MULTIMEDIA COMMUNICATIONS TECHNICAL COMMITTEE
IEEE COMMUNICATIONS SOCIETY
http://www.comsoc.org/~mmc

E-LETTER

Vol. 8, No. 4, July 2013

IEEE COMMUNICATIONS SOCIETY

CONTENTS

Message from MMTC Chair

Dear Fellow MMTC Members,

I am very happy to announce the winners of the MMTC service and paper awards this year.

MMTC has two types of service awards. **The MMTC Distinguished Service Award** is given to an individual who has made significant contributions to MMTC as a core leader over a substantial number of years. **The MMTC Outstanding Leadership Award** is given to an individual who has made significant contributions to MMTC as a current or past IG Chair/Co-Chair or Board Director/Co-Director.

This year, we called for nominations of the service awards through the MMTC email list on April 28, and received six nominations by May 13. MMTC Award Board reviewed the nominations and voted for the final winners by the end of May. I am very happy to announce that the 2013 IEEE ComSoc MMTC Distinguished Service Award goes to Dr. Haohong Wang from TCL Research America, and the 2013 IEEE ComSoc MMTC Outstanding Leadership Award goes to Dr. Shiwen Mao from Auburn University.

Dr. Wang was the Editor-in-Chief of IEEE MMTC E-Letter during 2009-2010, and then served as the Chair of MMTC during 2010-2012. Dr. Mao served as the founding Chair of MMTC Cross-Layer IG during 2010-2012, and now is serving as the MMTC E-Letter Director. Both Dr. Wang and Dr. Mao have made significant contributions to the growth of MMTC, and I am very glad that the MMTC community recognizes their contributions.

Different from the MMTC service awards, MMTC best paper awards are open for nominations all year around. Any candidate paper need to go through a rigorous three-stage process: nomination, review, and award selection. In particular, the MMTC Review Board takes the lead of the review process, and selects award quality papers published in the bi-monthly MMTC Review Letter. The MMTC Award Board takes lead of the award selection process, by voting on the eligible papers published in the MMTC Review letter since the last award selection. Detailed procedure can be found at http://committees.comsoc.org/mmc/awards.asp.

This year, the Award Board voted on the Best Paper candidates from the MMTC Review Letter (since the October 2011 issue), which consists of 24 papers in the journal/magazine category and 12 papers in the conference category. I am very happy to announce that the **2013 IEEE ComSoc MMTC Best Journal Paper Award** goes to the following paper:

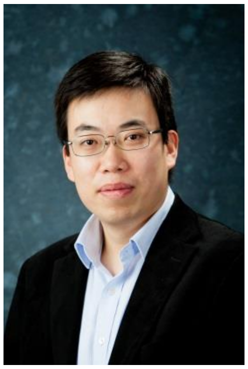
> P. Ndjiki-Nya, M. Koppel, D. Doshkov, H. Lakshman, P. Merkle, K. Muller, and T. Wiegand, “Depth Image-Based Rendering with Advanced Texture Synthesis for 3-D Video,” IEEE Transactions on Multimedia, vol. 13, no. 3, pp. 453-465, June 2011.

And the **2013 IEEE ComSoc MMTC Best Conference Paper Award** goes to the following paper:

> S. Shi, K. Nahrstedt, and R. Campbell, “Distortion Over Latency: Novel Metric for Measuring Interactive Performance of Remote Rendering Systems,” in Proc. of IEEE International Conference on Multimedia and Expo (ICME), pp. 1-7, July 2011.

I want to thank the Review Board and Award Board for the great services in terms of selecting the outstanding papers representing the latest impactful work in the MMTC community.

Regards,



Jianwei Huang

Chair, IEEE ComSoc Multimedia Communication Technical Committee

http://jianwei.ie.cuhk.edu.hk/

EMERGING TOPICS: SPECIAL ISSUE ON MULTIMEDIA SERVICES IN INFORMATION CENTRIC NETWORKS

Guest Editor: Luigi Alfredo Grieco, PhD
Department of Electrical and Information engineering (DEI)
Politecnico di Bari, Italy
a.grieco@poliba.it

Information Centric Networking (ICN) represents a novel way to look at the Future Internet, beyond the dominant IP-centric rationale. It is gaining the attention of researchers and practitioners worldwide, all interested in solving open issues raised by this revolutionary approach and designing the best ICN architecture that could gracefully (or not) replace the current Internet. Without any doubts, such research activities will be driven by the needs of multimedia services, which are expected to play a dominant role in next years. This is clearly reflected in the ICN literature, which strongly focuses on use cases based on multimedia contents.

