Internationalized Domain Names

preparing for APTLD non-technical training
Dubai, UAE
June 5, 2007

Tina Dam
Director IDN Program
ICANN

Email: tina.dam@icann.org
DNS functionality

- Domain Name Resolution Process:

http://www.example.tld

End-user / Client → Local Server

IP address returned

Local Server → Root Server

Root Server → .tld Server

.example.tld Server

End-user / Client

http://www.example.tld
Characters in the DNS

- The DNS can handle all US-ASCII characters
  - Examples:
    - (a...z), (0...9), (-)
    - ( ) SPACE
    - (!) EXCLAMATION MARK
    - ("") QUOTATION MARK
    - (#) NUMBER SIGN
    - ($) DOLLAR SIGN
    - (%) PERCENT SIGN
    - (&) AMPERSAND
    - (') APOSTROPHE
Characters for domain names

- All TLD registries have implemented the LDH rule
- Domain names can only contain:
  - (a,b,...z)
  - (0,1,...9)
  - (-)
- Before internationalization....
Why Internationalization?

- DNS handling US-ASCII character set
  - a natural choice at the time
  - no expectation to current commercial value
  - Unicode was not available

- IDNs a natural expansion for global usability
  - allow users to use domain names in local scripts
  - no need to learn US-ASCII
  - SLD IDN registration available across many TLDs
  - some applications have implemented IDNA
  - still need internationalization of TLD
Some IDN terminology

- **The A-label**
  - transmitted in the DNS protocol
  - ASCII-compatible (ACE) form of an IDNA string
  - Example: "xn--11b5bs1di"

- **The U-label**
  - should be displayed to the user
  - representation of the IDN in Unicode;
  - example "परीका " ("test" version in Hindi, Devanagari script)

- **LDH-label**
  - an all-ASCII label
  - obeys the "hostname" (LDH) conventions
  - not an IDN
  - example "icann" in the domain name "icann.org"
### Internationalization Overview

<table>
<thead>
<tr>
<th>Domain Names Based on ASCII / LDH Rule</th>
<th>IDN second level</th>
</tr>
</thead>
<tbody>
<tr>
<td>➢</td>
<td>➢ Internationalized top level</td>
</tr>
<tr>
<td>➢</td>
<td>➢ Application upgrades to get web access in local chars + IDN enabled emails…</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ASCII based browser/email clients/…</th>
<th>➢ Expected to continue to expand</th>
</tr>
</thead>
</table>

Content have been available in many languages for some time

- `example.test` ➔ `실례.test` and `실례.테스트`
  (stored form: `example.test` ➔ `xn--9n2bp8q.test` and `xn--9n2bp8q.xn--9t4b11yi5a`)

**Aim:** An internationalized Internet
Internationalization of the internet means that the internet is equally accessible from all languages and scripts.
IDNA – Protocol Functionality

• Domain Name Resolution Process:

1. User types in 실험.test in for example browser
2. 실험.test gets converted to codepoint
3. Case-folding and normalization
4. Stringprep filter
5. Punycode conversion → xn--9n2bp8q.test

IDNA is a client based protocol:
IDNA is the acronym for the IDN protocol, developed within the IETF and published in June 2003.

IDNA stands for:
- Internationalized Domain Names in Application.

Technical details are available in the IETF RFCs:
- RFCs 3490, 3491, and 3492

IDNA is currently under revision:
- RFC4690 and associated internet drafts suggesting revisions and solutions to some problems
- More about this later...
Historically the domain name you register is also the domain names stored and usable in the DNS.

This is changed with introduction of IDNs.

Usually the stored form usually gives no meaning:
- Example: فرسانهر.نفرية → xn--mgbtbg2evaoi.tld

However, there are exceptions:
- xn--gibberish - decodes into the Arabic characters ب٨٩٣
- xn--trademark - with different versions of trademarks
- This is coincidentally and hence not intentionally

xn-- prefix specifically designates a system called Punycode.

xn-- prefix indicates to application software that the label needs to be decoded back into Unicode for proper display to the user.
Why are we not there yet?

- Initial registration availability resulted in
  - visual confusion issues
  - damaging uniqueness principle of the DNS

- Different implementation in applications
  - security issues with IDNA2003
  - confusion of how to implement IDNA2003
  - different user experience per application
Why are we not there yet?

- display of `xn--mgbh0fb` instead of 株式会社
- display of `xn--mgb0dgl27d` instead of 株式会社
- display of `xn--1lqs71d` instead of 東京
- display of `xn--1lq90i` instead of 北京

→ Results in trademarks being displayed where the U-label version may be a different trademark

- more user confusion and fraud opportunity
  - Registration of mïcrosoft.<tld> ?

