Content-Centric Networking

Fabian Oehlmann
Supervisor: Heiko Niedermayer
Seminar Future Internet WS2012/2013
Chair for Network Architectures and Services
Faculty for Informatics, Technische Universität München
Email: oehlmann@in.tum.de

ABSTRACT

This paper introduces to the alternative internet architecture of information-centric networking as opposed to the common host-to-host oriented design approach. Since the rise of computer networking until the internet becoming an incremental part of people's everyday life, it has undergone severe changes in usage scenarios. Users are interested in content and services and not in locations where to find them. In order to reflect those better in the underlying architecture of the internet, *Content-Centric Networking* (CCN) tries to loosen content and services from addresses in the network of machines. Although being challenged by a new set of difficulties, the approach yields some desirable system-inherent consequences that alone make the research effort worthwhile. Despite of its different mindset, a development of the common internet architecture towards this direction does not seem unrealistic.

Keywords

content-centric, information-centric, internet architecture, future internet

1. INTRODUCTION

The internet architecture today is still strongly driven by the ideas and necessities of the early days of computing and communication. A central design principle in computer networks is the end-to-end argument. As described in [12], it formulates the necessity of a complete function implementation only at the endpoints of a communication system. It is a pledge for a well-defined boundary between the communication subsystem and the rest by a strict interface. From this derives the simplicity, flexibility, and eventually also the universality, which made the internet as successful as it is.

Evidently, the internet has changed since its origins, making it reasonable to reconsider even its founding principles [2]. In the beginning it was an internetwork between few institutions, connected for purposes of resource sharing, where there would be many users on one machine. Today it is a continuously growing, ubiquitous phenomenon. Not only a selected group of people, but everyone interacts with it over many devices. An astonishing plethora of new usage scenarios for the internet developed, introducing new stakeholders and interests, such as those of governments, ISPs or enterprises whose business model is built on the internet. Grave is the loss of trust towards the rest of the internet, as there is today a wide range of security issues that have to be handled, when using it. For this and other purposes there have already been introduced mechanisms, which are not consistent with the end-to-end argument, such as firewalls, traffic filters or network address translators.

Having identified the distribution of content, such as videos, music, and news as the main use of the internet today, several propositions suggest to further adapt the architecture accordingly [5, 15]. The idea is to replace locations, where information can be found, with the information itself as the central element of networking, thus breaking out of the mindset of the end-to-end argument. This is termed as content-centric or information-oriented networking. Much like peer-to-peer mechanisms and content distribution networks facilitate this in the current architecture, new propositions try to manifest this on lower levels. This comes with certain inherent advantages, but also poses new challenges to be solved.

The next Section gives an overview of the ideas of those alternative architectures with up- and downsides. Afterwards Section 3 explains the mechanisms and practicability of content-centric networking (CCN) as one incarnation of the proposals in closer detail. Section 4 then gives an overview of the related proposals. The work is concluded in Section 5.

2. INFORMATION-ORIENTATION

The concept of information-orientation internetworking is to replace the location of content (or services) in the architecture, i.e. IP addresses, with the content itself. Thus everything used for packet forwarding is some sort of identifier for the contained data. The motivation is to provide a network participant means to declare interest in a piece of information to the network, which then replies with the content, preferably in an efficient manner.

With this concept in mind a couple of suggestions [13, 10, 9, 14] developed more or less closely related to the publish/subscribe paradigm described in [6]. The common principle is to have a producing side (publishers) and a consuming side (subscribers). The latter subscribe to content, thus posing their interest, while the publishers somehow put content into the system, that is then automatically forwarded to the subscribers. Furthermore, the pub/sub paradigm demands a decoupling of those two actions in time, space and synchronization, which means that they can be performed independently from one another. Effectively, this realizes a pull model for the receiving side, which decides on what

doi: 10.2312/NET-2013-02-1_06

content to receive and nothing else, opposed to the packet forwarding mechanism in the current internet. As a consequence the network is given the task of finding and delivering the content over just providing a host-to-host connection. The counterpart models of the current internet architecture and those of an information-oriented one are depicted in Table 1.

