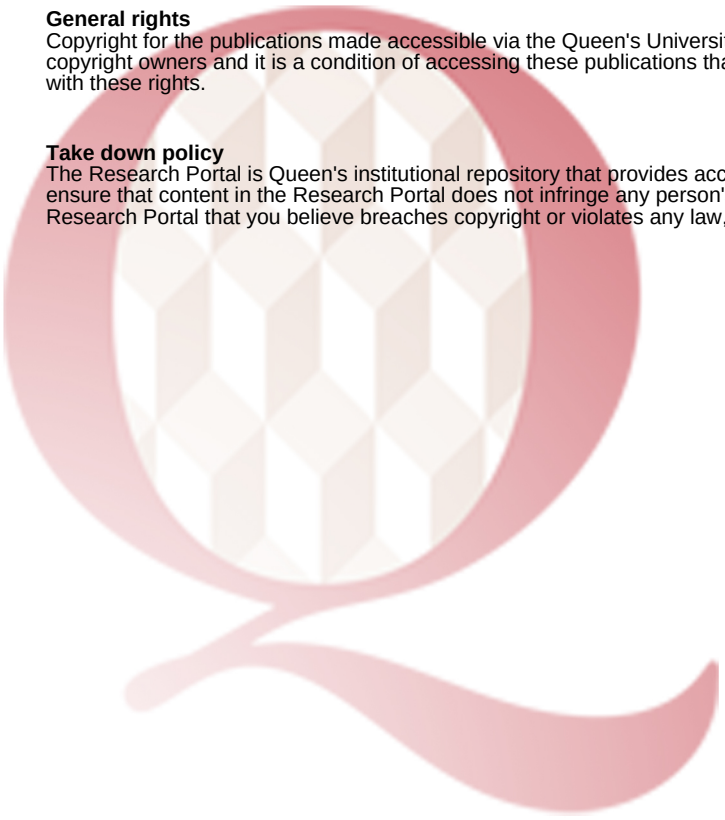
Implementation of Content Centric Networks Through Domain Name System

Khan, S. U., Khan, R., & Ali, A. (2015). Implementation of Content Centric Networks Through Domain Name System. In 2015 International Conference on Emerging Technologies (ICET): Proceedings.

Published in:
2015 International Conference on Emerging Technologies (ICET): Proceedings

Document Version:
Peer reviewed version



Implementation of Content Centric Networks Through Domain Name System

*Sarmad Ullah Khan, †Rafiullah Khan, *Arif Ali
*Electrical Engineering Department, CECOS University Hayatabad, Peshawar, Pakistan
†Electrical Engineering Department, Queen's University Belfast, Ireland, UK
Email: sarmad@cecos.edu.pk, rafiullah.khan@qub.ac.uk, engr.arif110@gmail.com

***Abstract*—Content Centric Network (CCN) is a proposed future internet architecture that is based on the concept of contents name instead of the hosts name followed in the traditional internet architecture. CCN architecture might do changes in the existing internet architecture or might replace it completely. In this paper, we present modifications to the existing Domain Name System (DNS) based on the CCN architecture requirements without changing the existing routing architecture. Hence the proposed solution achieves the benefits of both CCN and existing network infrastructure (i.e. content based routing, independent of host location, caching and content delivery protocols).**

I. Introduction

The basic purpose of internet is the distribution and sharing of contents among the users across the whole world. However, contents generation by users grow exponentially that forces the researchers to proposed a content based networking architecture to cope with this tremendous increase in contents generation and sharing. To this aim, [1], [2], [3], [4] have proposed a content centric networking architecture where the contents are addressed directly by their names instead of addressing hosts. This shift in architecture can provide many benefits in future but the replacement of TCP/IP architecture is a big challenge. However, major issues in the deployment of CCN architecture are contents addressing, caching, routing and security.

Although significant changes and improvements have been done to improve the internet architecture over the last decade for content distribution over the internet, however content centric networking architecture suits well to alleviate some of the problems associated with content distribution and dissemination. But CCN is evolved in a very disjoint and ad hoc manner.

