

ICNRG
Internet Draft
Intended status: Informational
Expires: April 12, 2019

Hongke Zhang
Fei Song
Wei Quan
BJTU
Jianfeng Guan
Changqiao Xu
BUPT
October 5, 2018

**Uniform information with a hybrid naming (hn) scheme
draft-zhang-icnrg-hn-09.txt**

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

This document may contain material from IETF Documents or IETF Contributions published or made publicly available before November 10, 2008. The person(s) controlling the copyright in some of this material may not have granted the IETF Trust the right to allow modifications of such material outside the IETF Standards Process. Without obtaining an adequate license from the person(s) controlling the copyright in such materials, this document may not be modified outside the IETF Standards Process, and derivative works of it may not be created outside the IETF Standards Process, except to format it for publication as an RFC or to translate it into languages other than English.

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>

This Internet-Draft will expire on April 12, 2019.

Copyright Notice

Copyright (c) 2017 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document.

Abstract

This document defines a hybrid naming scheme for unifying all kinds of information including resources, services and data. With many proposals of novel network architectures emerging, such as DONA, ICN, NDN, the location-based routing starts to transfer to the content based ones. Currently, it is incompatible that many different information naming schemes are adopted in different network proposals, respectively, i.e. flat names in DONA, hierarchical names in NDN. The proposed naming scheme adopts a hybrid naming structure, which includes hierarchical component, flat component and attribute component. The hybrid naming (hn) scheme enables to identify different routing information uniformly, and provides many great advantages, such as high aggregation, limited length, suffix holes remission, fuzzy matching support, high security and good compatibility with IPv4/IPv6, DONA, CCN/NDN and so on.

Table of Contents

1.	Introduction	3
1.1.	Hierarchical naming.....	3
1.2.	Flat naming	4
1.3.	Attribute naming.....	4
2.	Conventions used in this document.....	4
3.	Novel hybrid naming (hn) format.....	5
3.1.	Hierarchical component generating	6
3.2.	Flat component generating.....	6
3.3.	Attribute component generating	7
4.	Advantages	7
4.1.	High aggregation.....	7
4.2.	Limited length	8
4.3.	Suffix holes remission.....	8

4.4. Fuzzy matching support.....	9
4.5. Good compatibility.....	10
4.6. High security	10
5. Transition form IPv4 and IPv6.....	10
5.1. Case one	10
5.2. Case two	11
6. Compatibility	11
6.1. Compatibility with DONA.....	11
6.2. Compatibility with CCN/NDN.....	12
7. Formal Syntax	13
8. Security Considerations.....	13
9. IANA Considerations	13
10. Conclusions	13
11. References	13
11.1. Normative References.....	13
11.2. Informative References.....	14
12. Acknowledgments	15
Authors' Addresses	15

1. Introduction

1.1. Hierarchical naming

A readable naming mechanism based on the hierarchical structure by some emerging network architectures (i.e. Content-Centric Network (CCN) [1]/Named Data Networking (NDN) [2]) has been proposed. This kind of hierarchical name is very similar as identifying a web with a URL for example "/www.bupt.edu.cn/content/a.avi". In this example, "/" is the separator between adjacent components of the name.

As we know, there are many advantages in this naming scheme. First, it is well compatible with current applications or systems based on URL, which can decrease the difficulty of deploying the novel network. Second, it does well in aggregation which can reduce the number of routing information, thus, to improve lookup efficiency of routing information. Besides, the lookup mechanism of this mechanism has a good compatibility with the existing classless inter-domain routing (CIDR) [3].

However, there also exist some fatal disadvantages in the hierarchical name. Because it consists of a series of unlimited components. The number of components is changeable, with the length of each component being not restricted. All these features cause the length of names variable and relatively long [4]. In this way, the routing table and forwarding table may be very huge, which issues in low lookup efficiency.

On the side, when users search for a resource, they might not remember the long name of the resource. For example, users need the resource a.avi, but they might not know the official name "/www.bupt.edu.cn/content/a.avi" or "/www.bupt.edu.cn/movie/a.avi". Thus, hierarchical naming structure is difficult to support a fuzzy matching based on the attributes of names.

1.2. Flat naming

The flat naming mechanism has been used in other novel network architectures, such as DONA [5] and NetInf [6], of which the name can be produced by cryptographic hashing of the content or its attributes.

