



Names and Naming for the DNS
PRELIMINARY VERSION - © Internet Society, May 5, 2006

by S. Laing and J. Klensin

Introduction

Current international discussions among those who are thinking about the status and evolution of the Internet include complex topics related to the concepts of local and global naming. One practical problem shared by users around the globe is that when they use a local name in the global environment, they get results far from those expected. This article aims to help users understand the difference between the local and global environments, with reference to names and how they play out in navigation of the Internet.

The DNS: A Global Database

The Domain Name System (DNS) is the Internet's system for permitting there to be at least one globally unique and unambiguous name for every Internet resource. As a distributed database, the DNS supports these unique and unambiguous global names, which are called "fully qualified domain names" (FQDN). The FQDN is the name that goes all the way to the implied root and is actually present in the DNS. (For an overview of DNS technology and operations, please see Member Briefing #16, "The Internet Domain Name System Explained for Non-Experts," by Daniel Karrenberg.)

Resource Identifiers

End users rarely use only a domain name. Instead, a domain name is usually embedded in an email address, in a web URL (Uniform Resource Locator), or the more general URI (Uniform Resource Identifier) or IRI (International Resource Identifier).

A URI or an IRI usually consists of several elements; a protocol identifier (the very common "http" is an example); a domain name, sometimes with additional information such as the local part and "@" of an email address; the "tail" of the URI (additional qualification and directions); and syntax delimiter characters that identify and separate various fields, such as the aforementioned. (For more information about URIs and IRIs, please see RFC 3986 and RFC 3987.)

The length of a domain name increases along with the depth of the hierarchy. As the above paragraph would indicate, a URI or an IRI can become much longer. Most of the discussion below concerns the relationship between FQDNs and local names and usages,

which is very similar to the relationship between URIs (or IRIs) and their local abbreviations (e.g., “favorites” and “bookmarks.”)

Local Names: Human Convenience

The FQDN, especially when included in a URI, is typically longer than the other alternatives. FQDNs and full URIs are often difficult to remember and type exactly. That difficulty has been a strong motivation for users and those supporting them to have sought and found ways to create and effectively use more convenient names: local names. Type of local names include abbreviations, partial names and aliases. They are contextual in nature, and, to be useful, have clear and reliable referential value within the local environment. Outside the local environment, however, local names lose their meaning. This relationship between a convenient name and a fully qualified name can be seen in everyday human communication.

In a human context—broader than Internet—such as within a family (a local environment), one can typically refer to other family members by using just a first name or a nickname. Either may be considered to be a local name, since neither provides universal identification of the person involved, yet in that family (local) context, the first name or nickname may point to exactly the intended reference, rendering use of the full name unnecessary.

However, in conversation with someone outside the family (in the global environment), the speaker’s use of first name or nickname may cause the listener to think about a list of possible references, rather than pointing just to the unique individual to whom the speaker intends to refer the listener. Many people share the same first names, nicknames, and even full names, such that more information is required to pinpoint the exact individual reference. In contrast, no two Internet resources share the same name in the DNS, making the fully qualified domain sufficient to produce an accurate reference anywhere in the global environment.

Local Environments

With respect to the Internet, the local environment can be understood as delineated in terms of a host computer, a particular network, or a specific software configuration. The following examples are from the point of view of a user in the hypothetical local environment of *SomeUniversity*’s chemistry department.

In chemistry department, users make frequent use of the domain *pooh.chemistry.someuniversity.aka.uk*, gaining access to it by typing only the string *pooh*—a partial name. When they talk at work about that domain, they call it “*pooh*.” It’s a local name that works just fine among themselves, within the department.

A user from the chemistry department, at a friend’s computer in the physics department, types *pooh* and is surprised to see that the domain found is *pooh.physics.someuniversity.aka.uk*. What happened?

The person didn't realize that "pooh" is a local name within the chemistry department, and also didn't realize that the physics department is a separate local environment, though both departments are equal as members of a larger local environment—*SomeUniversity*.

Only after the local name *pooh* was shown to refer to a completely different host did the person gain understanding of the essential nature of a local name: its meaning is lost outside the local environment, and one does not necessarily know when one is communicating outside the local environment.

On a home computer, the chemistry department user types *pooh.chemistry* (another partial name, with more labels to distinguish it from *pooh.physics*), and finds that *pooh.chemistry* is a partial name contained within many other domain names. Expanding the partial name with yet another label (e.g., to *pooh.chemistry.someuniversity*) produces another list of domains with the labels *pooh.chemistry.someuniversity* contained in their FQDNs.

