Internationalized Domain Names: Technical Workshop

Prepared for:

LACTLD Meeting
Panama City, Panama
4 September 2008

Tina Dam
Director, IDN Program
tina.dam@icann.org

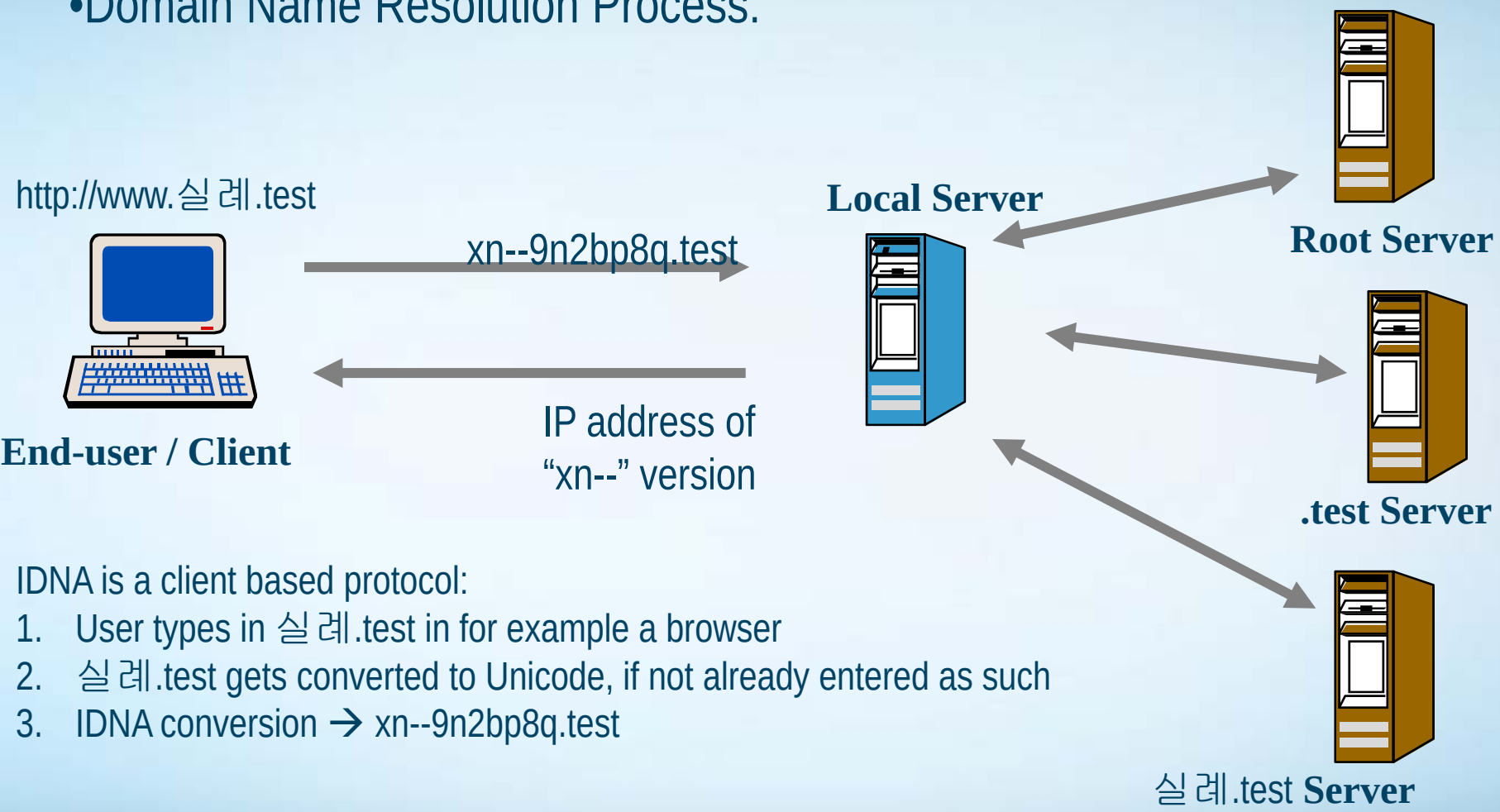
Workshop agenda

- Review of agenda – additions?
- How does IDNs work
  - Web and email demo's
- IDNA protocol revision
  - Rationale for the revision
  - Registration and resolution rules
  - Table properties
  - Fixing of bi-directional problems
- Confusability issues
  - Registration rules at registry level
  - Avoiding confusability at TLD level
- Status on policy implementation (if time/interest permits)

How does IDNs work?

IDN - Functionality

•Domain Name Resolution Process:



IDNA is a client based protocol:

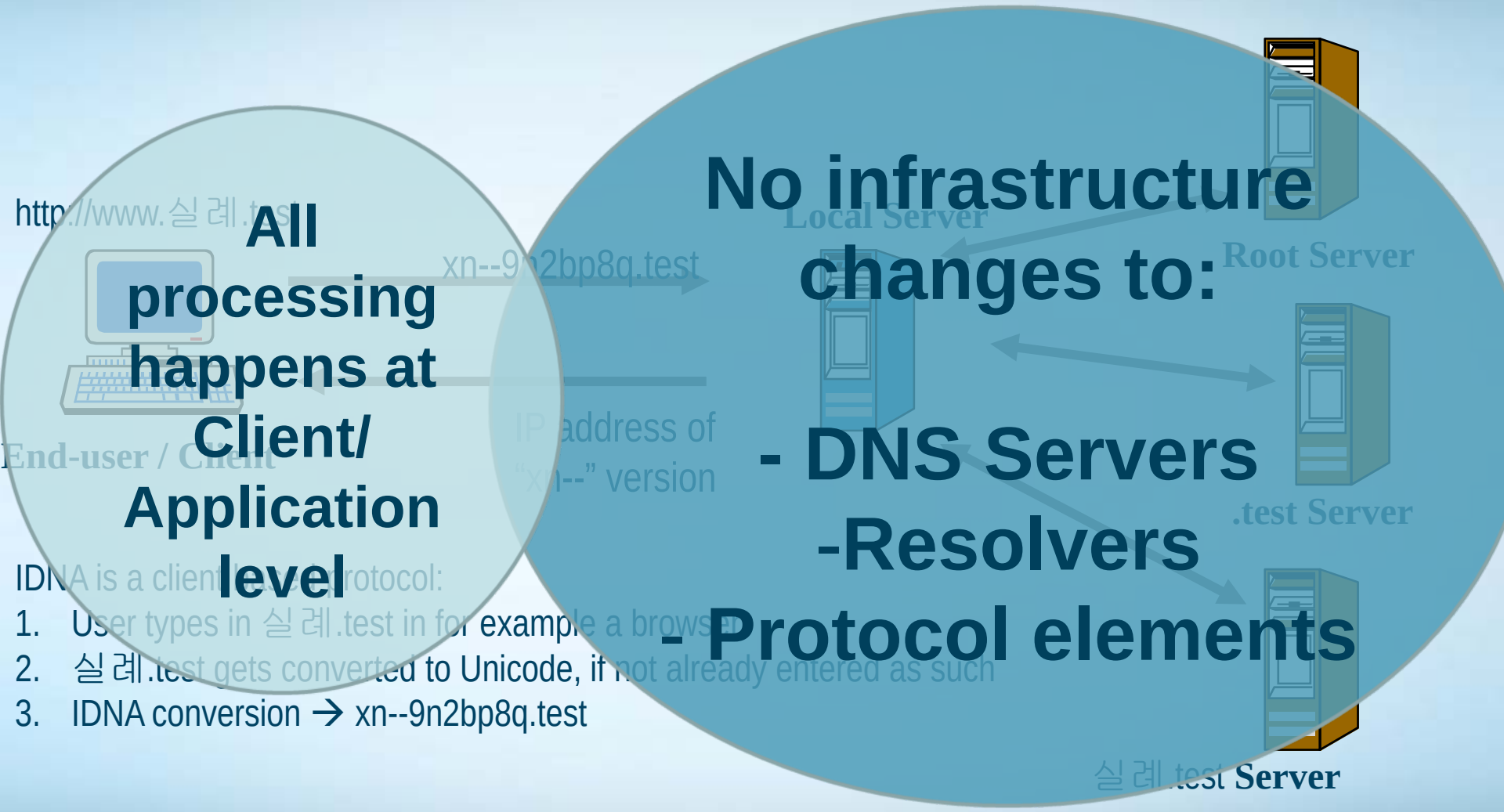
1. User types in 실례.test in for example a browser
2. 실례.test gets converted to Unicode, if not already entered as such
3. IDNA conversion → xn--9n2bp8q.test

