

Evaluating Unicode Scripts for Use in IDNs

DATE: 2022-01-05

Overview

This document explores the issues that might define feasibility for the support of the different Unicode scripts in domain names, particularly the Root Zone and the Second Level, and the potential interaction with categorization of the scripts in UAX#31: “Identifier and Pattern Syntax” [UAX#31].

Evaluating Unicode scripts with respect to IDN suitability involves both socio-linguistic and technical considerations. The result of such an evaluation would be to roughly sort the scripts by suitability for support in Internationalized Domain Names (IDNs). The first group of scripts is suitable for Second-level domain names but also suggested for consideration for inclusion in the Maximal Starting Repertoire [MSR] (for top-level domains) as part of the Root Zone LGR process [RZ-LGR]. The second group is considered of more limited suitability, but still feasible for domain names on the Second Level taking into account the nature of the DNS zone for which they are intended. Some scripts may be more suitable for single-script domains, for example. The third group is generally considered not suitable for domain names. For details of the reasoning behind these recommendations, see the description of the individual groups.

The background and spirit behind this work is defined by ICANN's desire to support access to domain names in scripts for native users across all communities. As the classification of code points in [IDNA2008] is based on Unicode properties, many recommendations from the Unicode Standard are inherited more or less directly into the derived IDNA 2008 protocol. An exception to this is [UAX#31] which is not a normative input to IDNA 2008, but instead is recommended in IDNA 2008 to registries in designing their policies [RFC5894]. Separately, the [Procedure] developed by ICANN for the [RZ-LGR] process makes explicit reference to [UAX#31].

UAX#31 sorts the list of scripts by Unicode Identifier_Type value into **Recommended** and **Excluded**, but also a third category of **Limited_Use**. That last category is somewhat ambiguous on its implications, with [UAX#31] stating: “some implementations may wish to disallow the limited-use scripts in identifiers”. The factors that underlie such a case-by-case decisions are discussed here in detail.

This document firstly explains how the categorization as **Recommended** and **Excluded** affect the support of a script in IDNs, but also delves into the special considerations that might be applied to a script in the **Limited_Use** category to see whether a particular script would be suitable for use in IDNs and whether it can be supported safely and securely in a given zone. The Unicode Identifier_Type is formally defined for “general-purpose” identifiers, and not specifically for domain names. There are a few scripts which may be less suitable for domain names than for general-purpose identifiers, but otherwise these classifications apply. We start the report by discussing the three rough levels of suitability for use with IDNs.

This is followed by a section that lays out the various factors that go into determining a script's suitability for use with IDNs. As noted there, a script's suitability is not an absolute value, but may depend on considerations such as the nature of the zone for which a script is to be supported. Accordingly, when we group scripts by suitability, we also point out whether the results would apply to the Root Zone, the Second Level or other public zones.

The final section sets out the requirements for defining label generation rules (LGR) sufficient for a safe and secure support of IDN labels in a given script.

Scripts Suitable for Use with Top-level IDNs

The determination of which scripts are suitable for use with top-level IDNs is reserved to the [RZ-LGR] process. In that process, the Integration Panel is charged with developing a Maximal Starting Repertoire [MSR] that positively lists all code points that are available to the various script-specific Generation Panels in developing LGR proposals for their script. Scripts for which no code points are included in the [MSR] are therefore excluded from the Root Zone. (Note that within each script many code points are excluded as well). The Root Zone process, as defined in [Procedure], balances the desire to accommodate a very wide range of user communities with the need for stability and security of the Root Zone.

The development of the [MSR] started with all the scripts that have Unicode Identifier_Type **Recommended**, which [UAX#31] asserts are scripts that are in “widespread everyday common use”. The [MSR] review confirms the suitability of all of these scripts, with the exception of **Bopomofo**, which the [MSR] regards as not suitable at the top level due to its restricted field of applicability. At the time of this writing almost all of the scripts shortlisted for the Root Zone have been thoroughly vetted for their suitability and desirability for IDNs. The basis for the recommendation to include them in the Root Zone has been extensively documented as part of the [RZ-LGR] effort, as has their subset of code points actually permitted for TLDs.

