASCII domains.

Similarity

- **String Similarity:** similarity in appearance, of a kind that contributes to the confusability between two labels, visual similarity.
  - Similarity is restricted to visual similarity in this context,
  - Impacted by User Interface Font Styles.

- **Confusable Similarity:** the degree of similarity required to make two similar labels “confusable”.
  - Depends on users’ background and context of presentation,
  - Subject to alternate forms such as italic style.

Name	Standard	Italics
Cyrillic small letter i	и	*и*
Latin small letter f	f	*f*
Latin small letter f with hook	ƒ	*ƒ*

Variants

- **Variant:** label accepted as “same as” the original label, not limited to visual similarity.
  - In Zone, mapping between single or multiple letters and across scripts,
  - Can apply not just visual appearance but also meaning or pronunciation,
  - Because it is an equivalence relation, variants are Symmetric and Transitive,
  - Disposition of variant mappings can be “blocked” or “allocatable”.
- **Original label:** label which the variant label is a variant of; also called Primary label.

Name	Code point	Glyph
Latin small letter f	0066	f
Latin small letter f with hook	0192	ƒ

Recognition Task and Context Dependence

- What looks similar and what can be confusable depends on the background of the user and expectations learned in reading the native language in the native script.
- Evaluating two labels for confusable similarity requires definition of a target user.

Target User and Recognition Context

- **Target User**: a typical user in terms of which confusability is defined. This user should:
  - Be natively familiar with script or language (of the intended label),
  - Have some familiarity with the ASCII subset (no matter the user's native script).
- **Recognition Context**: the context in which the Target User is asked to distinguish between the intended label and some alternative.
  - Assumption:
    - No requirement for simultaneous presentation,
    - Cross-script or cross-language scenarios,
    - No unfamiliar scripts,
    - Whole label, not character-by-character.

Recognition Considerations

- **Identifier Recognition vs. Reading**:
  - In reading primary task is to identify words in context to extract meaning,
  - Identifier recognition is related to spell checking,
- **Recognition applies to whole label:**
  - Word recognition is holistic,
  - Recognition does not proceed on a character-by-character basis.

Candidate Label and Review Targets

- **Candidate Label:** to be evaluated for confusable similarity against one or more other labels:
  - Already delegated labels and **their variants,**
  - Reserved words,
  - Other applied labels and their variants.
- **Review Target:** a label that is shortlisted for further review against the candidate label.
  - Typically pre-selected in a pre-screening process,
  - Each applied for labels is in turn considered the candidate label while others are treated temporarily as review targets.

Goal of String Similarity Review

- Ensuring that all labels are sufficiently distinct from each other,
  - By disallowing concurrent labels where either part or the entire label is substituted for a confusingly similar string,
  - Daunting to eliminate all strings that might possibly be confused by some readers ***some*** of the time,
  - Focus on substitutions where there is a strong probability of confusion **most** of the time.

Relation and Contrast To Variants

- How are they different?
- Where do they interact?

Comparing String Confusability with Variants

- Variant: a label accepted as “the same” as the original label.
  - Variant is an equivalence relation → symmetric and transitive:
- In contrast, “mere” similarity implies some distance in perception space,
  - Two labels similar to a third may not be similar to each other,
  - Similarity and confusability are not transitive.
- Variants not limited to visual similarity (unlike string confusability).
- Variants do not consider uppercase-based similarity,
  - But in scope for string similarity.
- Variants fall short in addressing confusability of whole strings.