Eventually, users realize that they are in the global environment, where only a global name will preserve the intended reference. The global name (FQDN) goes all the way to the implied root. Typing the FQDN (*pooh.chemistry.someuniversity.aka.uk*) at the home computer allows the chemistry worker at home to find the desired domain directly.

Users like to adopt abbreviations and aliases, in addition to partial names like *pooh*, above. For example, a user might want to call a favorite host by the name *freddy*.

If the user intends to substitute *freddy* as a local abbreviation for the full name *fred.example.com*, then *freddy* would be a local abbreviation akin to the type of substitutions commonly supported by web browsers (bookmarks or favorites) and email agents (nicknames in lieu of typing entire email addresses.)

Alternatively, the user may intend *freddy* to be a general synonym for *fred*, (so the user types *freddy* where others would type *fred*). That can be accomplished by creating a local alias to be substituted whenever the string *fred* appears in a DNS label. However, the potential for confusion with this technique is very high. Hosts whose FQDNs actually contained a label *freddy* would become inaccessible, at least without special "escapes" to turn off the substitution.

If the user wants *freddy.example.com* to be a global synonym for *fred.example.com*, that could be accomplished by the use of a DNS alias, called a CNAME record, which is entered into the DNS database, thereby making it public and global. The CNAME record associated with *freddy* in the *example.com* domain would point to *fred* in the same domain. It should be noted that the owner of the host *fred.example.com* could be offended by the alias *freddy*; the use of CNAME aliases requires care to avoid inadvertent creation of social problems.

As shown above, there are three different mechanisms to provide the "*freddy*" name. One substitutes for the entire FQDN (and applies equally well as a substitute for a URI). The second substitutes for a single name, locally. And the third involves an actual DNS entry for the alias. Each has advantages and disadvantages. The first two can be implemented by a single user or in that user's environment, or can be implemented more broadly (for the users in a particular department or of a particular ISP, for example.) The third requires global DNS implementation and access to the relevant zone to do that implementation. These relationships are discussed in more detail below.

Tradeoffs between Local and Global Names

While there are advantages to the local use of names that are shorter or more “user-friendly” than fully qualified domain names, the disadvantage comes to the user who travels or communicates outside the local environment and through multiple local environments. In one environment the local name will not identify any target at all. In another it will refer to A, and in yet another it will refer to B, where A and B have no relation to each other or to the object that the name refers to in the original local environment.

The use of local names presents no threat to the integrity of the Internet or the DNS namespace, because every host will always be represented in the DNS by an available and globally unique FQDN. Problems do arise when users try to pass *insufficiently qualified* names to colleagues in other environments. To be assured of accurate communication across environments, the FQDN is the name to use. In case one is unaware of having changed environments, finding an unexpected target in response to the use of a name is a call for the use of the FQDN.

How to Make a Local Name

Toward a better understanding of the power and limits of local names, below are several different techniques for creating local names. All of them have been in use in one context or another since the DNS came into being, and experience has shown that they work and cause no problems as long as users are aware of what is happening when that is necessary.

DNS Search Rules

Search rules can be used to find the intended reference of a local name, even when the name is potentially contained in other domains.

Back in the chemistry department again, in response to typing the string *pooh* the DNS resolver could look for the name as:

pooh.chemistry.someuniversity.aka.uk
pooh.someuniversity.aka.uk
pooh.aka.uk

One user in chemistry who frequently deals with the physics department might set up his system to look for:

pooh.chemistry.someuniversity.aka.uk
pooh.physics.someuniversity.aka.uk

The order of the search is very important. When the same name exists in two domains, as *pooh* does above, then the domain found first will be the domain listed first: the search proceeds in the order the domains are listed in the search rules.

The introduction of a new name may cause names that worked earlier to suddenly and unexpectedly produce references to different target hosts. By permitting a more complex search arrangement than is supported by most DNS resolvers, that problem can be reduced.

As a case in point, during the 1980s, the Internet community learned a pertinent lesson. Educational institutions all over the world had independently established the *.cs* subdomain for their computer science departments. One could use a partial name ending with the subdomain *.cs* and have success in locating the desired host. When a new country code, *.cs* for the then-nation of Czechoslovakia, came online, suddenly there was ambiguity: did *example-host.cs* refer to a host in the new top level domain, or did it refer to a host in the local institution?