For the current state of [RZ-LGR] development, the fact that a script is listed as **Recommended** in [UAX#31] effectively represents a necessary condition for considering a script suitable for use with top-level IDNs, but the final determination is made as part of the [RZ-LGR] process. The actual [MSR] further narrows each script so only code points in widespread everyday common use are included, with rare, historical, obsolete, technical and specialized additions to a script's repertoire excluded. As documented in [MSR] and [RZ-LGR], this involves a detailed examination of the code points actually used in living orthographies in vigorous use. One factor in that analysis is the [EGIDS] level for the language, which the [MSR] takes as a proxy for the likely use of the associated orthographies.

Section 2.3.2.3 of [RFC5890] specifies that because of the “*diversity of characters that can be used in a U-label and the confusion they might cause, such restrictions are mandatory for IDN registries and zones*” and points to this statement in [RFC5891]:

“In order to avoid intentional or accidental attacks from labels that might be confused with others, special problems in rendering, and so on, the IDNA model requires that registries exercise care and thoughtfulness about what labels they choose to permit. That issue is discussed in Section 4.3 of this document which, in turn, points to a somewhat more extensive discussion in the Rationale document [RFC5894].”

A key motivation behind the focus on active modern use in the [MSR] is that IDNs are intended to serve users as useful mnemonics in their own, familiar writing system. Extensions for rare or specialized use, including historic or extinct orthographies do not make good candidates for useful mnemonics, because ordinary users cannot read or type them. Most likely, they will confuse them with similar looking, more familiar code points, thus creating a security issue. [RFC5894] states this as follows:

“In general, users will benefit if registries only permit characters from scripts that are well-understood by the registry or its advisers. If a registry decides to reduce opportunities for confusion by constructing policies that disallow characters used in historic writing systems or characters whose use is restricted to specialized, highly technical contexts, some relevant information may be found in Section 2.4 (Specific Character Adjustments) of Unicode Identifier and Pattern Syntax [UAX#31], especially Table 4 (Candidate Characters for Exclusion from

Identifiers), and Section 3.1 (General Security Profile for Identifiers) in Unicode Security Mechanisms [UTS#39].”

There is a diminishing return in supporting scripts or code points outside the set defined in [MSR]: except for digits and a few other code points excluded from the Root Zone solely due to the Letter Principle, there are few users who will have strong needs for any additions, or even benefit from them.

All scripts suitable for use with top-level IDNs should also be considered suitable for use on the Second Level, assuming a set of Label Generation Rules has been developed that contain the [Required Elements for Label Generation Rules](#).

There are a small number of scripts not currently recommended for which the nature of their use is potentially undergoing a change, whether due to concerted efforts to recruit populations to adopt the use of the script or due to increased literacy and in particular digital use for a script, or both. Some are used for languages with large user communities, but that are written in a number of other scripts as well, making exact usage estimates uncertain and subject to changes. Thus the classification is not frozen, but can be expected to track the evolving script use over time. This would apply both to the [MSR] and to [UAX#31]. For further discussion, see [Determining a Script's Suitability for IDNs](#) below.

Scripts Suitable for IDNs on the Second Level

[UAX#31] states for scripts identified as *Limited_Use* that “some implementations may wish to disallow the limited-use scripts in identifiers”. The implication is thus that scripts of Unicode Identifier_Type *Limited_Use* (with the addition of **Bopomofo**) are generally suitable for IDNs on the Second Level (and any other level than the Root Zone), but a detailed case-by-case evaluation by the registry is expected on whether to support a given script for the intended zone.

Generally, scripts that are *Limited_Use* are in varying degree of active modern use. Unlike the case of recommended scripts, this category includes scripts with smaller populations, lower literacy rates or declining or emerging use; some are in competition with other recommended scripts for writing the same languages. Many of the scripts in this category share the same uncertainty about the exact state of their modern use, or the number of actual users. A few scripts, despite the classification, should probably be considered unsuitable for IDNs on technical and other grounds as explained below.

The choice of supporting any of these scripts would also depend on the nature of the zone. There are roughly two groups of zones. The first group consists of global zones where the requirements for implementing multiple scripts in the same zone are much the same as for the Root Zone. While many of the limited-used scripts would not be an issue when supported in a zone of their own, they often have confusability issues with scripts from the recommended set. Thus supporting certain combinations of multiple scripts in the same zone would require extensive constraints to address cross-script variants that could quickly prove unmanageable. On the other hand, a localized second zone, addressing a single language, country, or local market could more easily implement many of the limited-use scripts.