Comparing Variant Set with Confusables Cluster

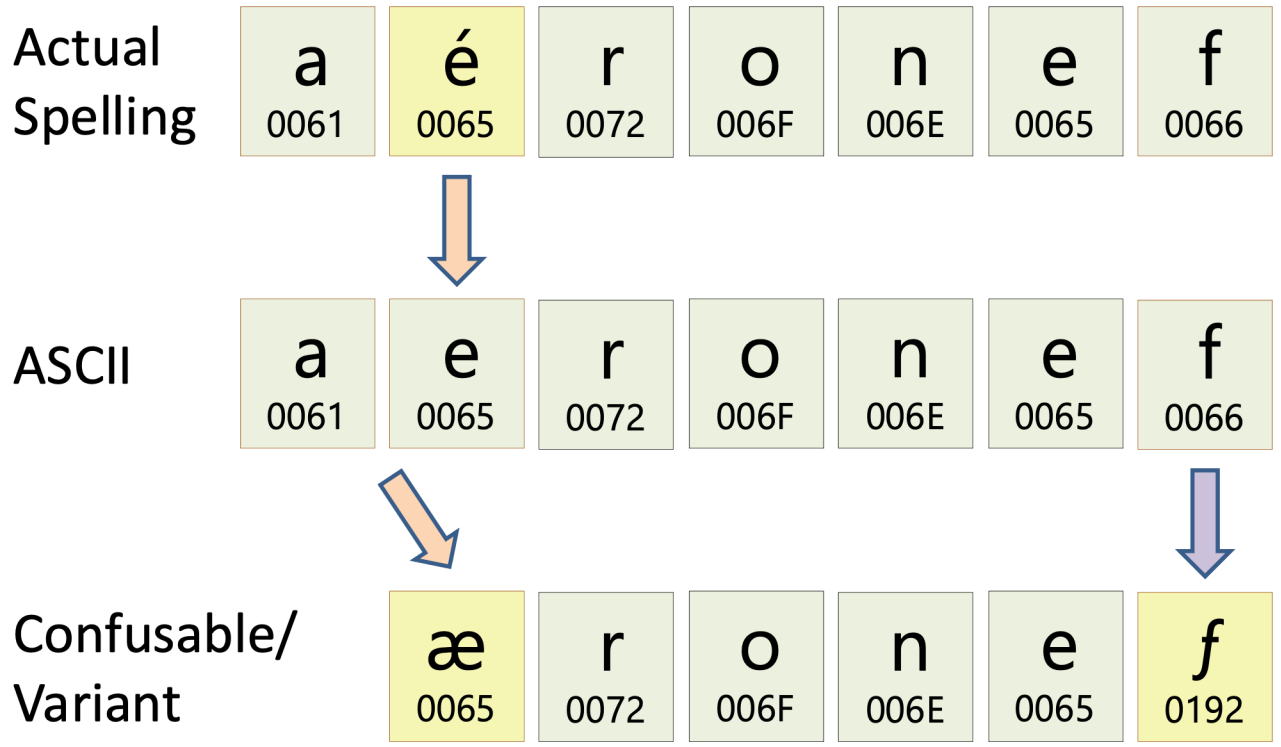
- **Variant set**: Collection of all variant labels or code points mutually variants of each other.
- **Confusables Cluster**: Confusables for a certain label cluster around it in perception space,
  - Applied to code points, represents all code points or sequences directly or by transitivity confusable with a given code point or sequence.
- A confusables cluster treats confusability as transitive,
  - Creates a superset that should include all confusable code points or labels,
  - But, may include non-confusable after more detailed review.

Interaction of String Similarity Review with Variants

- Some of the delegated labels may have allocatable variants also delegated,
  - Many more have many blocked variants.
- All variants of a Candidate Label may also be confusable with the delegated label, independent of whether applied for or not.
- A Candidate Label may be confusable with a variant of a delegated label, even if that variant is not delegated.
- A Candidate Label or its variant that is confusable with any variant of a delegated label should be evaluated like a Candidate Label that is confusable with the delegated label itself.

Labels Mixing Confusable and Variants

- Mixed cases need to be handled by String Similarity Review.