Original internet	**Information-oriented**
Sender	Content producer (publisher)
Receiver	Content consumer (subscriber)
Sender-based control	Receiver-based control
Client/Server communications	Publish/Subscribe sender and receiver uncoupled
Host-to-host	Service access/Information retrieval
Topology/Domain	Information scope
Unicast	Unified uni-, multi- and anycast
Explicit destination	Implicit destination
End-to-end	End-to-data
Host name (look-up oriented)	Data/Content name ("search" activity)
Secure channels, host authentication	Integrity and trust derived from the data

Table 1: The opposing concepts of the original and an information-oriented internet architecture [5]

2.1 Aimed benefits

Mostly following these principles the current designs promise a list of advantages over location-based communication. Depending on the design the advantages may be more or less inherent.

The first advantage of the concept is an easy way to implement communication primitives, such as multicast or anycast, while unicast is just a special case of the regular mechanism. Content that is subscribed by many end-points is just replicated according to the subscriptions, making this an inherent feature.

Secondly, mobility support is achieved by the lack of a need for receivers in a new network to obtain an address or identity. A sender can simply republish the offered content.

The freedom of location eases the use of many interfaces (multihoming), since just any connection can be used to publish or subscribe content, without further ado. Related to the feature of anycast support, this additionally improves persistence. Content can be published using one name, at many locations, over many channels.

Leaving out the destination in a packet header, makes it useful for many possible destinations. This renders the possibility of content caching at every intermediate node in a delivery tree. Similar to the inherent multicast, content that is demanded more than once, can be cached and replicated. This leads to a higher availability by reducing latency and increasing reliability through effective mirroring at nearby nodes.

The on-demand characteristic of the communication allows stateless connections. Interruptions in data transfer can therefore easily be tolerated, within the boundaries of application specific demands.

Especially from the pull-based messaging characteristic derives an increased security. DDoS, spoofing or spam could be practically eliminated, assuming a working authentication of the content, since the architecture only delivers packets a consumer has signaled interest in.

2.2 Challenges

The challenges for an implementation are manyfold and depend on a clear definition of the interface and therefore the tasks, that are to be implemented by the communication system. Over all the communication should in the end be more efficient in some way, making the development worthwhile.

The question of routing is the most crucial one, as it is the core of the architecture. The approaches taken are either inspired by existing routing protocols or build up on distributed hash tables. The difference to IP routing lies in the fact, that the names unlike IP addresses can not give a location hint as it is the case with IP prefixes and subnetting. This relates to the concept of introducing interface independent identities on top of the IP layer as in [3]. Furthermore, the mechanism has to be able to cope with the vast size of today's and the future's internet. Therefore a scalable solution is essential. Especially considering the fact, that there is more content than hosts in the network, an increased size of routing tables as to taken care of.

Additionally, a challenge is to meet requirements, which are driven by the internet's stakeholders like ISPs. Inter-domain routing policies have to be possible to be reflected in the new architecture as well. Preferable is also the possibility of an incremental deployment of the architecture, as well as the compatibility to existing machinery. A sudden replacement of the infrastructure would be out of scope, as costs would easily surpass the achievable advantages.

Lastly, ensuring security of the architecture has to be taken care of with highest priority. There could be risks unthought of in new routing designs, as opposed to the old ones, such as sibyl attacks in DHTs. Most importantly in a content-based architecture is guaranteeing authentication and integrity of the delivered content. If the proposed architecture can not give means to a consumer to be provided with the content she is asking for, its goal is not met. Thus, a strong protection against the compromisation of content is of vital importance.

3. THE CONTENT-CENTRIC NETWORKING DESIGN

This chapter outlines the general functioning mechanisms proposed by Jacobsen et al. in [9]. A first implementation of the concept is under development in the CCNx project[1].

The basic idea of content-centric networking is, as stated before, making content the central element in networking operations. Accordingly the architecture differs from the

[1]http://www.ccnx.org

common ISO/OSI layer model as depicted in Figure 1. Instead of IP, depicting source and destination of a packet, it is chunks of explicitly named content, that form the waist of the hourglass of the stack. By this, the content becomes the universal agreement between every network participant. It is decoupled from the end-hosts of the connection. Thus, on the new layer the source of the desired content is of no relevance anymore. Additionally, there is also a strategy layer and a security layer introduced below and on top of the content layer.