Domain Name System (DNS) is a mapping of domain names to the host address based on its location information. It is considered as a backbone of the current internet architecture. Hence the DNS is considered to be developed and deploy for content centric networks architecture without changing the current routing infrastructure. However, without changing the current DNS properties and making it to be adopted in CCN architecture without any impact on routing infrastructure and difficulty, given that intermediate DNS servers must forward records queries and response even if they don not recognize the record type, is a big challenge.

978-1-5090-0436-2/15/$31.00 2015 IEEE

In this paper, we describe that how the CCN flavor is achieve in current internet by making DNS to store content names along with host names and deliver contents using content-delivery protocol of CCN or develop a new protocol for content delivery for said modifications. Extending the DNS and accompanying protocols in this manner takes full advantage of the existing internet routing infrastructure and core DNS servers, and can be easily deployed from the edge in an "opt-in" fashion.

The rest of the paper is organized as follow: section II describes the literature review of CCN. Section III describes the proposed changes to the DNS while section IV describes the proposed changes to content delivery protocols. Section V discuss how the CCn objectives have been achieved. Section VI provides the analysis and evaluation of proposed changes to DNS while section VII concludes the paper.

II. Related Work

Various CCN proposals for content dissemination have one common objective i.e. follow a unique and persistent naming of contents, efficient distribution of content, self authenticated, secure retrieval of contents, supporting host mobility, disruption and multi-homing. However, to achieve these objectives, CCN proposals uses the same routing algorithms based on the content content names, discovering their locations, caching at intermediate routers and nearest replica routing [5], [6].

These common characteristics of CCN have some conflict with the proposed solution. In the proposed solution, preserving the IP routing for content forwarding conflicts with the CCN characteristics which has been proposed before the proposed solution. However the proposed solution is considered to be acceptable as it has the content centricity which is the key element of content centric networking. But they way of implementation is different.

Several Web technologies today (e.g., Content Delivery Networks (CDNs) and caches) have evolved to help scale content distribution. Though these technologies are not information centric, they implicitly support location-independent naming in that they serve the same data object from several locations. In this vein, HTTP itself can be considered to be information-centric in that URLs name a piece of content [7]. However, the host-name component of a URL is bound to a location in the network. Using the host

name in a URL literally inhibits true location independent naming, and is a significant obstacle in adapting these technologies to be more fully information-centric. Directing clients to other hosts (such as CDNs or mirrors) requires DNS redirection accomplished by changing the hostname (and by extension the URL), thereby fragmenting the namespace and encoding location-dependence into a content name. Fragmenting the namespace in this manner reduces the effectiveness of content caches, because a cache has no way of identifying that a particular piece of content is duplicated across multiple URLs. Additionally, given that HTTP requests are sent directly to the IP address of a host, local caches must be placed directly along the network path and sniff all HTTP headers to provide any benefit. Qualitatively, this design forces a fundamental trade-off between persistent content-naming and efficient content distribution. Moreover, Web technologies focus almost exclusively on these two points, and provide no mechanisms for content security (aside from securing the connection between two hosts) or support for client mobility.

III. Proposed DNS for CCN

In the proposed architecture, we have included the functionality of content names in DNS. Hence the DNS will represent the contents along with the host name. To do so, we have adopted the content delivery protocol. This change helps in achieving the location independent content naming and delivery.

The concept of content record is used as an additional service of DNS which represents the contents and their possible sources. Whenever a client need some content, it first retrieve the content record from the DNS. Once the content record is fetched, client gets the list of all possible servers hosting the content, protocols used to fetch those content and security parameters to verify the authenticity of those contents.

The format of content record is shown in figure 1. It includes content name instead of host name, type of content, its class and time to live, chache field stating whether content need to be cached or not, security field for the authentication of content, content delivery protocol identification field and the address field containing the possible content hosting servers. Address field may contain more than one address which indicates that the address might be the origin/source address or potentially a content hosting node/server. The security field

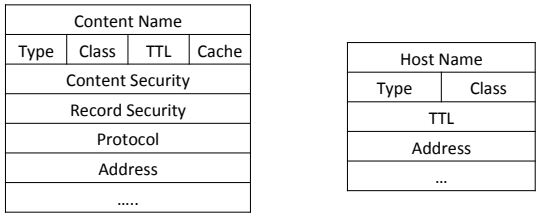


Fig. 1. Content Record vs Host Record

in content record may contain the hash value calculated over the content, or security key which might be the public key of publisher which ill be used for the authentication and verification purpose of digital signature. This field makes the content more secure from in the modification at the intermediate content hosting nodes. However the security of the content record itself is important. Without securing the content record, security field will not work properly as the attacker can publish the same content with its own pubic key and hash calculated over the content. Hence the content record can be secured by any existing security approach such as DNSSEC [8].