Types of Similarity

1. Identical or Near-Identical Labels
2. Highly Confusable Labels
3. Similar Labels
4. Distantly Similar Labels
5. Distinct Labels, not similar

EXAMPLE: “.cap” (Latin) *vs.* “.cap” (Cyrillic)

EXAMPLE: “.ρορ” (Greek) *vs.* “.pop” (Latin)

EXAMPLE: “.PAY” (Latin) *vs.* “.PAY” (Cyrillic)

EXAMPLE: 刀口爪丹工れ−𠂉丹乃モ𠂉

Simple Shapes

- Some characters have very simple glyph shapes giving little hint of script identity.

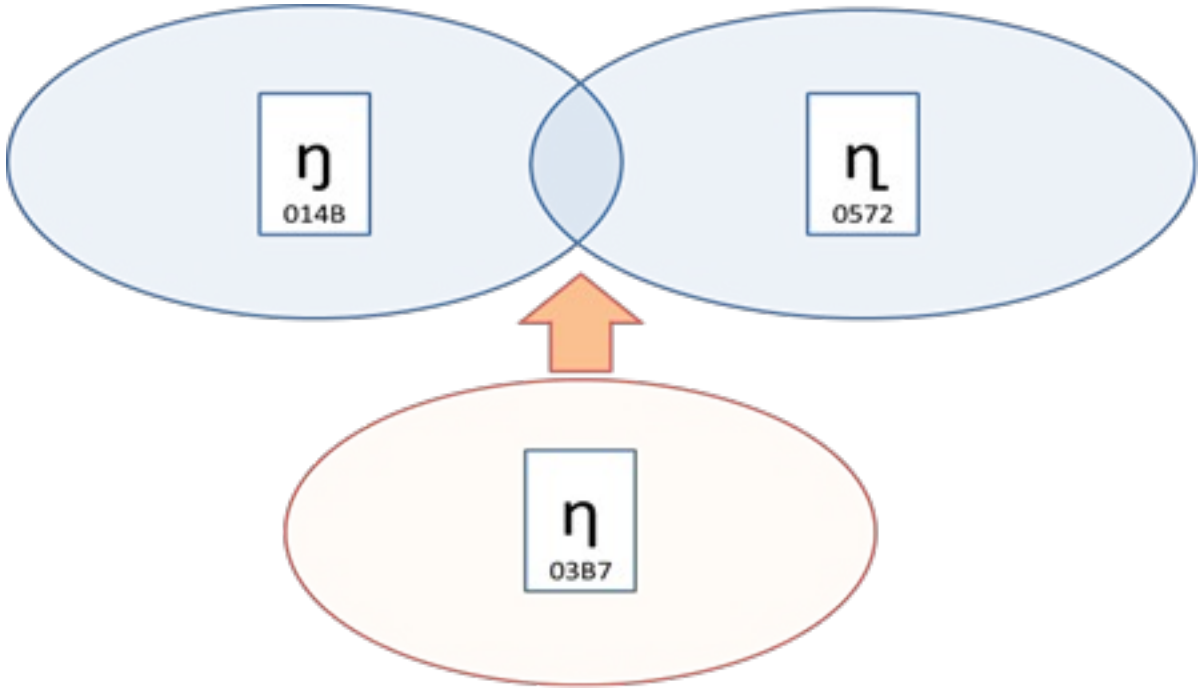
  Such as o, c, l, s, v, x.

  - U+006F (o) LATIN SMALL LETTER O
  - U+03BF (o) GREEK SMALL LETTER OMICRON
  - U+043E (o) CYRILLIC SMALL LETTER O
  - U+0585 (o) ARMENIAN SMALL LETTER OH
  - U+05E1 (ס) HEBREW LETTER SAMEKH
  - U+0B20 (ଠ) ORIYA LETTER TTHA
  - U+0D20 (ഠ) MALAYALAM LETTER TTHA
  - U+101D (ဝ) MYANMAR LETTER WA