The aim of this special issue of E-Letter is to shed some light on the recent progresses in the field of "*Multimedia Services in ICN*". It is the great honor of the editor to have six leading research groups, from both academia and industry laboratories, located in three different continents, that contributed at the definition of a clear, even yet not exhaustive, picture on the ebullient panorama of activities this special issue has been proposed for.

In the first article, titled "*Video Streaming over Named Data Networking*" by D. Kulinski, J. Burke, and L. Zhang, an overview of the *NDNVideo* tool is presented as a representative example of how content centric applications can be implemented in the pioneering *Named Data Networking* architecture.

The second contribution, titled "*Experimentation with large scale ICN multimedia services on the Internet made easy*" by A. Quereilhac, D. Camara, T. Turletti and W. Dabbous, focuses on the issues that arise when evaluating the performance of multimedia services in ICN and it proposes an experimental platform to ease such a kind of investigation.

With reference to Video on Demand and User Generated Contents, thoughtful design guidelines for sizing ICN networks are provided in the third letter, entitled "*Exploiting topology knowledge in Information Centric Networks: Guidelines and challenges*" by G. Rossini and D. Rossi.

The efforts of the previous letters are complemented in the fourth contribution that discusses issues and opportunities that arise when providing real-time TV services in NDN. It is titled "*Enabling real-time TV services in CCN networks*" by G. Piro and V. Ciancaglini.

The fifth paper, titled "*The Role of Virtualization in Information-centric Network Deployment*" has been contributed by J. Ren, K. Pentikousis, C. Westphal, W. Liu, and J. Wang. It presents a critical discussion on the importance that virtualization techniques can play on the deployment process of ICN architectures in real-life networks.

Last but not least, the sixth letter, titled "*The potential of Information Centric Networking in two illustrative use scenarios: mobile video delivery and network management in disaster situations*" by N. Blefari Melazzi and L. Chiariglione, presents the ICN architecture proposed within the FP7 EU project CONVERGENCE and discusses very relevant use cases related to this special issue.

While the present special issue is far from providing a full coverage of this hot research area, the six invited letters give an opportunity to address some relevant requirements linked to Multimedia Services in ICN and they may (hopefully) foster fruitful collaborations among different teams on common scientific targets. Finally, we would like to thank all the authors for their great contribution and the E-Letter Board for making such a special issue possible.

Luigi Alfredo Grieco (S'02–M'04–SM'12) received, with honors, the Italian Laurea Degree in Electronic Engineering in October 1999 from "Politecnico di Bari", Italy. Then, he received the Ph.D. in Information engineering on December 2003 from "Università di Lecce". During the years 2003-2004, he worked as research assistant at "Politecnico di Bari" with the support of a CNIT grant and of the research project PATTERN, founded by the Italian MIUR. Since January 2005, he holds an assistant professor position in Telecommunications at "Politecnico di Bari, Dip. Ing. Elettrica e dell'Informazione (DEI)". During the period March-June 2009 he has been visiting researcher at Planete group (INRIA - Sophia Antipolis, France), mainly working on modeling packet sampling techniques in Internet measurement systems. His main research interests include: congestion control, QoS in Wireless networks, energy efficient protocols in wireless ad hoc and sensor networks, industrial Internet of Things, real-time video processing, data aggregation, Internet monitoring infrastructures, and Information Centric Networks. Working within the frameworks of many regional, national, and inter-national research projects, he has authored more than 100 scientific papers about the afore-mentioned topics published on international journals and conference proceedings. Currently, he serves as Editor for the IEEE Transactions on Vehicular Technology (he has been awarded as top associate editor in 2012) and as Executive Editor for the Transactions on Emerging Telecommunications Technologies. He has been also involved in the organizing committees and technical committees for a number of IEEE Conferences.

Video Streaming over Named Data Networking

Derek Kulinski, UCLA Computer Science, kulinski@cs.ucla.edu
Jeff Burke, UCLA REMAP, jburke@remap.ucla.edu
Lixia Zhang, UCLA Computer Science, lixia@cs.ucla.edu

1. Introduction

Named Data Networking (NDN) is a proposed future Internet architecture that offers many advantages over TCP/IP [1], and holds significant promise for content distribution applications, such as video streaming. The TCP/IP architecture assigns IP addresses to hosts, making the Internet essentially a point-to-point communication system. NDN allows data consumers to retrieve desired content by directly using application-specified hierarchical data names, enabling a general-purpose distribution system. This approach, in combination with NDN's per-packet content signatures, permits any node in the network to cache named data packets and respond to requests for them. This is in sharp contrast to the current IP Internet, wherein a video producer sends data packets directly to every viewer, even when multiple viewers are watching the same video at the same time, and even when those consumers share the same upstream routers where the data could be easily cached.