Based on the above discussion, securing the content record should start from its generation time by the source node. To do so, every source is allowed to publish its content record through their predefined prefixes. For example, Alice will be allowed to publish her content through /uni/doc/alice and Bob as /uni/doc/bob. This type of security can easily be achieve by adopting any access control technique used by the content servers supporting multiple publishers (e.g. HTTP and FTP). Through these servers, only authentic publisher/user can publish its content because each publisher is assigned a username and password. Based on this approach, each DNS manage the record of its zone and different domain may handle security, registration and scalability differently without impacting the DNS itself.

Once the content is published by the publisher and content record is secured, it is publicly available to all the users throughout the internet. Whenever a client need a content, it asks the DNS for content record. After receiving it, client gets a list of one or more addresses hosting that content. However the addresses in that list have been ranked by the DNS server according to the availability, security and its location from the requesting client or some other parameters specified in [9]. So the client requests the content from the first address of the list. Client does not participate in ranking the addresses of content hosting nodes.

However directing the client to the nearest copy of content is the main point. To achieve this functionality in the proposed scheme, the local content server address is included in the list. Hence the ranking is done in such a way that this locality is reflected in the response. In this proposed architecture, any node in the response path is allowed to add the address record or reorder the address record to facilitate the DNS server in better understanding the client's environment.

This could lead to many DNS servers in return path but there are typically two DNS servers called (1) authoritative DNS server that holds the content record and (2) local DNS server. This local DNS server is has the main responsibility of directing the client to closer caches. This is because, local DNS server, after seeing the addresses of content hosting nodes, can efficiently perform the fine grain localization than the authoritative server that can only see the local DNS server address. Multiple studies [10], [11] have shown that this address is simply helpful for coarse-grained localization, and this limits the effectiveness of CDNs powered by DNS redirection.

IV. Content Delivery Protocols

After receiving the content record, client fetch the content from the most appropriate location using the records information where it is cached or replicated. The caching of content is divided into two categories: (1) long lived and (2) cached. Long lived usually try to guarantee the content availability while caching does not guarantee the availability of content. Caches usually provide ”best effort” reliability such that the requested contetn have never been cached or available or have been removed.

For long love content approach, that guarantee the content availability, a publisher usually add servers or mirroring locations or deploy a content delivery network to replicate the contents over different locations. However the content replication in content delivery protocol is done in ad hoc fashion while in the proposed architecture, it can be done by simply retrieving the content record from authoritative server by publisher and add new add server addresses to it hosting the content replicas.

The authoritative DNS server can reorder the addresses in any way after adding new addresses by the publisher. Authoritative DNS can provide a full record to the requesting server or only a small subset of addresses that are close to the requesting server's location. This is shown in figure 2. First publisher registers the content record with authoritative DNS server and then clients request the authoritative DNS server through their local servers for desired content. Local DNS servers receive different address sets according to their locations. Hence these servers request different content hosting nodes for the same content.

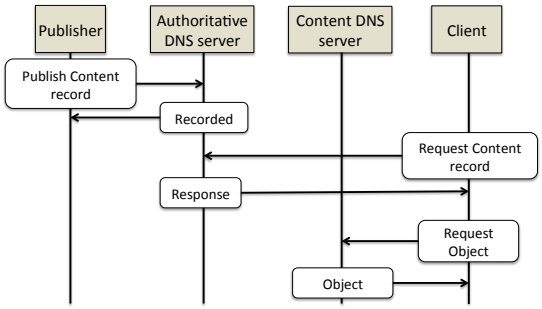


Fig. 2. Dynamic Record Generation

From security point of view, there must be difference between the new publishing content records and already published content records (i.e. cached and mirrors). In the already published content records, only the owner of that content record has the authority to do any changes in it. However this restriction is not applicable to the hosting or mirroring nodes. This is because, sometimes, nodes receive many request for the same content and are forced to mirror the content without informing the owner of content record. Hence these nodes can attach their own address to the content record without owner's permission. But these nodes are not allowed to do any changes in the existing content record metadata. This helps the client to easily identify any malicious content.

