Internationalizing Top-Level Domain Names: Another Look

ISOC MEMBER BRIEFING #18

September, 2004

by John Klensin

Abstract:

Over the last few years, rising interest in internationalized domain names has been accompanied by interest in using those names at the top level and, in particular, replacing or supplementing country-code based domain names with names in the language of the relevant countries. This memo suggests that actually creating such names in the DNS is undesirable from both a user-interface and DNS management standpoint. It then proposes the alternative of translating the names so that every TLD name is available to users in their own languages.

An observation is made in many cultures that one should understand the problem one is trying to solve before solving it. The conventional wisdom is usually that this produces better results and a solution that is a better match to the actual problem than a "solution first" approach. This article examines the question of top level DNS names that are expected to reflect the language of the user from that user standpoint and suggests a way to approach the problem that may be better – and serve a wider range of users – than the perhaps more obvious approach.

Restating the Problem

It is widely understood that users whose native languages are not English –or, more specifically, do not use simple Latin-based scripts—should be able to navigate the world wide web and email systems using their own languages and scripts. That implies the use of domain names that are entirely in those scripts since the notion of, eg., ГЛУПЫИ.ОТВЕТ.RU looks strange and inappropriate to almost everyone. From a user perspective, however, the issues are all about what is seen and typed, not what is in the DNS or the visual form of the URL. Even with IDNs, while奇怪. 概念..com might be a valid external ("native script") form of a name, current standards for email and URLs require that, to actually be used, it be written, e.g., as username@xn--mts68mse018a or, if it were part of an HTTP URL, as http://xn--mts68mse018a/. Those existing restrictions on email addresses and URIs (including URLs), also imply additional standardization work and deployment to internationalize the addresses and resource locators themselves. That work is in progress, but there are many difficult issues and it is not clear how quickly it will converge. In other words, internationalization of domain names may be a necessary requirement, but it is not sufficient to permit users to use the Internet in their native languages and scripts only.

If a user sees email addresses or URIs that contain non-ASCII characters today, and these work successfully without violating the standards, it is because user interface software is accepting characters in the script preferred for some other language and then mapping or transforming them

Expanded Coverage from ISOC
In-depth articles, papers, links and other resources on a variety of topics are available from the ISOC site at: www.isoc.org/internet/issues

Relevant IETF RFC's
RFC 3696 Application Techniques for Checking and Transformation of Names
RFC 3490 Internationalizing Domain Names in Applications (IDNA)
RFC 3491 Nameprep: A Stringprep Profile for Internationalized Domain Names (IDN)
RFC 3492 Punycode: A Bootstring encoding of Unicode for Internationalized Domain Names in Applications (IDNA) RFC 3454 Preparation of Internationalized Strings (stringprep)

IESG Statement
When the IDNA RFCs were approved as Proposed Standards and for publication, the IESG issued a statement that put that work in context. That statement has been posted at http://www.ietf.org/IESG/STATEMENTS/IDNstatement.txt.

into something the protocols will accept. It is important to understand that and its relationship to the IDN standards: they too, operate in the software on the user's machine and convert local characters and scripts to an especially-coded form; non-ASCII characters are not stored in the DNS or used in queries to it.

This paper suggests that, for almost all issues involving internationalization of the Internet, the correct question is "what should the user see (or enter) and what is the best way to accomplish that?" If, instead, we concentrate first on questions involving low-level functions in the network, such as "how do we change the DNS", we are likely to discover, as some people have discovered with IDN deployment to date, that they don't get quite the functionality they want and expect.

The Requirement for non-ASCII TLDs

As suggested in the discussion above, it is important to be able to write the names of TLDs, especially country-associated TLDs, in languages and scripts associated with those countries. From a user perspective, a reference to a web site which is located in China, whose content is in Chinese, and which uses Chinese labels for most of the domain name, should be, as much as possible, entirely in Chinese. We need to understand that this is not possible at a protocol level: the "http" in a URL is the name of a protocol; were it translated into a different language, it would be a different protocol.

But suppose a Russian-speaking community in Paris, or a Chinese-speaking one in San Francisco, registered second-level names in their own languages using the IDNA standards and, potentially, added more IDNA-based names at the third level and below. The same arguments would apply: users of those domains would presumably prefer to be able to reference .FR or .US (and, potentially, .BIZ, .COM, .INFO, and so forth) by using their own languages and scripts. If we see that as a useful goal, then the question is how best to accomplish it. Even were there were satisfactory mechanisms in the DNS for creating aliases for domain name trees, permitting each TLD to be utilized with a spelling in all of the world's languages would require many hundreds of aliases for each TLD.