- Protocol implementation experience and review showed other problems...
Language and Script

Languages are used by humans to interact
- estimate 5000-7000 languages worldwide
- 100-200 are mainly used
- RFC3066 discusses languages in more detail
- Examples: Arabic, Greek, Portuguese

Script is a set of graphic characters used for the written form of one or more languages (ISO10646 definition)
- Examples: Arabic, Cyrillic, Greek, Han

Computers don’t understand languages instead any characters will have an associated code-point
Same Script Different Language Issue

- Language specific character issues
  - Jorgen = Jørgen = Jörgen in Danish, Swedish, Norwegian
  - But users don’t always think that o equal ø and ö
  - ø is LATIN SMALL LETTER o WITH STROKE (U+00F8)
  - ö is 'LATIN SMALL LETTER o WITH DIAERESIS' (U+00D6)

- Not possible to make generic rule at the protocol level
- Need for specific rules at TLD registry level

- Some registries have submitted character tables to the IANA repository to show variants
  - Example: the .se table displays that:
    - The letter Å is not considered to be a variant of the letter A...Earlier practice substituted AA, which is no longer recommended but will still be encountered

- http://www.iana.org
  - (link to IANA Repository at bottom left of main page)
Some languages can be expressed by multiple scripts:
- Eastern European and Central Asian languages can be expressed in Cyrillic or Latin characters.
- African and Southeast Asian languages can be expressed in Arabic or Latin characters.
- Other languages are written in a combination of scripts—Kanji, Kana, Romanji for Japanese & Hangul and Hanji for Korean.

Hence, same word, same language can be expressed in different ways:
- Some words can only be expressed using a single script.
- Some words are expressed by mixing of scripts.

Result is that script definition is very important and sensitive in terms of IDNs.
Proposed Revisions to IDNA Protocol

- Revising the IDNA protocol will
  - build an “inclusion” based model for determining what scripts may be used for IDNs
  - increase available blocks of characters, via process
  - less mapping is result of characters not allowed
  - Non-unicode version dependant
  - fixing R-to-L error in Stringprep

- The revision effort is being managed through the IAB/IETF

- The Basic Framework was published Sept-06
  - RFC4690
Evaluation of IDN TLD Capability

- Laboratory test of DNS resolver and root-server software (Autonomica, ICANN)
  - February report showed no negative effect in laboratory environment
- Procedure for inserting and managing top-level labels;
- Emergency removal procedure;
- Tolerance measure for activating emergency removal:
  - Public comment period (2-22 June 07)
  - Consideration by ICANN Board in San Juan (June 07)
Looking forward

- Replication of laboratory test in live setting
  - to restate the laboratory result
  - need root server and community participation
  - plan under development (draft posting mid June 07)

- Evaluation facility for end-users and application developers
  - ICANN has no mandate over application development
  - plan under development (draft posting mid June 07)
Looking forward

- **IDN Repository**
  - added functionality for search and display (by San Juan, June07)

- **UI-Apps-DNS full description (Geoff Huston, APNIC)**
  - to illustrate and identify potential issues

- **IDN Security Issues Study (SSAC)**

- **IDN Policy initiatives**
  - Primarily from GAC, ccNSO, ccTLDs, GNSO
Policies and Processes

- Currently ICANN follows:
  - ISO3166 list for ccTLD delegation
  - GNSO developed process for introduction of new gTLDs

- None are adequate for IDN TLDs

- ICANN staff does not develop or decide on policy
gTLD considered policy issues

- Aspects on introduction of IDN gTLDs in relation to new non-IDN gTLDs
- IDN aspects on Geo-Political Details
- Aspects relating to existing gTLD strings and existing IDN SLDs
- Aspects relating to existing SLD Domain Name Holders
- Specific Techno-Policy Details relating to IDN gTLDs Particular IDN aspects relating to Privacy & Whois Details
- IDN aspects on Legal Details
GNSO IDN WG agreements

- **Avoidance of ASCII-Squatting:**
  - E.g. a new non-IDN gTLD “.caxap”, if accepted, would prohibit the acceptance of a later application for an IDN gTLD “.caxap” (in Cyrillic script and meaning “sugar” in Russian).

- **GAC Consultation on Geo-political Impact**

- **Language Community Input for Evaluation of new IDN gTLD Strings**

- **One String per new IDN gTLD:**
  - except when there is a need to cover script-specific character variants of an IDN gTLD string

- **Limit Variant Confusion and Collision & Limit Confusingly Similar Strings**

- **Priority Rights for new gTLD strings and new domain names**
  - do not derive from existing strings / may derive from IPR rights

- **IDN gTLDs may face challenges/objections**
  - for instance based on claims of intellectual property rights (IPR)

- **Suggested Approach towards Aliasing:**
  - address aliasing as a policy issue, rather than technical

- **Single Script Adherence – Conformance to IDN Guidelines**

- **Dispute Resolution for Domain Names in new IDN gTLDs (UDRP is ok)**
Agreement that other considerations in limiting scripts are:

- Official/significant languages in a country exist

- An IDN gTLD registry should limit the degree of script mixing and have a limit for the number of scripts allowed for its domain names. Such limits, with justifications, should be proposed by the IDN gTLD applicant and be evaluated for reasonableness.

- In all IDN gTLD applications, the applicant should adequately document its consultations with local language authorities and/or communities. See also 4.1.3.

- The way to define language communities is not in the purview of the IDN-WG, but CNDC and INFITT (representing Chinese and Tamil language communities, respectively) are some models to consider.

- ICANN should consult with the relevant language communities if in doubt whether an IDN gTLD string is in compliance with relevant tables.