While on the other hand, caching also play an important role in content distribution. DNS servers usually know the nearest caching nodes and can direct the clients directly to those caches by adding the their addresses in address set. This is considered to be the most efficient approach as the local DNS know the location of client and can direct client to the nearest cache easily. This process is shown in figure 3.

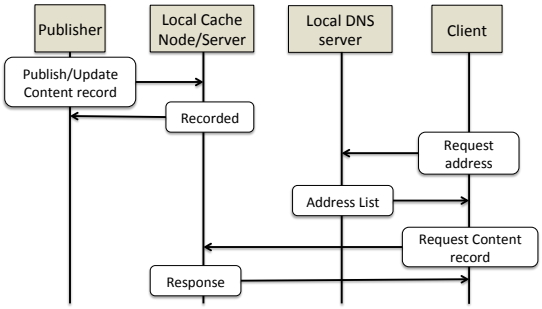


Fig. 3. Local Record Generation

V. Proposed Model vs CCN Model

Having provided a technical overview of proposed model, we return to the previously stated goals and benefits of CCN with the intent of showing that proposed architecture achieves all of the benefits of prior CCN proposals.

A. Location Independent Naming

In the proposed DNS architecture, uniqueness and consistency in the content name has been achieved through the hierarchical nature of the DNS. Also names were made independent from location by separating the addresses from content record and maintains a namespace that is not fragmented, even when content is moved or mirrored across different content servers.

There are many suggestions [12], [13] regarding the retrieval of content names. For example, whether it should be retrieved from flat name space or from hierarchical structure. However, flat name approach use peripheral name resolution service (NRS) that does the mapping of readable names to routable ones and vice versa.

Proposed DNS architecture does not force to use any of these approach for name retrieval. Content record provides a natural point of convergence for either approach. Since DNS follow a hierarchical structure, flat name-resolution protocols exist [14], [15], [16] and other protocols can be designed as necessary. Mapping a name to a content record would be main objective. However, NetInf [4] shows that flat name scheme can be used in DNS.

B. Efficient Content Distribution

The main objective of CCN architecture is reduce the congestion and enhance the efficient content distribution using caching and nearest replica routing. This has been achieved in

the proposed DNS architecture by allowing the servers along the DNS response path to do the required changes in address set.

Different caching policies have been introduced by the researchers for CCN architecture [5], [17], [18], [19] , such as ubiquitous caching compared to edge caching. Proposed DNS architecture can enable any caching policy based on the topology of intermediate DNS servers appending cache addresses.

C. Object Level Security

In CCN architecture, client can get the content from any content hosting node. This node might be a malicious node or some fake node. Hence a proper security approach is need where the content are authenticated instead of the content provider. This need some modification in the existing security schemes. In the proposed DNS model, content can be retrieved from any content hosting node instead of its generator, so a security field used for the content authentication in inserted in the content record.

Compared to other CCN proposals, an advantage of proposed DNS architecture is that its trust model depends only on the DNS. Since the content record is playing an important role in content retrieval, content record is secured using DNSSEC or some other protocol in necessary. If client receive the content from the address mentioned in secured content record, it assume the received content as authentic content. Also the verification of content at intermediate routers are not required in this model. This technique avoids an open problem in the CCN community, where many questions exist regarding the trust and feasibility of a universal PKI (or other such security protocol) deployed at intermediate routers, as well as the feasibility and scalability of performing content verification at each router.

D. Mobility

Proposed DNS architecture support the client mobility as well. Because the focus is on the content location instead of the host location. When nodes leave and rejoin the network, DHCP already provides them with the address of a local DNS server, and this information is all that is necessary to localize the client. The client then sends subsequent content record requests to the new local DNS server, which directs the client to nearby caches if they exist.