The introduced layers can be on top of the IP layer, but might as well be directly on top of the MAC layer or anything else delivering packets. The so called faces are managed by the strategy layer, which contains policies to when it is appropriate to use the correct one. Notably content is largely independent of the connection type it is served upon. Therefore it natively supports multihoming as well as it is tolerant towards disruptions of the connectivity. The authors chose the word faces instead of interfaces to point-out the possibility to have application processes work as those as well besides network interfaces. The security layer ensures the authenticity and integrity of the content, further described in Section 3.5.

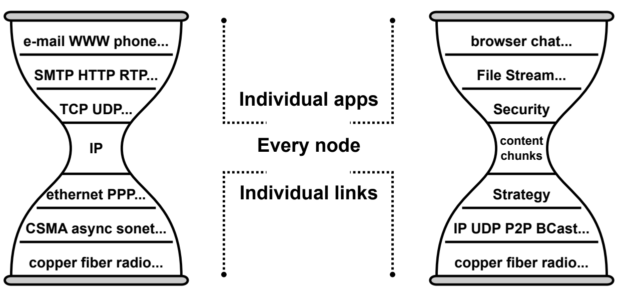

Figure 1: Changing the bottleneck of the communication stack to chunks of content [9]

3.1 Packet types

The CCN model knows two packet types. First is the *Interest* packet as depicted in Figure 2. By sending these over the outgoing interfaces, a node announces its demand for the content named by the packet. It is simply broadcasted on the available interfaces in hope to get the according data returned by the mechanisms of CCN. Naturally, the packet contains the name of the desired content. Additionally, it is accompanied with selection information, such as the scope within the network where the data should come from or certain filter information. Finally, it contains a nonce, used to detect duplicate interests.

In response to an incoming interest, *Data* packets are used as can be seen in Figure 3. The data packets are said to "satisfy" interest in that they maintain a one-to-one relation,

Content Name
Selector
Nonce

Figure 2: Interest packet

Content Name
Signature
Signed Info
Data

Figure 3: Data packet

where data consumes interest. This rule maintains a flow balance at each hop and prevents congestion in the middle of a connection path. The names in CCN are hierarchical with the consequence, that data only serves an interest, if its name-prefix matches the name of the interest. Apart from the name and arbitrary binary data, the packet also contains a digital signature of some cryptographic digest of the packet, as well as signed info. The last mentioned field gives additional information on the packet such as the publisher's ID, where to locate the key to check the signature or a timestamp. By these means of verfication it is to be ensured that the packet is authenticating and identifying itself and does not need legitimacy by the channel it was transferred over. This is further explained in Section 3.5. The design specifically allows interests for data that does not exist yet. To serve these "active names" publishers may generate content dynamically to meet the demands of the modern internet.

3.2 Forwarding engine model

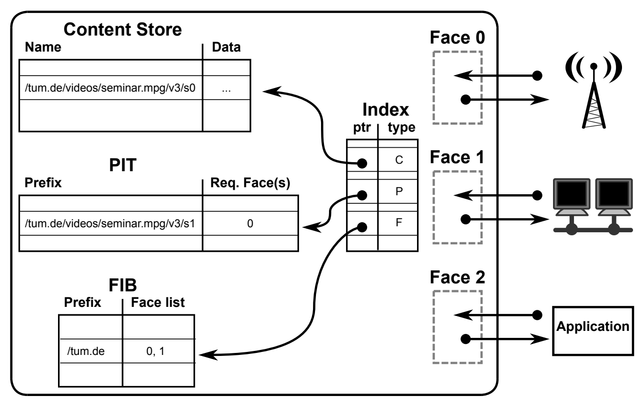

Figure 4: CCN node model [9]

A CCN node (see Figure 4) performs similar operations compared to those of a regular IP router. Packets that were delivered on a face undergo a longest-prefix matching on their name field and are processed according to the information stored in the three main data structures maintained by every node:

The content store (buffer memory) serves as a cache of content. Since content is self-authenticating and self-identifying each packet might be useful to any potential participant in the nearby network. The ability to serve content directly instead of generating further lookups minimizes overall bandwidth usage and latency. Compared to an IP router the most-recently-used replacement strategy of the buffer is replaced by a least-recently or least-frequently used strategy to increase the probability of a cache hit.

doi: 10.2312/NET-2013-02-1_06

The pending interest table (PIT) keeps track of Interests issued on the nodes faces. Hereby it does not matter whether the interest originates from the node itself or is one forwarded from another node. Interests are the only packets routed in CCN towards the source(s) of the content. As soon as an interest reaches a content source the PIT serves as a mark in the data's trail towards its requester(s). The authors compare this mechanism with "bread crumbs" consumed by the data on their way. By these means the data satisfies the interest and the entry in the PIT is removed. Unsatisfied interests time-out eventually and have to be reissued by the consumer.