The same considerations as stated above for top-level scripts apply to restricting the permitted code points (and their combinations) to those actually required for actively used languages. These are set forth below in the [Required Elements for Label Generation Rules](#), which are a prerequisite for implementing these scripts.

For a variety of reasons, registries might want to not support **Cherokee, Lisu, Mandaic, Osage** and **Syriac**. Cherokee and Lisu have the issue of letterforms being derived from and therefore highly confusable with uppercase ASCII, which makes their safe use in an IDN context somewhat challenging. Languages using the Cherokee, Mandaic and Osage scripts have very low numbers of speakers and are

considered threatened. Any speakers are likely to use a different language online. Syriac appears limited to religious use. For further discussion, see [Determining a Script's Suitability for IDNs](#) below.

Before supporting any of these scripts on the second or any other level, a registry is advised to work with the affected community to understand the issues involved; some of the scripts would require a fair amount of research, as well as technical and linguistic expertise in constructing the LGR. The LGRs for many of the South or South East Asian Brahmi derived Abugidas, for example, can be expected to require considerable investment in defining not simply the code point repertoire, but also the associated context rules that will make the LGR workable for a complex script.

There are a handful of scripts beyond this category that appear to have some indications of active, if rather limited modern use. If these indications solidify, the status for these scripts could be upgraded in the future. However, in all cases these would likely represent very modestly sized communities.

Scripts Considered Not Suitable for IDNs

The scripts in this category have Unicode Identifier_Type **Excluded** in [UAX#31], although registries might consider even some of the **Limited_Use** as not suitable, based on the discussion above.

The Unicode classification of “Excluded” is presented in [UAX#31] without detailing the basis for that determination on a per script basis. The primary factors that argue for considering a script not suitable for IDNs include whether a script is particularly problematic, for example, the encoding model is not fully worked out or stable, or the elements are not normally arranged in linear sequence. Such scripts, despite their code points being PVALID, should be categorically considered unfit for identifiers.

Beyond rejection on technical grounds, the absence of any modern use (other than by scholars or digital archivists) should be seen as sufficient rationale to also consider the script unsuitable. The legitimate audience for IDNs for such scripts is limited to specialists or academic users—for anyone else, labels in these scripts are uninterpretable and therefore incapable of expressing useful mnemonics. Instead, if carelessly supported in the same zone as modern scripts, they can make attractive tools for spoofing.

Many extinct scripts may be particularly problematic in this regard. Their letter shapes are often stylized; either by design or because modern fonts do not attempt to faithfully render the appearance, for example where the originals were incised in stone. In some cases, now defunct scripts devised for local orthographies by missionaries suffer from overlap in letter shapes with modern scripts that served as their template. In these cases, supporting such a script in the same zone as the modern script would require strong attention to cross-script variants; however, many of the historic and extinct scripts have not been well-researched in that regard.

For further discussion, see [Determining a Script's Suitability for IDNs](#) below.

Determining a Script's Suitability for IDNs

In reviewing scripts for their suitability for IDNs, it is generally *not possible* to reduce everything to a single “score” and then set some “suitability threshold”. The main reason why this approach is not successful lies in the multidimensional nature of the problem: there are many independent factors that may influence both the desirability and suitability of a script in both positive and negative ways. Depending on the overall nature of the script, one or more of these factors may take on greater importance and ultimately drive the decision to recommend a script or to consider it merely “suitable” in a generic way, or even to recommend that it not be supported as being unsuitable for use in domain names.

Different factors may come into play depending on the nature of the zone. As noted in [\[RFC6912\]](#) the Root Zone as a single global resource must be managed in a more restrictive manner than the Second Level or below. Zones that support only a single script or a single language are not exposed to some of the issues of cross-script, or cross-repertoire confusability and variants that can be relevant for the security of zones that allow labels from different scripts to coexist.

In the following, we list some of the general factors and their typical effect on the determination of script suitability. It has to be understood, though, that the actual determinations that go into classifications such as those presented above are the result of expert judgment influenced by these factors — and not the result of some testable “algorithm”. In a similar vein, a registry deciding to support a script in a particular zone would weigh these factors individually, making the necessary tradeoffs between any competing ones.