Proposed DNS architecture is different from other proposed CCN model. In this model, one node locate the content while other node fetch that content. While in CCN, single node send a request for content also fetch that content. Conceptually, this separates the act of locating content from the act of distributing it, and this split enables two separate topologies to coexist: one for content-location and the other for content-distribution. This design is a key strength of proposed DNS architecture, because it effectively supports ”near-replica routing” without relying on large content tables or a content-routing protocol. DNS names are simply routed swiftly without any localization or fragmentation, and then the content-request itself is routed over IP.

VI. Analysis and Evaluation

Accessing the content directly through DNS increases the burden on DNS by several order of magnitude to fulfill the requested queries [20]. To utilize the distributed nature of DNS efficiently, the content record and its management is divided among at least two different types of DNS. (1) Authoritative DNS is responsible for storing the content record and serving the request for it while (2) local DNS server is responsible for handling the queries for object, caching contents and forwarding records back to the clients.

This functionality increases the demand for storage as the number of content record increases, their storage and processing power also increases to satisfy the requested queries. Also increase in the name field further increase the storage requirements over DNS servers.

(1) Memory and Power Consumption: As the number of content records and content by itself increases the memory requirements which in-turn increases the energy consumption in handling those records and contents. however, if we look at the functionality of HTTP, we found that HTTP servers are already performing most of the same functionality. But current DNS architecture provide only names mapping and HTTP needs to manage the whole process of content name mapping, storage and delivering content to the requester. In the proposed architecture, DNS servers also keeps the content record in addition to the current functionality but they do not provide the contents.

Since the DNS is a tree based hierarchical distributed system, any update in the content record or the content itself by the publisher/owner or the organization will not slow down or affect the normal functionality of DNS. Hence this functionality motivate all the organization to successfully handle the content publication over the distributed server/nodes and keep the content records up to date easily.

(2) Delay and Referrals: DNS requests begin at the root and descend the hierarchy as necessary. As an example, with no cached info, DNS resolution for www.parc.com consists of three requests: the first to the root name-server, the second to the authoritative server for com, and therefore the last to the authoritative server for parc. Thus, if name contains more fields, more request will be generated.

This same approach is adopted by the DNS in which the request will be generated based on the name structure. Also the same record may result in different actions depending on the structure of name. To arrange the names in a meaningful way, some good assumption are necessary for their distribution and for their name structure. A good approach is to use the HTTP based naming structure like www.parc.com/index.html. This example contains four fields, ”www”, ”parc”, ”com” and ”index.html”. So the DNS server managing the www is responsible for keeping the content record for index.html. Such an assumption is safe and helpful, as HTTP does not mandate

the format of the path of object, and therefore the assumption permits us to draw conclusions from existing HTTP traffic.

Based on the following assumption, a large set of HTTP GETs requests have been evaluated. To do so, the host names have been removed and examined the rest of HTTP path for different fields extraction. For example, GET for parc.com/index.html has a value 1 and parc.com/label/index.html has a value 2. The proposed architecture results are shown in figure 4. Although prior analysis of DNS traffic has shown that DNS requests are largely reduced by the DNS cache. Jung et al [21] observe that the average DNS request leads to 1.2 referrals and a latency of roughly 60ms, despite the very fact that the average DNS name has 3.3 fields. These results are encouraging, because they illustrate the effectiveness of caching in improving DNS performance.

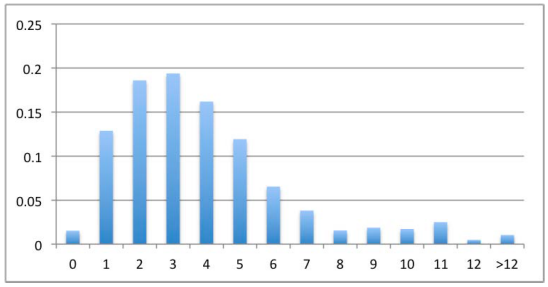


Fig. 4. Histogram of HTTP Path Components

It has been observe that caching and its optimization of DNS for host name give us very good results which indicates that it will be equally successful when applied to content objects.

