Report from Dagstuhl Seminar 14291

Information-Centric Networking 3

Edited by

Dirk Kutscher[1], Taekyoung Kwon[2], and Ignacio Solis[3]

1   **NEC Laboratories Europe – Heidelberg, DE,** `dirk.kutscher@neclab.eu`
2   **Seoul Nat. University, KR,** `tkkwon98@gmail.com`
3   **PARC – Palo Alto, US,** `Ignacio.Solis@parc.com`

——— **Abstract** —————————————————————————

This report documents the presentations and discussions of the 3rd Dagstuhl seminar on Information-Centric Networks. This seminar was focused on the deployment and scalability of ICNs. An overview of various ICN projects was used as a starting point for discussions. Participants provided a set of starting questions to cover with the rest of the group. The seminar increased the awareness on the state of the art in ICN research. Various topics on deployment and scalability were discussed. The opinions and comments presented here came directly from the notes taken at the seminar.

1   Executive Summary

Ignacio Solis

Information Centric Networks (ICN) has been a growing area of research in the past few years. The Dagstuhl ICN Seminar series has played a central role in forming the research community. The first seminar, Dagstuhl Seminar 10492, was the meeting point of the various ICN projects across the world; both from the academic perspective as well as the commercial perspective.

The community created at this event continued interacting. It was not long before the members created a set of academic workshops at the most important networking conferences; SIGCOMM, INFOCOM, etc. Following the success of the second Dagstuhl Seminar (12361), the community continued to coalesce and founded the ICNRG. The ICNRG, Information Centric Networking Research Group, was formed at the IRTF to evaluate the technology and to create a forum for companies discuss possible standardization efforts.

With the third iteration of this Seminar we've attempted to bring together the academic and commercial community together once more to discuss the state of the art in ICN. Specifically, we've focused on scalability and deployment. First, what are the problems we face in terms of scaling ICN. Are there technical limitations or political limitations. Second, what are the roadblocks in the path towards deployment. Since there will be no overnight switch, the technology must be deployed in controlled environments where interoperability can be slowly achieved.

2 Table of Contents

3 Overview of Talks

3.1 ENCODERS

Ashish Gehani (SRI – Menlo Park, US), Minyoung Kim (SRI – Menlo Park, US)

ENCODERS is part of the DARPA CBMEN program. CBMEN (Content-Based Mobile Edge Networking) tries to build an Information Centric approach for soldiers to network in the field. It focuses on a mixture of ICN and MANET. ENCODERS uses a mixture of push and pull. It incorporates a priority system that is used between the publisher and requester to negotiate queuing and caching resources. The core of the system is based on Haggle. Discussion questions

- What information to be protected how?
- Can the insiders leak enough information so that outsiders can become insiders?
- How do you handle revocation?
- What are the energy consumption for security?

3.2 CASCADE

Vikas Kawadia (BBN Technologies – Cambridge, US)

CASCADE is part of the DARPA CBMEN program. CBMEN (Content-Based Mobile Edge Networking) tries to build an Information Centric approach for soldiers to network in the field. It focuses on a mixture of ICN and MANET. CASCADE is a content-based architecture based on the notion of communities. Nodes keep data around per community and move the data by duplicating it between communities. It's built on top of TCP and UDP. CASCADE uses a put and get API.

It is similar to a DHT over a MANET where the DHT focuses on a specific region (community). The core notion is that the region is stable. Basically, network links are stable between members. This region can move; for example people on a bus.

Content management is done via machine learning using meta-data. This meta-data informs the system of which data must move and where. There is an assumption of high heterogeneity.

3.3 Green ICN

Tohru Asami (University of Tokyo, JP)

Green ICN is a project on using ICN in situations with low amounts of energy. Focusing on disaster situations. Two main scenarios being considered:

- Disaster management (aftermaths of disaster): the green part is essential to make devices last as long as possible after the disaster so to keep critical service running (intervention time may be long) => energy consumption is a priority in this scenario
- Video delivery: how to make video distribution at very large scale

Disaster scenario: After the Japanese earthquake some base stations were working, but the backhaul was not. One of the main goals is to enable communication between mobiles for 3 days (it took 3 days to get the backhaul up again).

