Hochschule für Angewandte Wissenschaften Hamburg
Hamburg University of Applied Sciences

Information-Centric Networking

a related work survey

Markus Vahlenkamp

Fakultät Technik und Informatik
Studiendepartment Informatik

Faculty of Engineering and Computer Science
Department of Computer Science

Contents

List of Figures

1 Introduction

Fostered by the steadily increasing amount of data that is transferred through the Internet, researchers take the opinion that the network should support optimised content dissemination through an enhanced content awareness. Today content dissemination techniques like Content Delivery Networks (CDNs) are already build on top of the network and explicitly facilitated through distortions of the DNS system or redirection mechanisms. The objective of the Information-Centric Networking (ICN) research community instead is to expose content information to the network and let the network itself figure out where to acquire the content from and how to handle it the best.
In this work, we will first start by taking a look at the general idea behind ICN in section 2, followed by a comparison of the three ICN concepts NDN/CCNx, NetInf and PSIRP in section 3. Section 4 then discusses in detail the difference of the ICN concepts introduced in the previous section. We finally close this work by a non-exhaustive collection of possible research directions and topics.

2 Concept of Information-Centric Networking

A couple of projects which try to evaluate what ICN should look like and how it could be implemented already exist. They all take slightly different approaches but share basic principles and ideas that will be described in the following.

Publish / Subscribe paradigm
: Content that is to be disseminated is made available through a publication process. Afterwards content consumers are able to find, request and retrieve the content by issuing a subscription.

Caching
: Through caching network infrastructure resources should be saved. Whenever some content is delivered to a content consumer, the content is cached within the network in order to satisfy subsequent requests from a nearby replica.
This behaviour carries different implications for the content distribution. On the one hand the network and server load is reduced. The content doesn't have to be transmitted all the way from the origin server to the client. Thus the work is offloaded to the caches holding the benefit of reduced network bandwidth utilisation as well as central server resource savings. On the other hand the delivery properties such as transmission delays caused by for instance network congestions are positively influenced through the use of a nearby cache. The overall Quality of Experience (QoE) for the end user will increase.

Naming
: Today DNS hostnames are used to reference content. Thus CDNs manipulate DNS responses and rewrite links to steer different users towards different spatially distributed

replicas. All this needs to be done due to the properties of Uniform Resource Locators (URLs). URLs identify the content, but they are also used to map the identifier to the contents location within the network.
This coupling of identifier and locator of URLs is for instance one of the reasons why content consumers suffer broken links and unreachable content caused by content, server or domain relocation. The content may still be available, but resides on a different server and is thus no longer accessible through its previous URL.
ICN names though need to provide content identification without the coupling to the contents location. This property is then exploited to better support the in network caching properties of ICN [21].

Security

Todays network security techniques, when it comes to secure data distribution, mainly consist of securing the communication channel, instead of securing the data itself. SSL and TLS are used to securely transmit the data end-to-end. Something that's not expedient when using intermediary caches, distributed all over the Internet. Thus some mechanism for a secure data dissemination is required that supports some kind of man-in-the-middle caches spread all over the network, without violating security or privacy properties.
For instance in the existing ICN projects mechanisms for data integrity checks are popular to be coupled with the naming of content objects. They provide, what is called self-certifiability, a technique where the names of a particular object reflects the hash values of the data it refers to. This is somehow comparable to the concept of cryptographically generated IPv6 addresses [8], where also parts of the addresses are generated through the use of cryptographic hashes. The use of cryptographic hash functions provide sufficient strength to be able to proof the data integrity today.

Routing and Forwarding

As suggested by the already published ICN proposals and prototypes [7], two general approaches to routing and forwarding emerge. The one-step resolve/retrieve method, where content requests are immediately routed towards an origin node, and the two-step resolve/retrieve, where a Name Resolution Service (NRS) is queried for the information that is needed to deliver the content request towards an instance of the content.

One step resolve/retrieve Figure 1 displays the one step resolve/retrieve mechanism. It is divided into two phases, the first phase, where a rendezvous between the request or subscription message and the content itself is aimed. The illustration depicts a request for some piece of content that is to be retrieved. This request arrives at Node1, where the name of the requested content is looked up in the name routing table to further be delivered towards the source. When the request arrives at the source, the second phase starts, which is delivering the requested content to the content consumer.

