

IETF IDN Working Group
Internet Draft
Document: [draft-ietf-idn-vidn-01.txt](#)
Expires: 2 September 2001

Sung Jae Shim
DualName, Inc.
2 March 2001

Virtually Internationalized Domain Names (VIDN)

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of [Section 10 of RFC2026](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/1id-abstracts.txt>

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

1. Abstract

This document proposes a method that enables domain names to be used in both local and English scripts, as a directory-search solution at an upper layer above the DNS. The method first converts virtual domain names typed in local scripts into the corresponding domain names in English scripts that comply with the DNS, using the knowledge of transliteration between local and English scripts. Then, the method searches for and displays domain names in English scripts that are active on the Internet so that the user can choose any of them.

The conversion takes place automatically and transparently in the user's applications before DNS queries are sent, and so, the method does not make any change to the DNS nor require separate name servers.

2. Conventions and definitions used in this document

The key words "REQUIRED" and "MAY" in this document are to be interpreted as described in [RFC-2119](#) [1].

A "host" is a computer or device attached to the Internet. A "user host" is a computer or device with which a user is connected to the Internet, and a "user" is a person who uses a user host. A "server host" is a computer or device that provides services to user hosts.

An "entity" is an organization or individual that has a domain name registered with the DNS.

A "local language" is a language other than English language that a user prefers to use in a local context. "Local scripts" are scripts of a local language and "English scripts" are scripts of English language.

A "virtual domain name" is a domain name in local scripts, and it is not registered with the DNS but used for the convenience of users. An "English domain name" is a domain name in English scripts. A "domain name" refers to an English domain name that complies with the DNS, unless specified otherwise.

A "coded portion" is a pre-coded portion of a domain name (e.g., generic codes including 'com', 'edu', 'gov', 'int', 'mil', 'net', 'org', and country codes such as 'kr', 'jp', 'cn', and so on). An "entity-defined portion" is a portion of a domain name, which is defined by the entity that holds the domain name (e.g., host name, organization name, server name, and so on).

The method proposed in this document is called "virtually internationalized domain names (VIDN)," as it enables domain names in English scripts to be used virtually in local scripts.

A number of Korean-language characters are used in the original of this document for examples, which is available from the author upon request. The software used for Internet-Drafts does not allow using multilingual characters other than ASCII characters. Thus, this document may not display Korean-language characters properly, although it may be comprehensible without the examples using Korean-language characters. Also, when you open the original of this document, please select your view encoding type to Korean for Korean-language characters to be displayed properly.

3. Introduction

Domain names are valuable to Internet users as a main identifier of entities and resources on the Internet. The DNS allows using only English scripts in naming hosts or clusters of hosts on the Internet. More specifically, the DNS uses only the basic Latin alphabets (case-insensitive), the decimal digits (0-9) and the hyphen (-) in domain names. But there is a growing need for internationalized domain names in local scripts. Recognizing this need, various methods have been

proposed to use local scripts in domain names. But to date, no method appears to meet all the requirements of internationalized domain names as described in Wenzel and Seng [2].

A group of earlier methods tries to put internationalized domain names in local scripts inside some parts of the overall DNS, using special encoding schemes of Universal Character Set (UCS). But these methods put too much of a burden on the DNS, requiring a great deal of work for transition and update of the DNS components and the applications working with the DNS. Another group of earlier methods tries to build separate directory services for internationalized domain names or keywords in local scripts. But these methods also require complex implementation efforts, duplicating much of the work already done for the DNS. Both the groups of earlier methods require creating internationalized domain names or keywords in local scripts from scratch, which is a costly and lengthy process on the parts of the DNS and Internet users. Further, domain names or keywords created in local scripts are usable only by those who know the local scripts, and so, they may segregate the Internet into many groups of different sets of local scripts that are less universal than English scripts.

VIDN intends to provide a more immediate and less costly solution to internationalized domain names than earlier methods. VIDN does not make any change to the DNS nor require creating additional domain names in local scripts.

VIDN takes notice of the fact that many domain names currently used in regions where English scripts are not widely used have their entity-defined portions consisting of English scripts as transliterated from the respective local scripts. Using this knowledge of transliteration between local and English scripts, VIDN converts virtual domain names typed in local scripts into the corresponding domain names in English scripts that comply with the DNS. In this way, VIDN enables the same domain names to be used not only in English scripts as usual but also in local scripts, without creating additional domain names in local scripts.

