Enabling Linguistic Diversity of the Domain Name System

Root Zone Label Generation Rules

Sarmad Hussain, Senior Director IDN and UA Programs

ICANN DNS Symposium 2022
16 November 2022

Overview

- Many of the Internet users globally use their own script and are not familiar with English letters used in ASCII encoding.
- Internationalized Domain Names (IDNs) allow for such users to navigate the Internet in their local languages and scripts, making the Internet more inclusive.
- Enabling IDNs requires clear rules for forming valid domain labels.
- Root Zone Label Generation Rules (RZ-LGR) define such rules for top-level domains (TLDs).
- The presentation provides details on the following aspects of RZ-LGR:
  - Need
  - Design principles
  - Development process
  - Scope
  - Solution for Repertoire, Variants and Rules

Basis for the Root Zone Label Generation Rules (RZ-LGR)

- *Internationalized Domain Names for Applications (IDNA): Definitions and Document Framework* (RFC5890) presents guidance on determining the IDNs.

- Section 2.3.2.3 states:
  - DNS zone administrators may impose restrictions, beyond those imposed by DNS or IDNA, on the characters or strings that may be registered as labels in their zones [including the root zone].
  - Because of the diversity of characters that can be used in a U-label and the confusion they might cause
    - such restrictions ["variant definitions and rules beyond those imposed by DNS or IDNA"] **are mandatory** [emphasis added] for IDN registries and zones.
    - even though the particular restrictions are not part of these specifications (the issue is discussed in more detail in Section 4.3 of the Protocol document [RFC5891]).

Basis for the RZ-LGR

- Section 4.4 of RFC 5890 states:
  - It is worth noting that there are no comprehensive technical solutions to the problems of confusable characters.
  - One can reduce the extent of the problems in various ways, but probably never eliminate it.
  - Some specific suggestions about identification and handling of confusable characters appear in a Unicode Consortium publication [Unicode-UTR36].
    - For example: combining mark order spoofing, inadequate rendering support, and others.

RZ-LGR – The Solution for the Root Zone

- For achieving the secure and stable definition of IDNs as top-level domains (TLDs) to support the different languages and scripts used globally, Root Zone Label Generation Rules (RZ-LGR) is needed.
  - Builds on the Internationalized Domain Names for Applications (IDNA) standards including RFCs 5890, 5891, 5892, 5893 and their successors.
  - Uses the principles outlined in RFC 6912.
  - Follows the LGR Procedure developed by the community.

- Uses the machine-readable XML based LGR formalism proposed in RFC 7940. Also published is the corresponding human-readable HTML form.
  - Description.
  - Repertoire of code points.
  - Variant code points with types (allocatable, blocked).
  - Label evaluation rules.

Principles Guiding the RZ-LGR Design

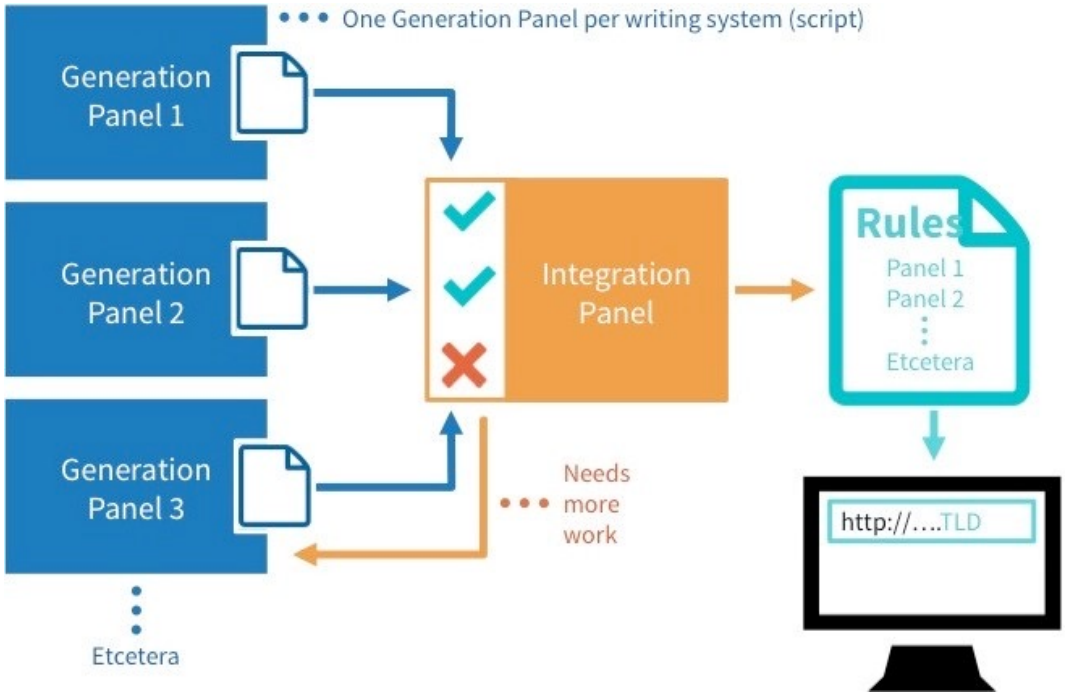
- *Principles for Unicode Code Point Inclusion in Labels in the DNS* (RFC6912) mentions:
  - most operators of zones should probably not permit registration of U-labels using the entire range.
  - presents a set of principles that can be used to guide the decision whether a Unicode code point may be included in the repertoire of permissible code points in a U-label in a zone.