One key factor is based on demand, or potential demand, by a given community, particularly one associated with a sovereign entity or other territory. In some ways, the anticipated demand is one of the more subjective factors, and one that is perhaps most subject to being affected by changes in the community over time: from increased literacy, to more thorough adoption of a given script, and finally to greater use of digital means of communication. Because direct statistics of the use of scripts are unavailable, the available information on the principal languages for the script, their population, their status (e.g. whether vigorous or threatened), as well as literacy of their speakers may be substituted.

Because IDNs are about accessing information and services that are provided online, some of the metrics (such as the prevalence of printed material) that would be used in gauging whether to encode a script are less useful in the present context. In contrast, the degree to which the script is used online in social media or publications and data intended for general audiences (and covering everyday topics) can serve as an indicator on favor of support in IDNs.

Issues that make a script particularly challenging from a security point of view may argue strongly against supporting a script. No amount of popular demand changes the technological obstacles based in the design and encoding of a script. However, some of these factors may have to be weighed differently based on the nature of the intended zone for which the script is to be used, and whether the script is used together with other scripts in the same zone. By this we mean whether labels could be registered in *either* of the scripts, not as mixed-script labels (outside the mixed script writing systems of the Far East).

In a few cases, a script may have an unusual feature not well-captured by these general factors.

Factors Supporting Use with IDNs

1. The script is in widespread, everyday, common use in a large user population. With one exception, this is equivalent to being classified with Identifier_Type **Recommended** in [\[UAX#31\]](#). The exception is Bopomofo, for which the common use is for phonetic transcription of Chinese in educational settings, and which based on this limitation has been excluded from the Root Zone LGR [\[RZ-LGR\]](#).
2. The script is in everyday, common use in a smaller population, either a growing one as adaptation of the script, literacy rates or degree of digital communications are changing, or it is used in smaller community, but with strong institutional support, for example an official script associated with a sovereign entity.
3. There is definite evidence of the script being used online, for both displayed user names and contents covering everyday topics on social media (such as Twitter and even the written descriptions and comments for YouTube videos); and there are general purpose online publications (news sites) or data collections in that script (Wikipedia). Online search readily discovers such content.
4. Supporting the script in IDNs presents few technical or security challenges: simple shapes without complex behavior that have a characteristic and not easily confused appearance.

5. For some zones, usage in a more limited domain, such as religious or scholarly contexts might not be an obstacle if the writing is well standardized and there is a community of users able to communicate in and actively use that writing system in ways that are relevant to IDNs.

Factors Making Use with IDNs less Desirable

1. The encoding model for the script is unstable. While the codes have been assigned in Unicode, the rules of how to interpret and render them may still be argued about. This is of course an exceptional situation. It is slightly more common for an individual code point to come into question. In all these cases, such instability should be seen as an absolute disqualification for the script or the code point.
2. The orthography for the principal language(s) using the script may be unstable. This should disqualify the use of this script for network identifiers for those languages.
3. The script may have been added to Unicode very recently. (At the time of this writing, the latest published version of Unicode is Version 13.0.0 and updates typically occur every year). It takes a few years, usually, before fonts and operating systems are updated to handle a newly encoded script. For scripts requiring complex rendering support, it is not enough for users to be able to load a font. Individual code points added to an existing script, may be supported sooner as they would “slot into” existing systems. However, many such extensions may be for specialized or limited use, and it would therefore not be harmful to “walk it slow” when it comes to updating a registry to a new Unicode version.
4. The IDNA2008 tables on the IANA registry [[IDNAREG](#)] may not be ready for the latest version. Because [[IDNA2008](#)] allows exceptions to the PVALID status computed from Unicode properties, it is a prudent thing to not support scripts or code points not yet in [[IDNAREG](#)].
5. The script may not be used for writing text. Some script codes are given to notational systems that by nature are unsuitable for IDNs.
6. The script is in limited use. For some scripts, the modern user community can measure in the few hundreds. Many scripts do not have living native users at all, but may have some degree of scholarly use. For some scripts, while there may be ongoing use in publication of religious texts, for example, there may not be all the other types of uses for which IDNs are so essential. There may be lack of effective demand for supporting that script.
7. There is limited evidence of the script being used online; or online search primarily brings up “how to learn this script/language” sites.
8. Issues around confusability or variants may be particularly challenging to settle and account for. Some scripts use combined forms for certain letter combinations. Unlike the familiar ligatures in the Latin script, these may be mandatory and number in the hundreds. The changes in shape, compared to the nominal appearance may be substantial.
9. Issues around required context and what sequences can be rendered reliably for a script may be difficult to ascertain (particularly for ancient or little used scripts), or the resultant context rules and other constraints may be challenging and complex. Unless there is a strong perceived demand for such a script, it may be better to “walk away”.