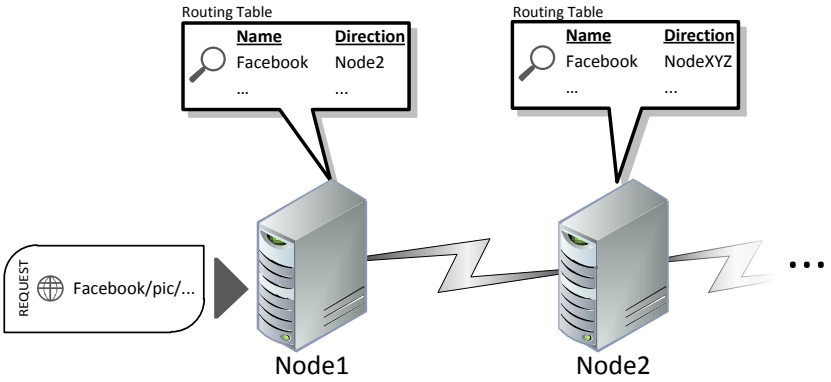

Figure 1: Conceptual view of one-step resolve/retrieve

Two step resolve/retrieve Figure 2 displays the two step resolve/retrieve mechanism. There exists a NRS, that is explicitly used to map content names to topological network addresses. Different options for these NRS are known today, as depicted in figure 2 they include for instance the use of distributed hash tables or distributed databases.

In contrast to the one step approach, the two step option consists of three phases. In the first phase, the name resolution service is utilised to map content names to topological addresses. These topological addresses are subsequently used to route the request towards a copy of the requested content. Finally in the third phase, the requested content is delivery towards the subscriber.

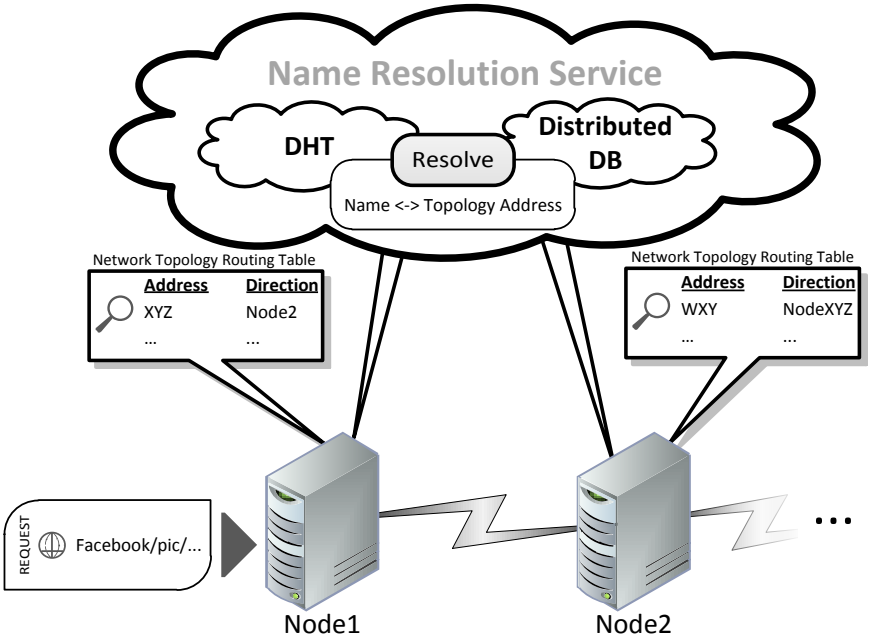

Figure 2: Conceptual view of two-step resolve/retrieve

3 Information-Centric Networking Prototypes

In this chapter, we will take a closer look at three actual ICN concepts and their implementation. In subsection 3.1 we introduce CCNx, the implementation of PARC's Named Data Networking (NDN) approach. Followed by NetInf, a prototype that is part of the 4Ward and SAIL research projects that belong to the European FP7 initiative. Finally the Publish-Subscribe Internet Routing Paradigm (PSIRP), also outcome of an European FP7 project that is continued as PURSUIT (Publish-Subscribe Internet Technologies) is discussed in subsection 3.3.

