ITU and UNESCO Global Symposium on Promoting the Multilingual Internet

Geneva, 9-11 May 2006

Paper Submitted by:	**The Syrian Arab Republic** on behalf of the League of Arab States' ADNS Working Group
Title:	**Arabic Domain Name System (ADNS)** Status and Issues
For Discussion in Session: 2	**IDN Operational experience Showcase**

Introduction

Arabic Domain Name System (ADNS) has been largely viewed as one of the most urgent issues which constitute a considerable hurdle against the development of Internet usage in the Arab world. Several workgroups were formed to bring a solution to this problem, and an Internet-draft was issued by ESCWA as an extra step towards a standard solution [5]. The Internet-draft contains a set of guidelines which are in line with the IETF IDN standards and takes into account Arabic-specific issues as recommended by SaudiNIC [4] and ICANN [7].

Given that an Internet draft is usually a first step towards a standard solution, its evolution is conditioned by the consensus of the concerned community about the correctness of the implementation and the consistency of this solution with international standards. These contributions can therefore be considered as the basis of a discussion bringing together the concerned stakeholders. And given the emergency of the issue and the proliferation of non-standard solutions (a more politically-correct term would be "vendor-specific standards"), the evolution should be solid and quick.

This document aims at defining all the "ingredients" needed to design and implement a successful, workable ADNS solution, which would be accepted by the Arab and Arabic-speaking countries. It represents the result of several meetings of the ADNS working group (ADNS-WG) formed by the League of Arab States (LAS), which carried a detailed discussion of most of the issues related to the Arabic domain names. The discussions were held with a broad participation of the concerned stakeholders, including those who issued the first Internet drafts. The suggested solutions are compatible with international standards and rules adopted by the IETF, and in particular the set of IDN standards as defined in RFCs 3490[1], 3491[2], and 3492 [3].

What is IDN

The goal of the Domain Name System (DNS) is to provide people with a clear and easy-to-use way of addressing sites connected to the Internet. The DNS was created in 1983 by Paul Mockapetris to address maintenance problems with the Internet hosts database, fondly remembered as HOSTS.TXT. It was originally defined in IETF RFCs 1034 and 1035, and then extended by numerous subsequent RFCs.

In one sense, the DNS remains the only Internet-wide deployed database used successfully, with more than 390.000.000 entries stored in 2006[1], which also makes it particularly hard to change.

For about 20 years, DNS was restricted to case-insensitive ASCII letters (a-z), digits (0-9) and hyphen (LDH). This restriction rapidly became a big obstacle against the globalization of the Internet and its wide spread among users who are not familiar with the Roman Latin alphabet. The character set is definitely not sufficient to meet the requirements of users who are native speakers of other languages, which resulted in an urgent demand for the "Internationalization" of the DNS, given born to "Internationalized domain names", or IDN [2].

The main objective of IDN was to allow the use of domain names which are not restricted to the mere 38 characters used in the original DNS. Instead, the IDN is associated with Unicode- (ISO-10646) based characters, which contains tens of thousands of possible "code points". The technical solution for the IDN was introduced with the RFCs 3491, 3491, and 3492 published in March 2003. These RFCs define a standard framework for the internationalization of the DNS. In sum, the technical solution relies on keeping the standard DNS character set "on the wire" for compatibility with the currently deployed DNS infrastructure and applications. Unicode representations used by the end user are encoded into ASCII Compatible Encoding (ACE), and a special string "xn--" was added in front of the encoded domain labels to indicate that it represents an ACE-encoded "internationalized" label. The preparation of the ACE string is commonly known as the "stringprep" phase and the encoding algorithm used to generate ACE strings is known as "Punycode" [3].

The ICANN followed the IETF trend and announced on its turn a set of rules for IDN registration which can be summarized as follows [7]:

- Must comply with RFCs 3490, 3491, and 3492.
- Must identify permissible Unicode code points and block non-compliant registrations.
- Must associate registration with one or more languages and employ language-specific registration rules (e.g., reservation of domain names associated with character variants).
- Registries and registrars should provide informational resources and services in all languages for which they offer IDN registrations.

More rules have been added recently to address an unexpected security problem related to the use of IDN [8]. The new rules aim at eliminating the threat of deceptive use of visually confusable characters from different scripts. This threat has been considered as a serious limitation against the wide deployment of IDN.

