

# Arabic Domain Name System (ADNS) Status and Issues

*Comments by the Syrian Work Team*  
November 2004

## Introduction

Arabic DNS has been largely debated as one of the most urgent issues which constitute a considerable hurdle against the development of Internet use in the Arab world. Several workgroups were formed in order to bring a solution to this problem, and two Internet drafts were lately issued as one extra step towards a standard solution. The first draft was issued by ESCWA, it contains a set of guidelines that are in coherence with the IETF IDN standard and take into account some Arabic-language specific issues just as recommended by ICANN and by Dr. Al-Zoman research work. The second draft was issued by the Arab Information engineering task force (AIETF).

Given that an Internet draft is usually a first step towards a standard solution, the evolution is conditioned by the consensus of the concerned community about the correctness of the implementation and the consistency of this solution with the international standards. These drafts 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 clearly defining all the “ingredients” needed to define and implement a successful, workable ADN solution, which would be accepted by all Arab countries. In this context, it should be definitely seen as a “work-in-progress” document, which should evolve towards a final, exhaustive and complete document. The final document would contain a clear list of all the issues, classified by nature, with the different solutions and approaches, the adopted solutions and the reasons behind this adoption. Achieving this status requires a broad participation of all the concerned people, including those who issued the first Internet drafts.

Hence, this document contains mainly a reiteration of the list of issues which need to be addressed in order to resolve the ADN problem, most of them are mentioned in the Internet drafts, but some of them are not. The list of issues mentioned here may grow vertically to present the discovery of new issues we failed to detect initially, or horizontally to present the different solutions brought to a particular problem. We do not endorse any particular solution at this moment, as we believe that the solutions should come as a result of the discussion and the consensus. We just mention the solutions suggested by the RFC draft (and remember that it is a draft, which means that all the solutions mentioned inside are merely proposals which need to be studied, evaluated, and eventually accepted as is or amended or eventually replaced). Needless to say that any suggested solution needs to be compatible with international standards and rules adopted by the IETF and ICANN, and in particular the set of IDN standards as defined in RFC 3490-3491-3492.

## What is IDN

DNS stands for Domain Name System, and its goal is to provide people with a clear and easy-to-use way of addressing computers connected to the Internet. 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, then extended by numerous subsequent RFCs.

In one sense, the DNS remains the only Internet-wide deployed data base used successfully, with more than 100.000.000 domain names stored, which also makes it hard to change.

For about 20 years, DNS was restricted to case-insensitive ASCII letters (a-z), digits (0-9) and hyphen (LDH). This became a definite obstacle against the globalization of the Internet and its generalization among users who are not familiar with ASCII. This character set was definitely not sufficient to meet the requirements of users who are native speakers of other languages, and that resulted in an increase of demand for the “Internationalization” of the DNS, also known as “Internationalized domain name”, or IDN.

The main objective of IDN was to allow the use of domain names which are not restricted to the mere 38 characters used originally in the DNS. Instead, the IDN is associated with Unicode (ISO-10646) based characters, which contains more than tens of thousands possible “code points”. The technical solution for the IDN was introduced with the RFCs 3491,3491,3492 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.

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

- 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.

## **IDN and ADN 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, we are talking about the Arabic language and its features. Such as the appropriate character set (Unicode code points), the use of diacritics (tashkeel), of 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 conflict remains on a few minor issues. These issues need to be discussed in the framework of the IDN standards and RFCs (3491,3491,3492).
2. *Organizational issues*, which are not covered by the IDN standards as they are much more related to ICANN activity 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. It is expected that most of the debate will concentrate on those issues. One example is the structure of the TLDs (gTLD and ccTLD) which affects heavily the structure of the ADNS. One another example is how to define registrars

and how to handle trademarks and how to avoid domain name reservation for the sake of speculation (like what happened in the Latin DNS, with more than 95% of Webster words being reserved).

## **ADNS technical issues**

Before we start discussing the technical issues in details, we must put forward a very important rule of thumb which would 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, there are quite a few rules of thumb which should be 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 like MIT !!). A long domain name means simply more errors typing the domain. The modern browsers are doing their best to 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 complications should be implemented. We need to stress here that (in our opinion) a domain name should not be considered as a legal Arabic sentence. In other terms, our goal is not the production of 100% correct Arabic phrase to be used as a domain name, but to produce a set of labels which would look familiar to the Arab user, and still keeps an acceptable level of conformance with Arabic language rules.
- Reduce the discordance between what is written at the 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 an ACE 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 these 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. In the Arabic language they affect heavily the meaning of the words. But it is largely observed that they are rarely used in the technical texts and documents, and they are written only when their absence may 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 or in the stored name.