Required Elements of Label Generation Rules

In evaluating scripts for their suitability with respect to IDNs, we are making the tacit assumption that this support is based on having defined a set of Label Generation Rules (LGR) for that script in a given zone (or for a language in that script). For more details on LGRs, see [[RFC7940](#)]. For examples, see the Root Zone LGR [[RZ-LGR](#)] and the Second Level Reference LGRs [[Level2-Ref](#)].

In particular, when addressing the suitability of a script for IDNs, we presuppose a well-designed LGR for that script containing the required elements outline here.

Repertoire: selecting a subset

Almost no script in Unicode consists only of characters that are available for identifiers as defined in [IDNA2008]. Instead, almost all script blocks in Unicode contain extensions for specialized use: phonetics, poetry, editorial or liturgical annotations, transcription of historic or archaic languages written in the script, and so on. Even users familiar with a script may not be familiar with such specialized characters and may have difficulty recognizing or entering them; or they may confuse them with similar looking commonly-used characters.

In several cases, Unicode even collects under the same script identifier several rather distinct writing systems, which may not be compatible with each other, or equally suitable for IDNs. Allowing any and all extensions for a given script could compromise the usability or security of a zone. As a result, not all code points are equally available or well-suited for IDNs.

In [RFC5890], IDNA 2008 makes it a mandatory requirement for registries to restrict IDNs to address confusability, security and stability concerns—beyond the algorithm for shortlisting characters mostly based on Unicode properties defined in [RFC5892]. The actual repertoire for a script's LGR should be constructed by adding only those code points strictly needed for IDNs and generally available to and recognized by users of the zone. That selection is assumed to follow the principles laid out in [RFC6912].

Outside the Root Zone, one of the items to be considered in selecting a subset is whether the script supports a set of native digits, and whether they are part of the modern use of the script and should therefore be included, whether alone, or in parallel with the European digits (ASCII digits).

Context and Whole Label Evaluation (WLE) Rules: adding constraints

Unlike the Latin script or the Han ideographs, for the many Unicode scripts random or arbitrary sequences of letters or marks are not supported. Users as well as rendering engines or fonts may not be able to process such sequences correctly or reliably. Some unexpected variations may result in sequences that are indistinguishable from the expected sequence (both may be rendered the same). For these reasons, LGRs for many scripts require carefully designed context rules that prevent the formation of sequences that are unsupported or “can't occur” in the script. This is different from sequences that merely “don't occur” in a given language; the latter would be too tight a standard for identifiers as it would amount to enforcing “spelling rules”.

In multiword identifiers that drop spaces, some atypical combinations might occur at word boundaries. If these do not lead to rendering issues, they can be allowed in identifiers, as can some subsets of arbitrary sequences that are tolerated by users and reliably rendered.

For scripts of the Abugida type, any code point sequences generally must represent well-formed syllables. Reading and writing for these scripts proceeds on the basis of the syllable, even if Unicode encodes the underlying letters or building blocks. For a lower-level zone that is under the control of and used within a single entity, it may be possible to rely on users to only enter legible and recognizable text. The same is not advisable, for example, for a public zone on the second or third level: the security implications would be significant. For some scripts, the required rules can be complex.

Right-to-left (RTL) scripts are displayed using the Unicode Bidi Algorithm [UAX#9]. Mixing directions in a fully qualified domain name (FQDN) can lead to ambiguous display, therefore, IDNs from RTL scripts must

conform to the additional restrictions laid out in [\[RFC5893\]](#):

“the most important example of non-permitted labels being labels that mix Arabic and European digits (AN and EN) inside an RTL label, and labels that use AN in an LTR label”.

In practice, these restrictions may be satisfied by a combination of repertoire restrictions and context or WLE rules.