IDN and ADNS issues

IDN issues can be classified into two large categories:

1. *Technical issues*, which are related to handling the technical specificities of the language *per se*. In our case, the Arabic language and its features; such as the appropriate character set (Unicode code points), the use of diacritics (*Tashkeel*), *Kasheeda*, and character folding. While these issues have been largely debated till now and a common agreement on a set of solutions is established, some conflicts remained on a few minor issues. The ADNS-WG discussed these issues in the framework of the IDN standards and RFCs (3491, 3491, and 3492).
2. *Organizational issues*, which are not covered by the IDN standards as they are much more related to ICANN activities rather than the IETF. These issues are largely *subjective* and are still quite open to discussion as there is no clear and adopted solution to them. Most of the future debate

[1] Source: ISC Domain Survey. www.isc.org

will concentrate on those issues. One example is the structure of the TLDs (gTLD and ccTLD) which affects heavily the structure of the ADNS. Another example is how to define registrars, and how to handle trademarks, and how to avoid domain name reservation for the sake of speculation.

ADNS technical issues

Before we start discussing the technical issues, we must put forward a very important rule to be followed through the discussion. A domain name is a set of labels which are used to identify a site on the internet in the easiest and most direct way. Hence, the discussion should not drift towards purely linguistic issues, as this might take us into endless debates (e.g., should the registrar accept misspelled but legal Unicode sequence for the domain name?) Our clearly stated goal is to define an Arabic domain name structure which will be accepted and adopted by all the users. In order to do so, a few rules which have been observed:

- Keep the domain name as short as possible. The Arabic language suffers already from the lack of acronyms (difficult to describe a university name into merely three characters, such as MIT!) A long sequence means simply more possible errors while typing the domain name. Modern browsers can help reduce the number of characters to be typed directly by the user through auto-complete and favorite sites functions. It is not expected that using ACE would result into a violation of the 255 characters limit of the whole domain name, but we should still remember that such a limit exists.

- Respect the Arabic language linguistic structure *as much as possible*. Arabic rules which can be respected without introducing too much complication should be implemented. We need to stress here that -in our opinion- a domain name should not be considered as a “valid” Arabic phrase. In other terms, our goal is to produce 100% natural Arabic phrases to be used as domain names, but to produce a set of labels which would *look familiar* to an Arabic-speaking user, and still keeps an acceptable level of conformance with Arabic language rules.

- Reduce the discordance between what is written at the graphic user interface (GUI) and what is stored at the registrar database. It is true that one of the main phases of IDN is the transformation of the Unicode domain name into a Punycode string, and that several processing rules could be applied (e.g., elimination of diacritics if they are kept at the GUI level). Still, if too many possible visual strings are converted into one stored ACE representation, then the inverse process (ACE to Unicode) would have a serious dilemma of selecting what is the correct visual string to display.

Having stated those basic assumptions, we can now list the technical or linguistic issues, proper to the Arabic language as follows:

- Diacritics (*Tashkeel*): Diacritics are legal Arabic characters which have their corresponding character codes. They affect heavily the meaning of the words; but they are rarely used in technical texts and documents as they are written only when their absence might result in a misunderstanding. The possible solutions here are the following:

a. Full support for diacritics;

b. Diacritics are supported visually but are not stored;

c. Diacritics are not allowed in the entered neither in the stored string.

The ADNS-WG adopts the position of supporting diacritics at the GUI level without storing them in the zone name. The main reason behind this is to avoid lengthy domain names and allow ease of writing. Allowing diacritics will only add another source of errors without giving the user any clear added value. This position was also adopted in the ESCWA’s Internet [5].

- *Shadda* (Double character): It is usually considered as part of the diacritics and actually represents a *real* character. For instance, a considerable number of names would have a totally different meaning if *Shadda* is ignored (consider سمّان، سمان؛ جبّان، جبان). Still, for reasons related to domain names simplification, the ADNS-WG decided to follow ESCWA's suggestion of not supporting *Shadda*.

- *Kasheeda* or *Tatweel* (Horizontal character-size extension): This extension is purely visual in the Arabic language, and has only calligraphic importance. It has a character code, but its presence (or lack of presence) does not affect the meaning of the word. Hence, the ESCWA's draft position suggesting not to support *Kasheeda* was adopted by the ADNS-WG.

- Character folding: This is the process where multiple letters having some similarity with respect to their shapes are folded into one single shape. Some folding examples are: folding *Teh Marbuta* and *Heh* at the end of a word, folding different forms of *Hamza*, folding *Alef Maqsura* and *Yeh* at the end of a word, and folding *Waw-Hamza* and simple *Waw*. The ESCWA's draft suggests strongly that character folding should not be supported. The only arguments why folding would be interesting to use are to avoid cyber-squatting, and to accommodate non-native Arabic speakers. Cyber-squatting is when someone registers ريفه.شركةظ so people who misspell ظريفة.شركة would get the squatting domain instead of an error message (this is a common practice in Latin DNS, e.g., Altaveesta.com instead of Altavista.com). The other problem is related to the use of ADNS by non-Arabic speaking people, for whom the distinction between different types of *Hamza* would be almost cryptic, idem when it comes to ي and ى, which are commonly folded in Egypt. The final decision, however, was not to support character folding.