4. VIDN method

4.1. Objectives

Earlier methods of internationalized domain names try to create domain names or keywords in local scripts one way or another in addition to existing domain names in English scripts, and put them inside or outside the DNS, using special encoding schemes or lookup services. These methods require a lengthy and costly process of creating domain names in local scripts and updating the DNS components and applications. Even when they are successfully implemented, these methods have a risk of localizing the Internet by segregating it into groups of different sets of local scripts that are less universal than English scripts and so diminishing the international scope of the Internet. Further, these methods may cause more problems and disputes on copyrights, trademarks, and so on, in

local contexts than those that we experience with current domain names in English scripts.

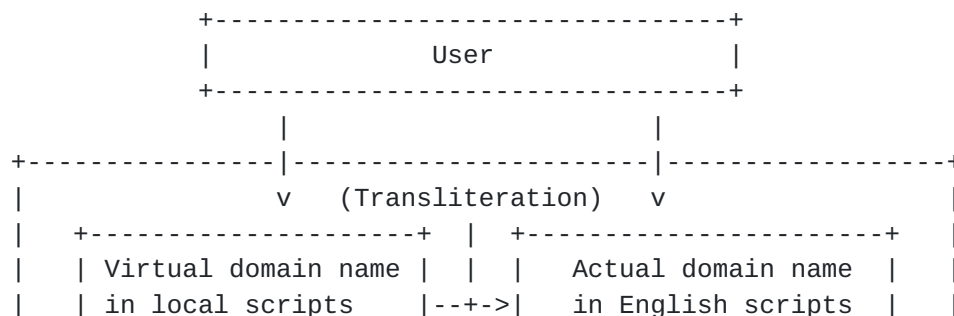
VIDN intends to provide a solution to the problems of earlier methods of internationalized domain names. VIDN enables the same domain names to be used in both English scripts as usual and local scripts, and so, there is no need to create domain names in local scripts in addition to domain names in English scripts. VIDN works automatically and transparently in applications at user hosts before DNS requests are sent, and so, there is no need to make any change to the DNS or to have additional name servers. For these reasons as well as others, VIDN can be implemented more immediately with less cost than other methods of internationalized domain names.

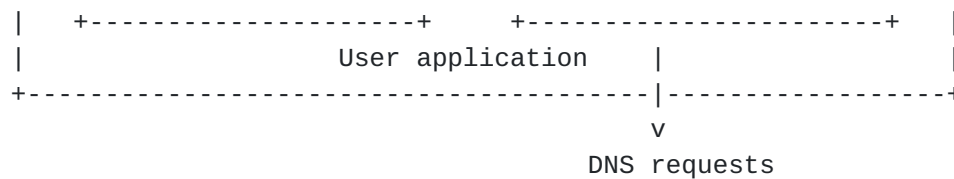
4.2. Description

It is important to note that most domain names used in regions where English scripts are not widely used have their entity-defined portions consisting of English scripts as transliterated from local scripts. Of course, there are many domain names in those regions that do not follow this kind of transliteration between local and English scripts. In such case, new domain names in English scripts need to be created following this transliteration, but the number would be minimal, compared to the number of internationalized domain names in local scripts to be created and registered under other methods.

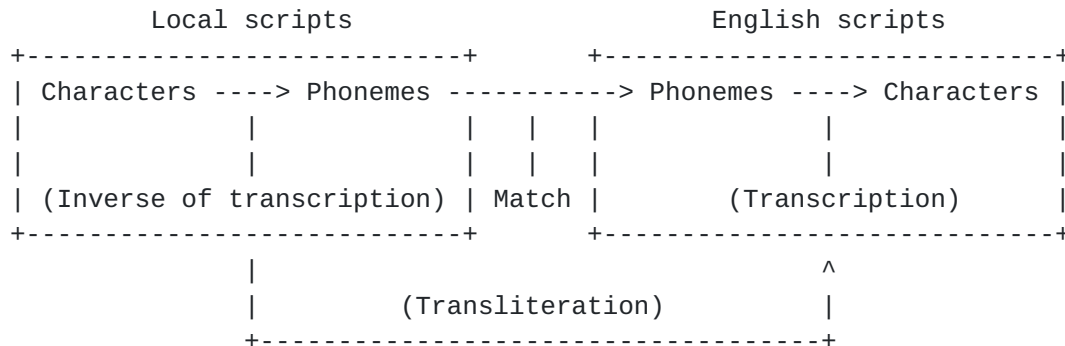
The English scripts transliterated from local scripts do not have any meanings in English language, but their originals in local scripts before the transliteration have some meanings in the respective local language, usually indicating organization names, brand names, trademarks, and so on. VIDN enables to use these original local scripts as the entity-defined portions of virtual domain names in local scripts, by transliterating them into the corresponding entity-defined portions of actual domain names in English scripts. In this way, VIDN enables the same domain names in English scripts to be used virtually in local scripts without actually creating domain names in local scripts.