The project considers more than just having some communication. For example, in the case of the government wanting to communicate with the citizens not only must there be a way to distribute the message but also a way for users to be able to authenticate the messages.

Notes:
- Feedback from 9/11 shows the amount of communication needed for basic services is small. But getting data from everybody and sharing that is expensive.
- Security and authentication might work against efficiency. It may be better to accept the noise and let the crowdsource information flow.
- Can there be better ways to summarize information? We don't need 10000 reports of something going wrong (or right).
- How do you prioritize traffic between he population and the government?

3.4 NetInf Update

Börje Ohlman (Ericsson Research – Stockholm, SE)

NetInf prepared a description of their project for live video streaming of the Ski World Cup Falun 2015. Ericsson worked on a "Virtual Arena" platform. It presented data and video streaming via web and other systems. The goal of the project was to see what NetInf technology brings to the table. Can it offer live streaming to Android as part of the virtual arena? This project is in the design stages but the plans are to open source the code. It's unclear what the success metrics for the project are at this point.

3.5 CCN 1.0 Update

Ignacio Solis (Xerox PARC – Palo Alto, US)

PARC has been working on CCN 1.0. It is a complete redesign of CCNx; both the protocol and the code has changed.

The base CCN protocol is the same. An interest message acts as a request for named data. Applications issue interests to retrieve named information. A content packet is a reply to this request. It carries a piece of named data. Names in CCN are a sequence of binary segments (or components). Each content packet (or content object) is named and contains a signature. This signature binds an identity, the name and the payload together into a

content object. When an application receives a content object it can check the validity by checking the signature.

Content Objects are network level elements, they are not the same as application level elements (like files). Large files are divided into chunks. Each chunk forms a content object. Chunks can be up to 64K in size but are normally the size of a regular network MTU (1500 bytes). Files are described using manifests. These manifests can be signed like other objects and list the set of content objects that make up the file. This list of content objects is created using self-certified names (hash based names of the content objects). This means that the content objects do not require a direct signature since they are implicitly signed from the manifest.

Changes from CCN 0.x to CCN 1.x:

- Packet format is now based on TLVs. 2+2 TLVs (2 bytes for T, 2 bytes for L).
- Packets now have a static header and some optional headers.
- Content Object matching is now done with exact match.
- Interests no longer have selectors.
- Interests can now have payloads.
- Both Interests and Content Objects have a unified format.
- The unified packet format contains the validation information at the end to make it modular.

The CCN 1.x software suite is being coded from scratch. It is written in C and uses modern software development practices. All code is unit tested and documented. It follows a specific coding style. Code complexity is low.

The current design uses a component based transport framework. This framework has various components that are used to instantiate stacks at runtime. Transport stacks can be configured to do different things. Some stacks deliver content in order and reliably. Other stacks might focus on pre-signed content, etc.

A set of APIs will be built on top of the transport framework. Applications will not need to interact with the network via sockets. The current plan is to have a KeyValue Store, a Messaging API and a streaming API. The native API currently used is called Portal. It allows applications to manually construct Interests and Content Objects.

3.6 CCN Lite

Christian Tschudin (Universität Basel, CH)

The goal of CCN Lite is to create a basic CCN forwarder in less than 1000 lines of C code. The current instantiation runs in user space and has basic interoperability with CCN 0.8 using the ccnb packet format. They are working on getting the capability to tunnel alien encoding formats. Code base is now part of the RIOT project (FU Berlin).

Christian is also working on Named Function Networking (NFN). This is a project to do lambda expressions over the network, basically asking the network to perform computation: "Network, please do this and give me the result".

4 Discussion Starters

We requested participants to submit various discussion starters for the breakout sessions. These discussion starters were presented to the group first to determine the most popular topics and to allow the participants to pick which topic they wanted to discuss.