The proposed architecture must be scalable at local DNS server as it put extra burden on local DNS server. For example, local DNS is responsible for directing the client to nearest content hosting node. In its simplest form, it appends the address to address set of DNS response for redirecting clients. This put less work than transparent caches when it checks HTTP header field. A number of schemes have been developed for cache load balancing but these are more complex and always exist a tradeoff between the complexity and efficiency of scheme. SUch a tradeoff is usually optimized for an application based on its requirements but not work for other scheme and need to optimize it again.

It has been mentioned that local DNS server keep the record. As the record increases, burden on DNS server increases which affect the performance. However, multiple studies [21], [22] have shown that memory requirement is not the only factor limiting its performance.This is because, content object is usually compressed which requires quite small memory. Thus there is ample room for DNS caching to expand before the effect on its performance is observed.

VII. Conclusion

Proposed DNS architecture is the first step to adopt the CCN architecture in practice. The proposed architecture is also compatible with the existing architecture to route and deliver the content with minor changes. Hence its compatibility ensure that this approach can be deployed in current scenario and can be extended to support more features of content centric networks. The analysis of proposed model and its functionality shows that it can be deployed at web scale and also it has achieved the benefits of content centric networks without increasing the complexity and communication and management overhead.

References

[1] T. Koponen, M. Chawla, B.-G. Chun, A. Ermolinskiy, K. H. Kim, S. Shenker, and I. Stoica, "A data-oriented (and beyond) network architecture," in *Proceedings of the 2007 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, ser. SIGCOMM '07. New York, NY, USA: ACM, 2007, pp. 181–192. [Online]. Available: http://doi.acm.org/10.1145/1282380.1282402

[2] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard, "Networking named content," in *Proceedings of the 5th International Conference on Emerging Networking Experiments and Technologies*, ser. CoNEXT '09. New York, NY, USA: ACM, 2009, pp. 1–12. [Online]. Available: http://doi.acm.org/10.1145/1658939.1658941

[3] N. Fotiou, P. Nikander, D. Trossen, and G. Polyzos, "Developing information networking further: From psirp to pursuit," in *Broadband Communications, Networks, and Systems*, ser. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering. Springer Berlin Heidelberg, 2012, pp. 1–13. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-30376-0-1

[4] C. Dannewitz, D. Kutscher, B. Ohlman, S. Farrell, B. Ahlgren, and H. Karl, "Network of information (netinf) - an information-centric networking architecture," *Comput. Commun.*, vol. 36, no. 7, pp. 721–735, Apr. 2013. [Online]. Available: http://dx.doi.org/10.1016/j.comcom.2013.01.009

[5] S. K. Fayazbakhsh, Y. Lin, A. Tootoonchian, A. Ghodsi, T. Koponen, B. Maggs, K. Ng, V. Sekar, and S. Shenker, "Less pain, most of the gain: Incrementally deployable icn," in *Proceedings of the ACM SIGCOMM 2013 Conference on SIGCOMM*, ser. SIGCOMM '13. New York, NY, USA: ACM, 2013, pp. 147–158. [Online]. Available: http://doi.acm.org/10.1145/2486001.2486023

[6] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher, and B. Ohlman, "A survey of information-centric networking," *Communications Magazine, IEEE*, vol. 50, no. 7, pp. 26–36, July 2012.

[7] L. Popa, A. Ghodsi, and I. Stoica, "Http as the narrow waist of the future internet," in *Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks*, ser. Hotnets-IX. New York, NY, USA: ACM, 2010, pp. 6:1–6:6. [Online]. Available: http://doi.acm.org/10.1145/1868447.1868453

[8] S. Weiler and D. Blacka, "Clarifications and implementation notes for dns security (dnssec)," 2013.

[9] R. Draves, "Default address selection for internet protocol version 6 (ipv6)," 2003.

[10] Z. M. Mao, C. D. Cranor, F. Douglis, M. Rabinovich, O. Spatscheck, and J. Wang, "A precise and efficient evaluation of the proximity between web clients and their local dns servers," in *Proceedings of the General Track of the Annual Conference on USENIX Annual Technical Conference*, ser. ATEC '02. Berkeley, CA, USA: USENIX Association, 2002, pp. 229–242. [Online]. Available: http://dl.acm.org/citation.cfm?id=647057.759950

[11] A. Shaikh, R. Tewari, and M. Agrawal, "On the effectiveness of dns-based server selection," in *INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 3, 2001, pp. 1801–1810 vol.3.