Principles Guiding the RZ-LGR Design

- More-restrictive rules going up the DNS tree.
- Principles Applicable to All Public Zones:
  - **Longevity** - properties of code point stable across Unicode versions.
  - **Least Astonishment** – support expected code points otherwise don't.
  - **Contextual Safety** – prevent where it can be used maliciously.
  - **Conservatism** – when in doubt, don't include a code point.
  - **Inclusion** – every code point excluded unless explicitly included.
  - **Simplicity** – rules to include a code point be simple to understand.
  - **Predictability** – rules to include predictable with requisite knowledge.
  - **Stability** – list of permitted code points to change slowly.
- Principle Specific to the Root Zone:
  - **Letter** – allowed code points should be alphabetic (e.g. not digits).

LGR Procedure to Develop the RZ-LGR

- Maximal Starting Repertoire as the starting point – by Integration Panel (IP).
- Three step process for RZ-LGR:
  - Develop script-based proposal – by community-based Generation Panel (GP) using MSR.
  - Review proposal – jointly by IP and GP.
  - Approval and integration into RZ-LGR – by IP.



Scripts Covered in RZ-LGR

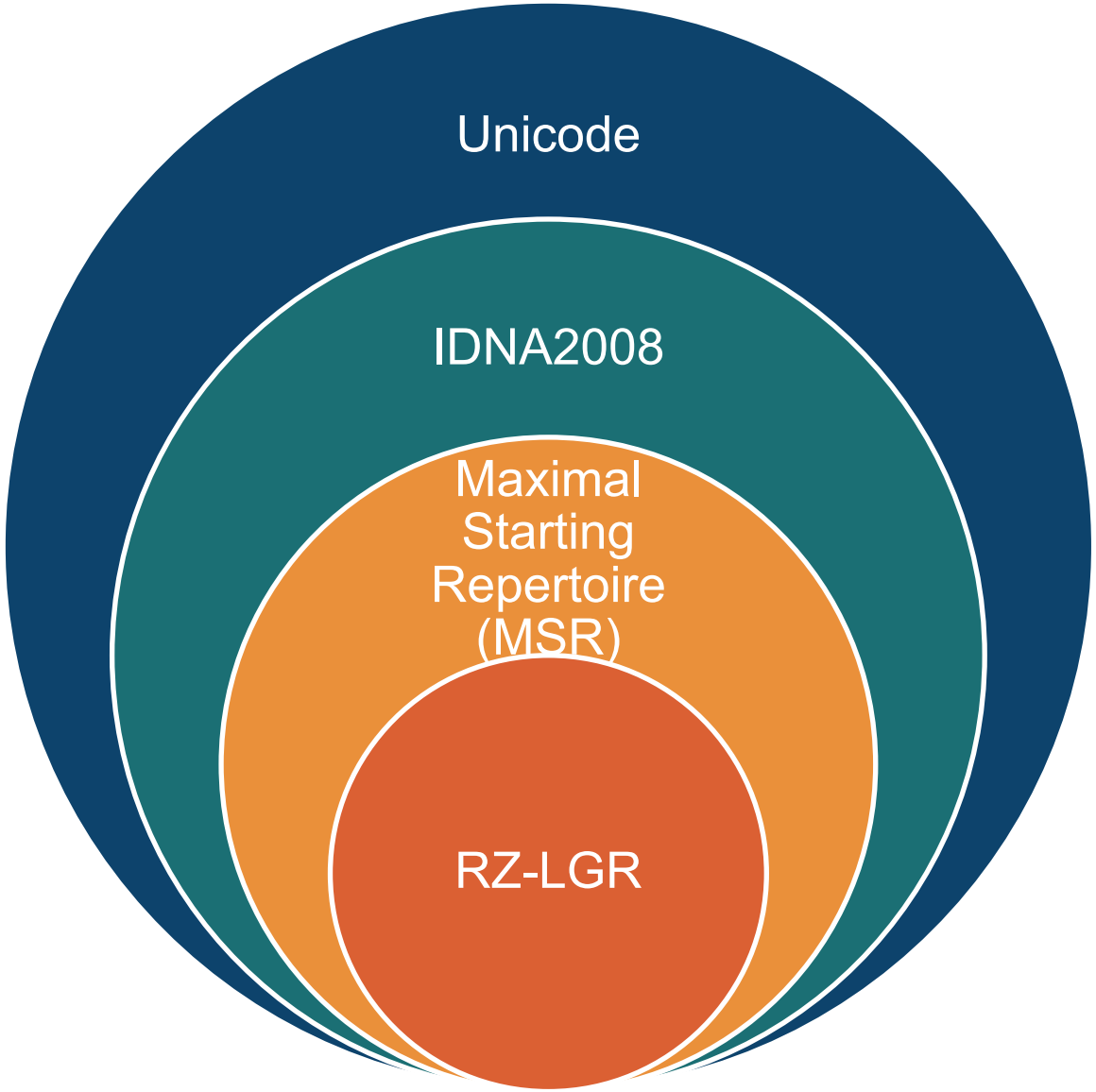
- MSR contains only 28 scripts aligned with scripts ”Recommended” for Identifiers by the Unicode standard out of the 159 encoded in Unicode 14.0.
- Does not include the following categories of script by Unicode standard. Complete script lists in UAX#31 (Tables 4, 5 and 7).
  - “Limited Use” scripts
  - ”Excluded” scripts.
- RZ-LGR-5 covers twenty-six scripts:
  - 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, Telugu, and Thai.
- The scripts covered may expand over time.

Language Status (using EGIDS) for Inclusion in RZ-LGR

- Many languages can be written using a script.
- Languages analyzed for RZ-LGR selected with a conservative criteria based on their status using Expanded Graded Intergenerational Disruption Scale (EGIDS):
  - 0-4 – included.
  - 5 – included on case-to-case basis.
  - > 6 – not included as their orthography may not be stable or well understood.
- The languages covered may expand over time.

Level	Label	Description
0	International	The language is widely used between nations in trade, knowledge exchange, and international policy.