IDN – Functionality

All processing happens at Client/ Application level

No infrastructure changes to:

- DNS Servers
-Resolvers
- Protocol elements

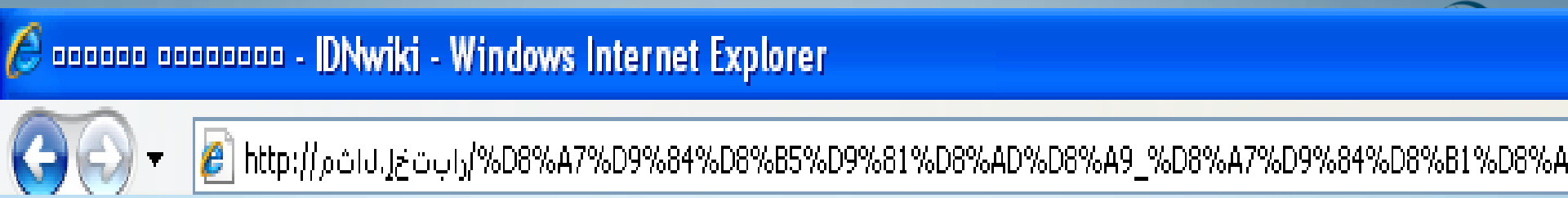


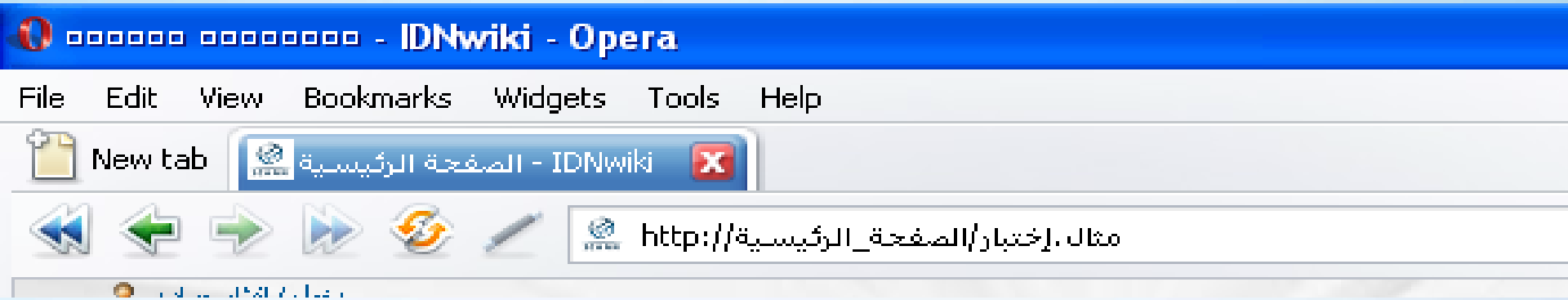
Implementation Overview

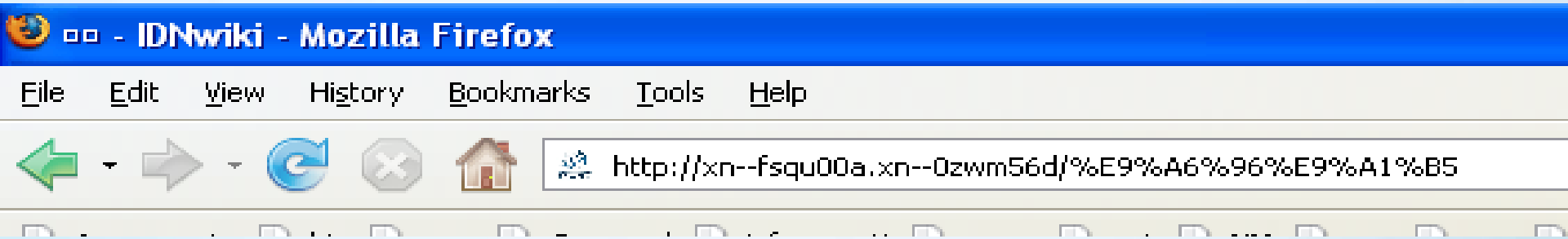
- Browser Implementation
  - IE 5.0+ with plug-in
  - IE 7.0+
  - Firefox 1.4+
  - Netscape 7.1+
  - Opera 7.11+
  - Apple Safari 1.2+
  - More…
- Email Implementation
  - Very limited due to experimental status on protocol
  - Possible IETF publishing of standard-version by Q309
  - Need to finish:
    - Downgrade
    - Mailinglist
    - POP and IMAP
  - See http://www.ietf.org/html.charters/eai-charter.html