- Numerals: The Numerals issue is somehow problematic, given that several Arab countries use the Hindi numerals (Eastern numerals) instead of the Arabic (Western) numerals. One potential problem with Hindi numerals is the similarity between the dot "." and zero. While a native Arabic speaker would most likely not be confused, these characters would be seen as identical by someone who is not well knowledgeable in the Arabic language (and therefore, subject to deceptive use). The ESCWA's Internet-draft suggests that the set of Hindi numerals should be folded into the Arabic ones, and this position was also adopted by the ADNS-WG.

- Separator: We need to distinguish between two types of separators: The label separator (traditionally dot '.', in the Latin DNS) and the separator between multiple words of the same label. These should not be confused as the first one has a meaning in the DNS system (a hierarchical interpretation) and therefore should not be touched, and the second one is treated as any other character. Given the nature of the Arabic language and especially the change of character's shape depending on its position in a word, it is very unlikely that collating words (such as iraqwar.com) would not be extremely confusing in Arabic (حربالعراق.شركة), especially if we don't use diacritics as suggested. Therefore, a separator is definitely needed, but space should not be acceptable, as it is not a legal Punycode character. Another problem with the use of space as separator is the possibility of the user entering multiple spaces by mistake. The biggest problem lies in the fact that space is an ASCII character, which means that it will pass unprocessed through the "stringprep" phase. The alternative separator suggested is the hyphen "-" character, which is also suggested in the ESCWA's Internet-draft. The ADNS-WG suggested, however, that space character is the "natural" separator in the Arabic language and that the standards need to be revised (if possible) to allow the introduction of this character at a later stage.

- Adopted character set: The international standard bodies consider that Unicode is the standard which should be used. The table suggested in the ESCWA's Internet-draft [5], which is the outcome of discussion with the ADNS-WG, is to be adopted.

As a result, the following table illustrates the set of technical issues discussed, the possible alternatives, and our suggestion.

Issue	*ADNTF/SaudiNIC position*	*ADNS-WG position*
Diacritics	Supported only in the user interface Not stored in DNS records	Not Supported in the user interface Not stored in DNS records
Shadda (U+0651)	Similar to Diacritics	Treated differently and requires processing
Kashida, Tatweel (U+0640)	Not supported	Not supported
Character Folding	Not supported	Not supported
Numerals	Arabic Hindi digits (U+0660 to U+0669) supported only in the user interface Not stored in DNS records Folded to ASCII digits (U+0030 to U+0039)	Idem as ESCWA/SAUDINIC
Word Separator	Hyphen-Minus (U+002D) Space (U+0020) preferred but not supported due to technical limitations	Hyphen-Minus (U+002D). Space (U+0020) preferred but not supported initially as it is not a legal Punycode character. If support can be added later, we should take into consideration the removal of repeated spaces.
Adopted Character Set	Unicode 3.1: U+0621 to U+063A U+0641 to U+064A U+0660 to U+0669 U+0030 to U+0039 U+002D U+002E	Unicode 3.1: U+0621 to U+063A U+0641 to U+064A U+0660 to U+0669 U+0030 to U+0039 U+002D U+002E

ADNS organizational issues

- ADNS structure: This means how to map the hierarchical structure of DNS into an acceptable Arabic scheme. While the technical issues are not problematic and need not be subject to much debate, this issue could be largely debated. The main problem tackled is the definition of gTLDs (equivalent to .com, .gov, .org, .info, etc.) and ccTLDs (equivalent to .fr, .uk, .eg, .sy, etc.)

- Regarding gTLD, the direct translation of gTLDs into Arabic is strongly opposed as the resulting domain name does not really fit with the Arabic languages structure and may look awkward for an Arabic language speaker (which is exactly the opposite of the goal of using ADNS). The following structure is proposed:

  <A-TLD>.<entity-name>

  Where, <entity-name> represents the Arabic name of the entity and <A-TLD> represents an Arabic TLD. Hex-coded Unicode values written below from left to right represent Arabic character originally typed from right to left. Example:

  المركز-التجاري.سورية

  u+0627 u+0644 u+0645 u+0631 u+0643 u+0632 u+02D u+0627 u+0644 u+062A u+062C u+0627 u+0631 u+064A u+002E u+0633 u+0648 u+0631 u+064A u+0629

  A major critic against this proposal is the alteration of the semantics of the gTLD which allows usually to define the nature of the concerned entity and was quite successful in the Internet to an extent that several new gTLDs were added (such as .info, .int, .biz, etc.) This information is then put inside the entity-name which has no semantics and is actually treated as a simple string by DNS resolving mechanism. This will have a tendency of flattening the DNS service and would make the domain name longer, and would eventually result in a longer resolving time.