[12] A. Ghodsi, T. Koponen, J. Rajahalme, P. Sarolahti, and S. Shenker, "Naming in content-oriented architectures," in *Proceedings of the ACM SIGCOMM Workshop on Information-centric Networking*, ser. ICN '11. New York, NY, USA: ACM, 2011, pp. 1–6. [Online]. Available: http://doi.acm.org/10.1145/2018584.2018586

[13] M. Bari, S. Chowdhury, R. Ahmed, R. Boutaba, and B. Mathieu, "A survey of naming and routing in information-centric networks," *Communications Magazine, IEEE*, vol. 50, no. 12, pp. 44–53, December 2012.

[14] K. V. Katsaros, N. Fotiou, X. Vasilakos, C. N. Ververidis, C. Tsilopoulos, G. Xylomenos, and G. C. Polyzos, "On inter-domain name resolution for information-centric networks," in *Proceedings of the 11th International IFIP TC 6 Conference on Networking - Volume Part I*, ser. IFIP'12. Berlin, Heidelberg: Springer-Verlag, 2012, pp. 13–26. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-30045-5-2

[15] V. Ramasubramanian and E. G. Sirer, "The design and implementation of a next generation name service for the internet," in *Proceedings of the 2004 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, ser. SIGCOMM '04. New York, NY, USA: ACM, 2004, pp. 331–342. [Online]. Available: http://doi.acm.org/10.1145/1015467.1015504

[16] T. Vu, A. Baid, Y. Zhang, T. D. Nguyen, J. Fukuyama, R. P. Martin, and D. Raychaudhuri, "Dmap: A shared hosting scheme for dynamic identifier to locator mappings in the global internet," in *Proceedings of the 2012 IEEE 32Nd International Conference on Distributed Computing Systems*, ser. ICDCS '12. Washington, DC, USA: IEEE Computer Society, 2012, pp. 698–707. [Online]. Available: http://dx.doi.org/10.1109/ICDCS.2012.50

[17] A. Ghodsi, S. Shenker, T. Koponen, A. Singla, B. Raghavan, and J. Wilcox, "Information-centric networking: Seeing the forest for the trees," in *Proceedings of the 10th ACM Workshop on Hot Topics in Networks*, ser. HotNets-X. New York, NY, USA: ACM, 2011, pp. 1:1–1:6. [Online]. Available: http://doi.acm.org/10.1145/2070562.2070563

[18] G. Xylomenos, X. Vasilakos, C. Tsilopoulos, V. Siris, and G. Polyzos, "Caching and mobility support in a publish-subscribe internet architecture," *Communications Magazine, IEEE*, vol. 50, no. 7, pp. 52–58, July 2012.

[19] W. K. Chai, D. He, I. Psaras, and G. Pavlou, "Cache "less for more" in information-centric networks," in *Proceedings of the 11th International IFIP TC 6 Conference on Networking - Volume Part I*, ser. IFIP'12. Berlin, Heidelberg: Springer-Verlag, 2012, pp. 27–40. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-30045-5-3

[20] D. Perino and M. Varvello, "A reality check for content centric networking," in *Proceedings of the ACM SIGCOMM Workshop on Information-centric Networking*, ser. ICN '11. New York, NY, USA: ACM, 2011, pp. 44–49. [Online]. Available: http://doi.acm.org/10.1145/2018584.2018596

[21] J. Jung, E. Sit, H. Balakrishnan, and R. Morris, "Dns performance and the effectiveness of caching," *Networking, IEEE/ACM Transactions on*, vol. 10, no. 5, pp. 589–603, Oct 2002.

[22] E. Cohen and H. Kaplan, "Proactive caching of dns records: Addressing a performance bottleneck," *Comput. Netw.*, vol. 41, no. 6, pp. 707–726, Apr. 2003. [Online]. Available: http://dx.doi.org/10.1016/S1389-1286(02)00424-3