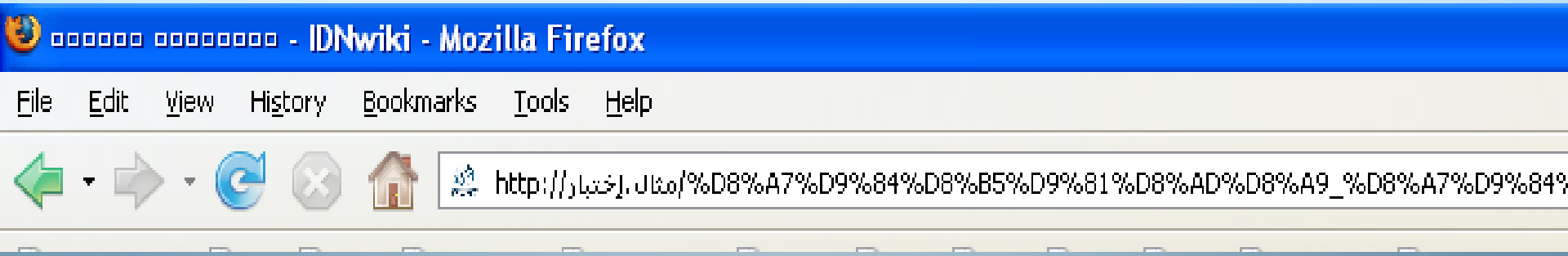
Demo's

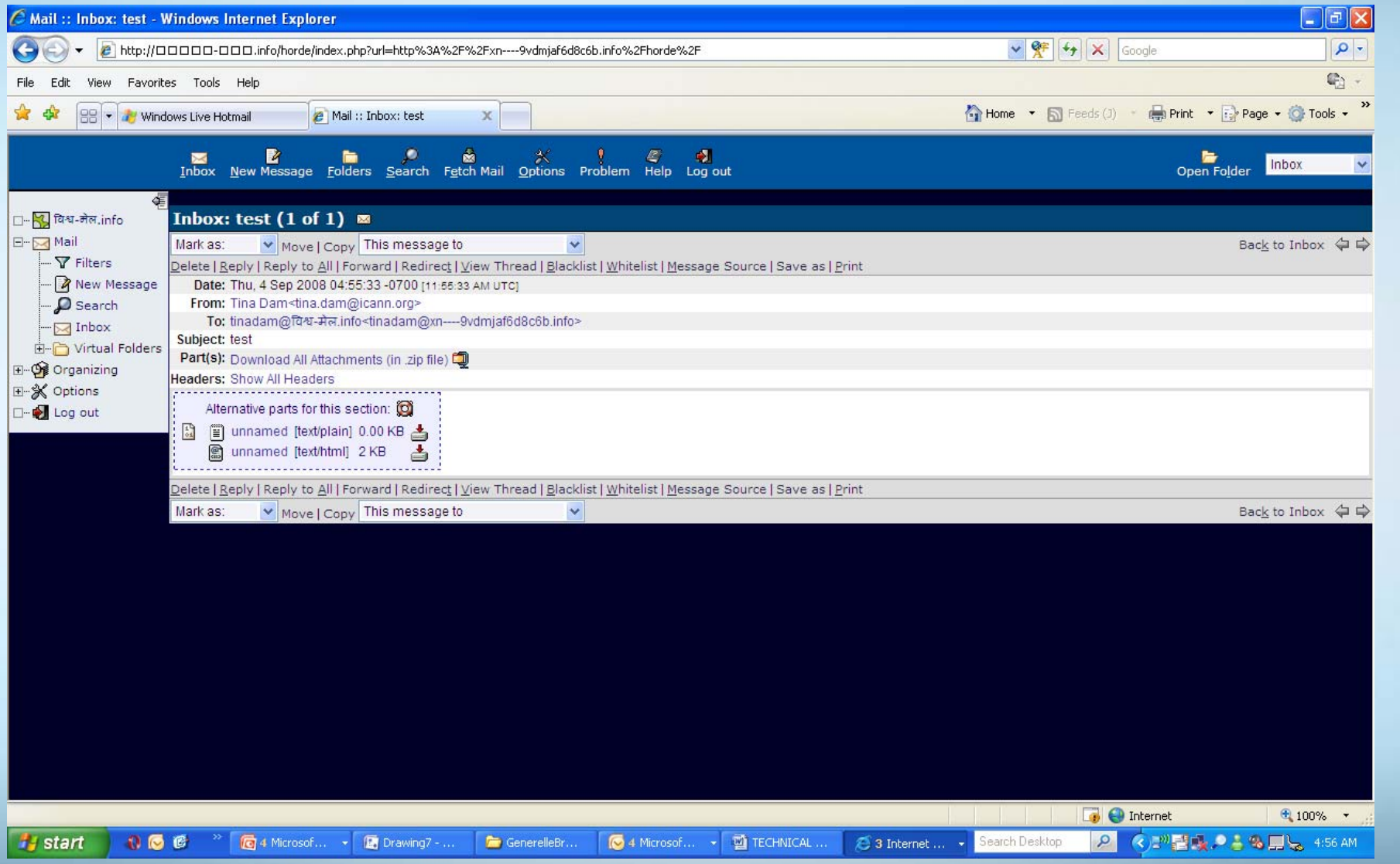
- Plan to demo:
  - Browser Applications:
    - IE
    - Firefox
    - Opera
    - Other browsers?
    - Perhaps some region-based browser?
  - Email clients
    - Afilias Global Email Sign-up
      - http://global-email.info/intro.html
    - Other email clients?


□□□□□□ □□□□□□□□ - IDNwiki - Windows Internet Explorer
http://مثال.إختبار/%D8%A7%D9%84%D8%B5%D9%81%D8%AD%D8%A9_%D8%A7%D9%84%D8%B1%D8%A


□□□□□□ □□□□□□□□ - IDNwiki - Opera
File Edit View Bookmarks Widgets Tools Help
New tab
الصفحة الرئيسية - IDNwiki
http://مثال.إختبار/الصفحة_الرئيسية


□□ - IDNwiki - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://xn--fsqu00a.xn--0zwm56d/%E9%A6%96%E9%A1%B5


□□□□□□ □□□□□□□□ - IDNwiki - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://مثال.إختبار/%D8%A7%D9%84%D8%B5%D9%81%D8%AD%D8%A9_%D8%A7%D9%84%


Mail :: Inbox: test - Windows Internet Explorer
http://□□□□□-□□□.info/horde/index.php?url=http%3A%2F%2Fxn----9vdmjaf6d8c6b.info%2Fhorde%2F
File Edit View Favorites Tools Help
Windows Live Hotmail
Mail :: Inbox: test
Home Feeds (J) Print Page Tools
Inbox New Message Folders Search Fetch Mail Options Problem Help Log out
Open Folder Inbox
विश्व-मेल.info
Mail
Filters
New Message
Search
Inbox
Virtual Folders
Organizing
Options
Log out
Inbox: test (1 of 1)
Mark as: Move | Copy This message to
Back to Inbox
Delete | Reply | Reply to All | Forward | Redirect | View Thread | Blacklist | Whitelist | Message Source | Save as | Print
Date: Thu, 4 Sep 2008 04:55:33 -0700 [11:55:33 AM UTC]
From: Tina Dam<tina.dam@icann.org>
To: tinadam@विश्व-मेल.info<tinadam@xn----9vdmjaf6d8c6b.info>
Subject: test
Part(s): Download All Attachments (in .zip file)
Headers: Show All Headers
Alternative parts for this section:
unnamed [text/plain] 0.00 KB
unnamed [text/html] 2 KB
Delete | Reply | Reply to All | Forward | Redirect | View Thread | Blacklist | Whitelist | Message Source | Save as | Print
Mark as: Move | Copy This message to
Back to Inbox
Internet
100%
start
4 Microsof...
Drawing7 - ...
GenerelleBr...
4 Microsof...
TECHNICAL ...
3 Internet ...
Search Desktop
4:56 AM

IDNA Protocol Revison

Rationale for the IDNA revision

- Proposed revision at IETF
  - RFC4690 requests the revision and provides suggestions to solutions to some problems

- Reasons and results of the revision:

Current Version	Revised Version
Unicode version 3.2	Unicode version independent
Some/New characters excluded	All characters in Unicode will have a status
Not all words can be represented	Not all words can be represented
Exclusion Based: - Table based	Inclusion Based: - Property and procedure based: - Protocol-valid (w/ context rules) - Disallowed - Unassigned
App developers have difficulty in understanding description of standard	Separates registration and resolution in detailed steps

Rationale for IDNA Protocol Revision

- Other issues was discovered during the revision process
  - For example: bidirectional problems
- Dynamic overview of documents:
  - by Patrik Faltstrom:
  - http://stupid.domain.name/idnabis/
    - Overall rationale and explanation
    - Protocol: registration vs. resolution
    - Tables and procedures
    - Bidirectional issues solutions

IDNA Revision

- Language working groups reviewing results and providing guidance
  - Arabic script working group
  - Additional future working groups
- Educational sessions on the difference
  - ICANN Paris meeting workshop for latest overview
  - http://par.icann.org/en/node/72
- Next steps:
  - The IETF Internet drafts to go into “Last Call”
  - Implementation by registries and application developers