DNS aliasing mechanisms that would be suitable for this purpose and for intense use do not exist. While an explanation is beyond the scope of this discussion, there are technical reasons involving the internal structure of the DNS why any form of such aliases would be problematic.

In principle, one could create extra TLDs, one corresponding to each desirable language for each TLD. If one assumed that countries (defined as entities now holding ccTLDs) would only require an average of two or three language-based additional TLDs each, and that gTLDs would not need versions in other languages, that would add on the order of 500 extra domains. The estimated value of two or three derives from two observations. First, while many countries have only one official language, some have several. Even among countries with only one official language, there might be considerable pressure to create domains in the languages of large minority groups, pressure that would be hard to resist if we retain our focus on the users and their needs. Second, the ISO 3166-1 code list on which the country code TLDs are based contains alphabetic codes in a subset of Roman-based characters only. As the group maintaining the ISO 3166 standard has pointed out, "translation" of the codes themselves – as

For More Information
"National and Local Characters in DNS TLD Names", http://www.ietf.org/internet-drafts/draft-klensin-idn-tld-03.txt, contains a discussion of this issue from a somewhat more technical point of view.

Related Organizations
www.ietf.org
www.rfc-editor.org

About the Author

Dr. John C. Klensin is now an independent consultant following a distinguished career as Internet Architecture Vice President at AT&T, Distinguished Engineering Fellow at MCI WorldCom, and Principal Research Scientist at MIT.

He served on the Internet Architecture Board from 1996-2002 and was its Chair from 2000 until the end of his term. Earlier, he served as IETF Area Director for Applications and was Chair, Co-chair, and/or Editor for IETF Working Groups focused on messaging and IETF process issues.

He was involved in the early procedural and definitional work for DNS administration and top-level domain definitions and was part of the committee that worked out the transition of DNS-related responsibilities between USC-ISI and what became ICANN. He distinction from translation of the official country names that also appear in the standard – is essentially meaningless. The codes are codes, not strings in some particular language. This suggests that the "multilingual" names of TLDs in would presumably be full country names or abbreviations of those names. Once that were permitted, even countries whose official names are normally written in Roman characters and whose ISO 3166-1 codes are obvious abbreviations for those names would undoubtedly find it desirable, and fair, to acquire at least an additional TLD based on a complete spelling or abbreviation of the country name.

It is unrealistic, however, to believe that additional names would not be wanted, and reasonably so, for some or all of the non-country TLDs. Equally important, we should apply the reasoning above and focus, not on the names on countries or domains as seen by their operators, but on users being able to utilize the Internet conveniently in their own languages. That would require each existing or newly-created TLD to be available in each language. If we assume that there were only two or three hundred such languages today – certainly low estimates – this would imply over 50 or 60 thousand TLDs just to support the current number of registries.

Of course, if we had all of those TLDs, the arrangement of them would raise some other issues. If they are operated independently, with the existence of a (second-level or below) domain in one not implying anything about its presence in a TLD that represented a translation of the same name, the problem for a user trying to guess which domain for a particular country or gTLD-concept to use would be immense. But linking them together would be very difficult or impossible. First, there is no reason to require that all of the subdomains of a domain name written in a particular language or script also be in that script. And second, whatever the cognitive or linguistic linkage between a particular subdomain of one TLD and a subdomain of another, they are separate as far as DNS administration is concerned, leading to many opportunities for confusion and differences in behavior as definitions and subdomain structures evolve in different ways for the different second-level domains and their subdomains.

It should be obvious by this point that permitting users to reference TLDs in their own scripts, or in the same script used for the associated second or third level domains, by installing non-ASCII synonymous TLDs would cause significant difficulties. Those difficulties might be worth accepting if there were no better alternative and if the extra TLDs also solved the rest of the URI or email problems. The latter is not the case. And, fortunately, there is a better alternative.