3.1 NDN/CCNx

The NDN concept [1] originates from the Palo Alto Research Center (PARC). It is used as the underlying concept for the implementation of ICN called CCNx [13], that is implemented and provided also by PARC.
Figure 3 depicts the general mode of operation of the NDN/CCNx prototype. Interest packets are created by a content consumer to request any content, the Interest packets are then routed in a hop-by-hop fashion towards a known source of the content. Every CCNx router uses its pre-populated name routing table to decide on which interface to forward the Interests. When the Interest reaches the source, the requested data is send back exactly the same path that the Interest packet took. This behaviour is caused by the architecture of the router. Interests that a router has forwarded are maintained in a list called Pending Interest Table (PIT). The PIT contains just records of Interests for which the corresponding content has not yet arrived. The entries consist of the names of the Interests as well as the id of the interface the request was received on. When content packets arrive, the router looks up the corresponding name of the data within the PIT and transmits the content on every interface that is listed within the particular entry. Besides forwarding the content to the consumers, the corresponding PIT entry is removed and the content is stored in the routers local cache. Through this step subsequent requests can be satisfied by the local cache itself instead of having to request it again from the source.
Figure 4 displays the structure of a CCNx router with its different components. The faces

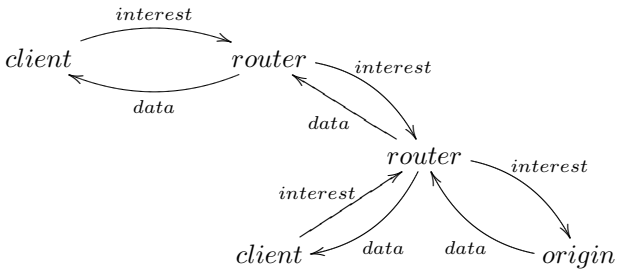

Figure 3: Abstract CCNx overview[7]

represent connections to other nodes or applications. The content store is utilised to cache previously forwarded content for subsequent requests. The Forward Information Base (FIB)

that holds the name routing information, where to forward requests that could not be satisfied by the cache, and finally the PIT, the data structure that is used to avoid duplicate Interest forwardings for the same content. Interests for the same name are aggregated through the

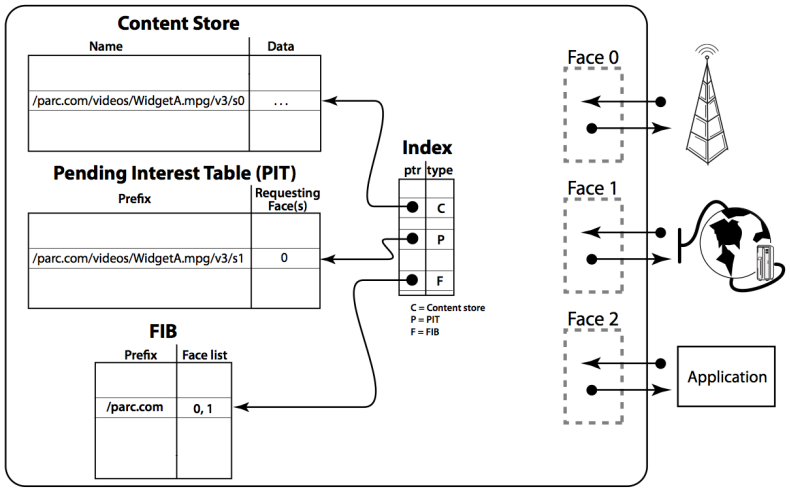

Figure 4: Conceptual CCNx router architecture[22]

use of the PIT, by adding the additional faces to the corresponding entry. Hence when the data arrives, it is duplicated locally by forwarding it out of the particular interfaces, realising an bandwidth efficient multicast like distribution behaviour.
The Names used in NDN/CCNx could for instance look like this: `ccnx:/parc/videos/intro.avi`. NDN/CCNx names follow the hierarchical structure of Uniform Resource Identifiers (URIs), name components are separated through slashes. When used to disseminate the actual content these names are extended by suffixes, that support naming of particular chunks of the content.
CCNx uses the one step resolve/retrieve process already described in section 2 to route Interests and acquire the requested content. This route lookup is done in a longest prefix match style. The longest full component matching entry of the routing table is used to forward the Interest out of the corresponding face with the best metric.