Not only does NDN enable the use of storage in the network to cache popular data that are frequently requested by multiple users—reducing bandwidth and improving performance—it also enables video producers to easily provide a variety of other functions. For example, producers can assign meaningful names to video data (e.g., timecode frame indexes using a well-established naming convention), so video consumers can simply request specific video content by names to seek (rewind or fast forward). Because of such properties, video streaming can benefit significantly from NDN.

In fact, video streaming has already received considerable attention in the CCN/NDN research community. Early developments include the CCNx VLC plug-in [2] and GStreamer plug-in [3] as test applications. A number of more recent research efforts have produced additional results. For example, Xu et al. compared HTTP live streaming with a CCNx-based approach on Android [8]. Others considered how devices can collaborate to share bandwidth for the same video [9] and rate adaptation [10], as well as additional topics in the context of NDN.

This paper provides an overview of NDNVideo, a complete software solution developed at UCLA for video and audio streaming over NDN that serves as a representative example of how content-centric applications can be implemented in this new architecture. NDNVideo takes advantage of NDN's features to provide highly scalable, random-access video from live and pre-recorded sources using a straightforward data publisher and consumer model without connection negotiation or session semantics. The application was built using PyCCN, Python bindings that we created for the CCNx library and software router by PARC [4, 7].

2. Design Goals

NDNVideo was designed to support live and pre-recorded video streaming with frame-accurate random access, while integrating persistent storage of live content and, eventually, enabling synchronized playback across multiple consumers. Driver applications are web video streaming services as well as applications in digital signage, live events, and professional media production.

Our design goals included 1) quality consistent with current Internet video expectations; 2) simple, low-latency random access into streams, based on actual location in the video stream (by video frame), using a timecode-based namespace (i.e., the hour, minute, second, and frame of the stream); 3) consumer-side synchronization of streams using the timecode-based namespace (for multiple consumers and future digital signage, multi-camera, and interactive applications), and scalability without impact on the original source of video; and 4) on-the-fly archival of live streams, making them indistinguishable from pre-recorded streams for the purposes of most playback applications. Additionally, NDN's per-packet signatures on ContentObjects (data packets) provided a starting point for content verification and provenance in video applications, and we employ them in this application.

We use the GStreamer framework for capturing and rendering video. A GStreamer based application for video streaming was developed in [3], but treats video as an arbitrary binary stream of data, and consequently does not fully use the potential of the NDN architecture. Their approach is similar to streaming video by downloading a file over the HTTP protocol, as opposed to using protocols that leverage the content type, such as RTP. In NDNVideo, we develop a namespace and protocol specifically for streaming video and audio.

3. Architecture

Overview. The NDNVideo protocol has two types of participants: publisher and consumer. The streaming relationship is one-to-many (i.e. a single publisher publishes data that is received by many consumers). Unlike IP, NDN is pull-based, in which data consumers issue request packets, or "Interests" that indicate the

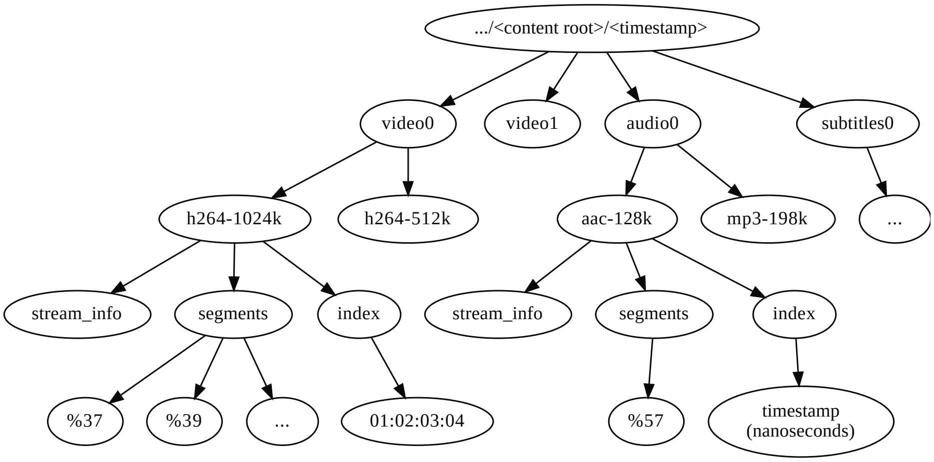


Figure 1. NDNVideo namespace, providing multiple encoding rates and both time and segment-based data access.