Deployment and adoption

- Networking today is like making a phone call. With a socket we connect somewhere and then exchange what we want. That's changing, but to what? Should we include ICN names directly in the programming languages?
- People feel proprietary in that they want to know who got their data, and also want to know from whom they got the data, thus motivating the security functions we have today (argument for a connection model?)
- How could ICN, through some short-term adoption, provide value for content providers and users, and thus start deployment?
- We need a killer app – "Killer app" in the sense that it can kill your project: debugging, monitoring, traffic engineering, etc; e.g, ping, and other management tools; this is dissemination critical – we have to get that right!

Architecture scalability

- Scaling the overhead of locating a content in a network; if things move around you need ways to handle this; if you want to find the nearest copy, how to you find it and what control overhead does that result in? how do you synchronize state in different routers?
- What is the role of centralized services (Google) in finding content?
- What is the current feasible line-rate?
- What do we need to do cache-aware routing – architecture for optimizing routing, to find a copy close to the requester; can use DHT or name resolution system to locate near copies.

Domains and traffic

- How can we scale ICN over multiple administrative domains? what protocols are needed, policy mechanisms, etc?
- How can we increase the efficiency of handling multimedia and in particular video using ICN?
- One of the problems with user experience is the dual control loops that don't know about each other; a different way is to get rid of one of them, running the application directly on top of a simpler ICN service; what are the fundamental constraints the encoders have? how would that map to ICN without trying to mimic how we today do it on top of TCP?

5 Discussion Groups

With the discussion topics picked we proceeded to breakout sessions.

Scalability

- Routing systems – what's the benefit of caching?
- How to utilize the cache in the routing system? One way is to look in your local domain, then forward to a peer with whom you have a bus request, else forward to origin domain.
- Is there a need to have locations and name locations?
- Will the resolution system resemble the current DNS?
- What's the overhead of interdomain caching?
- Will systems advertise prefixes to business partners?
- NetInf naming could potentially be used – covers cache-aware location names, allows to advertise location and comment.
- What's the relation of scalability to content names?
- Forwarding plane scalability has been address before, so not a lot of discussion on this.
- Integrating caching and forwarding – Cisco says can support forwarding @ 20Gbps. Haven't figured how to check cache at the same time at that rate. Infocom paper this year about cached video.
- Verify signature on each packet at each hop seems clearly infeasible. Need to design in a way that gives realistic assumptions.

API

Many participants were at previous ICN workshops, felt we circled around questions that had been circled around before. [Dave Joke: loop detection but not loop suppression.]

Philosophical argument: which "PI" are we talking about? Facing the application or top of the network service. Embedded in the discussion is question of which applications exist, who is responsible for what aspects of managing the content.

Three examples talked through – all have "application" and network.

- Publisher ("Hollywood") has created a static object (movie). Their view is they want to put it into the cloud and forget about it – network will distribute it. People who are interested in watching it can issue request/interest in the movie and get the movie back. Caching helps efficiency, maybe performance. There is an argument about where is the storage in the network.
- "Dark content" – content that doesn't already exist when it is requested (e.g., credit card account balance report). Publisher only produces the report when it is asked for – but routing system needs something so the network knows how to route queries for the object. Publisher (bank) gets interest, generates the report.
- traveler @ airport, dumps photos @ airport before getting on plane home, wants local repo to get it to home while traveler is on airport. Again, publisher vanishes after "putting".

Discussed about incentives and financial considerations. Does the API have to deal with that? That led to discussion about controlling access (movie scenario) and watermarking. Who pays the "custodian" in the photo scenario. What exactly is a first-class service? Is storage a first-class service?

What is supported by every node? What is the "core service"? What is "native" and what is "built in", but not "native".

Video

- Current state – HLS/Dash-based over ICN. Decouple producer and consumer. What are the real benefits?
- Scenarios – Netflix (least challenging), live event w/synch, video conference.
- Use case: anticipatory Video streaming – network (cell) operator knows where users are and how many – benefit: control greediness of users; helpful in managing buffer behavior. Modified manifest of HLS (similar to CCN manifest). This is anathema of ICN, but also in line with ICN. Can you do a receiver-based mechanism that provides benefits of ICN?
- Use case: video conference system: can I fast fwd through parts of the meeting that I missed? Time shifting video conference. ICN has in-network caching; utilize cache.
- In some cases (secure multicast) with "forward secrecy" you should not be able to decrypt anything that happened before you joined.
- Use case: infrastructure-less. storage on mobile devices, video clips to share, combine network fabric and database. Baseline is very low; lots of potential for improvement.
- Use case: massive public event (stadium scenario). Compute fns at the edge (fog computing), virtualize network elements... Good example for ICN b/c upload capacity is the bottleneck.
- Scalable video? Microsoft lync video coding ideal for ICN. Caching multicasting gain. Fine grained enough for rate control?

