# NETWORK ADDRESSING SYSTEM AND METHOD USING TELEPHONE NUMBERS

## Cross Reference to Related Application

This application claims the benefit of U.S. Provisional Application Serial No. 60/143,246 filed July 9, 1999.

## Field of Invention

The present invention relates to network addressing over internets and intranets. The invention finds particular application in converting telephone numbers into a multi-level domain name to enable devices on a network to communicate. It is to be appreciated however, that the present invention may find further application in private networks or public networks and using private protocols or internet protocols.

## Background of Invention

Originally, only numbers were used as network addresses on the internet. The numbers are called Internet Protocol numbers, or in short, an IP address. These numbers had virtually no meaning to anyone and were hard to remember.

Later, the Internet Assigned Number Authority (IANA) created a Domain Name System. Each domain name would use a more understandable and meaningful name to point to an IP address. For Example, "IBM.com" is a domain name that routes a user to a specific IP number or address such as 198.81.209.2 (an IBM IP address). Now, the entire world of the Internet users find themselves using this internet domain

name registration system which is a core component in getting an internet presence.

A domain name typically includes a name, plus a "." (dot) or separator, and a Top Level Domain (TLD) such as "com", "net", and "org".

While the domain name system made addresses easier for people to remember, the internet domain name registration system suffers from several current problems. For example, there is a limited supply of internet domain names. Only one company can own or register any given name on the internet such as Computers.com. Other top level domains can be employed such as Computers.net and Computers.org but these options are considered inferior to Computers.com. A second problem is the high cost of internet domain names. Name exclusivity has created a booming internet domain name market which has some domain names reportedly selling for millions of dollars. Also, the current monopoly and/or limited number of domain name registrants, have caused all domains in general to be more expensive.

A second problem is that currently internet domain names can take up to 72 hours to activate. This creates a problem for those internet users who need to set up their site quickly. Moreover, once registered, an internet domain name is fixed. A change of even one letter from the original domain name becomes a new name and must be reregistered. This need to register a domain name incurs additional cost and runs the risk that the new name may not be available. Indeed a cottage industry of domain name registrants or cyber-squatters has surfaced because they anticipate that eventually someone will need, and pay for, the name they are

registering. This "inflexibility" becomes a bigger issue with the convergence of the internet and telephone networks.

An attempted solution to some of the problems outlined above is a sub-directory based system of Domain Names called the "Forward Slash" method. For example, a company such as Realtor.com would add a Real Estate Agent's name to the end of the string of characters separated by a forward slash i.e. Realtor.com/AgentsNameHere. This would give the Agent a connection to an inexpensive web site because it is a part of the big company Realtor.com. There is theoretically, no limit to how many forward slashes could be added at the end of each string of characters. However, this method also has drawbacks, such as:

A. **Not IP Addressable** - This means that an individual "site" does not have the option to have a unique IP address and/or a shared one. The only option is a shared IP address with the primary domain name holder.

B. **Time Delay**- If a "Site" forwards a user to another site, the user has to wait for it to do so.

C. **Longer Addresses** - When using the "Forward-Slash" method, the address is almost always longer which creates more problems:

1. The longer address increases the complexity for users because there are more characters to type making more room for errors.
2. The longer address is harder to communicate to others such as a radio announcer saying "Check out Realtor dot com forward slash Agent's Name Here forward slash City."

3. It is too long to remember without writing it down and pen and paper are not always immediately available.

D. **Lack of Uniqueness** - Companies using the Forward Slash method need to use names that are relevant to the specific sub-site yet unique so as not to duplicate the sub-site names because each name can only be used once. This method is difficult in practice.

Business communications such as advertisements, literature, business cards etc. have to include both an internet address as well as a telephone number because of the success of the internet and the ubiquity of the telephone. Using both pieces of information not only creates more information to remember but also clutters the design of the piece of communication.

The above has led some companies to use a telephone number in their domain name. Using a telephone number in an internet domain name is not new. The approaches that the following companies use are severely limited to very specific applications or companies.

As people have started to participate on the internet, some companies have slowly started to integrate telephone numbers into an internet address. Very few individuals and/or organizations have registered an actual dial-able telephone number as an internet domain name.