The ESCWA draft adopts the Zoman position of supporting tashkeel at the GUI level without storing it in the zone name. The AIETF adopts a similar approach in suggesting that diacritics should be considered as legal Unicode characters that could be used to write the domain name, but delays the question of storing them at the registrar's data base. It is not clear if the use of diacritics would have any positive impact on the simplicity or ease of implementation of the ADNS. A more

reasonable approach would be the third option, i.e., not using diacritics at all. The main reason behind this recommendation is to avoid lengthy domain names. Allowing diacritics will only add another source of errors without giving the user any clear added value.

- Shaddah (U+0651) has been usually treated as a part of the diacritics, which might be linguistically correct, but they should not get the same treatment when it comes to domain names. For instance, a considerable number of names would have a totally different meaning if shaddah is ignored (consider سمان، سمان، جبان، جبان).
- Kasheeda or Tatweel (Horizontal Character Size Extension): This extension is purely visual in the Arabic language, and while it has a character code, its presence (or lack of presence) does not affect the meaning. Hence, the ESCWA's draft position suggesting not supporting the character extension is perfectly logical. The possible solutions here are the following:
  - a. Kasheeda is supported;
  - b. Kasheeda is not supported (Current recommendation).
- Character folding: which is the process where multiple letters (that may have some similarity with respect to their shapes) are folded into one shape. Some folding examples are: folding Teh Marbuta and Heh at the end of a word, folding different forms of Hamzah, folding Alef Maksura and Yeh at the end of a word, and folding Waw with Hamzah and Waw. The ESCWA's draft suggests strongly that character folding should not be supported because "It will lead to have only one form (word) out many other forms of words that are made by all the combination of folded characters". On the other hand side, the AIETF drafts suggests that the Alef Maksura (U+0649) should be folded with Yeh (U+064A). The only arguments why folding would be used are cyber-squatting avoidance, and accommodating non-native Arabic speakers. Cyber-squatting is when someone registers طرفه.شركة 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 the need to use 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. We believe that this issue needs further discussion before recommending the use of character folding or not.
- Numerals: The Numerals issue is somehow problematic, given that several Arab countries use the Hindi Numerals instead of the Arabic numerals. One potential problem with Hindi numerals is the similarity between the dot “.” Character and the 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. The ESCWA draft suggests that the set of Hindi numerals should be folded into the Arabic one, the AIETF draft suggests that only the Hindi numerals set should be used. A clear recommendation is needed regarding this particular point, but given the widespread use of both numeral sets, and the fact that stringprep would be required to transform the Hindi numerals into Arabic numerals, then we are not gaining anything by restring the use to Hindi numerals; therefore, the proposal of keeping both sets seems the most realistic one.
- 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 domain name 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 used, as it is not a legal ACE character, therefore it should be folded into another legal ACE character. One another problem with the use of space as a separator is the possibility of the user entering multiple spaces by mistake. This also can be avoided through reducing multiple spaces into one during the stringprep phase. The alternative separator suggested

is the minus-hyphen “-“ character, which is also suggested by the ESCWA draft, but opposed by the AIETF draft under the pretext “The use of dash character (-) to separate words is NOT acceptable according to IDNA (RFC-3490) which prevents the use of characters that belong to another language when reserving a name for a certain language”. We believe that this is due to a misunderstanding of the RFC-3490. The most flexible proposal would therefore be the following: allow the use of both hyphen and space characters as separators. Both characters would be transformed into a hyphen during the stringprep, which would avoid keeping the space in the final ACE string, as the legal ACE character set does not contain space. Hence, both domain names:

مؤسسة-الطيران.سورية

مؤسسة الطيران.سورية

Would be considered as equivalent as they will generate the same ACE string.

- Adopted character set: The international standard bodies consider that Unicode is the standard which should be used. Both drafts suggest a subset of the Arabic Unicode characters should be used (cf. table below). We believe that the table suggested by the ESCWA draft is the most accurate one.