The forwarding information base (FIB) acts like the routing table in a common IP router. It stores the information on which faces interests are to be forwarded upstream towards the source(s) of the content in question. The design hereby allows multiple entries that may be queried in parallel, because forwarding is not restricted to a spanning tree.

The processing of interest packets is managed according to the above order of the data structures. If the desired content is to be found in the cache, the node serves the request directly, thus, satisfying the interest. In case of a cache miss, the PIT is checked for an exact-match entry, and the arrival face is added to its list of requesting faces. This means an interest in that data has already been forwarded upstream. When it gets satisfied, the node copies the data on its way downstream. Lastly, it is matched for an entry in the FIB. The requesting face is removed from the FIB entry, an interest is sent out on the remaining faces of the entry and a PIT entry is stored. If the content name does not match any of the above the interest is discarded, as the node neither can satisfy it nor does it have the knowledge where to forward it. Duplicate interests might arrive at different faces, which is prevented by detecting and discarding ones that carry an already known nonce.

Data packets are processed in a similar way. First the name is matched with the content store. If it is already present on the node, it is a duplicate and can be discarded. Second the name is matched with the PIT. If there is no match, the data can be discarded as well, since there was no demand for it. In case of a match the data may be validated (see Section 3.5) and afterwards cached in the content store. Consequently it is forwarded on all faces in the list of the PIT entry.

3.3 Naming and transport

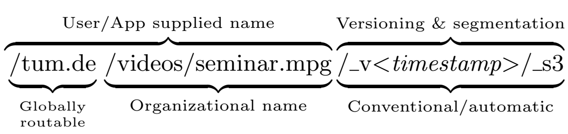

Figure 5: Human readable example data name

The naming scheme of CCN is designed to have a mostly human readable form for purposes of usability. At the same time it tries to comply with demands from routing, i.e. efficient routing capabilities on names, and is inspired by functioning mechanisms of TCP. Therefore, as can be seen in Figure 5, names are hierarchically structured and put together from several components. For notational convenience the human readable representation is chosen by the designers with / signs between the components as in common URIs. Notably, the binary representation differs from just the string representation of this. It is divided into a name provided by the user or an application, which consists of a globally routable name and the name of the content within the organizational structure of its origin. Secondly, the tail of the name is supposed to be a standardized naming convention reflecting the version of a certain content and its segmentation. The scheme allows to have a total order on the content that can be reflected in a content tree as in Figure 6. Accessing content means the traversal of the tree. An incremental feature obtained by imposing a total order is the ability to address content relative to known information. E.g. for accessing a video a consumer can issue an interest for '/tum.de/videos/seminar.mpg' and the selector primitive 'RightmostChild'. From having a certain chunk of video the standardized naming convention allows expressing interests in the following chunks by adding an offset according to the segmentation rules. The similarity to TCP's window size is hereby given with the number of the amount of next interests sent in parallel. The selective acknowledgement is given by the one-to-one serving mechanism of the chunks.

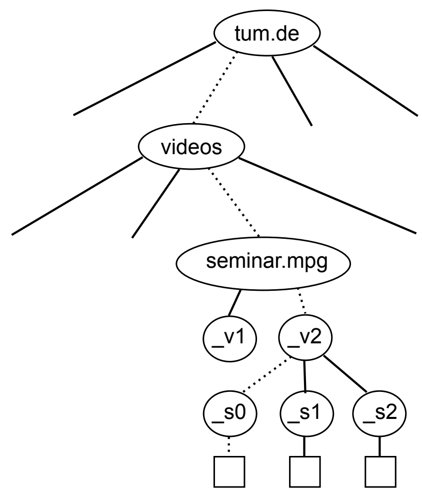

Figure 6: Name tree traversal [9]