As domain names in English scripts overlay IP addresses, so virtual domain names in local scripts do actual domain names in English scripts. The relationship between virtual domain names in local scripts and actual domain names in English scripts can be depicted as:





VIDN uses the phonemes of local and English scripts as a medium in transliterating the entity-defined portions of virtual domain names in local scripts into those of actual domain names in English scripts. This process of transliteration can be depicted as:



First, each entity-defined portion of a virtual domain name typed in local scripts is decomposed into individual characters or sets of characters so that each individual character or set of characters can represent an individual phoneme of the local language. This is the inverse of transcription of phonemes into characters. Second, each individual phoneme of the local language is matched with an equivalent phoneme of English language that has the same or most proximate sound. Third, each phoneme of English language is transcribed into the corresponding character or set of characters in English language. Finally, all the characters or sets of characters converted into English scripts are united to compose the corresponding entity-defined portion of an actual domain name in English scripts.

For example, a word in Korean language, '세기' that means 'century' in English language, is transliterated into 'segi' in English scripts, and so, the entity whose name contains '세기' in Korean language may have an entity-defined portion of its domain name as 'segi' in English scripts. VIDN enables to use '세기' as an entity-defined portion of a virtual domain name in Korean scripts, which is converted into 'segi,' the corresponding entity-defined portion of an actual domain name in English scripts. In other words, the phonemes represented by the characters consisting of '세기' in Korean scripts have the same sounds as the phonemes represented by the characters consisting of 'segi' in English scripts. In the local context, '세기' in Korean scripts is clearly easier to remember and type and more intuitive and meaningful than 'segi' in English scripts.

An entity-defined portion of a virtual domain name in Korean scripts, '야ahoo', is transliterated into 'yahoo' in English scripts, since the phonemes represented

by the characters consisting of 'ㅈㅇ' in Korean scripts have the same sounds as the phonemes represented by the characters consisting of 'yahoo' in English scripts. That is, 'ㅈㅇ' in Korean scripts is pronounced as the same as 'yahoo' in English scripts, and so, it is easy for Korean-speaking people to deduce 'ㅈㅇ' in Korean scripts as the virtual equivalent of 'yahoo' in English scripts. VIDN enables to use virtual domain names in local scripts for domain names whose originals are in local scripts, e.g., ' ' in Korean scripts, as well as domain names whose originals are in English scripts, e.g., 'ㅈㅇ' in Korean scripts. In this way, VIDN is able to make domain names truly international, allowing the same domain names to be used both in English and local scripts.

The coded portions of domain names such as generic codes and country codes can also be transliterated from local scripts into English scripts, using their phonemes as a medium. For example, seven generic codes in English scripts, 'com', 'edu', 'gov', 'int', 'mil', 'net', and 'org', can be transliterated from 'ㅇㅇ', 'Àí', ' ' | ', 'ÁðÈ«', ' Ĩ', 'þÚÈ«', 'ÀÁ Ú' in Korean scripts, respectively, which can be used as the corresponding generic codes of virtual domain names in Korean scripts. Based upon its meaning in English language, each coded portion of actual domain names also can be pre-assigned a virtual equivalent word or code in local scripts. For example, seven generic codes in English scripts, 'com', 'edu', 'gov', 'int', 'mil', 'net', and 'org', can be pre-assigned ' ㅈ ' (meaning 'commercial' in Korean language), 'Ĭ Ĭ þ' (meaning 'education' in Korean language), 'Âñ|ð' (meaning 'government' in Korean language), ' Âª' (meaning 'international' in Korean language), ' |À ' (meaning 'military' in Korean language), 'þÚÈ«' (meaning 'network' in Korean language), and '³ È' (meaning 'organization' in Korean language), respectively, which can be used as the corresponding generic codes of virtual domain names in Korean scripts.

