

Internet Operation (9)
IDN Pros and Cons
Advanced Internet Technology II by SOI Asia

26 Nov 2004

Yoshiro YONEYA <yone@jprs.co.jp>

Overview

- Background
- IDN Standards
- IDN Technical Operation
- IDN Service Implementation

Background

What is IDN?

- Internationalized Domain Name.
 - A protocol name standardized at IETF.
 - Most of current domain names are represented with ASCII alpha-numeric (*a-zA-Z0-9*) and hyphen (-) characters.
 - IDN was started as a technical challenge to represent domain name with not only ASCII but also NON-ASCII characters and accomplished.
 - The activity was initiated by Asian!

Why IDN?

- Increases of the Internet users who are not familiar with English.
 - Increasing demand for using their own language to access the Internet.
 - Easy to memorize, type in, etc.
- Drastic changes of usage of domain name.
 - Domain name is now used as not only host name but also signboard.

Drawback of IDN

- Loses global acceptability at end-user interface.
 - Hard to type in or display NON-ASCII characters without appropriate I/O devices and / or softwares.
- Cause impact to the operation.
 - Requires software update and / or additional processing.
 - Deployment issue.

IDN Standards

A series of RFCs

- RFC is a document that defines protocols used on the Internet.
 - Standardization organization is IETF.
 - IDN WG did the work.
- IETF published a series of RFCs as IDN standards on March, 2003.
 - RFC3490 (IDNA)
 - RFC3491 (NAMEPREP)
 - RFC3492 (Punycode)

Scope and priority of IDN standardization

- Standard track protocol.
 - Not to divide the global connectivity and communication of the Internet.
- Backward compatibility.
 - Compatibility with current DNS and application protocols to work with current Internet infrastructure.
- No localization.
 - Independent from certain regions, countries and / or languages.
 - Refer to existing universal standards.
 - Common framework essential to internationalization.

IDNA

(Internationalizing Domain Names In Applications)
RFC3490

- An architecture denotes how to process IDN.
 - Use Unicode which is upper compatible with ASCII as a character codeset.
 - Normalize internal representation of characters which has multiple code points such as upper/lower, full-width/half-width and composing characters, into a single representation to perform matching correctly.
 - Represent NON-ASCII characters which inputted or displayed at user interface as an ASCII Compatible Encoding (ACE) string on the Network.
 - Those processes be performed in application software.

Important point of IDNA

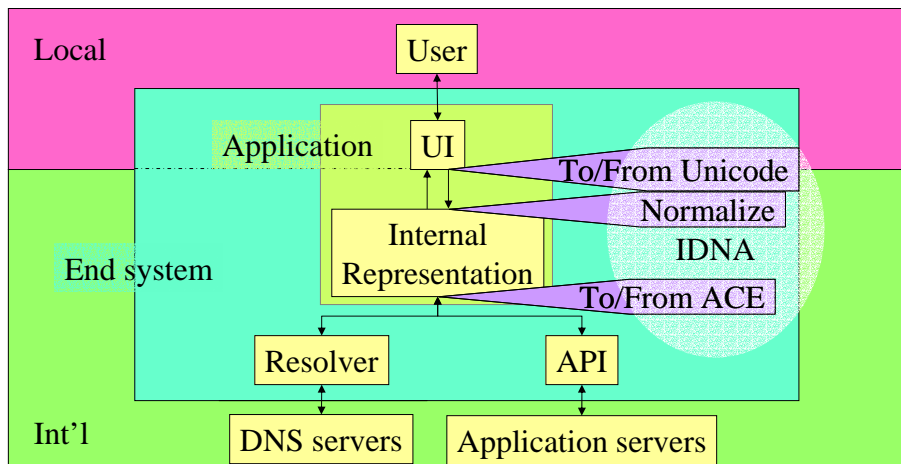
- Representation at the user interface layer and the network layer is different.
 - Though the same for ASCII domain names.
 - Internationalized form on display, ASCII form on the network.
 - To keep backward compatibility.
 - Comply with RFC2825 and RFC2826
- Application solution.
 - Least impact to the Internet infrastructure.
- Following domain names are equivalent (1 to 1 mapping) in meaning of IDN.