Because the flat name has not any structure restriction, it can be obtained and used more flexibly. Any string with a fix length, no matter whether it is readable or not, can be used as a flat name.

However, the flat name is hard to aggregate, which will increase the number of routing entries and reduce the expandability of routing table. In addition, it increases the probability for users to forget the official names of the desired information, because most of flat names is not readable. When users want to obtain contents, it needs an additional mapping system to connect readable names with unreadable names for users.

1.3. Attribute naming

The naming mechanism based on attributes of content is used in the CBCB [7]. It enumerates the attribute information of a resource, such as the category, format, date, feature, level and so on. This name is non-uniqueness, and is different from the former two mechanisms. The related content can be searched and located by dint of the key properties of resource.

The merits of this naming mechanism are that, it supports searching key words and provides benefits for the fuzzy matching of searching resources. However, there may be many similar properties for a set of certain resources. A number of attributes is hardly ensure the uniqueness of naming. Thus, to guarantee the uniqueness, the attributes stored in routing system will be very huge.

2. Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC 2119](#) [[RFC2119](#)].

In this document, these words will appear with that interpretation only when in ALL CAPS. Lower case uses of these words are not to be interpreted as carrying significance described in [RFC 2119](#).

In this document, the characters ">>" preceding an indented line(s) indicates a statement using the key words listed above. This convention aids reviewers in quickly identifying or finding the portions of this RFC covered by these keywords.

3. Novel hybrid naming (hn) format

According to the analysis of above three naming mechanisms in the field of advantages and disadvantages, a hybrid naming is suggested to emphasize the benefits of them and weaken their demerits.

Most important of all, three different mainstream naming schemes are adopted in different novel network architectures, which makes the networks be hardly compatible and implemented complexly.

There's one easy and all-benefit solution is to integrate them, and take each of them as a part of the hybrid naming solution. In other words, each of them takes some weight of the novel naming scheme.

We proposed a hybrid naming mechanism (named by "hn"), which combines the three naming mechanisms in a sequence, and creates a more powerful and universal naming format.

The hybrid naming format should include three components:

- o Hierarchical component
- o Flat component
- o Attribute component

Each part carries different information of name in diverse formats, and is combined to an entire name. The hybrid name is started by a symbol "hn://". The order of three parts should be as follows:

1. The first part of a name is very essential for the aggregation of routing entries, which adopts a hierarchical structure. The symbol "/" is used to split the hierarchical levels in this part.

2. The second part of a name is very important to identify the content uniquely. The second part uses a flat structure and a string with a fix length through hash computing.
3. The third part of a name is used to represent the extensive information of resources. The attribute-based structure is selected to use in the third part, which is composed of a series of attribute words. An example of the hybrid name for a movie is shown in Figure 1.

```
+-----+-----+-----+
|hn://www.bjtu.edu.cn/m|u584rnfiur324yh|movie:avi:1024:part1:kongfu|
+-----+-----+-----+
```

Figure 1 An example of hn for a movie

An example of the hybrid name for a picture is shown in Figure 2.

```
+-----+-----+-----+
|hn://www.bjtu.edu.cn/m|pic|fh84rnfiur324ru| jpg:300*500:prairie |
+-----+-----+-----+
```

Figure 2 An example of hn for a picture

3.1. Hierarchical component generating

Hierarchical component is the first part of the hn naming format. This part is supposed to be generated by a followed reference standard.

The string set in top level and second level and so on are defined by this standard, which is very useful to promote its aggregation greatly. One available but not complete reference standard for component of hierarchical naming is the naming scheme of DNS.

3.2. Flat component generating

Flat component is the second part of hn naming scheme. This part is suggested to identify the information using a string with a limited length, and it must be combined with the first part to identify the information uniquely.

Flat component can be generated by cryptographic hash algorithm by the information itself or some characters of the information. Even though this part has a low probability of aggregation, it emphasizes and ensures the uniqueness of name.

3.3. Attribute component generating

Attribute component is set as the third part of hn naming scheme. This part will take it over for the fuzzy matching and some advanced search, i.e. QoS guarantee. This part will also contribute to conduct some potential advanced application based on the useful attributes. It can be generated by extracting the features of the information, such as the format, issue time, file size, catalog, location, popularity, privacy level and so on.