- Some similar strings with simple shapes:
  - .coco Latin
  - .coco Cyrillic
  - .ᴄoᴄo Myanmar

Confusability and Proximity in Perception Space

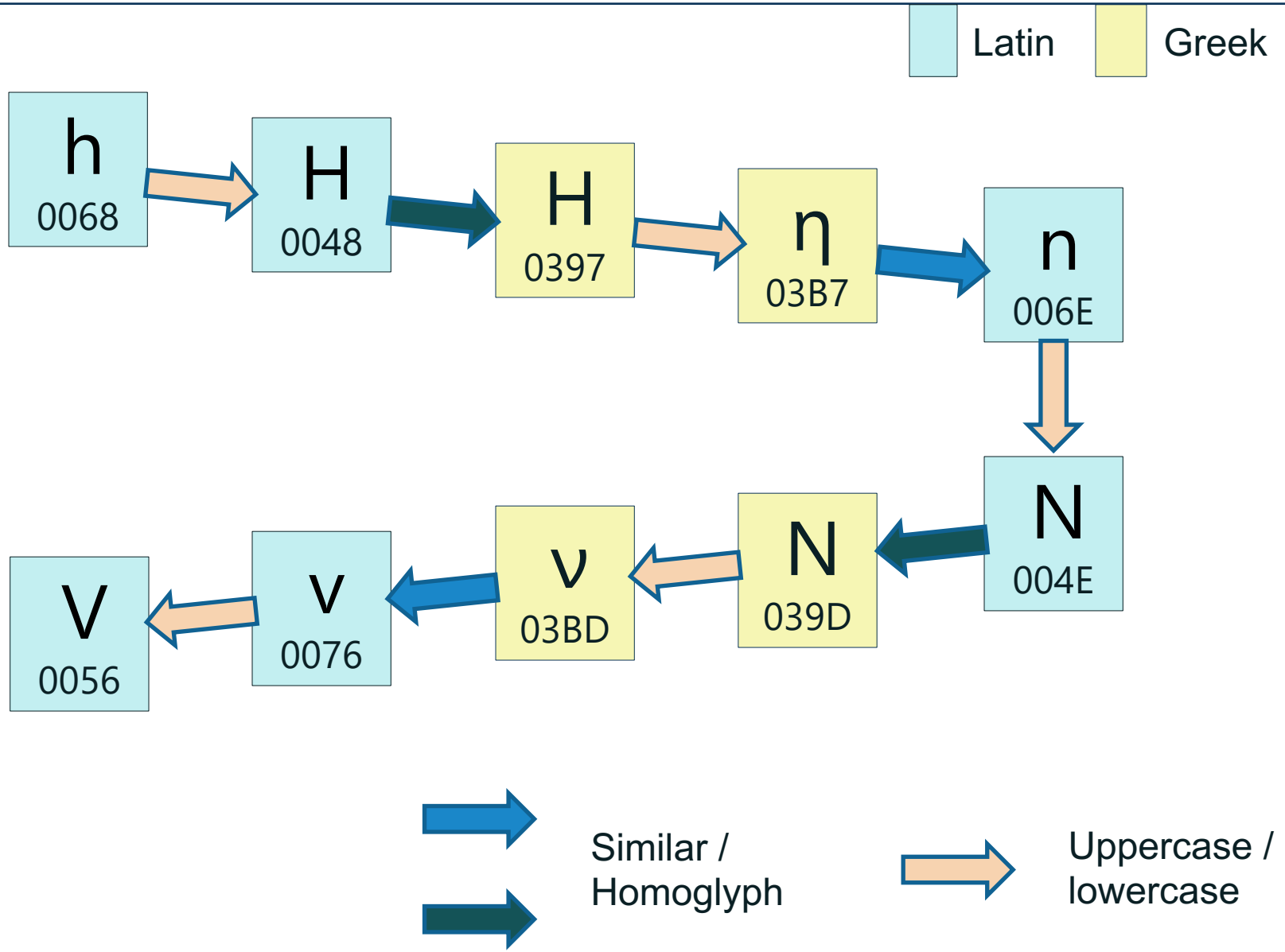
- Notionally, a label occupies a position in “perception space”, with labels that are similar to each other located in proximity.
  - Two labels that have identical appearance would coincide,
  - A threshold for confusability can then be considered as a radius around that position,
  - Where to draw these lines and how to estimate the distance between two labels is ultimately a judgment call.