日本語ドメイン名 ↔ xn--eckwd4c7c777u7mwo4bc84j

Display

Network

Image of the IDNA



NAMEPREP

(Stringprep Profile for Internationalized Host Names)
RFC3491

- Profile for STRINGPREP (Preparation of Internationalized Strings)
 - RFC3454
- Some scripts such as alphabet have multiple representation for a character.
 - Domain name is case insensitive.
- Normalization process to unify representation of strings that is the same in meaning or displaying into a single representation.
 - Case (upper / lower)
 - Compatible character (full / half width)
 - Composing character (accent / combination mark)

Important point of NAMEPREP

- Normalize representation of Internationalized domain name string to perform matching correctly.
 - ‘a’ vs ‘A’
 - ‘u’+‘’ vs ‘ü’
 - ‘ア’ vs ‘ｱ’

Processes in NAMEPREP

1. map
 - Case folding of upper/lower characters (UAX#21)
2. normalize
 - Normalize representation of string (UAX#15's NFKC)
3. prohibit
 - Check out inappropriate character as domain name.
4. Bidi check
 - Check out inappropriate mixing of bidirectional characters.

ACE

(ASCII Compatible Encoding)

- Represent NON-ASCII characters by ASCII characters.
 - Easy to apply current DNS.
 - Least impact to current applications.
- Decreases maximum characters in each label.
 - Penalty of using only 5-6bit to represent 8bit data.
 - Requires some sort of compression algorithm.

ACE prefix

- An explicit ACE-identifier.
 - To recognize the string is ACE.
 - Important at reverse conversion.
 - 'XN--' was selected as ACE Prefix.
 - IANA did selection on 14 Feb 2003.
 - <http://www1.ietf.org/mail-archive/ietf-announce/Current/msg22619.html>
 - Denoted in RFC3490 (IDNA).

Criteria of ACE selection

- Simple algorithm.
 - For ease implementation.
 - Interoperability.
- Effective compression mechanism.
 - To accommodate characters as much as possible.
- Bilateral corresponding between encoding and decoding.
 - To avoid existence of alternative encoded representation for one IDN.
 - Security consideration.

Punycode

A Bootstring encoding of Unicode for IDNA
RFC3492

- ACE algorithm for IDN.
- Two key concepts.
 - Compression.
 - Encoding/Decoding.
- Compression algorithm.
 - Extract characters by ascending order of codepoint.
 - Encode difference of codepoint from previously processed character's and the position into an integer.
- ASCII conversion algorithm.
 - Introduced new concept named 'Generalized variable-length integers'.
 - BASE36 (A-Z, 0-9).

Compression process of Punycode (simplified for understanding)

- Use “文字列例” as an example.
 - Compression.
 - 0:U+6587 1:U+5B57 2:U+5217 3:U+4F8B
- placement
(0 origin)
- ↓
- Sort and take diff
- ↓
- To integer
(diff*chars+ position)
- ↓
- 3:0x4F8B 2:0x28C 1:0x940 0:0xA30
- ↓
- 81455 2610 9473 10432
- (1FTM) (20H) (7A5) (81R)
- └──────────────────────────────────┘
BASE36 (Sample)

Generalized variable-length integers of Punycode

- 12345 in decimal is represented as $1*10^4+2*10^3+3*10^2+4*10^1+5*10^0$
- Digits in all place are 0-9, so components in sequential 12345 cannot distinguish 123 and 45 or 1234 and 5.
- Furthermore, 012345 and 12345 are the same value with different representation.
- GVLI (Generalized variable-length integers) is an idea to solve this problem.
- Defines threshold for each place, and recognize a number below the threshold is delimiter.
- Threshold is an appropriate number smaller than base number.

Encoding process of Punycode (simplified for understanding)

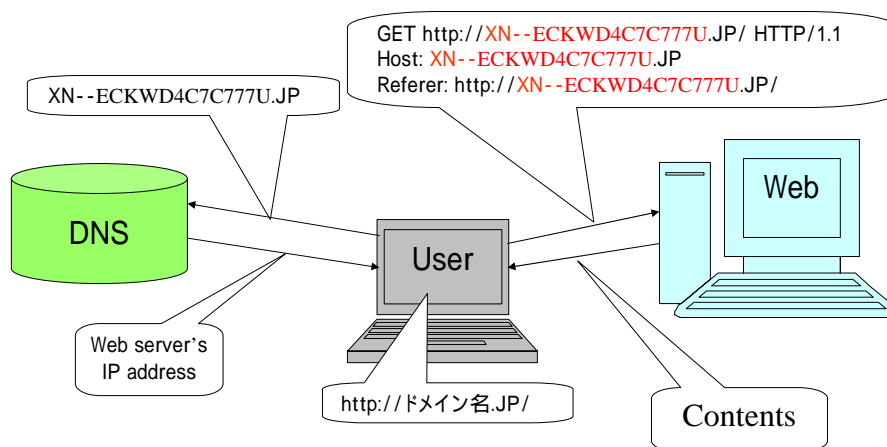
- Assign A-Z0-9 to GVLI.
 - Assume 36 for base, 10, 18, 25, 25 for thresholds.
- | | | | | |
|----|------------|------------------|-------------------|-------------------|
| 1. | 81455 | 2610 | 9473 | 10432 |
| | ↓ | ↓ | ↓ | ↓ |
| | 23*1+18*26 | 10*1+28*26+4*468 | 30*468 | 13*5148 |
| | ↓ | ↓ | ↓ | ↓ |
| 2. | NIUD | AS4 | | |
| 3. | | | 35*1+21*26+19*468 | |
| 4. | | | ↓ | 32*1+22*26+21*468 |
| 5. | | | | WML |
- 「文字列例」=>“NIUDAS4ZLJWML” (Pseudo Punycode)
 “1FTM20H7A581R” (BASE36)
 “FSQW5D78MBSK” (Real Punycode)