4. Advantages

4.1. High aggregation

The aggregation of naming is very important for the name lookup and storage. According to Google's report, the number of URLs it indexed was 26 million in 1998, which reached to one billion in 2000, and is currently 1 trillion [8]. In July 2011, these URLs could be aggregated to about 280 million domain names, among which 86 million are active.

It is a fact that there is a great aggregation for the first few levels of the hierarchical tree. Therefore, the hierarchical structure is placed in the first part of the hn. By this way, the routing entries can be reduced markedly and the aggregation of route can be improved. For example, there are two routing entries "/www.bjtu.edu.cn/m/movie/fhk562nfgjru056:kongfu:avi:1024p:part1 3" and "/www.bjtu.edu.cn/m/picture/fh84rnf213gjrru.jpg:300*500:prairie 3" which have the same forwarding port "3" and prefix "/www.bjtu.edu.cn/m". Therefore, the forwarding port and "/www.bjtu.edu.cn/m" can only be stored in routing table. Above all, it not only reduces the entries of routing table, but also reduces the length of each routing entries. An example of aggregation process is shown in Figure 3.

```
+-----+-----+-----+-----+
|hn://www.bjtu.edu.cn/m/movie|fhk562nfgjru056|kongfu 1024p part1|3 |
+-----+-----+-----+-----+

+-----+-----+-----+-----+
|hn://www.bjtu.edu.cn/m/picture| fh84rnf213gjrru |300*500 prairie|3|
+-----+-----+-----+-----+
```

```

+-----+-----+
|hn://www.bjtu.edu.cn/m| 3 |
+-----+-----+

```

Figure 3 An example of aggregation

4.2. Limited length

The length of name based on hierarchical structure is variable and relatively long, because it is formed by several parts and the number of component is changeable. Kelvin [9] has selected 6627999 URL in 78764 different domain names, and the statistics shows that the average length of URL is 76.97 bytes. In ICN, the name must be extracted to query in forwarding table or routing table. And the long name entry will lead to the lower query speed, hence, affecting the performance of routing.

The hn naming scheme uses flat naming as part of component in the name to ease this problem. A fix length flat part is embedded behind the hierarchical part. This design not only can prevent the length of names from being not too long, but also will reduce the effect of the aggregation. For example, if the average length of hierarchical part is controlled within 30 bytes, adopting a flat part with a fix length of 20 bytes, then, the whole average length will be restricted within 50 bytes. Compared to 76.97 bytes, the length is shortened by nearly 35%, which will improve the query speed of name greatly using the length dependent algorithms.

4.3. Suffix holes remission

The suffix hole is a well-known problem for the route of prefix matching. For example, a routing entry "/www.bjtu.edu.cn/movie/3" is stored in the route table for prefix matching. In fact, it is aggregated by "/www.bjtu.edu.cn/movie/a.avi/part1 3" and "/www.bjtu.edu.cn/movie/b.avi/part1 3". In this way, the forwarding packets will be forward from port 3, only if the prefix of name is "/www.bjtu.edu.cn/movie/". However, if packets with a name of "/www.bjtu.edu.cn/movie/c.avi" arrives in the router, it will also be forwarded from port 3. In fact, the network that port 3 connects only has a.avi and b.avi. This causes the so-called suffix holes [10].

In the proposed hn scheme, the problem of suffix holes can be solved by the flat part efficiently. For example, there are two resource names

"www.bjtu.edu.cn/movie/s83hho90oxn2783nde4r:kongfu:avi:1024p:part1 3" and

"www.bjtu.edu.cn/movie/8uh723k9ng556sgaesgs:love:rmvb:720p:part2:20 12-3-4 3". After route aggregation, the routing entry will become

"/www.bjtu.edu.cn/movie/ 3". The routing entry will be matched when a packet whose name is "/www.bjtu.edu.cn/movie/a932jfdjf2032942-jdd: control: avi: 1024p: part1: part2" arrives at this router.

However, it could not be forwarded from the port 3 based on hn scheme because of the incomplete prefix matching. There is a suffix list in each aggregating prefix, and the packet will be forwarded only when the requesting suffix exists in the suffix list. In hn scheme, it must assort a suffix list for each routing entries like "/www.bjtu.edu.cn/movie/ 3" to store the flat part of names. Even though the name of the new packet has been matched to the routing entries, its flat part "a932jfdjf2032942-jdd" does not exist in the suffix list "/www.bjtu.edu.cn/movie/ 3". The plat part will be used to decide whether it forwards the request packet when the prefix is matched. By this way, the problem of suffix holes can be resolved effectively. The lookup process of hn names is shown in Figure 4.