3.2 NetInf

The Network of Information, in short NetInf [2, 18], is a concept that emerged from the European FP 7 research projects 4Ward and SAIL. Figure 5 depicts an abstract overview of how NetInf works. Content that should be disseminated through NetInf needs to be registered within the NRS. Whenever a client wants to receive a particular piece of content it queries the NRS to acquire the topology based address of nodes holding the requested content. The data transfer itself is then realised through some transport protocol that is not specifically mandated by NetInf. When it comes to cache utilisation, this property carries an implication with it. As long as the utilized transport protocol itself doesn't provide some sort of cache awareness, just those cached copies explicitly registered within the NRS can be

taken into account. The name resolution and routing mechanism of NetInf follows the two-

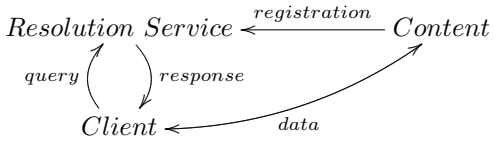

Figure 5: Abstract NetInf overview[7]

step resolve/retrieve approach introduced in section 2. The NRS utilises a Multilevel-DHT to save and organize the registered content and its sources.
The naming scheme that is used to address the content provides self-certifiability for the content. Through the self-certifiablility a content consumer is always able to verify if the retrieved content corresponds to the content that was requested. The naming structure is depicted in Figure 6. One can see that the name includes the hash of the content that is identified by the name.

3.3 PSIRP

The Publish-Subscribe Internet Routing Paradigm (PSIRP) [3] is also an outcome of a European FP7 research project that further continues its work under the name of Publish-Subscribe Internet Technologies (PURSUIT) [4].
Figure 7 depicts a high-level overview of the mechanics PSIRP is build upon. Content in PSIRP is also published by registering it within the rendezvous system. Whenever a content consumer wants to retrieve particular pieces of content he first queries the rendezvous system, that in turn resolves the name an constructs a Transport ID, that is than handed over to the content consumers client. Through attaching the obtained Transport ID to its subscription request the content consumer is able to steer its request towards the source he wants to utilise for the content transmission. Subscriptions then are forwarded in a hop-by-hop fashion towards a content source, whereas the sole information that is needed to forward the subscription is already contained in the Transport ID. The Transport ID represented by a so called zFilter describes the path the subscription should take, making PSIRP a source routing approach. The zFilter in fact follows the bloom filter approach. It is combining all IDs of the interfaces a packet has to traverse on its end-to-end path within one interfacemask. The mask is derived through the use of the bitwise OR-operation of the Interface IDs on the path. Displayed in Figure 8 are two possible Interface ID values, x_1 and x_2 that are combined to form a bloom filter, allowing the packet so traverse at least the two interfaces x_1 and x_2.
Through the introduction of bloom filters some blurriness is introduced into the routing process, this may lead to some amount of superfluous traffic, when interfaces matching the bloom filter, that the packet is not intended to traverse. The amount of this superfluous traffic

Type	Hash(PublicKey)	Label

Figure 6: Content id / name[5]

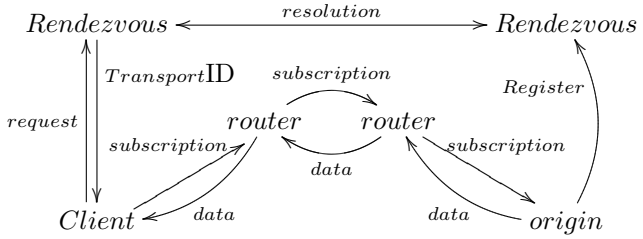

Figure 7: Abstract PSIRP overview[7]

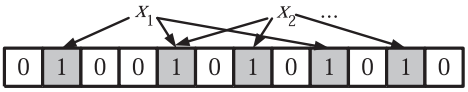

Figure 8: Bloom filter construction[9]

depends for instance on the length of the Interface IDs, the minimal amount of bits set as well as the amount of interfaces within the network. Through the right choice of these parameters the amount of unnecessary packet forwarding can be controlled. The blurriness of this approach further just leads to a false-positive forwarding behaviour. The packets are hence forwarded out of every interface they need to traverse but maybe also some additional interfaces. Figure 9 illustrates the whole process for a small network of nodes. It shows the links