VIDN does not create such complexities as other conversion methods based upon semantics do, since it uses phonemes as a medium of transliteration between local and English scripts. Further, most languages have a small number of phonemes. For example, Korean language has nineteen consonant phonemes and twenty-one vowel phonemes, and English language has twenty-four consonant phonemes and twenty vowel phonemes. Each phoneme of Korean language can be matched with a phoneme of English language that has the same or proximate sound, and vice versa.

Some characters or sets of characters may represent more than one phoneme. Some phonemes may be represented by more than one character or set of characters. Also, not every character or set of characters in local scripts may be neatly transliterated into only one character or set of characters in English scripts. In practice, people often transliterate the same local scripts differently into English scripts or vice versa. VIDN incorporates the provisions to deal with those variations that usually occur in particular situations as well as those variations that are caused by common usage or idiomatic expressions. More

fundamentally, VIDN uses phonemes, which are very universal across different languages, as a medium of transliteration rather than following a certain set of transliteration rules that does not exist in many non-English-speaking countries nor is followed by many non-English-speaking people.

One virtual domain name typed in local scripts may be converted into more than one possible domain name in English scripts. In such case, VIDN can search for and displays only those domain names in English scripts that are active on the Internet, so that the user can choose any of them. Further, VIDN can be used as a directory-search solution at an upper layer above the DNS. That is, the user can use VIDN to query a phoneme-based domain name request in local scripts, receive one or more corresponding domain names in English or ASCII-compatible scripts preferably, choose one based upon the results of that search, and make the final DNS request using any protocol or method to be chosen for internationalized domain names. In this regard of directory search, VIDN uses one-to-many map between virtual domain names in local scripts and actual domain names in English scripts.

VIDN needs the one-to-many mapping and subsequent multiple DNS lookups only at the first query of each virtual domain name typed in local scripts at the user host. After the first query, the virtual domain name is set to the domain name in English scripts that has been chosen at the first query. Any subsequent queries with the same virtual domain name generate only one query with the selected domain name in English scripts. Once the user selects one possible domain name in English scripts from the list, VIDN remembers the user's selection and directs the user to the same domain name at his or her subsequent queries with that virtual domain name. In this way, VIDN can generate less traffic on the DNS, while providing faster, easier, and simpler navigation on the Internet to the user, using local scripts.