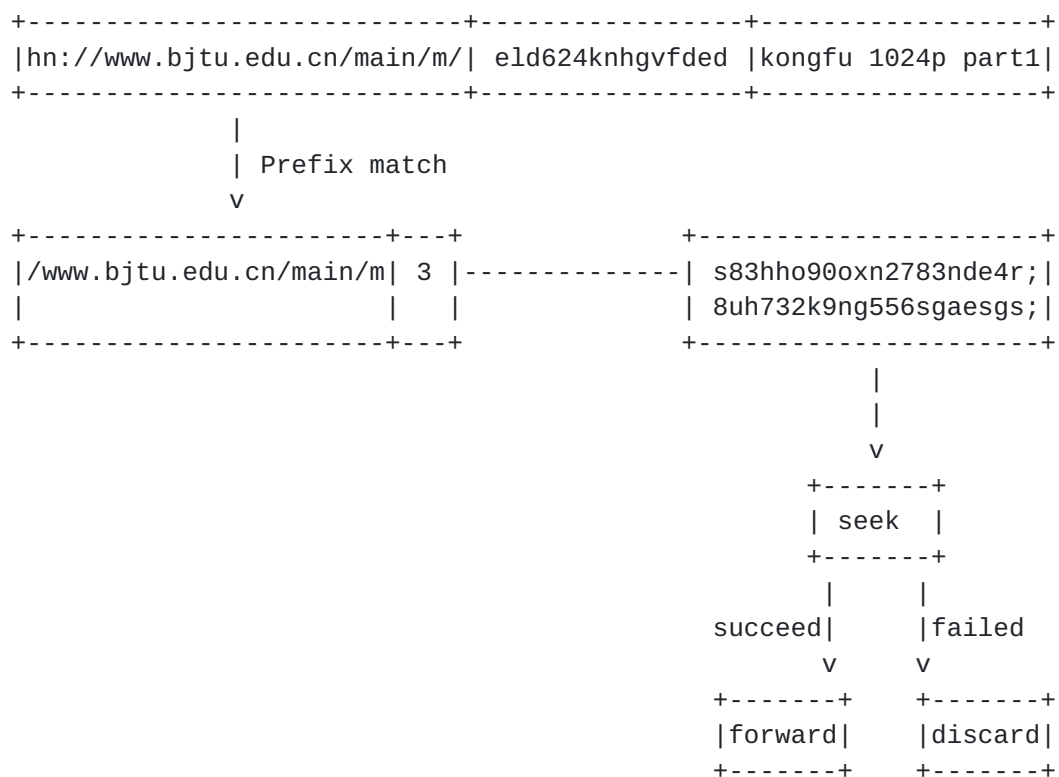


Figure 4 The hn lookup process

4.4. Fuzzy matching support

In the practical, it's an essential situation that the users may not know the full official resource name when they search for a resource. The hn naming scheme supports the fuzzy matching according to the function of the attribute component. For example, if the users need

the resource a.avi, they don't need to know the official name "hn://www.bjtu.edu.cn/m/|u584uu|89324ru|kongfu:movie:avi:1024p:part1". In this case, users only publish the information of video "kongfu" and the resolution ratio "1024p". Then the related resources can be found intelligently by fuzzy matching through the attribute component matching, which is the benefit of embedding attribute of resource in the end of name.

4.5. Good compatibility

This naming scheme provides a good compatibility for all three mainstream naming schemes, which are the subset of the hn naming scheme.

4.6. High security

It is very similar as identifying a web with a URL in the conventional hierarchical naming mechanism, for example "/www.bjtu.edu.cn/movie/a.avi". However, the name of components is variable. Although it is convenient to get information of every component of the resources, it results in bad security.