  As a result, no real agreement was reached yet on the gTLDs issue, given that the decision is not only dependant on the Arabic countries, but should also be discuss with ICANN. One suggestion is that a .arb or .arab gTLD should be created and all Arabic gTLDs should be registered under this domain. This is still the subject of ongoing discussions.

- Regarding ccTLDs, the discussion is related to whether a short or long form of the country name should be used, and the RFC suggests a root-server based solution which would allow the users to use any of the three possible forms (short, long, and long with *Al Tareef*). The stored string can be any of three forms, and the translation can be done during the preparation of the query.

- Operational issues: These issues concern mainly how registration information should be handled and how to define the registration structure. The ICANN model is recommended, where there are accredited registrars that can appoint resellers at a premium. One critical point to be addressed here is how the registrars should handle "variants" of a domain name, and the possibility of considering several domain names as equivalent. This would mean that the registration of a domain name would result automatically in the registration of several other domain names added to the same zone (or at least, blocking these other domain names from registration). This can be very useful in handling the *Hamza* problem, where we cannot guarantee that non-Arabic speaking user can enter a character like 'ؤ' correctly. So if all the *Hamza* variants are registered systematically, then this solution would tolerate user mistakes and allows to retrieve the correct domain even in the case of domain name misspell. If such an approach would be taken into consideration, then we need to implement a deterministic algorithm to allow the generation of the variants, so if it is applied to any variant of a domain name, it would always generate the same set of equivalent domain names.

- Legal issues: These issues are related to copyrights and trademarks, and should be discussed as early as possible in order to avoid similar situations which happened in the English speaking Internet, where more than 90% in the words which were in the Webster had been reserved for speculation purpose. Given the lack of coordination between the different Arab countries in legal issues, it is very probable that this particular subject should be discussed at the highest level possible.

Work to be done

The meetings of the ADNS-WG have resulted in an agreement on most of the technical issues. The following points are still to be addressed :

- ADNS structure, and mainly the gTLD structure which is still not “convincing”. The ccTLD system described in the ESCWA’s Internet-draft should meet the expectations of all users and would not be difficult to implement.
- Operational and legal issues, registrars, legalities, forbidden domain names (should there be such a thing?), trademark protection, etc.
- Migration from already existing ADNS proprietary schemes applied by some companies to the final standard ADNS. A list of these schemes needs to be prepared and contacts with their providers should be concluded in order to agree on a common migration path with a clear timetable and milestones. Eventually, a mechanism needs to be defined for resolving conflicts which may occur between companies registered with two proprietary registrars and would end with claiming the same domain name.
- How to interact with non-Arabic speakers, and how ADNS URL and email addresses can be sent to non-Arabic speakers which can still be workable. The RFC 3490 (Internationalizing Domain Names in Applications (IDNA)) could be a good starting point, but it is still not sufficient to address the Arabic problems because it requires that the user can still read and type Arabic characters.

References

[1] Faltstrom, P. Hoffman, P. and A. Costello, "Internationalizing Domain Names in Applications (IDNA)", RFC 3490, March 2003.

[2] Hoffman, P. and M. Blanchet, "Nameprep: A Stringprep Profile for Internationalized Domain Names (IDN)", RFC 3491, March 2003.

[3] A. Costello, "Punycode: A Bootstring encoding of Unicode for Internationalized Domain Names in Applications (IDNA)", RFC 3492, March 2003.

[4] Al-Zoman, "Supporting the Arabic Language in Domain Names", October 2003

[5] Abdel-Ati, et al, “ADN Task Force Guidelines for Arabic DNS”, Internet draft, June 2004.

[6] Bakleh et al, “Internationalized Domain Names Registration and Administration Guidelines for Arabic Characters Group of Languages”, Internet draft, Sept. 2004.

[7] ICANN Guidelines for the Implementation of Internationalized Domain Names, Version 1.0, 20 June 2003.

[8] ICANN Guidelines for the Implementation of Internationalized Domain Names, Version 2.1, 22 Feb. 2006.