3.4 Intra-domain Routing

Intra-domain routing in CCN is claimed to be very much compatible to routing schemes known in IP networks. Both rely on longest-prefix matching lookups, to hierarchically aggregate a more detailed connection information in the process of getting closer to the content. The similiarity is reflected in the close relation of the FIB and an IP routing table. The key difference is that the FIB has more than just one outgoing interface. The reason for this is the semantic difference of the entries. While an IP router can reach all hosts starting with a prefix via the stored interface, the CCN router only can reach some of the content with this prefix there. As a result the CCN router broadcasts interests to all of the faces on this entry to gather complete informa-

doi: 10.2312/NET-2013-02-1_06

tion. In an IP network this operation would lead to loops in the topology. In contrast, this is no problem in CCN, since neither interest nor data packets can loop. Thus, CCN is free from maintaining a spanning tree within the topology. For instance, if two nodes in a topology announce the same prefix, in IP this would mean that via both every content with such a prefix can be reached. Therefore, the forwarding mechanism has to pick the better one of them. In CCN, such two nodes would not both announce reachability to the whole content under this prefix, but maybe just a subset. Thus, both nodes have to be queried. This broadcast based mechanism raises questions of scalability [15].

Despite these differences, existing link-state routing protocols, namely IS-IS and OSPF, should be easily adaptable to the needs of CCN. The mentioned routing protocols basically handle two distinct tasks. The first is discovering local connectivities to adjacent nodes. The second is to announce available prefixes/resources throughout the network. By these means a CCN enabled node can distribute link-state-announcements containing the attached prefixes it can serve. The authors of CCN now propose that this is already possible with existing deployments of these protocols. Both of them announce connected resources using a general 'type label value' (TLV) scheme, which is capable of distributing CCN prefixes. Following the specification, nodes that do not support CCN-TLVs would simply ignore them. CCN enabled routers would directly add the distant CCN enabled router's IP address as a face. This way CCN routing could be implemented attached to that of IP. Additionally this can be done incrementally as not every router needs to be CCN enabled. CCN's interest broadcast mechanism would of course find better routes with an increased ratio of CCN enabled routers in the network.

3.5 Security

In the common internet architecture security of content is largely provided by authentication of the host from which content is delivered as well as securing the communications channel to prohibit man-in-the-middle attacks. The CCN architecture decouples content from its origin in a way that it does not matter by which means a consumer got hold of it. As a consequence authenticity and integrity of the content have to be possible to be checked just by the information provided alongside the data. In other words, the essential goals are to ensure that a certain piece of content was in fact published by the right source and that the content was not changed in any way. Thus, a digital signature mechanism has to be introduced. In [7, 9] the authors therefore demand publishers of content to maintain public and private key pairs to sign data packets. As given in Figure 3 data is transferred not only with its name, but also with the signature and some signed information. The approach is to have publishers not just sign the content, but both the content and its name together, in order to authenticate the linkage between them. The signed info does not need to be verifiable and may include the location of the public key or the public key itself. Thus, the consumer can verify whether the content in question was published with this name by a given key or not.

Eventually this mechanism does not provide trust in the key yet. As a solution a public key infrastructure is suggested. This can be easily realized by the CCN mechanisms themselves. Given a name tied to an identity, trust can be formed in the following way: An authority simply signs the name and public key of the identity as its content. This equals to an identity certificate issued by the authority. Additionally, this functioning allows fine grained certification of just a subtree of a given namespace.

The CCN design does not provide architectural instruments to protect content or enforce access control. The suggested solution is to rely on ordinary encryption of data, since CCN handles any binary sequence of data the same way.

3.6 Usability

Having described the basic architectural concepts, remains the question of the actual usability of the proposal in the real world.

3.6.1 Implementation

On the CCN project website one can find source code for an early stage implementation of the new architecture. At its core it consists of the ccnd linux userspace deamon, enabling basic CCN functionality for research purposes. Alongside to that an Android implementation, application libraries and several applications can be found. The latter prove the usability of the concept for the contexts of chatting, HTTP GET requests, media streaming and file transfer. Thus, it covers most of today's internet use cases.