Casing and Confusability

- IDN 2008 labels are lowercase, however, user agents will accept labels presented to the user as uppercase or mixed case.
  - Therefore, necessary to consider the uppercase and mixed case forms of labels,
  - Cannot be done using the standard variant specification, because transitivity would create undesirable mutual exclusion,
  - European scripts include many common uppercase shapes across scripts.
- Therefore, labels that can be mistaken for the uppercase form of an intended label should not be allowed concurrently.
- Design the process so more labels inside a given script can coexist,
  - As long as there are no labels delegated in the other script that could “bridge” them via combined upper- and lowercase mappings.

Example of Uppercase and Similarity



Confusable Sequences

- Accidental similarity such as ‘rn’ vs. ‘m’
- Related letters in sequences, such as ligature: æ,œ
- Repeated feature (when features are repeated and hard to detect)

  EXAMPLE: Myanmar script ကျွဲပင် *vs.* ကျွဲပင် (left is a typo, right is a name of a “giant reed”).

- Reordering

  EXAMPLE: Myanmar label “ဂO”, (U+101D (O) and U+1031 (ဂ◌) in that order) looks like “60” or “GO” .

Scripts and Similarity

- How is string similarity affected by the presence of multiple scripts?
- How does string similarity depend on a particular script?

Assumption on Script Guidelines

- Understanding of the typography and recognition of the script(s) by native readers.
- Native readers adept at interpreting a given string, resolving it into a word (a letter sequence).
- Recognition is highly efficient and follow patterns,
  - If a confusable alternative meets these expectations, it may be undetected,
  - Experience of the native readers matters.
- As a result, guidelines necessarily have a script-specific portion,
  - must be created in collaboration with native users of the scripts in question.

Script Dependence

- Similarity is in the eye of the beholder.
- Typographic and other features of the script influence the task of recognition of a label and discrimination from other labels.
- Some scripts have available data on suggested confusables (mostly provided by Script Generation Panel during the RZ-LGR creation process).

Script Families

- Some groups of scripts are closely related by history and features:
  - Latin/Greek/Cyrillic/Armenian: common origin, many common shapes.
  - Chinese, Japanese, Korean: overlapping repertoire, similar features.
  - Neo-Brahmi Scripts: common origin, similar structure, common shapes.
  - UI fonts (such as Segoe UI) show a tendency for harmonized glyphs.

Strongly Correlated Scripts

- Some script pairs show a particularly strong overlap in letter shapes:
  - Kannada/Telugu (34)
  - Latin/Cyrillic (28)
  - Devanagari/Gurmukhi (26).
- In contrast, Thai/Lao are strongly related, but typical font styles are distinct,
  - Latin/Greek share many uppercase forms,
    - ⇒ confusable if presented to the user in uppercase,
    - ⇒ LDH labels can look like Greek uppercase.