Standardization of IDN is just the start point of utilization

- End users uses IDN through application softwares.
 - Web, Mail, etc.
- IDNA requires application's correspondence.
- Must define how to deal IDNs in application protocols.

Standardization of IDN does not mean ready to use. Just a start point for applications incorporating new features.

Example of Web site browsing



Unresolved Issues

- The same characters, different characters.
 - Similar looking but different one.
 - A and , □ and □ and and □
 - Different looking but the same one (SC/TC).
 - 机 and 機, 叶 and 葉, 国 and 國
 - Language dependent issue.
- IDN-compliant application.
 - Vendors never be interested in without demand.
 - Recently, many of brand-new browsers are IDN-aware.
- Another identifiers.
 - Mail address, URI, and so on.
 - draft-duerst-iri-10.txt



IDN Technical Operation

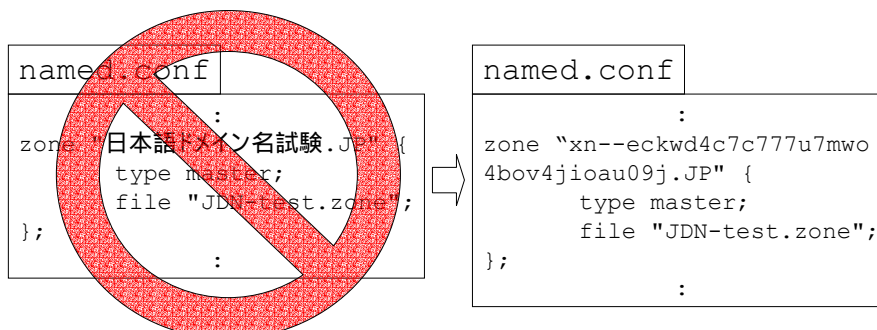


Overview

- No need to update name servers and / or resolvers.
- Local Encoding \leftrightarrow Punycode converter is required.
 - Such as idnconv in idnkit
- General procedure:
 - Edit configuration / zone file(s) using editors.
 - Convert encoding using converter.
 - Reload configuration / zone file(s) to name server.
 - Check the settings.

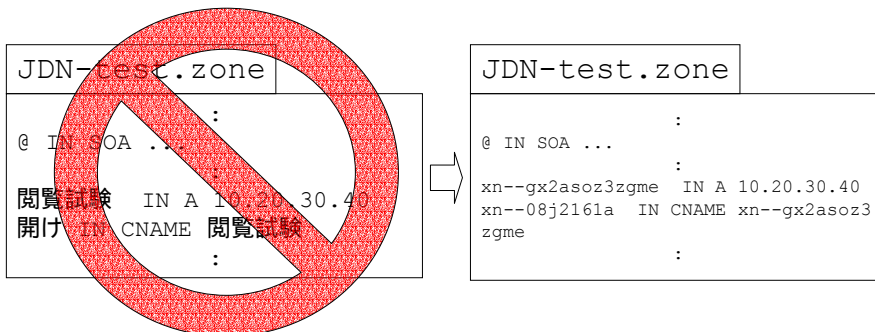
DNS Settings

(Example of named.conf of BIND)



DNS Settings (cont.)

(Example of zone file of BIND)



DNS settings (cont.)