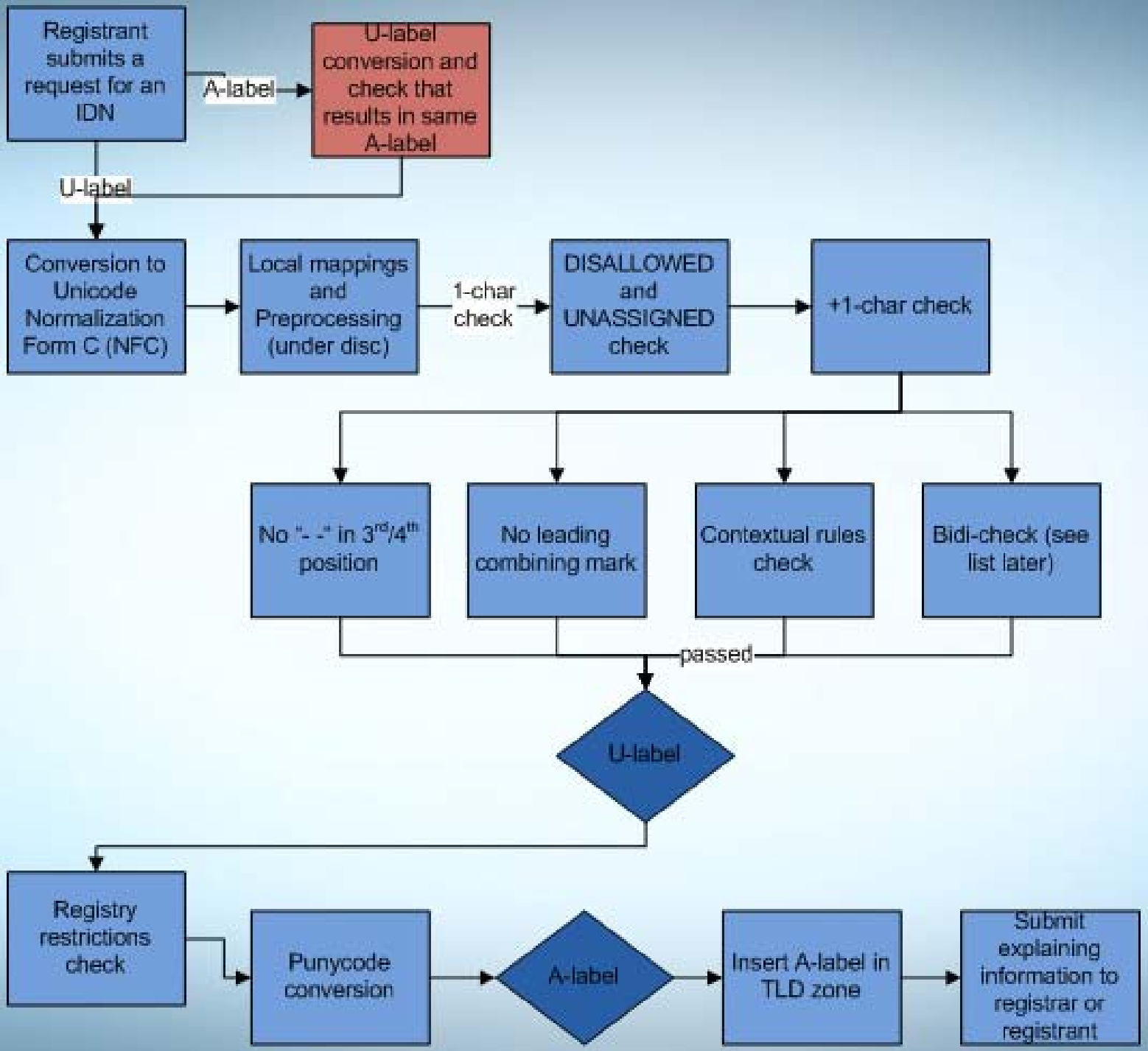
What are the rules in the “tables” doc?

- It reviews and classifies the collections of codepoints in the Unicode character set by examining various properties of the codepoints.
- It then defines an algorithm for determining a derived property value.
- It specifies a procedure and not a table of codepoints so that the algorithm can be used to determine code point sets independent of the version of Unicode that is in use.

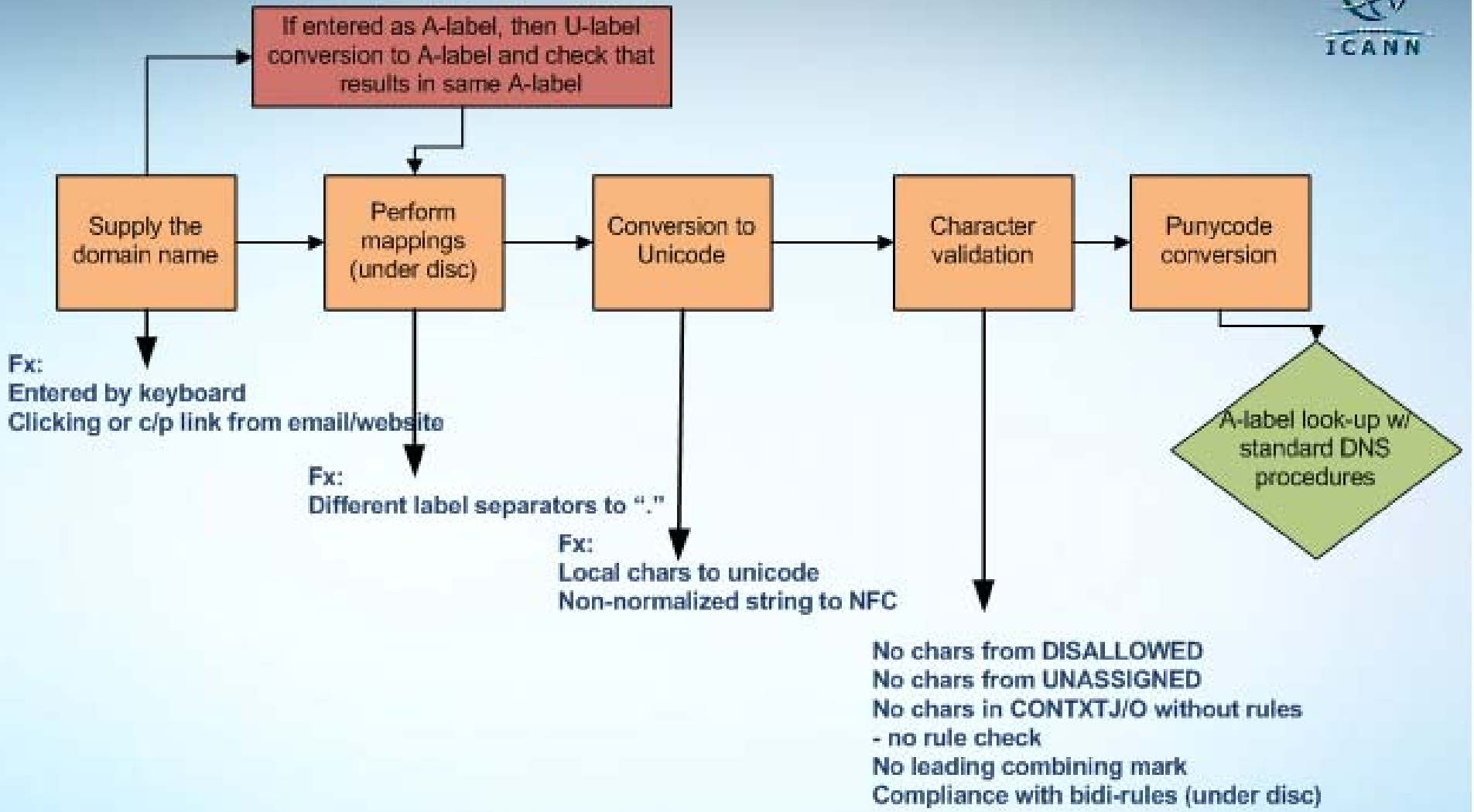
Registration with revised IDNA

- The registered labels are final
  - the protocol does not perform mappings
  - Registries can allow local mapping/pre-processing of mapping, but there is no resolution guarantee
  - users need to know what their requested label is mapped to if it is mapped before registration)
- Registry and registrars hold responsibility
  - Resolution is less restrictive than registration

