



Urdu Domain Names

ووو-اردو تحقیق نیٹ

Sana GUL,
Pakistan 2007



Center for Research in Urdu Language Processing



Background

- Access to information is mandatory for development of digitally divided population
- With HTML, Unicode and CLDR, deployment of multilingual content is possible
- Translating web content may not suffice as entry point to this access, the URL, is Latin based



Bottlenecks

- Internet Protocol maps onto address resolution based on 8-bit ASCII standard

The relevant function only allows ASCII inputs

Possible Solutions

- **Develop a system that works** independently of existing DNS
- **Develop a system that works** on top of existing infrastructure



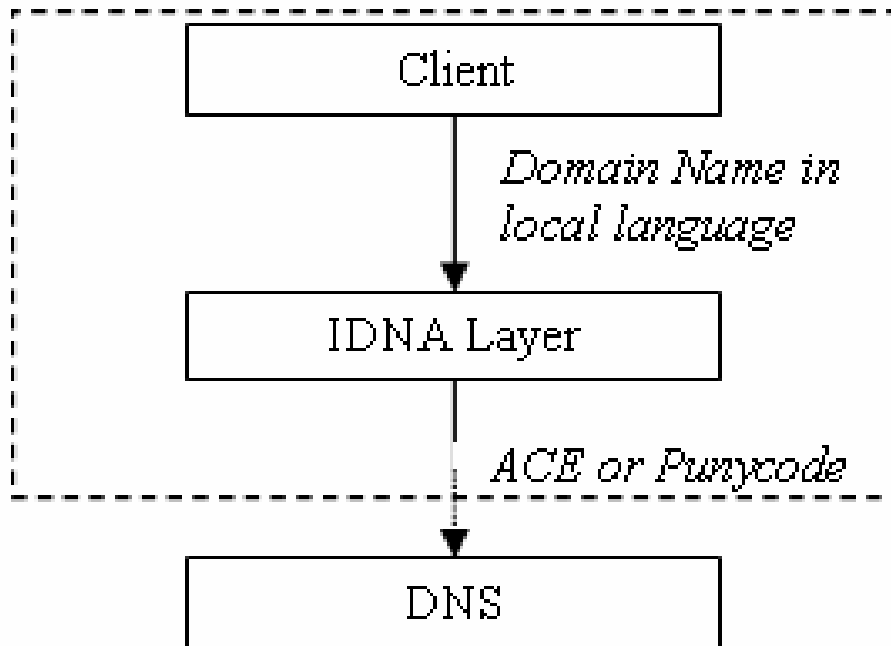
IDN Agent

- Proposed solution has been designed to work within existing Domain Name System
- An IDN layer is added between client and DNS
- The function of this layer is to convert the non-ASCII domain name to a DNS compatible ASCII equivalent code

Also ensures backward compatibility



Application Side



IDNA Layer

- As a separate server on network (*makes it browser and OS independent but causes network delays*)
- By embedding it within client side application (*makes it browser and OS dependant*)



Namerep Function

- Namerep function takes string in local language and converts it into a normalized Unicode string.
- The input string might be in other encodings like UTF-8, ISO 8859-x, Big5 (for Chinese). Namerep function recognizes the input encoding and converts it into Unicode.



Normalization

- Unicode standard has redundancy.
- The Unicode string has to be normalized in the second step of the process.

For example, á (U+00E1) can also be written as a combination of 'a' and ´ (U+0061 + U+0301)

- Without normalization spoofing poses a challenge to the security



Punycode Generation

- To make the hostname DNS compatible the Unicode string has to be converted to ASCII Compatible Encoding
- Punycode is a boot-string encoding mechanism that uniquely converts Unicode string to the allowed ASCII based encoding. This conversion takes place through an algorithm known as ToASCII()
- There is a possibility that the generated Punycode is already a registered domain name for example 'mgb' is generated for code '!' but www.mgb.com is a registered domain. To avoid such scenarios a four character prefix xn-- is appended to every punycode. So www.!.com is converted to www.xn--mgb.com



Research Issues for Urdu Domain Names Implementation

- Character Set
- Cursiveness
- Encoding
- Normalization
- GTLD
- CCTLD






Character Set

- Urdu character set include basic alphabet, digits, vowel marks, punctuation marks, honorifics, and special symbols.
- At least basic alphabets and digits must be allowed
- Diacritics are never or rarely used in Urdu text.
- Diacritics also raise spoofing issue because a native user would consider URL with and without diacritics equal. Therefore diacritics are not included in recommended character set



Contd..

- Honorifics like ,  and  are mandatory
- These should be allowed in domain names as variants of full forms and should be normalized in Namerep.
- Miscellaneous characters like footnote symbol and sign to indicate verse are not included



Cursiveness

- Urdu writing system is highly cursive
- Space character is not allowed in domain names. English words can be separated by hyphen or using upper case
- Even if they are written in a continuum they do not cause readability problem.
- Urdu neither has a hyphen nor capital letters. If the words are written in a continuum they will be less readable. Consider 'پنجابی لغت' vs. 'پنجابیپیلغت'
- Possible solution is to allow ZWNJ character and make it look like 'پنجابی لغت'
- Another solution is to allow space character which can be removed during Namerep process.



Encoding

- Minimally national encoding standard namely Urdu Zabta Takhti, UTF-8 and Unicode support must be provided.



Normalization

- Three kinds of normalization are required for Urdu:
 - Characters repeated for different languages for example there are two sets of one for Arabic and other for remaining languages (Farsi, Urdu, Sindhi etc.) So ۱۲۳ and ١٢٣ are visually same. Similarly Arabic Letter YEH ی (649) and Farsi letter YEH ی (6CC) are visually identical.
 - It is also important to note that these normalizations would not work across other languages (e.g. Sindhi, Pashto, etc.) and are only done in context of Urdu. Thus, these must be included in the language table at the registrar



Contd..

- Second, when base letters combine with some combining characters, their equivalent is also encoded directly in Unicode. Thus, آ can be written as U+0622 or a combination of U+0627 and U+0653. However, these sequences should be normalized.

Characters	Normalized Form	Recommended Form
آ+~	آ	آ
آ+ء	آ	آ
و+ء	و	و
ع+ء	ع	ع
ه+ء	ه	ه
ى+ء	ى	ى

Case fold normalization



-
- Finally, Unicode also lists many ligatures. These ligatures must be de-normalized into base characters as well. For example ‘الله’ should be normalized to ‘ا+ل+ل+ه’



GTLDS & CCTLDS

- Urdu need its own gTLD set and separate name space.
- Similarly ccTLDs also need to be translated.



Conversion of www and Label Separator

- An Urdu user can use 'ووو' instead of www in Urdu domain name. This is also converted back to 'www' in Namerep process
- An equivalent replacement for DOT (.) in Urdu is Arabic Full stop (-).
- BUT it has its associated problems