As an example shall be given a closer insight on the most sophisticated application, given by the VoIP alternative VoCCN, as depicted in [8]. The application realizes phone calls between two parties using the Session Initiation Protocol SIP and RTP media streams. Used was a modified version of the Linphone VoIP client. Standard SIP connection establishment is done using a signaling path, which is an indirection infrastructure maintained by proxies, forwarding the connection details between the caller and the callee. By this a direct media path is built up between the two parties. VoCCN arguments to simplify this process by merging the signaling and the media path. A callee Bob simply announces to be the content source for the call, e.g. using the name '/tum.de/sip/bob/invite'. The caller Alice then sends out an interest for the SIP invite message with the same prefix, which is served by Bob. There he defines the connection details, i.e. the detailed name under which to route the mediastream. Thus, VoCCN makes use of the active names, described in Section 3.1, to serve unpublished content. This then sets up a simple bidirectional media-stream by the means of delivery depicted in the end of Section 3.3

3.6.2 Feasibility

The CCN design removes intelligence from the edges, the endpoints of connections, to the core of the network itself. This results in higher resource demands particularly considering the content caching functions, as well as an increased size of the routing tables. There is a general consent in that content caching poses high potential in increasing the efficiency of networking, by eliminating duplicate transmissions in the delivery tree. Yet, today's internet infrastructure is highly optimized on low levels to support packet forwarding mechanisms in host-to-host connections. Unfortunately, this means the underlying functions and data structures of the new proposals cannot simply be deployed on the existing infrastructure. A new hardware design for content routers is necessary. The question answered by [1] and [11] is, whether present-day technology is capable of processing the different workload. Naturally, due to the early stage of the research, those answers rely only on vague estimations regarding the dimensions of the requirements and are supported by simulation. Eventually, they point out that depending on the size of the implementation a deployment is not unrealistic. Up to the size of CDNs or ISPs the concept might very well be feasible. The scalability up to the size of the whole internet, although, is with current possibilities to be doubted.

3.6.3 Evaluation

Content-centric networking is a very radical approach to change the modern internet architecture. Yet, at its current stage it is a research effort, which demands a lot more testing and refining. Prototype deployments in real testbeds yet have to prove the correct functioning and scalability. Particularly the simplistic routing mechanism is a reason to doubt the practicability of CCN. Furthermore, the security aspect of the proposal is a crucial element. If the mechanism to provide authentication and integrity is corrupted, the architecture is defect in its substance. Delivery of the correct content could not be guaranteed anymore, potentially rendering the architecture useless. Another aspect are new threats rising from the mechanism itself, such as the possibility to render denial of service attacks by flooding interest packets, which can only be mitigated by heuristic countermeasures.

In summary, CCN is a yet very theoretic approach, displaying a certain elegance, which leads to a number of desirable advantages. However, it is unlikely to ever be deployed in its initial form. Nonetheless, it is a good research effort for the evolution of the internet. In what ways it will have an impact, will ultimately be decided by its economic benefits.

4. RELATED WORK

There are several proposals, which were designed to replace the current internet architecture. The one closest to the one of CCN is the Data-Oriented Network Architecture DONA [10]. It introduces a new network entity, the resolution handlers (RH), one for every autonomous system. Content is registered at those using flat, human-unreadable names, that are self-certifying. Publishers sign the data, that a consumer can verify with their public key, the signature and the name. Consumers issue find commands towards their RH, which locates the content. It is then delivered using a regular IP connection.

The internet indirection infrastructure (i3) [14], based on early experiences with distributed hash tables (DHT), similarly proposes to form an overlay network by a DHT. The DHT organizes nodes to take care of content identifiers. A node interested in such content places a trigger at this node. Publishers send packets for that identifier to that node, which then forwards the data to the addresses, who registered a trigger for the identifier. The data connection is done again using IP.

In the Publish-Subscribe Internet Routing Paradigm (PSIRP) project[2] [13] follows to implement a pure publish subscribe model as mentioned in Section 2. Much like DNS a directory structure is introduced mapping human-readable names on content-labels. The architecture would then rely on three Network modules. *Rendezvous*, which matches publications and subscriptions, *Topology*, which forms content-delivery-trees and *Forwarding* of labels. *Mediation* depicts the physical data transmission between nodes.

The Network of Information (NetInf) project[3] [4] also makes use of DHT for a name resolution system. While it also implements a publish-subscribe interface, its focus lies more on giving an abstract information model and naming framework.

5. CONCLUSION

The internet architecture as it evolved until today does not reflect the standard usage scenarios of the majority of its users anymore. Content-based networking seems to be a promising approach to simplify networking operations, reflecting today's requirements towards the internet in a more natural manner. A long list of potential enhancements could significantly increase the efficiency of the internet, while simplifying it at the same time.