(Example of BIND)

- Using Makefile makes maintenance ease.
- Divide editing file and configuration file, then generate latter one from former one.

```
.SUFFIXES:      .conf .conf-j .zone .zone-j
.conf-j.conf:
    idnconv $< > $@
.zone-j.zone:
    idnconv $< > $@
all:
    named.conf JDN-test.zone
```

How to check DNS settings

- Using DNS lookup tool and code converter:
 - Using shell alias or function is convenient
 - csh: `alias idig 'dig `echo ¥!* | idnconv`'`
 - bsh: `idig () { dig `echo "$@" | idnconv` }`
 - Output in ACE format


```
idig 日本語ドメイン名試験.jp
```
 - Output in IDN format


```
idig 日本語ドメイン名試験.jp | idnconv -r
```

How to operate other servers

- IDNs can be used with existing servers.
 - Web, mail, etc.
- Principle
 - IDNs MUST be represented with ACE format (Punycode) in configuration file.
 - Ex. Virtual host name in `httpd.conf`

IDN Compliant clients & implementations

- Netscape7.1 / Mozilla-1.4 or higher
 - <http://channels.netscape.com/ns/browsers/download.jsp>
 - <http://www.mozilla.org/products/mozilla1.x/>
- Mozilla Firefox-0.8 or higher
 - <http://www.mozilla.org/products/firefox/>
- Opera7.2 or higher
 - <http://www.opera.com/>
- i-Nav (A plug-in) for Internet Explorer 5 or higher
 - <http://www.idnnow.com/>
- JPNIC idnkit
 - <http://www.nic.ad.jp/ja/idn/mdnkit/download/>
- GNU libidn
 - <http://www.gnu.org/software/libidn/>
- VeriSign SDK
 - http://www.verisign.com/nds/naming/idn/sdk_form.html

IDN Service Implementation

Association with Language

- IDN itself has no language information.
- At registration service layer, IDNs **MUST** be associated with certain Language(s).
 - To reduce possible confusion due to [unresolved issues](#).
 - Define a Language table that includes:
 - Language Name
 - List of acceptable codepoints
 - Variants (if any)

IDN Registration Guideline

- An algorithm to be applied at an IDN registration.
 - Administration guideline for zone managers.
- Originally developed by Joint Engineering Team of CN, JP, KR, TW and a few Experts.
 - JET Guidelines for IDN Registration and Administration for CJK (RFC3743)
- And other related guidelines.
 - draft-klensin-reg-guidelines-05.txt

ICANN Guideline

- Conditions that ICANN permits IDN registration.
 - Targets are TLD registries with ICANN contract.
 - Following IDN Registration Guidelines.
 - Prohibits using marks.
 - Version 1.0 was published on 20 June.
 - <http://www.icann.org/general/idn-guidelines-20jun03.htm>

Case study of JP

- JPRS defined “Japanese Domain Name” and started its registration on Feb 2001.
- The first “Language Domain Name” registration service in the world.
- More than 40,000 registration.

Japanese Domain Name's registration technical rules

- 1st edition published on 06 Nov 2000.
 - Prior work to any IDN registration guidelines.
- Regulates:
 - Definition of Japanese Domain Name.
 - Maximum number of characters in a label.
 - Normalization rule at the registration.
 - Encoding for the resolution.
 - Reserved names and strings including ACE prefixes.
 - Acceptable characters' table for registration.
 - Referral standards.

ja-JP

- Derived work of the JDN technical rules.
 - According to IDN-Admin format.
- Registered as IDN Language Table at IANA.
 - <http://www.iana.org/assignments/idn/jp-japanese.html>

Reserved words

- Over 8000 of:
 - Government-and-municipal-offices names.
 - Names of Internet-related organization.
 - Suffixes indicating schools.
 - Prefecture names, big city names.
 - Normal words.
 - ... and so on

Duplication exclusion in sunrise and concurrent registration periods

- Sunrise period was to protect IPR.
 - One month period started at 22 Feb 2001.
- Concurrent registration period was to avoid rushing.
 - Three weeks period started at 2 Apr 2001.
- Lottery system for duplication exclusion.

Dispute Resolution Policy

- Preparation of DRP applicable to JDN.
 - JP DRP is defined by JPNIC.

References

- Unicode Consortium
 - <http://www.unicode.org/>
- Terminology Used in Internationalization in the IETF (rfc3536)
 - <http://www.ietf.org/rfc/rfc3536.txt>
- IDN and related standards
 - <http://www.ietf.org/rfc/rfc3454.txt>
 - <http://www.ietf.org/rfc/rfc3490.txt>
 - <http://www.ietf.org/rfc/rfc3491.txt>
 - <http://www.ietf.org/rfc/rfc3492.txt>
- IANA IDN Language Table Registry
 - <http://www.iana.org/assignments/idn/>
- JET Guideline for IDN
 - <http://www.ietf.org/rfc/rfc3743.txt>

Live IDN examples

- Vietnamese
 - <http://tênmiềntiếngviệt.vn/>
- Thai
 - <http://ทีเอชที.th/>
- Taiwanese (Traditional Chinese)
 - <http://台網中心.tw/>
- Chinese (Simplified Chinese)
 - <http://中国互联网信息中心.cn/>
- Korean (Hangeul)
 - <http://.kr/>
- Japanese (Kanji)
 - <http://日本語.jp/>