Weakly Correlated Scripts

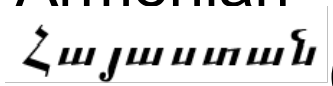
- These scripts have relatively small sets of shapes that are visually correlated; some of them may bear an intrinsic relation, while others may be accidental:
  - Han vs. Katakana/Hirgana (8)
  - Tamil/Malayalam (6)
  - Katakana/Hiragana (3)
  - Hangul/Han (3)
  - Devanagari/Bengalii (2)
- More Distant:
  - Hebrew/Latin and Greek (1 – accidental)
  - Myanmar/Georgian (1)
  - Malayalam/Latin (2)
- None
  - Arabic, Ethiopic, Gujarati, Khmer, Lao, Sinhala, Thai

Some Examples of Scripts Related Confusability

- Kannada/Telugu

  ಜಬ *vs.* ಜಬ (Kannada/Telugu)

- Armenian

   (traditional) *vs.* Հայաստան (modern, sans-serif)

- Greek

  αλφάβητο (traditional serif) *vs.* αλφάβητο (modern, sans-serif)

  Greek phi: φ *vs.* ϕ (alternate forms)

- Ethiopic/Armenian/Latin (accidental)

  ՈՈՈ (Ethiopic) *vs.* ՈՈՈ (Armenian)

  UUU (Ethiopic) *vs.* UUU (Latin) *vs.* UUU(Armenian)

Scalability of String Similarity Review

- Brute force approaches don’t scale.
- For many labels it is not possible to enumerate all variants.
- It is also not possible to compare all variants of a candidate label with all delegated labels and their variants.
- Scalability of the String Similarity as well as size of the registry matters.
- Possible approach – a two stage process:
  - Data driven pre-screening to rule out all unlikely pairs and reduce the number of labels to be verified,
  - Focused expert driven review of remaining candidates.

Size of the Target Registry

- Current Root Zone Database: 1590 entries (1422 ASCII and 168 IDN entries. (source: https://www.iana.org/domains/root/db)
  - Many variants (blocked or not), most from Latin to others (such as ‘aaa’, ‘aarp’, ‘ac’, ‘aco’, etc.), but also from other scripts to Latin (such as Cyrillic ‘opr’)
- Main sets: Latin (1424), Han-CJK (66), Arabic (36),
  - Others: Cyrillic (18), Kana (10), Devanagari (7), Hangul (5), Tamil (4), Bengali/Bangla (3), Greek (3), Hebrew (3), Thai(2)
  - Singletons: Kannada (1), Oriya/Odia (1), Telugu (1), Sinhala (1), Gujarati (1), Georgian (1), Lao (1), Malayalam (1), Gurmukhi (1), Armenian (1)
- Latin and even more ASCII subset is a focus point.
- Root Zone size should be small enough to support the strategies discussed below.

Pre-screening Implementation Hints

- Re-use some of the methodology from the variant computation using Index Label:

  Index (Variant) Label (or also "skeleton"): a label computed from any label such that all labels that are defined as variants of each other result in the same Index Variant, and no other label will result in the same Index Variant.

- Treat confusables as superset of variants to cover interactions.

- Leverage a modified variant mechanism to identify potentially confusable labels,
  - Score labels by suspected similarity based on letter similarities,
    - Dissimilar, Weak, Similar, Strong, Identical.

  ⇒ May include some that shouldn't be considered confusable.

What Kind of Data is Available?

- Some data is available for L/G/C/A scripts (single code points):
  - Byproduct of Root Zone LGR development.
  - Additional data from registry policies.
- Unicode confusables data:
  - Current status not consistently maintained.
  - Future direction being reviewed.
  - Need vetting and alignment with Root Zone.
- Generally poor data for sequences:
  - ⇒ Handicap for scripts that write in “clusters” (Neo Brahmi).
  - But sequences not limited to Neo-Brahmi (.corn vs .com).

Outcome of the String Similarity Review

Status	Description
Ineligible	The label fails to meet the validity criteria for a label to enter String Similarity Review.
Pass	The label is not confusable with any of the labels it has been reviewed against. Or confusable only with labels delegated to or applied for by the same entity and delegation is permitted in the relevant policy.
Contention	The label is confusable only with another applied-for label. To be resolved by further process.
Fail	The label is confusable with any existing or reserved label, no longer candidate.