Variants: maintaining identifier uniqueness

Several scripts in Unicode have two or more letters or digits of exactly the same appearance, and some may have multiple code point sequences that look alike. The same can be true for letter shapes or sequences across closely related scripts. The underlying reason is that Unicode considers letters distinct when they are part of different case pairs, have different positional shaping or digit properties as well as have different script properties. Certain basic letter forms (line, circle, half circle) may be indistinguishable even across otherwise unrelated scripts if presented out of context.

Other scripts may have alternative characters that are the same when sounded out, or have the same, interchangeable meaning, despite different appearance. All of these scripts require an LGR that addresses the security and usability issues posed by such variants. [\[RFC5894\]](#) states:

“For many scripts, the use of variant techniques such as those as described in the JET specification for the CJK script [\[RFC3743\]](#) and its generalization [\[RFC4290\]](#), and illustrated for Chinese by the tables provided by the Chinese Domain Name Consortium [\[RFC4713\]](#) may be helpful in reducing problems that might be perceived by users.”

The variant mechanism in [\[RFC7940\]](#) provides a generalized and fully machine-processable approach to prevent collisions between labels that are composed entirely of look-alike or otherwise equivalent characters. See [\[RFC8228\]](#) for guidance on defining LGRs for variants and [\[RZ-LGR\]](#) or [\[Level2-Ref\]](#) for further discussion and examples.

Beyond look-alikes or characters that can be seen as equivalent according to the communities who use them, there is a wider set of confusability issues. For scripts that have a Root Zone LGR [\[RZ-LGR\]](#) or Second Level Reference LGR [\[Level2-Ref\]](#), the issues of look-alikes and other variants have been extensively researched and codified; in some cases, additional observations on confusability have been documented. For most other scripts, all that information would have to be developed by any registry looking at safely using that script for IDNs, particularly on the Second Level, or in any zone supporting multiple scripts.

Conclusion

The report presents the classification of scripts for use in identifiers undertaken by the Unicode Standard in [\[UAX#31\]](#) and explains how this is applicable to determining the suitability for use of a script with IDNs on the top-level or for other levels. While the decision on the scope of the Root Zone lies with the [\[RZ-LGR\]](#) process, the results tally with the set of **Recommended** set of scripts from [\[UAX#31\]](#) (with one exception). All of these scripts are also suitable for the Second Level. Within each script as defined in the Unicode Standard, there are many code points that are not themselves recommended. Registries wishing to support these scripts can leverage the [\[MSR\]](#), the LGRs developed for the Root Zone [\[RZ-LGR\]](#) or Second-level Reference LGRs [\[Level2-Ref\]](#) when applying the restrictions mandated in [\[RFC5890\]](#).

With few exceptions, scripts classified as **Limited Use** are generally suitable for the Second Level (and all levels other than the Root Zone), but their support should be evaluated on a case-by-case basis by the

registry. The report presents the factors that would go into making such a decision and sets out the required elements for any LGR implementing such a script. When developing an LGR for these scripts, registries would have to work with the relevant communities to research and understand how to apply the restrictions mandated in [RFC5890].

Scripts in the **Excluded** set combine scripts that are categorically unsuitable for identifiers on technical grounds with scripts that may be challenging to implement and for which limited information is available. Common to all is the absence of a modern native user community that would be able to use these scripts for useful mnemonic identifiers in a familiar language. Generally, the only users of these scripts are specialists, such as scholars or digital archivists, who conduct their everyday online transactions in some modern language instead. Together with the often problematic and little understood nature of these scripts, that makes them unattractive targets for developing the requisite label generation rules.

Contributors

This report was created by Asmus Freytag and Michel Suignard, with ICANN staff support by Sarmad Hussain and Pitinan Kooarmornpatana.

Table of References

The following lists the references cited in this report.

