

Reprise: Early technical IDN approaches

- **“Just use” UTF-8 or 8859-N or GB2312, or Big5, or KOI-8, or...**
- **Tagging problem w/ DNS**
- **The IDNA Approach**
 - **Name format no one uses.**
 - **Efficient for script-homogeneous strings (UTF-7 and UTF-8 are not, especially for East Asian characters)**

Some DNS physics

- DNS performance depends critically on caching “near” the site of the query
- Consistent and predictable DNS operations depends on caching only complete RR sets
- All known-possible methods for guaranteeing integrity of DNS data, including DNSSEC, are quite sensitive to non-conforming handling of queries and responses.
- “Trick servers” are, to at least some extent, a problem for each of these.

Problems Internal to IDNA and Issues It Does Not Address

Nameprep Issues

- Eliminates/normalizes some lookalikes & font forms
- Try to preserve case-mapping rule
- Cannot be completely successful partially due to characters shared among scripts or languages but used differently
- Unavoidably does one-way mappings badly (e.g., a German IDN may be registered with ä, ö, or ü, but not ß)
- Important to understand that these properties are the result of tradeoffs – the alternatives are worse.

Applications Issues

- Email addresses
 - Local-parts more important than domain-part?
 - DNS advantage with LDH
 - Unrestricted local-parts, so ACE-like encoding cannot be completely safe
 - Envelope – header (transport) issues
- URL definition
 - Strict ASCII
 - IRI proposal and `http://...`
 - Status of IRIs

Traditional DNS: What Goes In, Comes Out

- Case-insensitive mapping
 - If “A” is registered, a query for “a” matches, but returns “A”.
- With IDNA,
 - “Ü” can be looked up, but not registered
 - If “ü” is registered, but the query is for “Ü”, the query will match, but “ü” will be returned.

depending on the application, this difference may result in some user astonishment.

Unicode Complications

- **Unified CJK**
- **Separate European**
- **Font-specific chars**

IDNA helps with some of this, but not much

Traditional and Simplified Chinese

- **Characters with semantics**
- **Relationship to case mapping**
- **Cannot process Kanji and get Simplified Chinese**

The Character Variant Model

- **JET: Registry restrictions, variants, and reserved strings**
 - Adoption in CJK ccTLDs
 - No actual variants, yet, in two of them.
 - Analogies to alphabetic languages
- **The ICANN Guideline**
 - Language base
 - Registration of tables
- **Implementations and Issues**

Dispute Resolution or Conflict Prevention

- Key principles
- Character variants and other evolving systems: prevention of conflicting/confusing registrations
- Dispute resolution policies and mechanisms: “register first, then straighten it out”

Variant Roman Character

Example

- Suppose we have two people with surnames
Müller and Quinoñes
- And they have historically registered the obvious
ASCII domain labels
Mueller and Quinones
- Now, when IDN registrations are permitted,
should others be permitted to register the IDNs
with the correct spellings, or should those names
be reserved? If not, how is the restriction
managed?

The Meaning of “Language”

- JET, IETF, ICANN, etc., use the term “language” to describe tables and rules.

this is *not* the normal usage

The Meaning of “Language”

- Really Zone-Language-Script
 - No one really knows what the limits of a “language” are, although governments can make decisions within their territories.
 - “Scripts” overlap in strange ways. Neither Unicode Consortium nor ISO have been able to rigorously define scripts associated with particular languages (there are some broad, descriptive, definitions)
 - For example., for some zones in Western Europe the appropriate language-script has been “generic European”, i.e., “Latin-1”. For others, more specific lists of characters may be needed.

Authoritative Policies about Scripts

International Bodies: Consensus about Language

- Authority
 - National sovereignty issue for ccTLDs
 - Rules generally cannot be enforced below level two or three (similar to trademarks)
 - International issue for gTLDs
- Scripts and Languages
 - If one script is used by several languages, language authority is not sufficient

Authoritative Policies about Languages

- If a good-quality recommendation is available, will registries use it?
 - Foolish not to: saves a lot of work, trouble, and looking silly
 - Compulsion is another matter
- Multiple-language scripts can be a major gTLD challenge

Major Issues with variant models

- “Multilingual” strings
- Labels and “names”
- Variant charging in JET-like models
 - Cost of a reserved label
 - Cost of activation given that the label has no value to anyone else
- DNS as an administrative hierarchy
- New types of conflict/ dispute problems

Technical Interoperability

- IDNA is entirely a client algorithm and procedure, hence depends on correct client implementations and is hard to verify.
- JET Guidelines and similar approaches are registry-dependent
 - They do not raise interoperability issues.
 - May raise user experience ones

Administrative Hierarchy Issues

- Policy and trust relationships
- No cross-tree cross-references to branches of hierarchy
- Maintaining parallel trees
 - Workable if really identical and have a single coordinating database.
- Organizational branding
 - <http://www.product.tld/> or
 - <http://www.organization.tld/product>

New Dispute and Resolution Issues

- ICANN-WIPO UDRP assumes
 - Homogeneous scripts and language characters
 - Conflicts about rights to identical names
- but not...
 - Labels constructed from line or box-drawing characters
 - Look-alike characters and strings from different scripts unless they meet trademark-like criteria for “confusingly similar”
 - Translations, transcriptions, transcodings
- Is the relevant “name” the IDNA encoding or its display/presentation form?