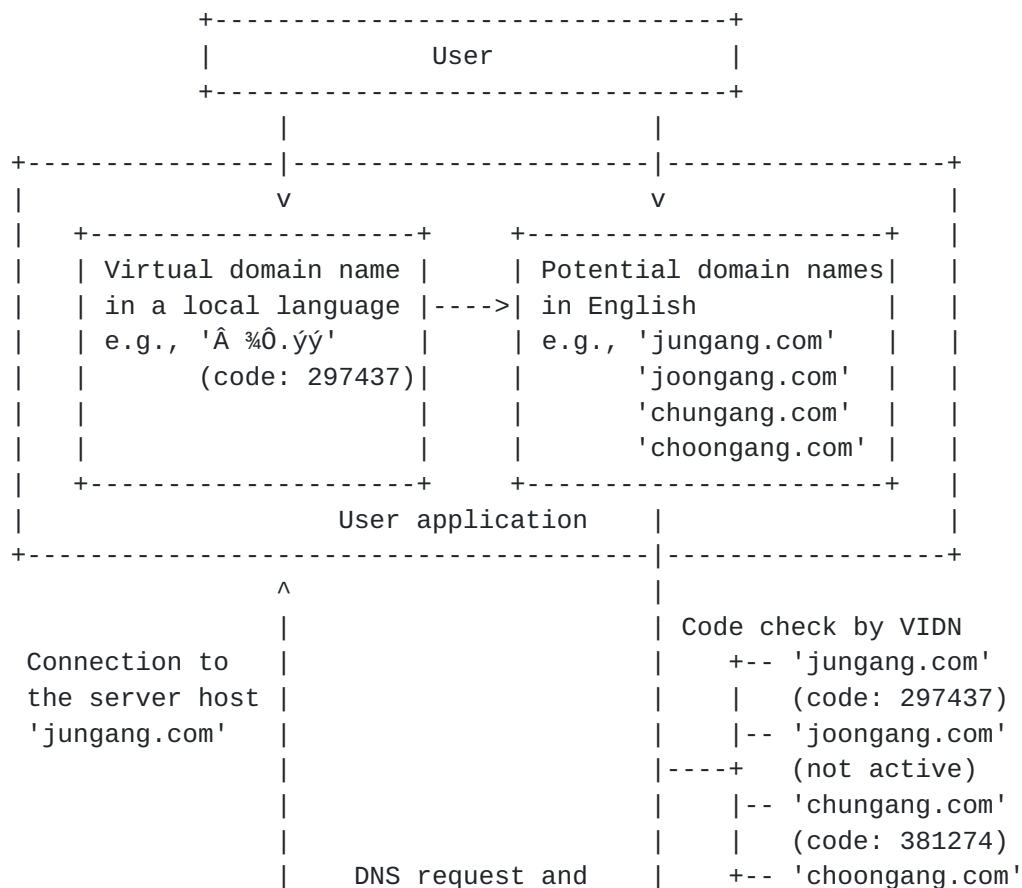
Utilizing a coding scheme, VIDN is also capable of making each virtual domain name typed in local scripts correspond to exactly one actual domain name in English scripts. In this coding scheme, a unique code such as the Unicode or hexadecimal code represented by the virtual domain name, is pre-assigned to one of the corresponding domain names in English scripts and stored in the respective server host, so that both the user host and the server host can support and understand the code. Then, VIDN checks whether the code at each server host matches with the code generated at the user host. If one of the servers stores the code that matches with the code generated at the user host, the virtual domain name typed at the user host is recognized as corresponding only to the domain name of that server host, and the user host is connected to the server host. The domain names of the remaining server hosts that do not have the matching code are also displayed at the user host as alternative sites.

Because a unique code is assigned to only one of the domain names in English scripts, it does not cause any domain name squatting problem beyond what we experience with current domain names in English scripts. Unique codes do not need to be stored in any specific format, that is, they can be embedded in

HTML, XML, WML, and so on, so that the user host can interpret the retrieved code correctly. Likewise, unique codes do not require any specific intermediate transport protocol such as TCP/IP. The only requirement is that the protocol must be understood among all participating user hosts and server hosts. For security purpose, this coding scheme may use an encryption technique.

For example, 'Â ¤Ï.ýý', a virtual domain name typed in Korean scripts, may result in four corresponding domain names in English scripts, including 'jungang.com', 'joongang.com,' 'chungang.com', and 'choongang.com', since the phonemes represented by characters consisting of 'Â ¤Ï.ýý' in Korean scripts can have the same or almost the same sounds as the phonemes represented by characters consisting of 'jungang.com', 'joongang.com,' 'chungang.com', or 'choongang.com' in English scripts. In this case, we assume that the server host with its domain name 'jungang.com' has the pre-assigned code that matches with the code generated when 'Â ¤Ï.ýý' in Korean scripts is entered in user applications. Then, the user host is connected to this server host, and the other server hosts may be listed to the user as alternative sites so that the user can try them.

The process of this coding scheme that makes each virtual domain name in local scripts correspond to only one actual domain name in English scripts, can be depicted as:



	response		(not active)
+-----+			

Since VIDN converts separately the entity-defined portions and the coded portions of a virtual domain name, it preserves the current syntax of domain names, that is, the hierarchical dotted notation, which Internet users are familiar with. Also, VIDN allows using a virtual domain name mixed with local and English scripts as the user wishes to, since the conversion takes place on each individual portion of the domain name and each individual character or set of characters of the portion.

While VIDN preserves the hierarchical dotted notation of current domain names, the principles of VIDN are applicable to domain names in other possible notations such as those in a natural language (e.g., 'microsoft windows' rather than 'windows.microsoft.com'). Also, the principles of VIDN can be applied into other identifiers used on the Internet, such as user IDs of e-mail addresses, names of directories and folders, names of web pages and files, keywords used in search engines and directory services, and so on, allowing them to be used interchangeably in local and English scripts, without creating additional identifiers in local scripts. The conversion of VIDN can be done between any two sets of scripts interchangeably. Thus, even when the DNS accepts and registers domain names in other local scripts in addition to English, VIDN can allow using the same domain names in any two sets of scripts by converting virtual domain names in one set of scripts into actual domain names in another set of scripts.