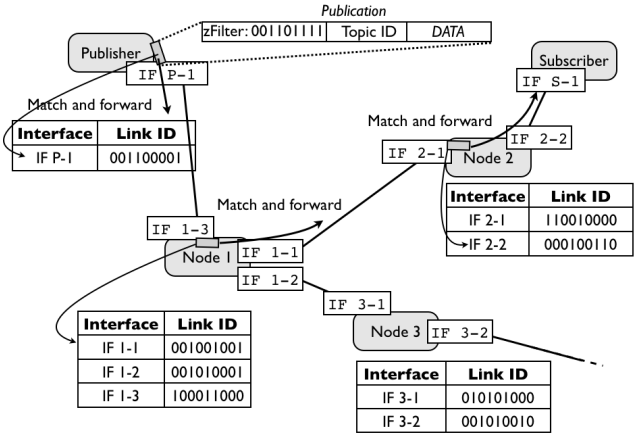

Figure 9: zFilter based forwarding[12]

and the Interface IDs for every node in the network. At the top a packet is shown, that is to be send through the network from the publisher to the subscriber along with its corresponding zFilter. The thin arrows indicate which entry matches the zFilter that the packet carries with it. The thick arrow indicates the paths the subscription takes.

As depicted, the packets are disseminated in a hop-by-hop fashion. Thus packets can be cached on every hop along the transmission path. To use these cached copies in an efficient manner, PSIRP registers every cached copy within the NRS, what in turn leads to a higher burden for the NRS, because it leads to a great number of additional entries as well as an

increased update frequency of the NRS.
The naming scheme used by PSIRP utilises non human-friendly names to identify the content. It is split into different abstraction levels, as shown in Figure 10. Application IDs are used by publishers and subscribers to identify the content. The Application ID is further resolved to Rendezvous IDs, that identify the network level identity of a publication. Rendezvous IDs are then, along with there associated Scope IDs handed over to the network where they are mapped to Forwarding IDs, the zFilters, that define the path through the network.

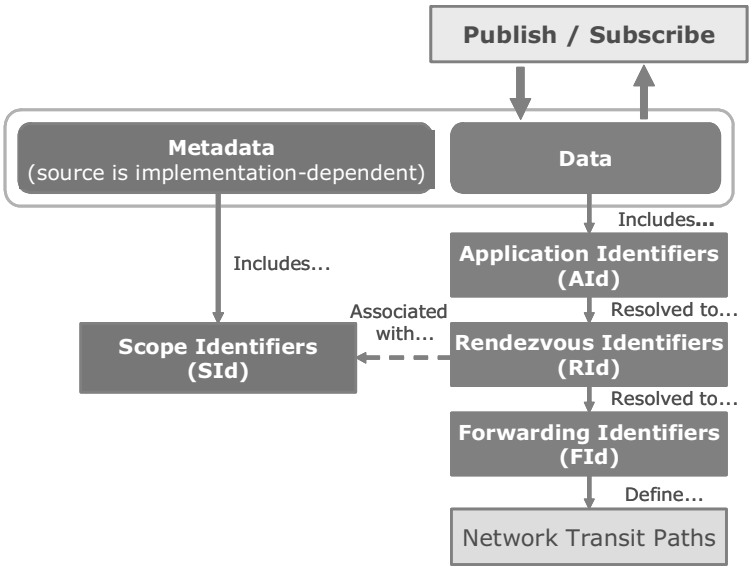

Figure 10: PSIRP ID coherence[17]

4 Comparison

Having introduced three different ICN approaches with their individual properties, we will compare them regarding certain vital aspects. We start by comparing the data path properties in subsection 4.1, followed by the comparison of the network states (4.2), naming (4.3), versioning (4.4) and close the section with a closer look at scoping in subsection 4.5.

4.1 Data path

In NDN/CCNx the requested data can just flow along the reverse path the interest packet took beforehand, this is due to the Reverse Path Forwarding (RPF) approach the NDN/CCNx designers took.
As opposed to NDN/CCNx, the PSIRP and NetInf concepts allow for different paths for the back and forth traffic. PSIRP allows this through the use of different zFilters provided by the topology manager for requests and responses. NetInf doesn't define the delivery protocol for request and response packets, hence through choosing an appropriate protocol this path-diversity can be achieved.

4.2 Network states

PSIRP utilises zFilter attached to the packets to steer them through the network in a source routing fashion. Where the zFilters are created by the NRS, thus the NRS has to handle the state information where all the pieces of content reside. NDN/CCNx on the other hand utilises soft-states, created by each interest packet for every chunk of data on each router on the path from subscriber to the source or just up to the point where a cached copy is available, what could lead to serious problems, as shown in [20].
Since NetInf doesn't rely on any underlying transport protocol in particular, one can make no general statement about the network states, that depend on the transport protocol used.