1	National	The language is used in education, work, mass media, and government at the national level.
2	Provincial	The language is used in education, work, mass media, and government within major administrative subdivisions of a nation.
3	Wider Communication	The language is used in work and mass media without official status to transcend language differences across a region.
4	Educational	The language is in vigorous use, with standardization and literature being sustained through a widespread system of institutionally supported education.
5	Developing	The language is in vigorous use, with literature in a standardized form being used by some though this is not yet widespread or sustainable.
6a	Vigorous	The language is used for face-to-face communication by all generations and the situation is sustainable.
6b	Threatened	The language is used for face-to-face communication within all generations, but it is losing users.
7	Shifting	The child-bearing generation can use the language among themselves, but it is not being transmitted to children.
8a	Moribund	The only remaining active users of the language are members of the grandparent generation and older.
8b	Nearly Extinct	The only remaining users of the language are members of the grandparent generation or older who have little opportunity to use the language.
9	Dormant	The language serves as a reminder of heritage identity for an ethnic community, but no one has more than symbolic proficiency.
10	Extinct	The language is no longer used and no one retains a sense of ethnic identity associated with the language.

Repertoire Analysis for RZ-LGR



Repertoire Not Shortlisted by MSR

- Historic and phonetic extensions to modern scripts.
- Code points that pose special risks, e.g., due to instability of encoding.
- Code points with strong justification to exclude:
  - Archaic, historic, symbolic, and have little chance to gain use in modern context.
  - PVALID as unintended consequence of the IDNA2008 algorithm.
  - Highly confusable with an existing and common punctuation character.
  - Exclusively used for phonetic, liturgical or other specialized purposes.
- Non-spacing combining marks, where precomposed forms are also encoded.
- Digits.