Probably the most popular telephone number/internet domain Name is "1-800-Flowers.com". This was a natural use for the company since it is their company name, as well as their phone number. This approach once again has a number of the drawbacks of internet domain names in general, as described above. Of interest is the fact that only a single

sub-domain, i.e. "1-800-flowers", is employed. Specifically, this approach has a higher cost than that of the forward slash method and has the additional problem of a potential lack of availability. Additional problems with this approach are more fully discussed below, as they pertain to network providers. Other similar references are "www. 411.com", and "www.1-800-555-1212.com", both having similar challenges.

The present invention contemplates an improved method and apparatus for a network addressing system which includes a telephone number as a component of the address.

## Summary of the Invention

In accordance with one embodiment of the present invention, a method includes receiving a telephone number portion which identifies a device with which communication is desired. The telephone number portion is converted into a multiple level domain name which statically represents the device on the network. A component of the multiple level domain name is the received telephone number portion. Communication is then established with the device over the network.

In accordance with another aspect of the present invention, the converting step includes adding domain separators to the received telephone number portion at determinable locations in the received telephone number portions.

In accordance with another aspect of the present invention, the received telephone number portion includes a separator, and the converting step includes parsing the received telephone number portion for the separator and inserting a domain separator therefor.

In accordance with another aspect of the present invention, additional domain levels are appended to the converted telephone number portion to complete the multiple level domain name.

In accordance with another embodiment of the present invention, an apparatus to establish communication between at least two devices over a network includes a processor which receives from a first device a telephone number portion identifying a second device. The processor then converts the telephone number portion into a static multiple level domain name sufficient to identify the second device on the network.

In accordance with another aspect of the present invention, the apparatus further includes a table which matches the static multiple level domain name to a static IP address.

One advantage of the present invention resides in the ability to have internet domain names that are meaningful and simultaneously more economical than traditionally used.

Another advantage of the present invention resides in the ability to convey both an internet address and a telephone number compactly and economically in a communication.

Still further advantages of the present invention will become apparent to those of ordinary skill in the art upon reading and understanding the following detailed description of the preferred embodiments.

## Brief Description of the Drawings

The invention may take physical form in certain parts and arrangements of parts, and in certain steps and arrangements of steps. The drawings are only for purposes of

illustrating the preferred embodiments and are not to be construed as limiting the invention.

FIGURE 1 is a generic representation of a typical network addressing system or domain name structure;

FIGURE 2 is an exemplary U.S. telephone number;

FIGURE 3 is the telephone number of Figure 2 following conversion into a multiple level domain name;

FIGURE 4 is a flowchart detailing a logical flow that suitably practices the present invention; and,

FIGURE 5 is a graphical depiction of device connectivity across a network as provided by the present invention.

## Detailed Description of the Preferred Embodiments

As used herein an internet telephone number or numbering system is defined as a domain name or set of domain names that enables network addressing system through the use of telephone numbers or internet/intranet networks.

A telephone number portion is herein defined as at least a part of a telephone number which can include the exchange, the area code, and/or the country code. While telephone numbers generally have gradually increasing geographic specificity with the most generic portion preceding more definite portions, (i.e. the country code precedes the area code which precedes the exchange, etc.) the particular order in which a telephone number portion is converted or presented as a static multiple level domain name is immaterial according to the present invention. That is, the received telephone number may be rearranged in any sequence without materially affecting the scope of the subject invention so long as all are likewise arranged or determinable.

As use herein the terms source, target, and/or device are intended to refer to any variety addressable devices interconnected via any of a variety of networks. For example, devices amenable to the present invention include without limitation, computers, storage devices, output devices, telephones, personal information managers, laptop, palmtop, or watch based computers, and the like. Interconnecting networks include without limitation, intranets, or the internet, wireless and/or wireline telephone networks, either public or private, hardwired, infrared, optical, or electro-magnetic networks and the like.