  Key takeaways:
- ICN Video is a service of the network
- Allows better composition of different elements
- Would benefit from scalable coding
- Need to understand how ICN arch would affect video quality
- Interactive is hard!
- Synch is important issue
- Strategy layer would be useful [yes indeed]
- Content-dependent meta-info is important

Instrumentation

- What is the meta-architecture / philosophy behind instrumenting ICNs?
- Is there a distinction between inside the network to outside the network. (Different than policy permissions)
- Is there a difference between regular users and managers?
- Are we introducing security issues?
- Do we need more than one naming scheme?
- Are you allowed to talk to the cache system?
- Can users have external policies?
- What's the equivalent of ping in NDN/CCN; reach a namespace? reach anyone?
- Do we need host names for instrumentation purposes?
- Do you use relative or global names for nodes?
- Management might open attacks. We have to deal with those in the same way as we treat other types of attacks.
- Every element of the system has to be able to act as a provider to make it's state available.
- What is the topology in this system?
- What's the effect of multi-path?

Security

- scanning vs interest packets
- redirect questions
- ICMP redirect is different
- general concern re: Internet-based DoS attacks
- constraining interests
- make interest expensive or rate limit
- how to protect PIT?
- out of bound mechanisms (make it easy for IT department tofind you)
- banking, time constraints

Participants

- Bengt Ahlgren
Swedish Institute of Computer Science – Kista, SE
- Tohru Asami
University of Tokyo, JP
- Kenneth L. Calvert
University of Kentucky, US
- Antonio Carzaniga
University of Lugano, CH
- György Dán
KTH Royal Institute of Technology, SE
- Elwyn Davies
Trinity College Dublin, IE
- Anders Eriksson
Ericsson Res. – Stockholm, SE
- Suyong Eum
NICT – Tokyo, JP
- Kevin R. Fall
Carnegie Mellon University, US
- Xiaoming Fu
Universität Göttingen, DE
- Massimo Gallo
Bell Labs – Nozay, FR
- Ashish Gehani
SRI – Menlo Park, US
- Volker Hilt
Alcatel-Lucent – Stuttgart, DE

- Jussi Kangasharju
University of Helsinki, FI
- Holger Karl
Universität Paderborn, DE
- Vikas Kawadia
BBN Technologies – Cambridge, US
- Minyoung Kim
SRI – Menlo Park, US
- Dirk Kutscher
NEC Laboratories Europe – Heidelberg, DE
- Taekyoung Kwon
Seoul National University, KR
- Stefan Lederer
Alpen-Adria Universität & BITMOVIN – Klagenfurt, AT
- Eiichi Muramoto
Panasonic Corporation – Yokohama, JP
- Edith Ngai
Uppsala University, SE
- Börje Ohlman
Ericsson Res. – Stockholm, SE
- David Oran
Cisco Systems – San Jose, US
- Craig Partridge
BBN Technologies – Cambridge, US

- Diego Perino
Bell Labs – Nozay, FR
- Ioannis Psaras
University College London, GB
- Damien Saucez
INRIA Sophia Antipolis – Méditerranée, FR
- Thomas C. Schmidt
HAW – Hamburg, DE
- Glenn Scott
Xerox PARC – Palo Alto, US
- Jan Seedorf
NEC Laboratories Europe – Heidelberg, DE
- Ignacio Solis
Xerox PARC – Palo Alto, US
- Christian Tschudin
Universität Basel, CH
- Ersin Uzun
Xerox PARC – Palo Alto, US
- Matthias Wählisch
FU Berlin, DE
- Cedric Westphal
Huawei Technologies – Santa Clara, US
- George Xylomenos
Athens University of Economics and Business, GR