CCN is a thoroughly designed approach towards the new paradigm. It is in an advanced status allowing ongoing research, which is necessary to reveal unthought of consequences and challenges. Nonetheless it shows possible flaws in respect to scalability when facing the internet's size. Moreover, by this example can be seen, on how many levels considerations have to be taken into account, when tackling the challenge of changing a huge system like the internet at its core.

Aside the technological consequences one may also not forget the surrounding implications the architectural change may have. Breaking up the end-to-end argument of [12] by shifting the computations from the end-hosts to the core of the network, is also a shift in which party has power over the network. As pointed out in [2], the end-to-end argument is a big reason for the internet's flexibility and base for innovation. A major redesign, thus, also has to be taken under careful consideration not only regarding its economic, but also its social benefits. The internet's stakeholders are numerous and represent opposing interests regarding its open nature. Thus, future developments of the internet should also be viewed cautiously from a non-technical perspective.

6. REFERENCES

[1] S. Arianfar, P. Nikander, and J. Ott. On content-centric router design and implications. In *Proceedings of the Re-Architecting the Internet Workshop*, ReARCH '10, pages 5:1–5:6, New York, NY, USA, 2010.

[2] M. S. Blumenthal and D. D. Clark. Rethinking the design of the internet: the end-to-end arguments vs. the brave new world. *ACM Trans. Internet Technol.*, 1(1):70–109, Aug. 2001.

[2] The PSIRP project has ended and is today followed by the PURSUIT project, http://www.fp7-pursuit.eu

[3] http://www.netinf.org

[3] M. Caesar, T. Condie, J. Kannan, K. Lakshminarayanan, and I. Stoica. Rofl: routing on flat labels. In *Proceedings of the 2006 conference on Applications, technologies, architectures, and protocols for computer communications*, SIGCOMM '06, pages 363–374, New York, NY, USA, 2006.

[4] C. Dannewitz. NetInf: An Information-Centric Design for the Future Internet. In *Proceedings of the 3rd GI/ITG KuVS Workshop on The Future Internet*, Munich, Germany, 2009.

[5] C. Esteve, F. L. Verdi, and M. F. Magalhães. Towards a new generation of information-oriented internetworking architectures. In *Proceedings of the 2008 ACM CoNEXT Conference*, CoNEXT '08, pages 65:1–65:6, New York, NY, USA, 2008.

[6] P. T. Eugster, P. A. Felber, R. Guerraoui, and A.-M. Kermarrec. The many faces of publish/subscribe. *ACM Comput. Surv.*, 35(2):114–131, June 2003.

[7] V. Jacobson and D. K. Smetters. Securing network content. Technical report, PARC, 2009.

[8] V. Jacobson, D. K. Smetters, N. H. Briggs, M. F. Plass, P. Stewart, J. D. Thornton, and R. L. Braynard. Voccn: voice-over content-centric networks. In *Proceedings of the 2009 workshop on Re-architecting the internet*, ReArch '09, pages 1–6, New York, NY, USA, 2009.

[9] 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*, CoNEXT '09, pages 1–12, New York, NY, USA, 2009.

[10] 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*, SIGCOMM '07, pages 181–192, New York, NY, USA, 2007.

[11] D. Perino and M. Varvello. A reality check for content centric networking. In *Proceedings of the ACM SIGCOMM workshop on Information-centric networking*, ICN '11, pages 44–49, New York, NY, USA, 2011.

[12] J. H. Saltzer, D. P. Reed, and D. D. Clark. End-to-end arguments in system design. *ACM Trans. Comput. Syst.*, 2(4):277–288, Nov. 1984.

[13] M. Särelä, T. Rinta-aho, and S. Tarkoma. RTFM: Publish/subscribe internetworking architecture. ICT Mobile Summit, 2008.

[14] I. Stoica, D. Adkins, S. Zhuang, S. Shenker, and S. Surana. Internet indirection infrastructure. In *Proceedings of the 2002 conference on Applications, technologies, architectures, and protocols for computer communications*, SIGCOMM '02, pages 73–86, New York, NY, USA, 2002.

[15] D. Trossen, M. Sarela, and K. Sollins. Arguments for an information-centric internetworking architecture. *SIGCOMM Comput. Commun. Rev.*, 40(2):26–33, Apr. 2010.

doi: 10.2312/NET-2013-02-1_06