4.3 Naming

The naming schemes can be divided into two general categories, human-friendly, like used by NDN/CCNx and non human-friendly like those used by PSIRP or NetInf [10].
Another property coming along with the human-friendliness of the NDN/CCNx hierarchical naming scheme is the ability to aggregate names at their hierarchical boundaries which may be used to lower the size of the routers FIB. This method is on the opposite not simply applicable when using hash values of the content as part of their names, in the way that for instance NetInf does.

4.4 Versioning

Since the network should focus on the content itself, different versions of a particular piece of content may exist. These different content versions need to be distinguished by the network. It might be that the content consumer is interested in an older version of the content or explicitly in the most recent published edition. NDN/CCNx and NetInf inherently include mechanisms supporting the ability to distinguish different versions of the same content in the network, whereas PSIRP leaves this up to the application itself, that can define the Application ID scheme to implement some sort of versioning.

4.5 Scoping

Not all information available in a network are intended to be available globally. Thus scoping, or restricting the availability of content is desired. The three projects presented in this work provide different approaches to scoping the content availability. PSIRP utilises its Scope IDs mentioned earlier. NDN/CCNx can use export policies to prevent particular namespaces from being announced to other routers. This works similar to Border Gateway Protocol (BGP) export-policies, where IP prefixes are filtered before they are announced to adjacent BGP neighbours.
NetInf has no mechanism of restricting the availability scope of content so far, but possible approaches are shown in [7].

4.6 Cache placement

The rendezvous point in the PSIRP approach doesn't have to be on the path from content consumer to the content source, because the cached copies are also registered within the NRS, thus they are announced to the content consumers in the same way as the origin server. Whereas CCNx just checks its local cached content store when receiving packages thus only local cache content of the actual router is taken into account. The FIB contains just origin servers.
NetInf may work both ways, registering the copy within NRS. Depending on which transport protocol is utilised, a local cache lookup may also be available.
Publication [11] provides detailed information about on- and off-path caching in ICN.

5 Future Work

Throughout this work we've presented general properties of ICN (section 2), introduced three ICN prototypes (section 3) and described there general approach regarding some essential design parts. Finally we want to close this elaboration by pointing out remaining challenges and further topics in the field of ICN research.

Scalability & Performance Due to the vast amount of content that is already available in the internet and the continuing growth, ICN is required to perform very efficient and scalable. To give an example, Googles index reached the size of one trillion unique URLs in the summer of 2008[1]. Every website consists of different pieces of content that all need to be reachable by name, leading to an even larger number of entries within the NRS or FIB. ICN needs to handle the already existing content amount as well as further increases, to be future proof [6, 19, 14].

Non human-friendly names Some kind of a secure mapping service is needed for systems that do not provide human-friendly content names by default.

Scoping of content Mechanisms are required to scope content, to limit it's reachability. Some content is maybe just intended to stay within a company's intranet, other content should maybe be publicly available. ICN has to account for this.

Mobility IP introduced mechanisms like mobile IP [15, 16] to support node mobility without loosing connection or making the application aware of mobility. ICN also needs to support node mobility to be able to compete with IP. May it be origin or client mobility, with support for stored, as well as real-time content.

Security Beside the mechanisms for data integrity checks, capabilities for author and/or origin authentication need further research hence to validate if data that is expected to be created by an entity is definitely originated by it and not by maybe an attacker.

[1] http://googleblog.blogspot.de/2008/07/we-knew-web-was-big.html

References

[1] "The Named Data Networking Homepage," http://www.named-data.net, 2012.

[2] "The NetInf Homepage," http://www.netinf.org, 2012.

[3] "The PSIRP Homepage," http://www.psirp.org, 2012.

[4] "The PURSUIT Homepage," http://www.fp7-pursuit.eu, 2012.

[5] B. Ahlgren, M. D'Ambrosio, C. Dannewitz, A. Eriksson, J. Golic, B. Grönvall, D. Horne, A. Lindgren, O. Mämmelä, M. Marchisio, J. Mäkelä, S. Nechifor, B. Ohlman, K. Pentikousis, S. Randriamasy, T. Rautio, E. Renault, P. Seittenranta, O. Strandberg, B. Tarnauca, V. Vercellone, and D. Zeghlache, "Second NetInf Architecture Description," ser. Deliverable D6.2. 4WARD Project, 2010.