Problems IDNs Don't Solve

- Registration policy issues
 - “This language is more important”
 - The gTLD problem
- Applications and local character sets
- Even JET Guidelines won't eliminate all confusion, just some of it
- DNS is a poor “search” mechanism... and getting worse.

The Whois Policy Issues

- Registration in non-ASCII and data in ???
- Searching of a multilingual/ multiscrypt database
- Reading the records
- Information about variants and IDN
Package contents

Competition and Policy

- Policy tradeoff between
 - More flexibility of registrations
 - Less risk of conflicts, deception, or fraud
- Each domain or zone will need to develop its own policy, and there will probably be wide variations.
- Implications of a country deciding to go its own way with, e.g., local character codings.
- User-exposed punycode between people using very different scripts is probably forever.

What was that Problem Again?

- **Domain-name guessing is becoming less useful**
 - Effectiveness reduced with more names
 - Effectiveness reduced with more possibly-relevant TLDs
- **Guessing in a multiple script (“multilingual”) environment will be *much* harder.**

The Application Interface Problem and Unicode

- Windows, Internet Explorer, Outlook, and...
 - Winsock and UTF-8 conversion of UTF-8
 - Localized versions with local character codings and different behavior
- Better if you have a Mac
- Maybe better if you have a Unix or Linux system
- Windows may get fixed, but not this year

Global Interoperability Again

- Giving up the ideas of
 - Any two Internet users being able to communicate, regardless of language
 - Any Internet user being able to access any public host, using a globally-available namewould make many of these problems much easier, but...
- It would be a high price to pay.

For some of us...

This is where
“being frightened”
will rapidly give way to
“being depressed”

The Cure for that Depression

Working cooperatively with each other to both

- internationalize and
- preserve global interoperability

And We Still have not Solved The Problem

- If IDNs are this hard
and do not solve the problem
– and slogans do not solve it either
- Maybe it is time to go back to the problem
and do some serious thinking about models
and approaches.

Questions for Thought

- Several studies indicate that search engine use is rising rapidly and even replacing name-guessing in some areas. Does that suggest opportunities?
- Can we get past the marketing hype, scaling problems, and need for a name-conflict “judge” and take another look at alternate naming systems with fewer constraints about characters and cross-references than the DNS?

(More) Questions for Thought

- Is it time to look again at “yellow pages”-like systems, perhaps with the multihierarchical structure of contemporary classification systems, as an alternative to both the DNS and search engines for some purposes?
- Are IDNs of primary importance for communication within a country or language rather than between them? Can we accept the use of Roman-based characters – or even ASCII or IA4 – between language groups?

(Still more) Questions for Thought

- Should we be giving serious consideration to inter-language translation of DNS names in applications in addition to IDNA mapping to and from DNS names in those applications?
- If IDNA had been designed with knowledge of the registry restriction and variant models, would its mappings and restrictions be the same? If not, is it too late to fix?

Major Issues We Have Barely Touched

- Email addresses
- Names and domains in digital certificates
- A fully internationalized alternative to the URL or URI
- Special problems with “multilingual” TLD names
- Hundreds or thousands of other protocols and how to internationalize applications that use them
- Finding and navigating to resources with non-ASCII names
- User interface issues

Summary – The Protocol Foundation

- From a technical/ protocol standpoint, IDNA is ready to deploy today and being deployed.
- IDNA is ultimately rooted in Unicode, which can represent, in some plausible way, almost every character in contemporary use for writing a language in today's world.
- IDNA is essentially a coding standard, not a “solution”.

Summary – The Policy Challenge

- Interesting issues and opportunities are best found by examining the user experience at the application interface: putting names in the DNS and getting them out is easy and always has been.
- Avoiding or dealing with confusion and name conflicts will require a good deal of thought.
- Whatever is done, must be done with great sensitivity to cultures and traditions
- It may be time to think about “non-DNS” or “above-DNS” approaches that really do solve the problems.

Internationalization of the Internet

A Great Opportunity

A Great Risk of Fragmentation

and a Great Challenge for all of us.

Internationalization of the Internet

A Great Opportunity

A Great Risk of Fragmentation

and a Great Challenge for all of us.

References and Additional Reading

- The IDNA Standard: Faltstrom, P., Hoffman, P. and A. Costello, "Internationalizing Domain Names in Applications (IDNA)", RFC 3490, March 2003; Hoffman, P. and M. Blanchet, "Nameprep: A Stringprep Profile for Internationalized Domain Names (IDN)", RFC 3491, March 2003; Costello, A., "Punycode: A Bootstring encoding of Unicode for Internationalized Domain Names in Applications (IDNA)", RFC 3492, March 2003.
- Variant names and registry restrictions: 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; Klensin, J., "Registration of Internationalized Domain Names: Overview and Method", work in progress, July 2004 version is draft-klensin-reg-guidelines-04.txt.
- Uses and abuses of the DNS: Klensin, J., "Role of the Domain Name System (DNS)", RFC 3467, February 2003.
- A different view of the non-ASCII TLD issue: Klensin, J., "National and Local Characters in DNS TLD Names", work in progress, May 2004 version is draft-klensin-idn-tld-03.txt.