String Similarity Review Steps

1. Verify prerequisites (valid label), if invalid, result is “Ineligible”,
2. Pre-screen Test Label to establish that it is potentially confusable; if not confusable, skip to step 4,
3. Pre-screen for existing labels that are potentially confusable with Test Label,
4. Add manually requested Review Targets,
5. Review the Test Label against all Review Targets,
6. Further processing of Test Labels, result in “pass”, “contention”, or “fail” status.

Screening Heuristics

- Assumption that similar labels are clustered in perception space, and that such clusters are disjoint.
- Identify all labels belonging to the same cluster as the target label.
- Scoring two labels in the same cluster would be to assign as score the highest value in:
  - 1-Identical, 2-Highly confusable, 3-Similar, 4-Distant similar, 5-Distinct.
- ‘Variant’ types can be used to express various degrees of similarity.
- Using a unique cluster index that can be calculated for each label (like Index Label).
- Heuristic system based on LGR mechanism (RFC 7940).

Preparatory Steps in Heuristic Pre-screening

A. Create a similarity map on the code point level using LGR for the full repertoire
   - Distinct code points have no ‘var,’
   - Not dissimilar code points are given a ‘var’ mapping type based on classification above,
   - All code points get a reflexive variant type ‘r-identical’.

B. Create “actions” corresponding to step 3 of the Processing Steps (next slide).

C. Collect a list of all delegated labels and reserved, not assigned words/labels,
   - Implicitly include variants using cluster index.

D. Compute cluster indexes for all labels mentioned in C.

Processing Steps in Heuristic Pre-screening

1. Compute cluster index for the target label.
2. Collect all existing labels belonging to same cluster.
3. Compare each label to the target label,
   - Either not confusable, likely confusable (strongly similar), or further evaluation.
4. For any label that has not been resolved in step 3, perform manual evaluation against the target label.

Alternative Metrics for Similarity Distance

- Instead of confusability cutoff, use a mechanism of similarity based on spelling distance (such as the Levenshtein distance):
  - Define a number of basic editing algorithms that, if carried out in succession, morph one string into another,
  - Consider only direct operations, not chains.

- Steps:
  1. Determine all existing labels that are in the same confusables cluster as the target label,
  2. For each label in the cluster, compute a distance metric calculated for each possible partition of a label,
  3. Distance computed for each possible partition of the label
  4. Define two cutoff points with three results:
     a) above: distinct,
     b) middle: manual review,
     c) below: confusable.

Post-Screening Process: Manual Evaluation

- How to get to a yes/no decision: “is it confusable”?
- Guidelines not algorithm.
- Steps in manual evaluation
  - Verify score from pre-screening,
  - Optionally add more review labels based on expert opinion,
  - Reach a decision concerning confusable similarity (maybe based on adjusted score),
  - Document decision based on used considerations.
- Final decision left to String Similarity Review Panel:
  - Panel may consult script experts,
  - Panel will not use focus groups.

Post-Screening Process: Procedural Aspects, Outcomes

- Procedural aspects
  - Minimum number of reviewers (3),
  - Minimum qualification (familiarity with script, DNS security),
  - Ability to override computed score.
- Outcomes
  - Rejected if confusable with existing or pre-existing labels,
  - Added to contention set, if confusable with another applied for label or variant,
  - Otherwise: Accepted for further processing.
- Documentation
  - Identify label and outcome,
  - Record the decisions and number of experts.

SSR Next Steps

- Finalize initial report and collect initial data set.
- Public comment of initial report.
- If the methodology acceptable, then
  - Trial-run of pre-screening on sample labels.
  - Refine data sets.
- Community input on test results and finalized methodology.

Thank You!

- Public Comment Review link:
  https://www.icann.org/en/public-comment/proceeding/string-similarity-review-guidelines-07-02-2024
- String Similarity Review Guidelines
  https://itp.cdn.icann.org/en/files/strategic-initiatives/string-similarity-review-guidelines-07-02-2024-en.pdf
- Questions?