IDN Variant Issues Project
Initial Definitions Document

July 15, 2011

Note to readers

The Initial Definitions document has been developed by the ICANN IDN Variant Issues Project (VIP) team to provide a set of definitions of the terms to be used in the project.

The definitions are culled from various public sources, and are provided to the project's six -- Arabic, Chinese, Cyrillic, Devanagari, Greek, and Latin -- community based case study teams.

The purpose of the definitions document is to define and reserve a number of unambiguous terms. The definitions are not required to be used by the case study teams; but if the teams wish to use these terms, they are asked to use them as defined in this document and not in any other way. If the case study teams use these terms, they will be interpreted as they are defined here.

The VIP project team expects that there may be problems with these definitions. Indeed, early feedback from the case study teams has indicated that the definitions may not be adequate.

These definitions have three changes compared to the original set of definitions provided the teams. First, Valid Code Point was clarified to indicate that evaluation of it is performed during registration. Second, Character Variant Label is clarified to indicate that it corresponds to a Fundamental Label. Third, Preferred Variant Label is clarified to indicate that it corresponds to a Fundamental Label.

We invite ICANN community experts to advise on whether these definitions are correct or not, or badly defined, or inadequate for use in dealing with IDN Variant issues in various scripts.

Background

The ICANN Variant Issues Project is investigating issues around variants and the global DNS, with particular attention to the issues for top-level Internationalized Domain Names using IDNA2008 (and, therefore, Unicode).

The Project provides the following definitions as a starting definition both as part of the definition of terms used in the Questions for Case Study Teams, and as terms to be used across all Case Study Teams.

The Project asks that Case Study Teams use the terms only as defined below. If a term is defined in a way that the Team finds undesirable, or if there is not a term for something the Team needs to discuss, the Project prefers the Team to define a new term (if only by adding a subscript to the term as defined here), so that it is always clear what a given term means. The goal of this restriction is to minimize the difficulty later during the harmonization step.

To avoid reinventing the wheels, we borrowed / reused some definitions from the following documents:

- Internationalized Domain Names for Applications (IDNA): Definitions and Document Framework (RFC 5890[1])
- Terminology Used in Internationalization in the IETF (draft-ietf-appsawg-rfc3536bis-01[2])
- The Unicode standard including the standard annexes[3]
- Joint Engineering Team (JET) Guidelines for Internationalized Domain Names (IDN) Registration and Administration for Chinese, Japanese, and Korean (RFC 3743[4])

Definitions

Abstract Character: A unit of information used for the organization, control, or representation of textual data. (Unicode Standard, section 3.4, D7)

A-label: An ASCII-Compatible Encoding form of an IDNA-valid string. It must be a complete label: IDNA is defined for labels, not for parts of them and not for complete domain names. This means, by definition, that every A-label will begin with the IDNA ACE prefix, "xn--", followed by a string that is a valid output of the Punycode algorithm (RFC 3492) and hence a maximum of 59 ASCII characters in length. The prefix and string together must conform to all requirements for a label that can be stored in the DNS including conformance to the rules for LDH labels (See RFC 5390, Section RFC 2.3.1). If and only if a string meeting the above requirements can be decoded into a U-label is it an A-label. (RFC 5890)

Allocation: In a DNS context, the first step on the way to Delegation. A registry (the parent side) is managing a zone. The registry makes an administrative association between a string and some entity that requests the string, making the string a label inside the zone, and a candidate for delegation. Allocation does not affect the DNS itself at all.

[1] See http://tools.ietf.org/html/rfc5890.
[2] See http://tools.ietf.org/html/draft-ietf-appsawg-rfc3536bis-01
[3] See http://unicode.org/versions/Unicode6.0.0/
[4] See http://www.ietf.org/rfc/rfc3743.txt

Assigned Code Point: A mapping from an Abstract Character to a particular Code Point in the code space. See Unicode Standard, section 2.4. Not to be confused with Valid Code Point.

Character Variant: In a Language Variant Table, a second list of Code Points corresponding to each Valid Code Point and providing possible substitutions for it. Unlike the Preferred Variants, substitutions based on Character Variants are normally reserved but not actually registered (or "activated"). Character Variants appear in column 3 of the Language Variant Table. The term "Code Point Variants" is used interchangeably with this term. (RFC 3743)

Character Variant Label: A U-label generated from a Fundamental Label by use of Character Variants. The Character Variant Label must contain at least one Character Variant, but need not contain all the Character Variants possible for the Fundamental Label. This definition differs from that in RFC 3743 by specifying "U-label" rather than "label".

Code Point: A value in the Unicode code space. The meaning here is restricted to meaning D10 in the Unicode Standard, section 3.4.

Delegation: In a DNS context, the act of entering parent-side NS (nameserver) records in a zone, thereby creating a subordinate namespace with its own SOA (start of authority) record. See RFC 1034 for detailed discussion of how the DNS name space is broken up into zones.

Fundamental Label: A U-label that consists only of Valid Code Points. In practice, this is the U-label requested to be registered.

Fundamental TLD: The Fundamental Label form of a Variant TLD Set.