It is all about the User Interface

As discussed above, the presentation and interpretation of strings being used in email addresses, web locators, and other references is largely up to the user interface software and need be only loosely coupled with the protocols used over the network. It is still important to have some standardization of the forms of the strings (as presented) in order that users be able to share them with each other independent of their software environment. That issue is just an instance of a classic set of tradeoffs about optimizing interfaces. For example, assuming that either can be made to "work", optimizing an interface for a particular homogeneous group of users makes that interface less convenient for others, while designing the same interface for international use tends to make it more or

became ICANN. He participated actively in a number of efforts to expand internationalization of the Internet, ranging from early character set requirements work, through the efforts to permit and identify non-ASCII character sets in email, and including IDN efforts in the IETF, ICANN, and elsewhere.

He was also involved in the design and development of the Internet's email system from its beginnings, and has been heavily involved in internationalization issues, for the Internet and in other areas, for many years.

Acknowledgments
The ISOC Member Briefing series is made possible through the generous assistance of ISOC’s Platinum Program Sponsors: Afilias, APNIC, ARIN, Microsoft, and Ripe NCC, Sida. *More information on the Platinum Sponsorship Program*: http://www.isoc.org/isoc/membership/platinum.shtml

less compatible with everyone but not optimal for anyone. This is the reason why it has also been observed that, at the user interface level, no one wants internationalization: internationalization is only a useful tool for constructing versions of interfaces that localized to local language, script, and cultural habits.

In a localization model in which browsers, mail user agents, and other applications software are tailored to the needs and preferences of the local user, another solution emerges to the multilingual TLD requirement. That solution is to translate or map TLD names locally, rather than trying to make language-specific names global, and is described in the next section.

Translating TLD Names

Fortunately for the designer of user interfaces, there are only about 260 country-code TLD names, and another 14 generic names. The country-code list changes only very slowly. ICANN plans for the generic name list to grow moderately, but not dramatically, in the foreseeable future. Maintaining a translation table in which around 300 names are kept together with convenient local forms is a fairly simple matter of programming. In general, user interface software would examine a presumed TLD name and, if it were in the local character set, attempt to translate it to the standard (ASCII) form using that table. Similarly, it would be feasible to translate standard-form names to local ones for user convenience.

The implications of this approach is that a user in China could not only refer to the .CN domain in Chinese, but could also refer to the .FR, .US, or .MUSEUM domain in Chinese. Similarly, a user in France could refer to .CN using a French name for China, and so on for every other country, language, and TLD. Moreover, in France, the local name for Japan in French is much more useful for most users than the name of Japan in Japanese characters.

Limitations

This approach would not work for second-level domains or domains further down the tree: there are too many of them and they change too quickly. But, in general, IDNs are a satisfactory solution for those domains. They are usually spoken of as separate domains with a non-ASCII name, rather than as an alias or additional name for an existing domain.

As with any attempt to localize, or otherwise optimize a system for use within a specific community, the technique proposed makes global interoperability more difficult. Just as is the case with IDNs themselves, the user sees strings that are not the ones being passed across the network and that may not be globally comprehensive. If a user of one language passes a domain name containing IDNs that are expressed in their native script to another user, the second user may not be able to read them or key them back into a computer and, at least with the state of the technology today, a cut-and-paste operation on the characters from, say, an email message, may or may not work as intended. So a user who wishes to pass an IDN to a user of a very different language, in a different part of the world, is be well-advised to pass the IDNA "punycode" form of the name, at least as supplemental information. For TLD names handled according to this proposal, users will need to be aware that only the

About the Background Paper Series

Published by:
The Internet Society
1775 Wiehle Avenue, Suite 102
Reston, Virginia 20190 USA
Tel: +1 703 326 9880
Fax: +1 703 326 9881

4, rue des Falaises
CH-1205 Geneva
Switzerland
Tel: +41 22 807 1444
Fax: +41 22 807 1445

Email: info@isoc.org
Web: http://www.isoc.org/

Series Editor: Martin Kupres

standard, ASCII, form of the TLD names will be generally usable outside the local environment, just as any local form of, e.g., protocol names in URLs would need to be translated back to the standard form before being transmitted to foreign users.

Summary

This document proposes an approach for presenting and processing native language TLD names that provides a better user experience and less load on the DNS than trying to install multiple names for each domain in the DNS itself. The approach also avoids many complications in DNS administration and interoperation.

For Additional Reading

"National and Local Characters in DNS TLD Names", http://www.ietf.org/internet-drafts/draft-klensin-idn-tld-01.txt, contains a discussion of this issue from a somewhat more technical point of view.