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

Issue	ADNTF / SaudiNIC	AIETF	Arab Work Team
<b>Diacritics</b>	Supported only in user interface Not stored in DNS records	Supported only in user interface Initial Phase: Not stored in DNS records Next Phases: Stored in DNS records	Not Supported in user interface Not stored in DNS records
<b>SHADDA (U+0651)</b>	Similar to Diacritics	To be treated differently and requires some algorithm	treated differently and requires some algorithm
<b>Kashida TATWEEL (U+0640)</b>	Not supported	Not supported	Not supported
<b>Character Folding</b>	Not supported	Only YEH (U+064A) folded to ALEF MAKSURA (U+0649)	Needs further discussion. In principal refused, except the YEH to ALEF MAKSURA

<b>Numerals</b>	ARABIC-INDIC DIGITS (U+0660 to U+0669) supported only in user interface  Not stored in DNS records  Folded to ASCII DIGITS (U+0030 to U+0039)	Only ARABIC-INDIC DIGITS (U+0660 to U+0669) supported in user interface and stored in DNS records	Idem as the ESCWA/SAUDINIC
<b>Word Separator</b>	HYPHEN-MINUS (U+002D)  SPACE (U+0020) preferred but not supported due to technical limitations	SPACE (U+0020)	HYPHEN-MINUS (U+002D).  SPACE (U+0020) preferred but not supported initially as it requires processing during the stringprep phase. Support can be added later, and should take into consideration the removal of repeated spaces.
<b>Adopted Character Set</b>	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+0020 U+0621 to U+063A U+0641 to U+064A U+064E to U+0651 U+0660 to U+0670	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  Eventually U+0020 if space is accepted as a separator.

### **ADNS organizational issues**

- ANDS structure, which means how to map the hierarchical structure of DNS to an 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 by the RFC and derived from work by Dr. Zoman is the definition of gTLDs (equivalent of .com, .gov, .org, .info, etc.) and ccTLDs (equivalent of .fr, .uk, .eg, .sy, etc.)
  - Regarding gTLD, the direct translation of gTLDs into Arabic is strongly opposed as the author does not really fit with the Arabic languages structure and may look awkward for Arabic language speakers (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

Actually, this structure is no structure! The proposal boils into removing the semantics of the gTLD which allowed usually to define the type of the concerned entity and was quite successful in the Internet to an extent that several new gTLDs were added (like .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. While we agree that the resulting strings would not sound in perfect harmony with Arabic pronunciation rules, it is still better structured than the current proposal.

- Regarding gTLDs, it is discussed 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, mainly concerning how registration information should be handled and 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 of 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 to handle the hamzah problem, where we cannot guarantee that non-arabic speaking user can enter a character like ‘ؤ’ correctly. So if we register all the hamzah variants systematically, then this solution would tolerate user mistakes and allows him to retrieve the correct domain even if he misspell the domain name. 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, regarding copyrights and trademarks. Which should be discussed as early as possible in order to avoid similar situations which happened in the English speaking Internet, as more than 90% in the words which were in the Webster had been reserved for speculation purpose. Given the lack of coordination between Arab countries in legal issues, it is very probable that this particular subject should be discussed at the highest levels possible.

## Work to be done

The Internet drafts are a good basis towards the achievement of a finally operational standard ADNS setup. Therefore, it is suggested that any taskforce to be formed would start its work by finalizing the issues which are clearly stated as pending or which are not fully covered in the ADNS. Actually, it is obvious that purely technical issues are well-addressed and need very minimal further work (if any). The remaining issues are rather subjective, political, and organizational rather than technical. These issues are the following:

- ADNS structure, and mainly the gTLD structure, as the ccTLD system described in the ESCWA draft should meet the expectations of all users and would not be difficult to implement. But the gTLD is not really convincing and should be studied more carefully.
- 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 done in order to agree on a common migration path with a clear calendar 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 can we send ADNS URL and email addresses to non-Arabic speakers who can still use it and be workable. The RFC 3490 (Internationalizing Domain Names in Applications (IDNA)) can be a good start, but it is still not sufficient to address the Arabic problem because it requires that the user can still read and type the Arabic letters, which is not evident at all).

## References

- Faltstrom, P., Hoffman, P. and A. Costello, "Internationalizing Domain Names in Applications (IDNA)", RFC 3490, March 2003.
- Hoffman, P. and M. Blanchet, "Nameprep: A Stringprep Profile for Internationalized Domain Names (IDN)", RFC 3491, March 2003.
- Costello, "Punycode: A Bootstring encoding of Unicode for Internationalized Domain Names in Applications (IDNA)", RFC 3492, March 2003.
- Al-Zoman, "Supporting the Arabic Language in Domain Names", October 2003
- Abdel-Ati, et al, "ADN Task Force Guidelines for Arabic DNS", Internet draft, June 2004.
- Bakleh et al, "Internationalized Domain Names Registration and Administration Guidelines for Arabic Characters Group of Languages", Internet draft, Sept. 2004.