With reference now to Figure 1, the existing domain name structure is generically illustrated. This structure includes a Base Level Domain (BLD) **10**. The base level domain **10**, sometimes called a top level domain or first level domain, is not associated with an IP address itself but is a logical grouping used to distinguish between Countries (e.g. .US, .CA, .UK, .HK, etc.); Colleges (e.g. .EDU); US Military (e.g. .MIL); US Government (e.g. .GOV); Corporations (e.g. .COM, .ORG); and ISP's (e.g. .NET). These base level domains manage any inquiries to the second level sub-domains **12**. The second level sub-domain **12** is normally associated with an IP address when used in conjunction with a BLD **10**. An example of a second level domain name is IBM as used in IBM.COM, or OSU as used in OSU.EDU. The second level domain **12** usually manages any inquiries to the third level sub-domain **14**. The third level sub-domain **14** is also normally associated with an IP address when used in conjunction with both .2LD.BLD **12**, **10**. An example of a third level domain name is SUPPORT as used in SUPPORT.IBM.COM. The third level sub-domain **14** usually manages any inquiries to the fourth level sub-domain

16. The fourth level sub-domain 16 is usually associated with an IP address when used in conjunction with .3LD.2LD.BLD 14, 12, 10. Some examples of a fourth level domain name are PC or AIX as used in PC.SUPPORT.IBM.COM or AIX.SUPPORT.IBM.COM. The fourth level sub-domain 16 usually manages inquiries to the next domain 18 and so on. The Nth level domain 18 represents any and all higher level domains where the N represents integer intervals of higher level domains (i.e. 2, 3, 4, 5, 6,...). All domain names can contain words or phrases consisting only of letters, numbers, and the dash, (a...z, 0..9, '-').

All domain name levels 10-18 are separated by the 'dot', the domain separator 30. This separator 30 is used to designate managing control of higher domain levels to their next lower domain level. Complete domain names consist of two parts: a domain name segment 34 and a base level segment 38. A base level segment 38 contains a .BLD 10, and contains any number of upper level sub-domains. Some examples of base level segments 38 are single level segments (e.g. .COM, .NET) and multi level segments (e.g. IBM.COM; ABC.DEF.ORG; one.two.three.four.five.six.NET). Domain name segments are any group of one or more discrete 'names' separated by a dot 30 that does not contain a .BLD 10. Some examples of domain name segments are single level segments (e.g. ABC, IBM, telenumber) and multi-level segments (e.g. ABC.DEF, Support.IBM, one.two.three.four.five.six.etc). These segments, when combined, form a complete domain name. A domain name tree 42 is any number of domain name segments 34 appended by a base level segment 38.

With reference now to Figure 2, a telephone number portion 50 is entered for translation into a single or

multiple level domain name segment **34** enabling that segment to be grafted onto any existing domain name tree **42** at any point in that tree. The translation of the telephone number portion **50** consists of reducing the number into discrete pieces **52, 54, 56, 58** based on the natural separators **70, 72, 74** of that number, (e.g. dash, parenthesis, dot, or the like). These separators **70, 72, 74** are replaced with a domain separator **30** and the discrete pieces **52, 54, 56, 58** become domains and sub-domains.

Referring now to Figure 3, the multi-level domain name resulting from the entry illustrated in Figure 2 is shown. The telephone number portion **50** was parsed for separators **70, 72, 74** and assuming left to right scan, the separators will likewise be replaced from left to right. Thus Figure 2 separator **70** becomes domain separator **80** in Figure 3. Likewise Figure 2 separators **72, 74** are substituted with domain separators **82, 84** in Figure 3.

Continued reference to Figure 3 also illustrates appending of additional domain levels onto a converted or translated telephone number portion **50** (Figure 2). In the illustrated example, to complete the multiple level domain name with respect to a particular web server, the additional sub domain, "telenumber" **90** and the top level domain, ".com" **92** are appended. Those skilled in the art will now appreciate that domain levels may be appended anywhere in the multiple level domain name without departing from the spirit of the present invention. Moreover, further domains or sub-domains can be used for other country codes, area codes, telephone exchanges, etc.

With respect now to Figure 4, an exemplary process is illustrated by which the telephone number portion of Figure 2

for example is converted into that of Figure 3. The user enters a query or a domain name which includes a telephone number portion suitable to identify a target or desired device across a network, as seen in step **100**. An ambiguity may be introduced at this point when the processor determines where to place the domain separators **30**. In the illustrated embodiment, the ambiguity may be resolved, as seen by decision block **102**, depending on whether the user enters separators in their query. If so, the processor then substitutes domain separators **30** for the user entered (manually or according to some predetermined protocol) organic separators **70**, **72**, **74** as seen in step **106** and illustrated in Figures 2 and 3. On the other hand, if no user entered separators are detected, or if an insufficient number of organic separators are included as determined by decision block **102**, then domain separators **30** will be inserted at determined locations based on information from, or data stored with the user (e.g. in cookies, or other predetermined protocol), assumptions made based on the particular protocol in use, or between determined numbers of digits or the like, as illustrated by step **108**.