In the proposed hn scheme, this security problem can be solved. For example, one hn resource name called "/www.bjtu.edu.cn/s83hho90oxn2783nde4r: kongfu: avi: 1024p: part1 3", and another conventional name "/www.bjtu.edu.cn/movie/a.avi 3". The attacker can know every component when he/she sees the conventional name. On the contrary, the hn name does not have this problem. In the hn naming scheme, people can just know the few components of the resources, thus, the attacker could not attack the components easily. Therefore, this kind of naming scheme has a better security than hierarchical naming mechanism. Also, MD5 algorithm can be applied to the hn naming in order to encrypt the resources displayed in the flat component.

5. Transition form IPv4 and IPv6

5.1. Case one

In TCP/IP networks, IPv4 and IPv6 addresses are used to represent the resource locations. IPv4 and IPv6 addresses can also be used to fetch the desired information uniquely with association of the port information and content directory. We consider the hybrid naming scheme transiting from IPv4 and IPv6 networks.

The IPv4 or IPv6 address is the hierarchical as the first part of the hybrid name. The port number is placed as the second part of the

hybrid name. The third part of hybrid name is the content directory set. An illustration of transition from IPv4 and IPv6 is shown in Figure 5.

```
+-----+-----+-----+-----+
|hn://192.168.100.100|8080|m:picture:library:west:computer:book |3 |
+-----+-----+-----+-----+
+-----+-----+-----+-----+
|hn://2001.da8.215.a815.c492.d445.3489.ec8c|8080|m:picture:book |3 |
+-----+-----+-----+-----+
```

Figure 5 Illustration of case one

5.2. Case two

Another case of transition from URL is shown in Figure 6. For example, the url is "http://www.baidu.com:80/s?wd=icbc&rsvp_bp=0&tn=baidu&spt=3&ie=utf8", in which the symbol "?" is followed by a sequence of attributes information. The hn format is shown as following.

```
+-----+-----+-----+-----+
|hn://www.baidu.com|80/s?|wd:icbc rsvbp:0 tn:baidu spt:3 ie:utf8|3 |
+-----+-----+-----+-----+
```

Figure 6 Illustration of case two

6. Compatibility

6.1. Compatibility with DONA

Data-Oriented Network Architecture (DONA) converts the location-based routing to the content-based routing. The hybrid naming scheme is well compatible with DONA and the specific transformation process is manifested as below.

- (1)The hierarchical component is transferred into a flat id with a shorter length, which is distinct with the original flat component.
- (2)This new flat id can be produced by some relevant authorities, which is similar with the domain-name providers. Besides, this flat id enables to represent huge amounts of hierarchical names by increasing its length constantly. However, it is typically much shorter than the previous name.
- (3)According to the variable length of hierarchical components, an integer identifier is designed to identify the length of transferred component. This mechanism is similar to the partition method of subset.

- (4)The symbol "/" is used for splitting this identifier with flat component.

For example, there is a routing entry

"/www.bjtu.edu.cn/m/movie/fhk562nfgjru056:kongfu:avi:1024p:part1 3".
The first component "www.bjtu.edu.cn/m/movie" is transferred to a unique flat name "dllta", which is settled before the flat component. Meanwhile, we get an identifier "5" to illustrate that the first 5 characters represent the length of transferred hierarchical name. It is significant that the name can be restored easily by their one-to-one mapping. This transformation process is shown in Figure 7.

```
+-----+-----+-----+---+
|hn://www.bjtu.edu.cn/m/movie|fhk562nfgjru056|kongfu 1024p part1|3 |
+-----+-----+-----+---+
+-----+-----+---+
|dona://dlltafhk562nfgjru056/5|kongfu 1024p part1| 3 |
+-----+-----+---+
```

Figure 7 An example of the transformation for hierarchical name

6.2. Compatibility with CCN/NDN

A readable naming mechanism based on the hierarchical structure has been proposed in CCN/NDN. The hybrid naming scheme is also well compatible with CCN/NDN. The specific transformation process is shown as below.

- (1)The hierarchical component of hn structure will not be changed as the first unit.
- (2)The flat component is transferred to one unit followed by the first unit, and uses "/" as separation label.
- (3)The attributes component is separated as many units, which are separated by the label "/".
- (4)The transformation between the hybrid naming structure and CCN/NDN hierarchical naming structure can be easily accomplished.