name of the content that they wish to receive. Any entity on the network may respond with a corresponding data packet. This request/response approach enables the video publisher and consumers to utilize the Interest-Data exchange of the NDN protocol directly. NDNVideo requires no direct interaction between consumer and producer; the publisher simply prepares data packets, signs them, and stores them in a repository (persistent store) for retrieval by consumers. Consumers no longer need to establish a session with the publisher, nor inform the publisher about their Quality of Service requirements. Instead, they are in full control of how much data they receive, and at what rate. If, for any reason, a consumer needs to upgrade or downgrade its bit rate, it can do so seamlessly by requesting data from a different namespace for the same timecode in the video.

Namespace. Like all NDN applications, a fundamental facet of NDNVideo's design is its namespace, illustrated in Figure 1. During packet preparation, the video producer segments the stream using semantically meaningful names, e.g. frames for video and samples for audio, organized by bitrates and encodings that can be enumerated by the consumer via a metadata file to list all children nodes in a namespace tree. Such structuring of data provides many benefits, one of which is enabling the publisher to uniquely name every frame and, in turn, allows the consumer to easily seek to a specific place in the stream. A key design goal was to provide this random-access via video timecode, in which video is indexed by frames (e.g., HH:MM:SS:FF). To allow more efficient playback after seeking, the data is also provided using consecutive segment names.

Encoding and Packet format. NDNVideo supports all of the many encoding formats supported by GStreamer, although our implementation focuses on H.264 video and MP3 audio. The NDNVideo publisher generates signed ContentObjects from an input video stream, which can be either live or pre-recorded, and places them immediately into a local repository—a persistent, disk-based storage that is another fundamental component in the NDN architecture. Consumers' data requests go to that repository or are otherwise satisfied by caches in the network.

NDNVideo handles large (video) and small (audio) samples by packing data in two layers. The inner layer contains all information necessary for the playback of a data buffer, such as timestamp, duration, and length in bytes. The timestamp tells the player at what point in time the buffer is supposed to be played, while the duration tells how long, making NDNVideo compatible with variable frame rate codecs. Then, multiple buffers are put inside one packet by trying to fill it completely. If a buffer does not fit into one packet, it is split into multiple packets. The outer layer of the packet contains two additional fields: offset and count. The count indicates how many buffers begin in a given packet, and the offset is used to tell where the first buffer starts. The offset is used only when there is a packet loss, and it lets the consumer quickly resume processing from the next available buffer.

Random access. A player seeks in NDNVideo by simply issuing an Interest in the index namespace corresponding to the desired timecode (e.g., "HH:MM:SS:FF"), with Interest "selector fields" set to return the nearest keyframe[1] ContentObject, which in turn maps timecode to segment number using a simple ASCII text payload. NDN's Interest selector fields enable a consumer to express requests that are more sophisticated than just name prefixes; they are described in detail in [5]. The *ChildSelector* is set to RIGHTMOST, instructing the responding node to return the last element it has in the index namespace, and *AnswerOriginKind* is set to NONE, to tell the

---

[1] The publisher indexes only keyframes, since the closest preceding keyframe is needed to properly decode a specific frame of the video.

library to fetch content from the network rather than any local cache. Finally, the *Exclusion Filter* is used to tell the network to return the latest index entry, just before the desired frame. (This requires timecode be expressed in a format that sorts properly according to the NDN architecture.) For example, to seek to 00:00:05:00, the Interest excludes all indexes after this point and, through the rightmost child selector, requests the next largest index. Thus, nodes on the network will respond with the nearest keyframe ContentObject they have, without any direct negotiation with the publisher.

5. Live Streaming

With NDN, the video publisher is far simpler than the corresponding one in IP; even for live streaming, it simply puts frames in a network-addressable content repository immediately after capture. Cooperation between publisher and consumer to maintain QoS (as done in RTP) is no longer needed. In the NDN case, the consumer knows exactly whether any data packet is lost, and there is no need to inform the publisher of how fast it needs to send the data, since it controls its own data fetch. However, some complexity is shifted to the stream consumer for certain cases, such as live streaming. The video consumer needs to pipeline Interest packets so that the data are fetched continuously as they are produced. However, it should not fetch the data too quickly and request segments that do not yet exist. To address this challenge, the player must determine what is the latest data and at which rate it should be requested.