Registration Steps


Registrant submits a request for an IDN
A-label
U-label conversion and check that results in same A-label
U-label
Conversion to Unicode Normalization Form C (NFC)
Local mappings and Preprocessing (under disc)
1-char check
DISALLOWED and UNASSIGNED check
+1-char check
No "- -" in 3rd/4th position
No leading combining mark
Contextual rules check
Bidi-check (see list later)
passed
U-label
Registry restrictions check
Punycode conversion
A-label
Insert A-label in TLD zone
Submit explaining information to registrar or registrant

Resolution Steps


If entered as A-label, then U-label conversion to A-label and check that results in same A-label
Supply the domain name
Perform mappings (under disc)
Conversion to Unicode
Character validation
Punycode conversion
A-label look-up w/ standard DNS procedures
Fx:
Entered by keyboard
Clicking or c/p link from email/website
Fx:
Different label separators to "."
Fx:
Local chars to unicode
Non-normalized string to NFC
No chars from DISALLOWED
No chars from UNASSIGNED
No chars in CONTXTJ/O without rules
- no rule check
No leading combining mark
Compliance with bidi-rules (under disc)

IDNA classification of codepoints

What are the rules in the “tables” doc?

- It reviews and classifies the collections of codepoints in the Unicode character set by examining various properties of the codepoints.
- It then defines an algorithm for determining a derived property value.
- It specifies a procedure and not a table of codepoints so that the algorithm can be used to determine code point sets independent of the version of Unicode that is in use.

In other words….

- It specifies rules for determining whether a codepoint can be used in IDNs or not.
  - that is, outside any specific registry requirements
  - in isolation
    - the “bidi” I-D provides requirements around the context of use

The table in draft-ietf-idnabis-tables-01.txt

- http://stupid.domain.name/idnabis
  - Appendix A holds the list of codepoints and their values, Unicode 5.1 ***(non-normative)***
    - Protocol valid
    - Contextual rules
    - Disallowed
    - Unassigned
  - Document procedures are ***normative*** and is what must be used
    - can be used on any Unicode version
    - backward compatible w/ old versions
  - In IDNA2003 the table was normative

The various classes of codepoints

LetterDigits (A)

- The good codepoints that we allow in IDNs
- The attribute is
  - generalCategory (cp) for codepoints in
    - Ll – Lowercase_Letter
    - Lu – Uppercase_Letter
    - Lo – Other_Letter
    - Nd – Modifier_Letter
    - Mn – Nonspacing_mark
    - Mc – Spacing_Mark
  - Metadata is from the Unicode database
  - IDNA2003 also allowed graphics characters, etc.

Unstable (B)

- Codepoints that are not stable under normalization and casefolding
  - In the DNS you can look up both upper-case and lowercase
    - Works for US ASCII, not for IDN
    - IDNA2003 required only lowercase
  - toNFKC(tocaseFolded(to NFKC(cp))) ! = cp
    - the codepoint will stay the same through casefolding and normalization
    - Uppercase letters are not stable
    - Lowercase letters are stable

IgnorableProperties (C) Blocks (D)

- This is catching codepoints with properties we want to ignore, such as
  - Default ignorable codepoints
  - White space
  - Noncharacters
    - Property(cp) is in { Default_Ignorable_code_Point, White_Space, Noncharacter_Code_Point }
  - Blocks to ignore
    - Block(cp) in { combining Diacritical marks for Symbols, Ancient Greek Musical Notation, Private Use Area}

LDH (E)

- ASCII letters, Digits, Hyphen
- Ensuring that these are still to be used
  - cp is in {002D, 0030..0039, 0061..007A}

Exceptions (F)

- Codepoints that need special attention
  - Special rules → allowed
  - List is under discussion

    -
      - 002D; CONTEXTO # HYPHEN-MINUS
      - 00B7; CONTEXTO # MIDDLE DOT
      - 02B9; CONTEXTO # MODIFIER LETTER PRIME
      - 0375; CONTEXTO # GREEK LOWER NUMERAL SIGN (KERAIA)
      - 0483; CONTEXTO # COMBINING CYRILLIC TITLO
      - 05F3; CONTEXTO # HEBREW PUNCTUATION GERESH
      - 05F4; CONTEXTO # HEBREW PUNCTUATION GERSHAYIM
      - 06FD; PVALID # ARABIC SIGN SINDHI AMPERSAND
      - 06FE; PVALID # ARABIC SIGN SINDHI POSTPOSITION MEN
      - 0F0B; PVALID # TIBETAN MARK INTERSYLLABIC TSHEG
      - 3005; CONTEXTO # IDEOGRAPHIC ITERATION MARK
      - 3007; PVALID # IDEOGRAPHIC NUMBER ZERO
      - 303B; CONTEXTO # VERTICAL IDEOGRAPHIC ITERATION MARK
      - 30FB; CONTEXTO # KATAKANA MIDDLE DOT

Backward Compatible (G)

- Currently empty
- Needed for new version of Unicode that create incompatibility but where we want to make an exception and specify value to a codepoint
- Adding characters, requires revision of the RFC

JoinControl (H)

- Requires special attention in Registration and Resolution
- For example, non-spacing mark
  - Property(cp) is in { Join_Control }

Unassigned (J)

- Cp is in {Cn} and property{cp) is not in {noncharacter_Code_Point}
- Unassigned codepoints per the Unicode defintion

Algorithm order

- All codepoints belong to one or more categories
- Stop when hitting a match, in order of:
  - Exceptions, see the list
  - BackwardsCompatible, see the list
  - Unassigned, UNASSIGNED
  - LDH, PVALID
  - JoinControl, CONTEXT J
  - Unstable, DISALLOWED
  - IgnorableProperties, DISALLOWED
  - IgnorableBlocks, DISALLOWED
  - LetterDigits, PVALID
  - Not LetterDigits (the rest), DISALLOWED

→ non-normative table as output

Contextual rules registry examples

- 002D; HYPHEN-MINUS; F;
  - must not appear at beginning or end of a label
  - [^^]\u002D|\u002D[^$]
- 200C; ZERO WIDTH NON-JOINER; T;
  - Between two characters from the same script only. The script must be one in which the use of this character causes significant visual transformation of one or both of the adjacent characters
  - [\p(Script:Deva)\p(Script:Tamil)]\u200C[\p(Script:Deva)\p(Script:Tamil)]
- 00B7; MIDDLE DOT; F;
  - Between two 'l' (U+006C) characters only, used to permit the Catalan character ela geminada to be expressed
  - \u006C\u00B7\u006C
- More….