For example, there is a routing entry

hn://www.bjtu.edu.cn/m/picture|fh84rnf213gjrru|300*500 prairie 3".
The components "fh84rnf213gjrru|300*500 prairie" is transferred to several unique units "id=fh84rnf213gjrru/300*500prairie". It is essential that the name can be restored easily according to their one-to-one mapping. This transformation process is shown in Figure 8.


```

+-----+-----+-----+-----+
|hn://www.bjtu.edu.cn/m/picture| fh84rnf213gjrru |300*500 prairie |3|
+-----+-----+-----+-----+
+-----+-----+-----+-----+
|ccn://www.bjtu.edu.cn/m/picture/id=fh84rnf213gjrru/300*500prairie|3|
+-----+-----+-----+-----+

```

Figure 8 An example of the transformation for flat name

7. Formal Syntax

The following syntax specification uses the augmented Backus-Naur Form (BNF) as described in [RFC 5234](#) [[RFC5234](#)].

8. Security Considerations

The proposed hn naming scheme has potential benefits for the security. The hierarchical prefix has a high aggregation, which can avoid the security issues of rapid expansion in routing or forwarding table, such as DoS attack. The users' privacy and the content secrets can be protected by the flat component from readable names. The attributes component can improve the management for the secure contents by using some encryption key.

9. IANA Considerations

This document presents no IANA considerations.

10. Conclusions

This document defines a novel hybrid naming scheme for unifying all kinds of information (including resources, services and data). This hybrid naming scheme owns many advantages, which can provide a better compatibility for existing naming schemes.

11. References

11.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), DOI 10.17487/RFC2119, March 1997, <<http://www.rfc-editor.org/info/rfc2119>>.

11.2. Informative References

- [1] Jacobson, V., Smetters, D., Thornton, J., et al. "Networking named content", Proceedings of the 5th international conference on Emerging networking experiments and technologies. ACM 2009 pp. 1-12.
- [2] Zhang, L., Estrin, D., Jacobson V., et al., "Named Data Networking (NDN) project," Technical Report, NDN-0001, 2010.
- [3] Yu, J., Varadhan, K., Li, T., et al, "Classless inter-domain routing (CIDR): an address assignment and aggregation strategy", [RFC 4632](#), September 1993.
- [4] Ding, S., Chen, Z. and Liu, Z., "Parallelizing FIB Lookup in Content Centric Networking", Networking and Distributed Computing (ICNDC), 2012 Third International Conference on. IEEE, 2012 pp. 6-10.
- [5] Koponen, T., Chawla, M., Chun, B., et al, "A data-oriented (and beyond) network architecture", ACM SIGCOMM Computer Communication Review. ACM, 2007 pp. 181-192.
- [6] Dannewitz, C., "NetInf: An Information-Centric Design for the Future Internet," Proc. 3rd GI/ITGKuVS Workshop on The Future Internet, Munich, Germany, May 2009.
- [7] Carzaniga, A., Rutherford, M. and Wolf, A., "A routing scheme for content-based networking", INFOCOM 2004. Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies. IEEE, 2004 pp. 918-928.
- [8] <https://observatorio.iti.upv.es/resources/new/542>
- [9] <http://www.supermind.org/blog/740/average-length-of-a-url-part-2>
- [10] Perino D. and Varvello M., "A reality check for content centric networking", in Proc. ACM SIGCOMM workshop on Information centric networking, 2011 pp. 44-49.

12. Acknowledgments

Meng Zhang and Liang Zhu contributed to discussion and revision of this document whilst working at Beijing University of Posts and Telecommunications, Beijing, China.

This document was prepared using 2-Word-v2.0.template.dot.

Authors' Addresses

Hongke Zhang
Beijing Jiaotong University (BJTU)
Beijing, 100044, P.R.China

Email: hkzhang@bjtu.edu.cn

Fei Song
Beijing Jiaotong University (BJTU)
Beijing, 100044, P.R.China

Email: fsong@bjtu.edu.cn

Wei Quan
Beijing Jiaotong University (BJTU)
Beijing, 100044, P.R.China

Email: weiquan@bjtu.edu.cn

Jianfeng Guan
Beijing University of Posts and Telecommunications (BUPT)
Beijing, 100876, P.R.China

Email: jfguan@bupt.edu.cn

Changqiao Xu

Beijing University of Posts and Telecommunications (BUPT)

Beijing, 100876, P.R.China

Email: [cxqu@bupt.edu.cn](mailto:cqxu@bupt.edu.cn)