Repertoire Shortlisted by Script Community

- Include only general purpose and common use code points.
- Code points may not be included for many reasons:
  - Historic use or no longer common use:
    - Arabic: ڈ 0690 – historic use only, now replaced by ڈھ.
    - Kannada: ಱ 0CB1 - obsolete character, not used in modern Kannada.
    - Gurmukhi: ◌ਃ 0A03 - limited or declining use.
  - Special purpose:
    - Devanagari: U+0929 ऩ - not in any spoken language; transcribes Dravidian alveolar n.
  - Usage not known in any language included for RZ-LGR:
    - Cyrillic: ӛ 04ED – possibly used in Sami with EGIDs 8b.
    - Arabic: ﭙ 069B – no evidence found of active use.

Summary of Community Analysis for Repertoire

Script tag[26]	Script Name	MSR-5	LGR-1	LGR-2	LGR-3	LGR-4	LGR-5
Arab	Arabic	241	128	128	128	128	128
Armn	Armenian	38					38
Beng	Bengali	64				62	62
Cyrl	Cyrillic	93					86
Deva	Devanagari	92			84	84	84
Ethi	Ethiopic	364		311	311	311	311
Geor	Georgian	37		33	33	33	33
Grek	Greek	36					36
Gujr	Gujarati	66			65	65	65
Guru	Gurmukhi	61			56	56	56
Hang	Hangul	11 172					11 172
Hani	Han Ideographs	19 855				19 685	19 844
Hebr	Hebrew	46			27	27	27
Hira	Hiragana	89					86
Kana	Katakana	92					88
Khmr	Khmer	78		71	71	71	71
Knda	Kannada	68			62	62	62
Laoo	Lao	53		51	51	51	51
Latn	Latin	312					197
Mlym	Malayalam	73			70	70	70
Mymr	Myanmar	102					99
Orya	Oriya	66			62	62	62
Sinh	Sinhala	79			72	72	72
Taml	Tamil	49			48	48	48
Telu	Telugu	67			63	63	63
Thaa	Thaana	50					
Thai	Thai	71		69	69	69	69
Tibt	Tibetan	80					
Zinh	INHERITED	21					7
Total		**33 515**	**128**	**663**	**1 318**	**21 019**	**32 987**

Understanding Variant TLDs

Code points technically distinct but considered the “same” by the script community – non-deterministic!

- Security
- Usability

Categories of Variant Code Points - Identical

- Visually identical.
  - Same:
    - Armenian, Cyrillic, Greek, Latin: օ o o o 0585 043E 03BF 006F
    - Japanese: へ へ 3078 30D8
  - Same in a joined form.
    - Arabic:
      - 0643: ك كب بك بكب
      - 06A9: ک کب بک بکب
    - Khmer:
      - ស្ + ត 179F 17D2 + 178F = ស្ត
      - ស្ + ដ 179F 17D2 + 178A = ស្ដ

Categories of Variant Code Points - Similar

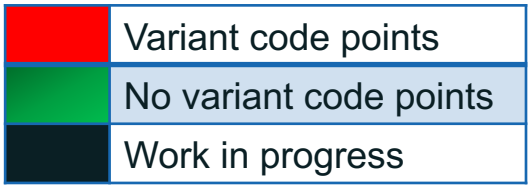
- Similar but not identical.
  - Visually similar:
    - Devanagari and Gurmukhi: उ ਤ 0909 0A24
    - Kashmiri vowel signs in Devanagari: ॲ ॵ 0973 0905+0902
    - Korean Hangul and Hanja: 슴 舍 C2B4 5408
  - Similar due to cursive/handwriting form:
    - Latin: f ƒ 0066 0192
  - Similar with stylistic variation:
    - Arabic: ک ے 06A9 06AA
  - Similar in marks:
    - Latin: ğ ǧ 011F 01E7 (breve and caron)

