Confusingly Similarity

Current rule:

String confusion exists where a string so nearly resembles another visually that it is likely to deceive or cause confusion.

Proposed Rule:

Visually confusable strings refer to two different strings of Unicode characters whose appearance in common fonts in small sizes at typical screen resolutions is sufficiently close that people who are unfamiliar with the script easily confuse one for the other.

Questions relating to description

Question 1: which of these cases of confusion are considered relevant i.e. should be addressed?
Suggestion is to include it explicitly in description.

Examples of user confusion:

- Latin script user confuses Latin script string with a string in Cyrillic (Person who is familiar with script confuses string with a string in another script he or she is not familiar with).
- Latin script user confuses string in Greek with string in Latin script (Person who is not familiar with script confuses string in scrip he is or she is familiar with).
- Latin script user confuses string in Greek script string with Cyrillic script string (Person confuses string in script he or she is not familiar with string in another script he or she is not familiar with).

Question 2: What is impact of case mapping on criteria "common fonts in small sizes"?

Should common fonts in small sizes be read as in lower case or does "small sizes" refer to the font size?

(RFC 5894 is not an Internet Standards Track specification; it is published for informational purposes):

4.4. Case Mapping and Related Issues
In the DNS, ASCII letters are stored with their case preserved. Matching during the query process is case-independent, but none of the information that might be represented by choices of case has been lost. That model has been accidentally helpful because, as people have created DNS labels by catenating words (or parts of

words) to form labels, case has often been used to distinguish among components and make the labels more memorable.

Since DNS servers do not get involved in parsing IDNs, they cannot do case-independent matching. Thus, keeping the cases separate in lookup or registration, and doing matching at the server, is not feasible with IDNA or any similar approach. Matching of characters that are considered to differ only by case must be done, if desired, by programs invoking IDNA lookup even though it wasn't done by ASCII-only DNS clients. That situation was recognized in IDNA2003 and nothing in IDNA2008 fundamentally changes it or could do so. In IDNA2003, all characters are case folded and mapped by clients in a standardized step.

Even in scripts that generally support case distinctions, some characters do not have uppercase forms. For example, the Unicode case-folding operation maps Greek Final Form Sigma (U+03C2) to the medial form (U+03C3) and maps Eszett (German Sharp S, U+00DF) to "ss". Neither of these mappings is reversible because the uppercase of U+03C3 is the uppercase Sigma (U+03A3) and "ss" is an ASCII string. IDNA2008 permits, at the risk of some incompatibility, slightly more flexibility in this area by avoiding case folding and treating these characters as themselves. Approaches to handling one- way mappings are discussed in Section 7.2.

Because IDNA2003 maps Final Sigma and Eszett to other characters, and the reverse mapping is never possible, neither Final Sigma nor Eszett can be represented in the ACE form of IDNA2003 IDN nor in the native character (U-label) form derived from it. With IDNA2008, both characters can be used in an IDN and so the A-label used for lookup for any U-label containing those characters is now different. See Section 7.1 for a discussion of what kinds of changes might require the IDNA prefix to change; after extended discussions, the IDNABIS Working Group came to consensus that the change for these characters did not justify a prefix change.

Question 3: What are “common fonts”? Can this be determined or demarcated?

Question 4:

Should the test for determining whether similarity exists be included in the policy?

From the Fast Track Implementation Plan and blog on confusingly similarity

The determination whether strings present a risk for user confusion is done on a case-by case basis and takes into consideration among others, but is not limited to, the documentation provided by the IDN ccTLD requester, and language/equivalence tables available.

String confusion issues can involve two or more strings that are visual identical or are so visual similar that for reasons of stability and security they cannot coexist in the DNS, such as:

- Requested IDN ccTLD strings against any combination of two ISO 646 Basic Version (ISO 646-BV*) characters[1].

- Requested IDN ccTLD strings against existing generic TLDs and reserved names;
- Requested IDN ccTLD strings against other IDN ccTLD strings; and
- Requested IDN ccTLD strings against applied-for gTLD strings.

To avoid the risks of string confusion with any combination of two ISO 646 Basic Version (ISO 646-BV) characters (letter [a-z] codes), a conservative standard is used to assess confusability. To determine the risk of confusability with two letter in the the following ranking is used:

[6] Both characters are visually identical to an ISO 646 Basic Version (ISO 646-BV) character.

[5] One character is visually identical to, and one character is visually confusable with, an ISO 646-BV character.

[4] Both characters are visually confusable with, but neither character is visually identical to, an ISO 646-BV character.

[3] One character is visually distinct from, and one character is visually identical to, an ISO 646-BV character.

[2] One character is visually distinct from, and one character is visually confusable with, an ISO 646-BV character.

[1] Both characters are visually distinct from an ISO 646-BV character.

To avoid the risks of string confusion with existing TLD strings: Assessments are made in the String Similarity Evaluation Process for IDN ccTLD requests and in the Initial Evaluation step for new gTLD applications[2]. The following rules provide the thresholds for solving any identified contention issues between the two processes:

[1] International Organization for Standardization, "Information Technology – ISO 7-bit coded character set for information interchange," ISO Standard 646, 1991

[2] This implies that the initial evaluation under both the IDN ccTLD and new gTLD has to be conducted by one and the same entity.

A. A gTLD application that is approved by the ICANN Board will be considered an existing TLD in inter-process contention unless it is withdrawn.

B. A validated request for an IDN ccTLD will be considered an existing TLD in inter-process contention unless it is withdrawn.

For the purpose of the above contention rules, an IDN ccTLD string request is regarded as validated once it is confirmed that the string is a meaningful representation of the country or territory and that the string has passed the DNS Stability review process.