Complicating things further at that time, there was a convention of assuming that any DNS name containing at least two dots was fully qualified. Before the *.cs* TLD (top level domain) came online, the old “two dot test” would find nothing and the search rule would be applied. After the *.cs* TLD came online, if there was a host in the *.cs* TLD (and maybe even if there wasn’t, depending upon how the rules were constructed), the hosts in the local computer science department essentially disappeared overnight.

In 1989, the DNS specifications with regard to this sort of searching were updated and clarified as part of a general effort to identify and clarify issues with key specifications. It upgraded the earlier recommendation for support of the “trailing period” convention to a requirement, and also suggested that DNS resolvers treat the name as an FQDN if more than two periods appeared in a name, and therefore not attempt any searching. (For more information about these specifications, please see RFC 1123.)

The two rules helped, but were not sufficient: unambiguously global references require names that are unambiguously fully qualified domain names, rather than strings that are dependent on local context and convention.

DNS Aliases: The CNAME Resource Record

An alias can be placed into the DNS so that a name points to another name, which would then point to an Internet resource. This sort of alias, the CNAME alias, is part of the DNS, and therefore is available in the global environment.

Thus, it is possible to provide a global alternate name for a host. However, it is important to be aware of the fact that various application protocols require users to be clear about which name is the primary name and which one is the alias. Here are two examples:

- 1) The email transport protocol (SMTP) requires that only primary names be used in the announcements of the sender (client) host.
- 2) Implementations of the main web transport protocol (HTTP) could have configuration issues if DNS-based CNAME aliases are used in URLs, especially if they refer to servers that have multiple names.

Local Aliases

In some circumstances, local aliases are a better choice than the global reference provided by CNAME aliases. For example, a host may be known within a local workgroup by an abbreviation or alias that could or would likely be considered offensive to others outside the context of that local environment. Also, users may find it helpful to those within their local environment to create aliases with spellings that reflect local linguistic habits, or with formations designed in some particular way to reduce locally predictable, cognitive confusion.

The mapping, from the string that the user types to the Internet resource, is performed either in the local resolver that serves the local environment, or in the application program itself. If a stub resolver (one that hands all requests off to another resolver rather than performing the DNS lookups itself) is deployed, then the mapping is performed by the DNS forwarder. Either way, local names can be freely used, because the servers see only the fully qualified names that actually exist in the DNS.

Still, the traveling member of the local workgroup must know the underlying names for the local aliases, or be able to find them, because outside the local environment, the local alias won't work or, worse, may identify another resource entirely.

Internationalized Domain Names and Alias Names

An Internationalized Domain Name (IDN) may be thought of as a type of local alias. The user inputs a convenient string in local characters—the local alias. Using the "IDNA" protocol, the DNS resolver (or a mechanism in front of it) converts that string to the form actually used in the DNS—the FQDN. The DNS subsequently uses that form in lookups and communications at a low level with other systems.

IDNA differs from the aliases discussed above because it specifies a global convention for making the conversion. In theory, the same non-ASCII string typed anywhere in the world should result in the same query to the DNS. However, at present, the user traveling to an environment where different characters or different keyboards are supported would be wise to make the (internal – so-called "punycode") FQDN available as well as the native-character alias form. This applies also if domain names are to be

sent to people who can't read or copy the characters involved. Confusion of many possible sorts would be avoided.

Other Types of IDN Translations

In some cases, it may be sensible to depart from IDNA to use a different type of local alias, one that is more nearly related to the translation of a name rather than an algorithmic remapping of local characters. This would be particularly important if the common name of a host or subdomain might reasonably be reflected in multiple languages and different in each.

Considering that there are over 6000 languages in the world, and that some countries claim to have over 200 languages in active use, providing aliases for all of the languages that might usefully refer to a given domain may become impractical. It is especially impractical for the DNS root, where both operational and administrative considerations argue for keeping the total number of name entries relatively small. For second level domains, adding hundreds or thousands of names for each one that exists today may still be operationally problematic, but, within limits, doing so is primarily a matter of update time and economics.

One proposal for doing this involves making the name, in the language and script of the user who operates the computer, completely local to that user and computer or to the user's workgroup. This could be accomplished as a special case of the "local aliases" approach outlined above, applying such local aliases to either the entire DNS name (the FQDN) or to one or more of its label components. From another perspective, it is a variation of something that many users have been doing for years: bookmarks or favorites in browsers, address book entries for email programs, and similar arrangements for other applications, typically permit the user to specify a name for the entry that is different from that selected by the party who created the target host or site. That name can reasonably be in a non-ASCII script; few contemporary systems require that it be in ASCII.