Categories of Variant Code Points – Visually Distinct

- Considered equivalent even when not visually similar.
  - Phonetically same or similar:
    - Arabic: ه ة 0647 0629
    - Ethiopic: ሀ HA ሐ HHA ኀ XA (1200 1210 1280)
  - Alternate writing convention:
    - Arabic Western (African) vs. Conventional: ڤ ق 0642 06A7
    - Chinese Simplified vs. Traditional: 万 萬 4E07 842C
  - Spatial rotation of dots:
    - Arabic: ت ٮ 062A 067A
  - With or without marks:
    - Greek tonos and dialytica: ι ί ϊ ΐ 03B9 03AF 03CA 0390
  - Contextual variation:
    - Hebrew normal and final form: פ ף 05E4 05E3
  - Semantically same:
    - Chinese: 叢 欉 53E2 6B09

Scripts With Variant Code Points

- Arabic
- Armenian
- Bengali
- Cyrillic
- Devanagari
- Ethiopic
- Georgian
- Greek
- Gujarati
- Gurmukhi
- Han
- Hebrew
- Japanese
- Kannada
- Khmer
- Korean
- Lao
- Latin
- Malayalam
- Myanmar
- Oriya
- Sinhala
- Tamil
- Telugu
- Thaana
- Tibetan
- Thai



Code point variants are as defined in the RZ-LGR - deterministic.
3,763 variant sets in RZ-LGR.

Types of Variant Code Points

- Two main types of variant mappings:
  - Blocked – label with a code point with this type cannot be delegated.
  - Allocatable – label with only this type of code points can be considered for delegation.
- Design consideration: Maximize blocked variant labels (for end-user security) and minimize allocatable variant labels based on usability (for manageability).
  - Blocked - by default.
  - Allocatable: In few cases, where there is a clear usability requirement documented by the script community.
- 58 allocatable and 5,806 blocked variant mappings (excluding CJK).

Label Evaluation Rules

- Context of a character.
  - Complex scripts are inherently rule based.
    - Different categories of character: consonants, vowels, tone marks, others.
    - In Abugida scripts, there is a specific structure of a well-formed orthographic syllable.
  - Script users apply these rules when writing, and so same rules are needed when encoding labels.
  - Multiple code point sequences can generate the same orthographic syllable.
  - Out of context code points.
    - Not predicted by users.
    - May not be supported in fonts.
    - May have unpredictable rendering by rendering engines.

Label Evaluation Rules

- Context of a character.
  - Lao vowel placement rules:
    - A vowel-before precedes the main consonant cluster C.
    - A vowel-above or a vowel-below follows the main consonant C.
    - A vowel-after follows the main consonant C or a tone mark or a vowel-above.
  - Thai tone mark rule:
    - A tone mark can only follow a consonant, an above-vowel or a below-vowel.
  - Tamil Virama rule:
    - Virama must be preceded by a consonant.

Label Evaluation Rules

- Place of a character.
  - Lao repetition sign ໆ (0EC6) can only occur 0-3 times at the end of a label.
- Reducing variant labels.
  - Arabic:
    - Cannot mix ه wtih ھ 06C1 0647
    - Cannot mix ة with ۃ 0629 06C3
  - Myanmar:
    - No mixing from two sets of code points to limit the number of variants generated.

Label Evaluation Rules

- The rules fix the order and place of characters to only well-formed and predictable options.
  - Supported by fonts and rendering engines.
- The rules also help minimize allocatable variant labels, as needed.
- Number of rules140:
  - Used as context rule - 100.
  - Place of character - 98.
  - Used to trigger actions - 28.
  - Used only in another rule - 14.

Conclusions for RZ-LGR

- Creates a solution for top-level domains in multiple scripts and languages, balancing different design principles, while being sufficiently conservative.
- Developed by the relevant script community with expertise of the script.
  - Includes the repertoire needed for common and general-purpose use.
  - Gives a deterministic definition of variant labels.
  - Allows for domain names which are well-formed for the community.
- Provides a solution which is technically viable and secure for the end-users.
  - Repertoire.
  - Variant labels.
  - Context rules.
- Enables a solution which addresses the usability of domain names.
- Provides a solution which can evolve in a stable manner (e.g., adding support of more languages and scripts).

Engage with ICANN

Thank You and Questions

Visit us at **icann.org**
Email: IDNProgram@icann.org

@icann

linkedin/company/icann

facebook.com/icannorg

slideshare/icannpresentations

youtube.com/icannnews

soundcloud/icann

flickr.com/icann

instagram.com/icannorg