Examples of Latin char result

- 0000..002C ; DISALLOWED # <control>..COMMA
- 002D ; CONTEXTO # HYPHEN-MINUS
- 002E..002F ; DISALLOWED # FULL STOP..SOLIDUS
- 0030..0039 ; PVALID # DIGIT ZERO..DIGIT NINE
- 003A..0060 ; DISALLOWED # COLON..GRAVE ACCENT
- 0061..007A ; PVALID # LATIN SMALL LETTER A..LATIN SMALL LETTER Z
- 007B..00B6 ; DISALLOWED # LEFT CURLY BRACKET..PILCROW SIGN
- 00B7 ; CONTEXTO # MIDDLE DOT
- 00B8..00DF ; DISALLOWED # CEDILLA..LATIN SMALL LETTER SHARP S
- 00E0..00F6 ; PVALID # LATIN SMALL LETTER A WITH GRAVE..LATIN SMALL
- 00F7 ; DISALLOWED # DIVISION SIGN
- 00F8..00FF ; PVALID # LATIN SMALL LETTER O WITH STROKE..LATIN SMAL
- 0100 ; DISALLOWED # LATIN CAPITAL LETTER A WITH MACRON

Bidirectional issues and solutions

IDNAbis – right-to-left problem

- Draft-alvestrand-idna-bidi-00 (need renewal)
  - Discusses problem resulting from a constraint on the use of combining characters at the end of an RTL domain label resulting in errors
    - Stringprep: If a string contains a RandALCat character, a RandALCat MUST be the first character and the last character in the string
      - Results in some words being invalid as IDN labels and at least one case an entire language

IDNAbis – right-to-left problem

- Example: Dhivehi
  - official language of the Maldives, Thaana script
  - Vowels are indicated by diacritical marking of the consonantal base – as a requirement

computer = konpeetaru = ކ ް ނ ް ޕ ް ޓ ަ ރ ު

  - not displyaed correctly here, but with markings separate to illustrate:
    - ◌ް directional class is NSM, i.e. not R or AL, hence IDNA will determine not RandALCat and error is returned

IDNAbis – right-to-left problem

- Fix to IDNA, RFC 3454:
  - for characters that have category R, AL, L the category is fixed;
  - for characters in category EN, ES, ET, AN, CS, NSM, BN, B, S, WS, ON the category is determined by applying the algorithm described in UAX#9, section 3.3. to the string
  - …and RandALCat character is a character that, after this determination has unicode bidirectional categories R or AL; and Lcat character is a char with Unicode bidi category L.

- Other problems:
  - » Digraphs
  - » Display of mixtures of LtR and RtL strings
  - » Digits are “jumping”

Confusability Issues

IDN TLD/SLD launch considerations

- Which characters should be offered
  - Formal language, survey users, legal matters…
- Launch procedure, registration policy
  - IP rights, existing registration rights, fcfs?
  - Variant table, blocking registrations, pre-rights or packaged registrations
- IDN Guidelines and protocol adherence
- Registrar and user education & assistance
  - Web only based on application uptake, no email yet

Language and Script

- Languages are used by humans to interact
  - Best guesses estimate 5000-7000 languages worldwide, of which 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
- IDNA is based on Unicode character set and code-point properties

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 Ü is referred to in Swedish as a # "German Y" and is considered to be a variant of the letter Y.
    - 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
- IANA Repository holds the variant tables
  - http://www.iana.org

Same Language Multiple Scripts Issues

- 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 use 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

Visual Confusion Issues

- Well-known example: paypal.com
  - Second character is U+0430, Cyrillic small a
  - Looks like Roman/ASCII “a”
  - This is now prevented by “one label, one script” rule per the IDN Guidelines with exceptions for mixed script languages
- Other example:
  - Russian ccTLD is .ru
    - Cyrillic “r” and “u” is: p and y
    - Which looks like p y (in latin) is ccTLD for Paraguay
    - **Note: Russia did not ask for .py, this is just an example**
  - Process needed to determine labels matching
    - ccTLDs, gTLDs, TLD labels under application

How are these issues being solved?

- Not all issues can be solved but some can:
  - SWORD Algorithm
    - to avoid confusingly similar TLD strings
  - Protocol revision
    - Future proof solution, fixing right-to-left script issues, adding contextual rules to some characters
  - Variant table requirements and guidelines updates
    - Eliminating more confusable characters
    - Potentially with linguistic support/clearing house
  - Local initiatives creating common registration policies
    - CJK JET guidelines
    - Arabic script working group on variant table
    - Cyrillic group in initial stages to get launched

Draft TLD label restrictions

- General Requirements
  - The label must be a valid domain name, as specified in technical standards [RFC]. This includes:
    - The label must be 63 characters or less, in wire format.
  - The label must be a valid host name, as specified in technical standard [TBD]. This includes
    - The label must commence with a letter "a" through "z".
    - The label must be wholly comprised of letters, digits and hyphens.
    - The label must not conclude with a hyphen.

Cont…

- The label must not be likely to be confused for an IP address or other numerical identifier by application software. For example, representations such as "255", "O377" or "0xff"; representing decimal, octal and hexadecimal strings; can be confused for IP addresses. As such, labels must not:
  - Commence with "0x", case insensitive.
  - Commence with "o", and have the remainder of the label wholly comprised of of digits between 0 and 7.
  - The label may only include hyphens in the third and fourth position if it represents a valid internationalized domain name in its ASCII encoding.

Requirements for Internationalized TLDs

- The label must be a valid internationalized domain name, as specified in technical standards [RFC]. This includes the following, non-exhaustive, list of limitations:
  - Must not contain any Unicode code points that are disallowed or unassigned.
  - Must not contain code points other than those identified in Unicode as Letters or combining marks.
  - Must not contain code points that are not NFC compliant.
  - The label must meet the relevant criteria of the ICANN IDN Guidelines .
  - The label must not reasonably be known to cause any rendering or operational issues.

Generic and ccTLD specifics

- Requirements for Generic Top-Level Domains
  - The label must be comprised of three or more visually distinct letters or characters, as appropriate in the script.
- Requirements for Country Code Top-Level Domains
  - The label must be comprised of two or more visually distinct letters or characters, as appropriate in the script.

Policy Implementation Status
(if time/interest permits)

IDN ccTLD Fast Track Process

- To introduce a limited number of non-contentious IDN ccTLDs that:
  - are associated with the ISO3166-1 list
  - will meet near term demand in territories and countries that are ready
  - preserve stability of the DNS
  - do not pre-empt the IDN ccPDP
  - are not based on characters from Latin script

IDNC Fast Track Reports

- Several reports posted over time for public review and comments
- Latest comment period ended 15 Aug 08:
  - http://www.icann.org/en/public-comment/public-comment-200808.html#final-idnc-wg
- Information available at:
  - http://icann.org/topics/idn