[6] B. Ahlgren, M. D'Ambrosio, M. Marchisio, I. Marsh, C. Dannewitz, B. Ohlman, K. Pentikousis, O. Strandberg, R. Rembarz, and V. Vercellone, "Design Considerations for a Network of Information," in *Proc. of Re-Architecting the Internet Workshop (ReARCH)*, ser. ReARCH '08. New York, NY, USA: ACM, 2008, pp. 66:1–66:6.

[7] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher, and B. Ohlmann, "A Survey of Information-Centric Networking (Draft)," Dagstuhl Seminar Proceedings, Tech. Rep. 10492, 2011.

[8] T. Aura, "Cryptographically Generated Addresses (CGA)," IETF, RFC 3972, March 2005.

[9] A. Z. Broder and M. Mitzenmacher, "Survey: Network applications of bloom filters: A survey," *Internet Mathematics*, vol. 1, no. 4, pp. 485–509, 2003.

[10] C. Dannewitz, J. Goliólic, B. Ohlman, and B. Ahlgren, "Secure Naming for a Network of Information," in *Proc. of the IEEE Global Internet Symposium*. Piscataway, NJ, USA: IEEE, 2010.

[11] A. Ghodsi, S. Shenker, T. Koponen, A. Singla, B. Raghavan, and J. Wilcox, "Information-Centric networking: Seeing the Forest for the Trees," in *Proc. of the 10th ACM HotNets Workshop*, ser. HotNets-X. New York, NY, USA: ACM, 2011.

[12] P. Jokela, A. Zahemszky, C. E. Rothenberg, S. Arianfar, and P. Nikander, "LIPSIN: Line Speed Publish/Subscribe Inter-networking," in *Proc. of the ACM SIGCOMM 2009*. New York, NY, USA: ACM, 2009, pp. 195–206.

[13] PARC, "The CCNx Homepage," http://www.ccnx.org, 2012.

[14] D. Perino and M. Varvello, "A Reality Check for Content Centric Networking," in *Proc. of the ACM SIGCOMM WS on Information-centric Networking (ICN '11)*. New York, NY, USA: ACM, 2011, pp. 44–49.

[15] C. Perkins, “IP Mobility Support for IPv4, Revised,” IETF, RFC 5944, November 2010.

[16] C. Perkins, D. Johnson, and J. Arkko, “Mobility Support in IPv6,” IETF, RFC 6275, July 2011.

[17] S. Tarkoma, M. Ain, and K. Visala, “The publish/subscribe internet routing paradigm (psirp): Designing the future internet architecture.” in *Future Internet Assembly*, G. Tselentis, J. Domingue, A. Galis, A. Gavras, D. Hausheer, S. Krco, V. Lotz, and T. Zahariadis, Eds. IOS Press, 2009, pp. 102–111. [Online]. Available: http://dblp.uni-trier.de/db/conf/fia/fia2009.html#TarkomaAV09

[18] D. Trossen and G. Parisis, “Designing and realizing an information-centric internet,” *IEEE Communications Magazine*, vol. 50, no. 7, pp. 60–67, 2012.

[19] M. Wählisch, T. C. Schmidt, and M. Vahlenkamp, “Backscatter from the Data Plane — Threats to Stability and Security in Information-Centric Networking,” Open Archive: arXiv.org, Technical Report arXiv:1205.4778v1, 2012. [Online]. Available: http://arxiv.org/abs/1205.4778v1

[20] ——, “Bulk of Interest: Performance Measurement of Content-Centric Routing,” in *Proc. of ACM SIGCOMM, Poster Session*. New York: ACM, August 2012, pp. 99–100. [Online]. Available: http://conferences.sigcomm.org/sigcomm/2012/paper/sigcomm/p99.pdf

[21] W. Wong and P. Nikander, “Secure Naming in Information-centric Networks,” in *Proc. of Re-Architecting the Internet Workshop (ReARCH ’10)*. New York, NY, USA: ACM, 2010, pp. 12:1–12:6.

[22] L. Zhang, D. Estrin, J. Burke, V. Jacobson, and J. D. Thornton, “Named Data Networking (NDN) Project,” PARC, Tech.report ndn-0001, 2010.