IDNA Symmetry Constraint: A-label/U-label transformation must be symmetric: an A-label A1 must be capable of being produced by conversion from a U-label U1, and that U-label U1 must be capable of being produced by conversion from A-label A1. (RFC 5890)

Language Character Repertoire: A set of Code Points identified by some identifier (such as a tag for identifying language as defined in RFC 5646). The definition of the Language Character Repertoire is ideally performed in a way appropriate to some community of language users, and might colloquially be understood as "the characters used to write a language". In most cases, all the Code Points in a Language Character Repertoire will come from the same Script Table.

Language Variant Table: A three-column table for each Language Character Repertoire permitted to be registered in a zone. The columns are known, respectively, as "Valid Code Point", "Preferred Variant", and "Character Variant", which are defined separately. (This definition differs from RFC 3743 in the subsitution of Language Character Repertoire for "language".) Note that in the rest of this document "Table" and "Variant Table" are *not* used as short forms for Language Variant Table, as they are in RFC 3743. Note also that it is logically possible a

U-label would be consistent with more than one Language Variant Table. What to do in such a case is a matter of registry policy.

Preferred Variant: In a Language Variant Table, a list of Code Points corresponding to each Valid Code Point and providing possible substitutions for it. These substitutions are "preferred" in the sense that the variant labels generated using them are normally registered in the zone file, or "activated." The Preferred Code Points appear in column 2 of the Language Variant Table. "Preferred Code Point" is used interchangeably with this term. (RFC 3743)

Preferred Variant Label: A U-label generated by use of Preferred Variants. The Preferred Variant Label must contain at least one Preferred Variant, but need not contain all the Preferred Variants possible for the Fundamental Label. This definition differs from that in RFC 3743 by specifying “U-label” rather than “label”.

Preferred Variant TLD: The Preferred Variant Label form(s) of a Variant TLD Set.

Reserved Variant TLD: The Character Variant Label form(s) of a Variant TLD Set.

Script Table: A Script Table is a table of Unicode Code Points all having the same script property value. See Unicode Standard Annex #24.

U-label: An IDNA-valid string of Unicode Code Points, in Normalization Form C (NFC) and including at least one non-ASCII character, expressed in a standard Unicode Encoding Form (such as UTF-8). It is also subject to the constraints about permitted characters that are specified in Section 4.2 of RFC 5891 and the rules in the Sections 2 and 3 of RFC 5892, the Bidi constraints in RFC 5893 if it contains any character from scripts that are written right to left, and the IDNA Symmetry Constraint. (RFC 5890)

Valid Code Point: In a Language Variant Table, the list of Code Points that is permitted at registration time for that language. Any other Code Points, or any string containing them, will be rejected. The Valid Code Point list appears as the first column of the Language Variant Table. (RFC 3743) Note that Valid Code Points are always both Assigned Code Points and Variant Members.

Variant Character Collection: All the characters listed in a single row of a Language Variant Table, as any of Valid Code Point, Preferred Variant, or Character Variant. (RFC 3743) It is important to recognize that the relationship may not be reciprocal (that is, if *foo* is a Valid Code Point and *bar* is a Character Variant, that does not mean that *foo* is a Character Variant for Valid Code Point *bar*).

Variant Label Set: A set of U-labels consisting of one Fundamental Label, zero or more Preferrred Variant Labels, and zero or more Character Variant Labels.

Variant Members: Code Points that appear in a Language Variant Table. The code point may appear in any of the Valid Code Point, Preferred Variant, or Character Variant positions.

Variant TLD: A Variant Domain Name Label corresponding to an A-label that appears or is intended to appear immediately below the root in the global DNS. Note that this definition includes TLDs that do not actually exist in the DNS at a given point in time. More informally, a Variant Domain Name Label that appears or intended to appear immediately below the DNS root. Because the actual labels in the DNS are all A-labels, this informal use is not strictly true; but because A-labels and U-labels are symmetric, it amounts to the same thing.

Variant TLD Set: A set of Variant TLDs consisting of one Fundamental Label, zero or more Preferred Variant Labels, and zero or more Character Variant Labels.

Other items to note

The definitions of variants above is different from other meanings of “variant” sometimes heard in an ICANN context. Those alternate meanings for the term “variant” include the following:

- Different abstract characters (see Unicode Standard section 3.4 D7) that may be visually confusable. Example: Arabic-Indic Digit Seven (U+0667, “٧”) and Latin Small Letter V (U+0076, “v”).
- Characters that are normally graphically identical, but that have different Assigned Code Points. Example: Cyrillic Small Letter A (U+0430, ”a”) and Latin Small Letter A (U+0061, “a”).
- Orthographic differences within a language. Many languages have alternate choices of spellings or spellings that differ by locale. Users of those languages generally recognize the spellings as equivalent. An example is "color" and "colour" in English.

Of course, different choices of fonts or other forms of visual representations, including handwriting, can cause any pair of characters in any language to look more or less alike. For example, while Greek Lower Case Alpha (U+03B1) is usually considered visually distinguishable from the Latin and Cyrillic Lower Case A characters mentioned above, there are typefaces for basic Latin characters that make the lower case A look almost like Greek Lower Case Alpha. None of these other meanings of “variant” are the kind of variant discussed in the ICANN Variant Issues Project.