With the domain separators in place from either step **106** or **108**, a quasi-multi level domain name exists. Optionally then, the processor may affix any prefixes or suffixes needed to generate a complete multiple level domain name or as may be required based on known factors, as seen in step **110**. For instance, an originating user may omit the country code, for example, of their target or desired device. In this case, the processor will recognize the insufficiency or ambiguity resulting from too few sub-domains in the address and substitute or append the country code of the originating

user. Similar additions can be made by suffix if desired (e.g. automatic appending of a base level domain). The target device is then queried across the network at the multiple level domain name generated, as seen in block **112**. If the target device is available, it will respond in a like fashion at which point communication can be established between the source and the target devices, as seen in step **114**.

With reference now to Figure 5, a generic system **120** is shown connected to a network **122** for suitably practicing the invention. A device **124** is connected to server **120**. A processor **120a** within the server receives from the device **124** a telephone number portion **50** (Figure 2) address intended to identify a second device **128**. The processor **120a** converts the telephone number portion into a static multiple level domain name **96** (Figure 3) identifying the second device **128** through the network **122**. For illustration purposes, the device **128** is illustrated as also being connected to the network **122** via its own server, however numerous other interconnections are envisioned and apply with equal facility. The processor **120a** determines the presence or lack of organic separators and, as discussed above, generates a complete multiple level domain name **96** suitable to identify device **128**. The completed multiple level domain name **96** is then optionally converted in memory **120b** to a standard IP address in this example. The target device **128** is then queried at the converted IP address location, and if available, responds to the query. The details of packet switching and call setup through the network along with path selection through the network are known to those in the art and proceed here conventionally.

The invention has been described with reference to the preferred embodiment. Obviously, modifications and alterations will occur to others upon the reading and understanding of the specification. It is our intention to include all such modifications and alterations insofar as they come within the scope of the appended claims or the equivalents thereof.

Having thus described the preferred embodiments, the invention is now claimed to be:

1. A method comprising:

receiving a telephone number portion identifying a device;

converting the telephone number portion into a multiple level domain name identifying the device over a network, the multiple level domain name comprising the telephone number portion and a base portion; and

establishing communication with the device via the multiple level domain name over the network.

2. The method as set forth in claim 1, where the telephone number portion of the multiple level domain name is subordinated to the base portion.

3. The method as set forth in claim 2, where the base portion of the multiple level domain name comprises a base level domain.

4. The method as set forth in claim 1, where the converting step comprises:

adding domain separators to the received telephone number portion at determinable locations in the received telephone number portion.

5. The method as set forth in claim 1, where the received telephone number portion comprises a separator, the converting step comprising:

parsing the received telephone number portion for the separator; and

inserting a domain separator for the parsed separator.

6. The method as set forth in claim 1, further comprising:

appending additional domain levels to the converted telephone number portion to complete the multiple level domain name.

7. A method of communicating over a network comprising:

receiving from a first device at least a portion of a static, multiple level domain name including a telephone number portion identifying a second device;

determining availability of the second device on the network; and

in response to the determining step, selectively establishing communications from the first device to the second device.

8. The method as set forth in claim 7, further comprising:

establishing communications from the second device to the first device.

9. The method as set forth in claim 7, where the determining availability step comprises:

querying the second device over the network; and

receiving a response from the second device indicative of second device availability.

10. An apparatus to establish communication between at least two devices over a network, the apparatus comprising a processor which receives from a first device a telephone number portion identifying a second device, and which converts the telephone number portion into a static multiple level domain name identifying the second device on the network.

11. The apparatus as set forth in claim 10, where the processor further establishes communication with the second device over the network.

12. The apparatus as set forth in claim 10, further comprising a table which matches the static multiple level domain name to an IP address.

13. The apparatus as set forth in claim 10, where the processor further adds domain separators to the received telephone number portion at determinable locations to result in the static multiple level domain name.

14. The apparatus as set forth in claim 10, where the received telephone number portion comprises a separator, and where the processor parses the received telephone number

portion for the separator and inserts a domain separator for selected instances of the parsed separator.