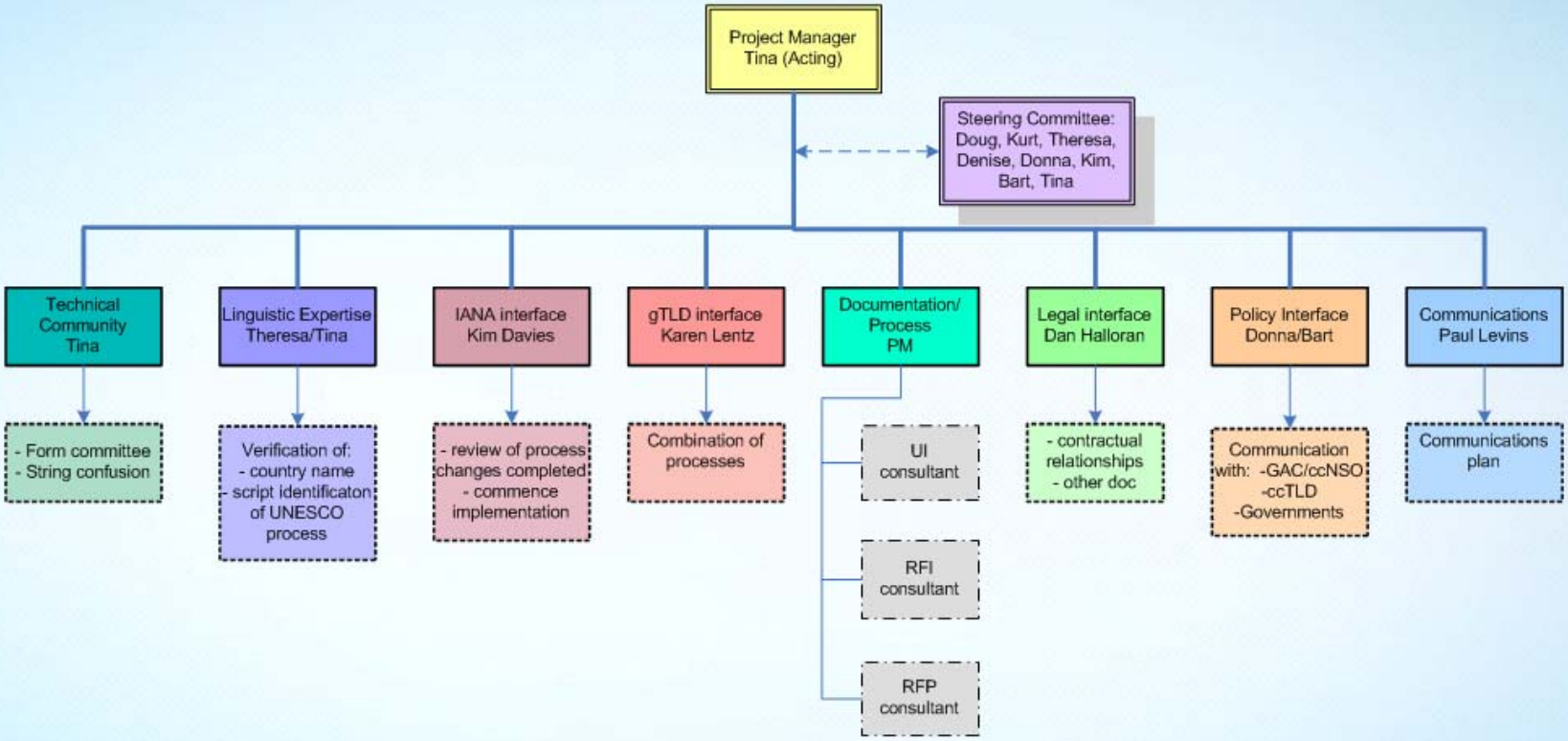
Fast Track Implementation

- ICANN Board Resolution, Paris, 26 June 08:
  - post the IDNC WG final report for public comments;
    - Comment period ended 15 Aug 2008
  - commence work on implementation issues in consultation with relevant stakeholders; and
  - submit a detailed implementation report including a list of any outstanding issues to the Board in advance of the ICANN Cairo meeting in November 2008

FT – IDNC Report Main Focus

- Preserve the security and stability of the DNS;
- Comply with the IDNA protocols;
- Take input and advice from the technical community with respect to the implementation of IDNs; and
- Build on and maintain the current practices for the delegation of ccTLDs, which include the current IANA practices.

IDN Fast Track Implementation Team



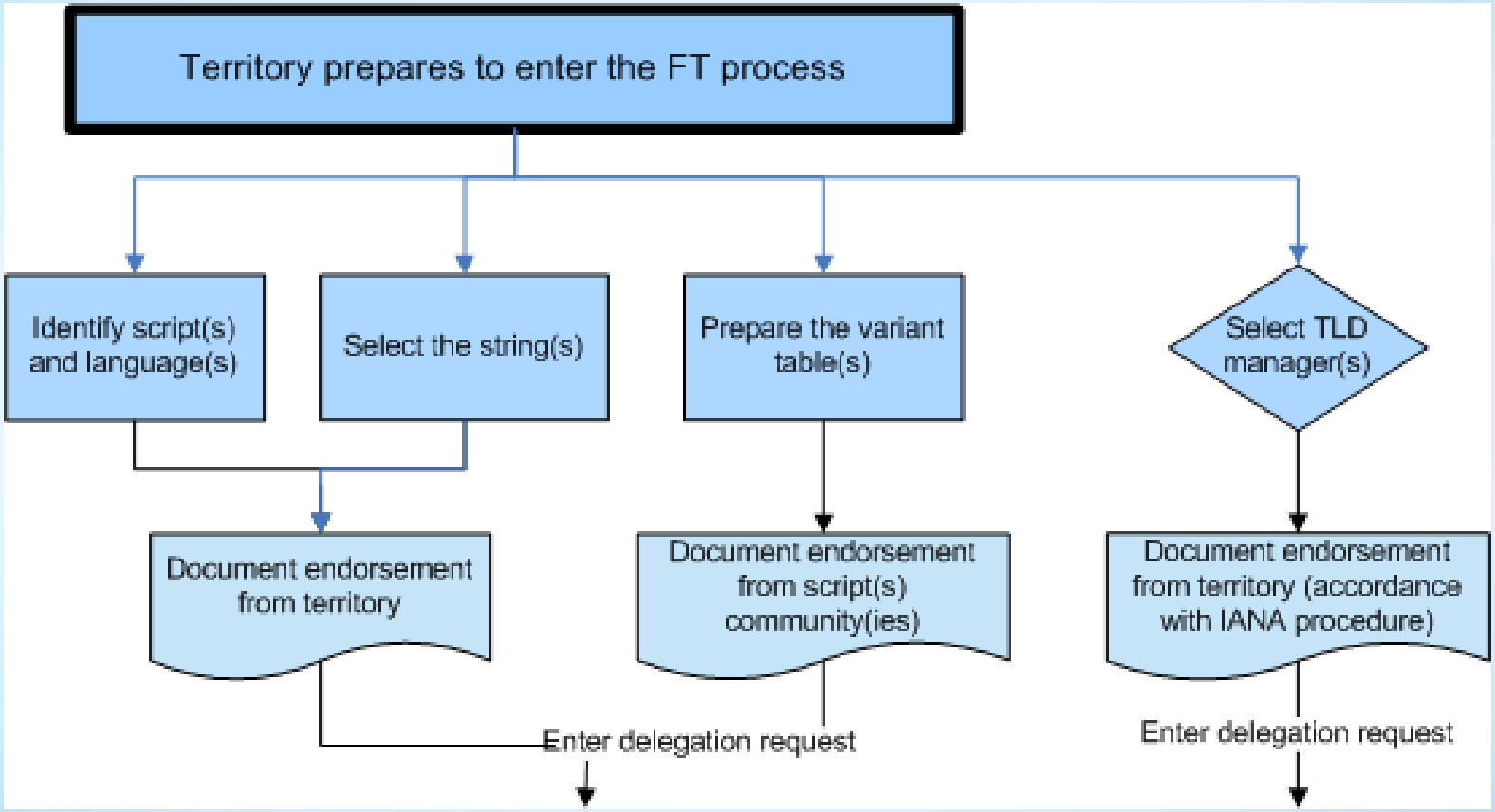
FT – Staff Focus and Activities

- Draft project plan, with focus on:
  - Review of received comments (per 15aug08)
  - Applicant eligibility
    - Issues with ISO list not covering all existing ccTLDs
  - Request for information letter
    - To be submitted to all governments and ccTLDs
  - Endorsement requirements
    - Territory support; string; and variant table criteria