4.3. Development and implementation

In a preferred arrangement, the development of VIDN for each set of local scripts may be administered by one or more local standard bodies in regions where the local scripts are widely used, for example, Korean Network Information Center for Korean scripts, Japan Network Information Center for Japanese scripts, and China, Hong Kong and Taiwan Network Information Centers for Chinese scripts, with consultation with experts on phonemics and linguistics of the respective local language and English language. Also, the unique codes for one-to-one mapping between virtual domain names in local scripts and actual domain names in English scripts can be administered by a central standard body like IANA. Alternatively, the unique codes for each set of local scripts may be administered by one or more local standard bodies in regions where the local scripts are widely used, as with the development of VIDN.

VIDN is implemented in applications at the user host. That is, the conversion of virtual domain names in local scripts into the corresponding actual domain names

in English scripts takes place at the user host before DNS requests are sent. Thus, neither a special encoding nor a separate lookup service is needed to implement VIDN. VIDN is also modularized with each module being used for conversion of virtual domain names in one set of local scripts into the corresponding actual domain names in English scripts. A user needs only the module for conversion of his or her preferred set of local scripts into English scripts. Alternatively, VIDN can be implemented at a central server host or a cluster of local server hosts. A central server can provide the conversion service for all sets of local scripts, or a cluster of local server hosts can share the conversion service. In the latter case, each local server host can provide the conversion service for one or more sets of local scripts used in a certain region.

Because of its small size, VIDN can be easily embedded into applications software such as web browser, e-mail software, ftp system, and so on at the user host, or it can work as an add-on program to such software. In either case, the only requirement on the part of the user is to install VIDN or software embedding VIDN at the user host. Using virtual domain names in local scripts in accordance with the principles of VIDN is very intuitive to those who use the local scripts. The only requirement on the part of the entity whose server host provides Internet services to user hosts is to have an actual domain name in English scripts into which virtual domain names in local scripts are neatly transliterated in accordance with the principles of VIDN. Most entities in regions where English scripts are not widely used already have such domain names in English scripts. Finally, there is nothing to change on the part of the DNS, since VIDN uses the current DNS as it is.

Taken together, the features of VIDN can meet all the requirement of internationalized domain names as described in Wenzel and Seng [2], with respect to compatibility and interoperability, internationalization, canonicalization, and operating issues. Given the fact that different methods toward internationalized domain names confuse users, as already observed in some regions where some of these methods have already been commercialized, e.g., Korea, Japan and China, it is important to find and implement the most effective solution to internationalized domain names as soon as possible.

4.4. Current status

VIDN has been developed for Korean-English conversion as a web browser add-on program. The program contains all the features described in this document and is capable of listing all the domain names in English scripts that correspond to a virtual domain name typed in Korean scripts so that a user can choose any of them. The program can cover more than ninety percent of the sample. That is, the results of testing indicate that more than ninety percent of web sites in Korea can be accessed using virtual domain names in Korean scripts without creating

additional domain names in Korean scripts. The remaining ten percent of domain names are mostly those that contain acronyms, abbreviations or initials. With improvement of its knowledge of transliteration, the program is expected to cover more domain names used in Korea.

5. Security considerations

Because VIDN uses the DNS as it is, it inherits the same security considerations as the DNS.

6. Intellectual property considerations

It is the intention of DualName, Inc. to submit the VIDN method and other elements of VIDN software to IETF for review, comment or standardization.

DualName has applied for one or more patents on the technology related to virtual domain name software and virtual email software. If a standard is adopted by IETF and any patents are issued to DualName with claims that are necessary for practicing the standard, DualName is prepared to make available, upon written request, a non-exclusive license under fair, reasonable and non-discriminatory terms and condition, based on the principle of reciprocity, consistent with established practice.

7. References

1 Wenzel, Z. and Seng, J. (Editors), "Requirements of Internationalized Domain Names," [draft-ietf-idn-requirements-03.txt](#), August 2000

8. Author's address

Sung Jae Shim
DualName, Inc.
3600 Wilshire Boulevard, Suite 1814
Los Angeles, California 90010
USA
Email: shimsungjae@dualname.com