- [EGIDS] Lewis and Simons, “EGIDS: Expanded Graded Intergenerational Disruption Scale,” documented in [SIL-Ethnologue] and summarized here:
[https://en.wikipedia.org/wiki/Expanded_Graded_Intergenerational_Disruption_Scale_\(EGIDS\)](https://en.wikipedia.org/wiki/Expanded_Graded_Intergenerational_Disruption_Scale_(EGIDS)).
- [IDNA2008] Faltstrom, P., Ed., “The Unicode Code Points and Internationalized Domain Names for Applications (IDNA),” RFC 5892, August 2010,
<https://www.rfc-editor.org/info/rfc5892>.
- [IDNAREG] IANA Registry: “IDNA Parameters”. For Unicode 11.0 available at:
<https://www.iana.org/assignments/idna-tables-11.0.0/idna-tables-11.0.0.xhtml>.
- [Level2-Ref] ICANN, Reference LGRs for the Second Level,
<https://www.icann.org/resources/pages/second-level-lgr-2015-06-21-en>.
- [MSR] ICANN, Root Zone Maximal Starting Repertoire (MSR),
<https://www.icann.org/resources/pages/msr-2015-06-21-en>
- [Procedure] Internet Corporation for Assigned Names and Numbers, “Procedure to Develop and Maintain the Label Generation Rules for the Root Zone in Respect of IDNA Labels.” (Los Angeles, California: ICANN, March, 2013)
<http://www.icann.org/en/resources/idn/variant-tlds/draft-lgr-procedure-20mar13-en.pdf>.
- [RFC3743] Konishi, K., Huang, K., Qian, H., and Y. Ko, “Joint Engineering Team (JET) Guidelines for Internationalized Domain Names (IDN) Registration and Administration for Chinese, Japanese, and Korean”, RFC 3743, April 2004.
- [RFC4290] Klensin, J., “Suggested Practices for Registration of Internationalized Domain Names (IDN)”, RFC 4290, December 2005.
- [RFC4713] Lee, X., Mao, W., Chen, E., Hsu, N., and J. Klensin, “Registration and Administration Recommendations for Chinese Domain Names”, RFC 4713, October 2006.
- [RFC5890] Klensin, J., “Internationalized Domain Names for Applications (IDNA): Definitions and Document Framework,” RFC 5890, August 2010,
<https://www.rfc-editor.org/info/rfc5890>
- [RFC5891] Klensin, J., “Internationalized Domain Names in Applications (IDNA): Protocol,” RFC 5891, August 2010,
<https://www.rfc-editor.org/info/rfc5891>

- [RFC5892] Faltstrom, P., Ed., "The Unicode Code Points and Internationalized Domain Names for Applications (IDNA)," RFC 5892, August 2010,
<https://www.rfc-editor.org/info/rfc5892>
- [RFC5893] Alvestrand, H., Ed., and C. Karp, "Right-to-Left Scripts for Internationalized Domain Names for Applications (IDNA)," RFC 5893, August 2010,
<https://www.rfc-editor.org/info/rfc5893>.
- [RFC5894] Klensin, J., "Internationalized Domain Names for Applications (IDNA): Background, Explanation, and Rationale," RFC 5893, August 2010,
<https://www.rfc-editor.org/info/rfc5894>.
- [RFC6912] Sullivan, A., et al., "Principles for Unicode Code Point Inclusion in Labels in the DNS," RFC 6912, April 2013. = IABCP,
<https://www.rfc-editor.org/info/rfc6912>.
- [RFC7940] Davies, K. and A. Freytag, "Representing Label Generation Rulesets Using XML," RFC 7940, August 2016,
<https://www.rfc-editor.org/info/rfc7940>.
- [RFC8228] A. Freytag, "Guidance on Designing Label Generation Rulesets (LGRs) Supporting Variant Labels, August 2017,
<https://tools.ietf.org/html/rfc8228>.
- [RZ-LGR] ICANN, Root Zone Label Generation Rules (RZ-LGR),
<https://www.icann.org/resources/pages/root-zone-lgr-2015-06-21-en>.
- [UAX#9] The Unicode Consortium, "UAX #9: The Unicode Bidirectional Algorithm". An integral part of The Unicode Standard. Most recent version available from
<http://www.unicode.org/reports/tr9/>.
- [UAX#31] The Unicode Consortium, "UAX #31: Unicode Identifier and Pattern Syntax". An integral part of The Unicode Standard. Most recent version available from
<http://www.unicode.org/reports/tr31/>.
- [UTS#39] The Unicode Consortium, "Unicode Technical Standard #39: Unicode Security Mechanisms" Most recent version available from,
<http://www.unicode.org/reports/tr39/>.