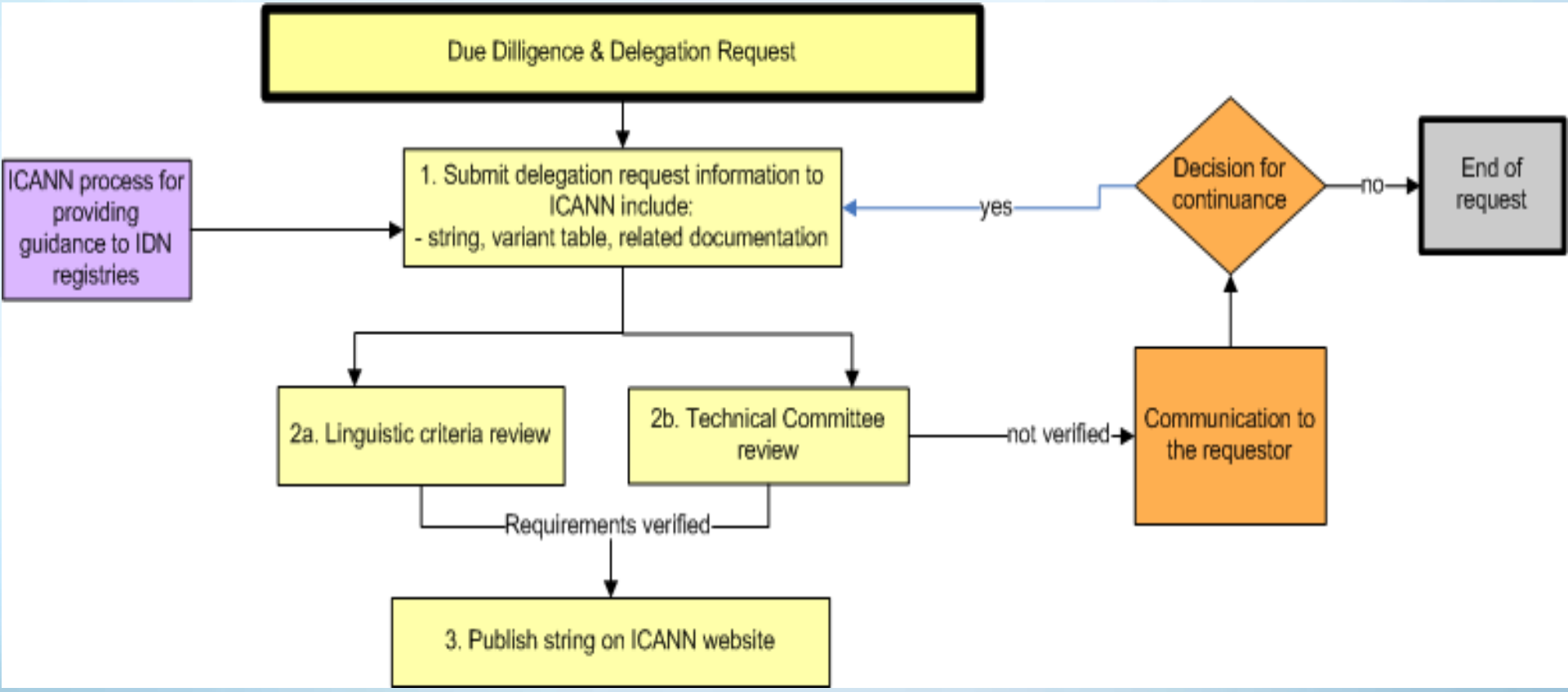
FT – Staff Focus

- Issues not covered in IDNC report
  - String contention
  - Financial considerations
  - Documentation of mutual obligations
    - Technical stability considerations
    - Consensus policy considerations

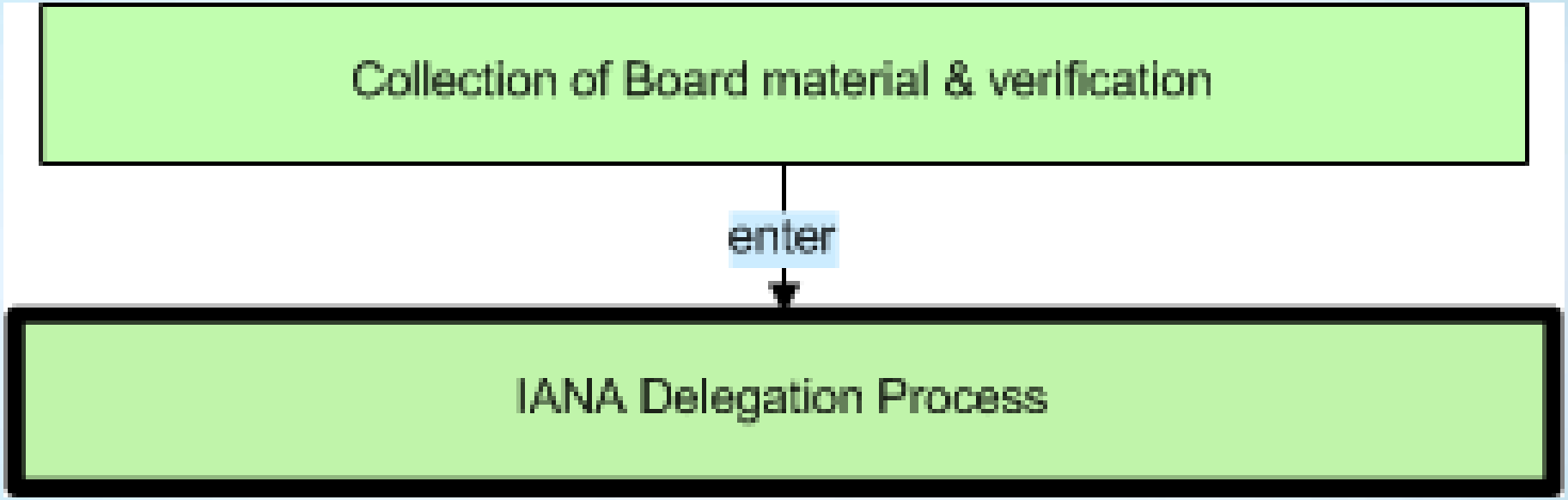
Process Flow 1 – Territory Preparation



Process Flow 2 – Submission of delegation request



Process Flow 3 – IANA and ICANN Board Process



IANA management of IDN TLDs

- Process for insertion of IDN TLDs in root
  - exists for test domains only (IDN .test)
    - Developed w/ RSSAC & SSAC recommendations
  - includes emergency removal procedure
    - for test IDN TLDs only, not for production
    - to be closed down with RSSAC agreement
  - Initial review of process showed need for additional information from applicant:

1. A-label	**2. U-label**
3. Short-form of string (English)	**4. Language of label (ISO630-1)**
5. Language of label (English)	**6. Script of label (ISO 15924)**
7. Script of label (English)	**8. Unicode code points (list)**

Timing Considerations

- IDN TLD launch
  - ccTLD and gTLD aiming at same launch time
  - although, one process's delay will not delay the other
- IDNA revision finalization
  - Preferred to be finalized before IDN TLD launch, but not a requirement
- Currently implementation efforts are aiming towards Q2-2009