So both those providing software interfaces for the user, and the users themselves, have a wide range of options for aliasing, most of which can support internationalized names independent of what actually appears in the DNS.

Summary and Conclusion

There are many purposes for which the names that are actually placed in the DNS are critically important. However, the vast majority of those situations involve the identification of network resources for use in computer-to-computer communications. If one is going to communicate across a global Internet, such communications require that DNS entries be unique, unambiguous, and global.

Ambiguity of references makes global inter-working, at the computer and networking level, impossible. Resolving ambiguous names by specifying the root in which they are listed is at odds with the design of the DNS. However, the names by which users refer to

network resources need not be the same as the names used to communicate between those resources. A variety of mechanisms are available for permitting alternative or alias names to be used.

These mechanisms have been in use, in one form or another, since the DNS was first deployed and, in some cases, for much longer, giving many years of evidence that they do not, in and of themselves, cause either network fragmentation or other serious side effects. They do require that a user who changes environments have access to the underlying names.

Recommendations

The mechanisms for DNS names have been in use in the Internet since the time the DNS was deployed, and even earlier. The local abbreviations for URIs date back to the first modern web browsers. Such names can present the user with convenient abbreviations, local spellings or terminology, and other conveniences.

Internet users, and those designing interfaces and systems for them, should be aware that there are a variety of different ways to provide and support names that are more user-friendly than typical FQDNs or URIs. Their cost is that users who shift from the environments in which the names are defined to environments where those names are not supported, or who share information across those boundaries, must understand how to derive and use fully qualified names and full URLs.

As URLs become more complex, as hierarchy deepens, or as more TLDs are introduced into the DNS, the use of one or another type of local alias is likely to increase. Rather than intensifying theoretical concerns regarding the relation between aliases and network fragmentation or other serious side effects, those who are concerned with Internet development could choose to recognize the value of local aliasing and seek to improve upon existing mechanisms for making them work.

For example, the elements of local environments that support local aliases could be made exportable with one set of mechanisms, easily allowing users to take aliases, search rules, and the like from one computer or environment to another. Exporting an environment that is supported by an ISP on behalf of a user may be less advantageous because it is less under the user's control and less easily re-established in a different ISP context. Finding ways to keep the aliases as close to the user as possible would, for example, argue in favor of renewed protocol work to permit users to download applications and environmental settings from convenient servers rather than carrying them around on portable media or portable machines.

While concerns about Internet fragmentation deriving from the use of user-friendly names are unsubstantiated by decades of Internet experience, there is an important challenge to Internet development posed by the problems faced by users working outside their local environments. In keeping with the DNS design principle of permitting at least one unique and unambiguous name for an Internet resource, and in acknowledging the

human user's proclivity for local naming, the best solutions to the problems that local users have in the global environment may be found in new or improved protocols.

For Further Reading:

Member Briefing #16:

“The Internet Domain System Explained for Non-Experts”

by Daniel Karrenberg

Signposts in Cyberspace, National Research Council, 2005

Chapter 3, for DNS basics

Member Briefing #18:

“Internationalizing Top Level Domain Names: Another Look”

by John Klensin

RFC 4185:

“National and Local Characters for DNS Top Level Domain (TLD) Names”

J. Klensin, October 2005

RFC 3986:

“Uniform Resource Identifier (URI): Generic Syntax”

T. Berners-Lee, R. Fielding, L. Masinter, January 2005

RFC 3987:

“Internationalized Resource Identifiers (IRIs)”

M. Duerst, M. Suignard, January 2005

RFC 1123:

“Requirements for Internet Hosts—Application and Support”

R. Braden, October 1989

RFC 2672:

“Non-Terminal DNS Name Redirection”

M. Crawford, August 1999

RFC 3490: I

“Internationalizing Domain Names in Applications (IDNA)”

P. Falstron, P. Hoffman, A. Costello, March 2003

RFC 3491:

“Nameprep: A Stringprep Profile for Internationalized Domain Names (IDN)”

P. Hoffman, M. Blanchet, March 2003

RFC 3492:

“Punycode: A Bootstring encoding of Unicode for Internationalized Domain Names in Applications (IDNA)”

A. Costello, March 2003

Verisign White Paper: A Proposal for DNAME Equivalence Mapping for TLD Strings,
found at www.icann.org