When an NDN node receives an Interest, it does a longest-prefix match, also incorporating specified selectors, to see if it can be answered with data contained in its Content Store, only forwarding Interests to neighboring nodes when it cannot satisfy the request. Because different nodes can have different data in their Content Stores, the same request might result in different responses from different nodes. To accommodate this behavior, the consumer first issues Interests periodically during its video playback to determine the ever-increasing duration of the stream, by checking for the latest entry in the index namespace. It sets the exclusion filter in the Interest packet to only request Data packets with index name components *greater* than the last index it has seen. This forces the node providing the previous response to forward the Interest to its peers. Without this parameter, the consumer's Interest would retrieve previously received content cached in the network. Different nodes may respond with their own notion of what is “latest,” so these Interests must be issued more than once. In the worst case, this approach may take N queries, where N is number of connecting nodes accessed, before converging on the correct “latest position.”

Second, to enable the consumer to determine the rate at which it should issue Interests, in addition to the timestamp of the buffer, each individual packet also contains a local time at which the packet was generated. While the consumer may or may not have a clock in sync with the publisher, this information is still useful, because it can be used to calculate the time difference between packets. The consumer can then estimate the mean time interval and dynamically adjust the rate of Interests after starting with the latest segment of the live stream previously determined.

If the player does not receive data fast enough to play back at the correct rate, instead of pausing the playback, it skips to the most recent segment and continues playback from there. To do so, it calculates the local time at which given content is supposed to be played back, as well as determining whether to send the data to the decoder or not.

Compared to the complexities in live video stream seeking and pipelining, archival/recorded playback is more straightforward; the video’s length is fixed. The only additional information needed for streaming from a pre-recorded file (or a live stream that is already completed) is a marker of the end of the stream. By convention, the publisher sets the value of the FinalBlockID field in the ContentObjects to the last segment number to signal the end of the stream. If the player uses multiple streams (e.g. audio and video), it stops playback after receiving the EOS (End of Stream) signal from all the streams.

6. Handling Packet Loss

Given the pull-based nature of NDN, NDNVideo lets the consumer be fully in charge of data it is receiving. Each data packet is named with the segment name plus segment numbers to make the data names predictable and enable the consumer to pipeline requests for the data. This is necessary to provide sufficient playback quality, especially when latency between the publisher and the consumer is high. If a single buffer is contained in multiple packets, the packet header information is used to put the buffer back together. In case of packet loss, the consumer can either request the same data again or issue an Interest for the next segment. If the segment is considered lost, the offset field is used to determine the point at which the next buffer starts. The code does not need to wait for the buffers that start at the beginning of packets (e.g., those of keyframes).

In an ideal case, the consumer will get responses to all the Interests it issues. Unfortunately, packets can be dropped due to network congestion or other causes. In order to provide seamless playback, it is important for the consumer to know how long to wait before assuming that an Interest or the corresponding data packet is lost. Interests for the data can then be quickly reissued or assumed unavailable, and the consumer can move on to the next segment. The NDNVideo consumer adjusts its Interest timeout based on previous RTT values, smoothed using a low pass filter similar to what is defined in RFC 2988 for TCP.

7. Implementation and Testing

The implementation is written in Python, and uses the GStreamer [6] multimedia framework. It employs PARC's CCNx [4] implementation of the NDN architecture and our PyCCN [7] bindings. Both NDNVideo and PyCCN are open source (as are CCNx and GStreamer) and can be retrieved from GitHub.

After a variety of tests using pre-recorded files and live sources, the system was demonstrated "live" to a large audience in March 2012. A live, standard definition H.264-encoded stream (@ 1Mbit/sec) from a musical performance in the UCLA School of Theater, Film and Television's TV Studio #1 was published over the NDN testbed to our Washington University in St. Louis collaborators' demonstration for the GENI Engineering Conference in Los Angeles. Broadcast quality audio and video feeds from three cameras were mixed live and published to a CCN repo at UCLA. The WUSTL team displayed the video using the NDNVideo player. In this and other tests with standard definition, H.264 video, the streaming works well for end-users. We have recently added support for high-definition ("1080p") resolution. Additionally, we have deployed webcams connected to application servers at several geographic locations on the NDN testbed, which use NDNVideo.

8. Current issues and future work

The interval-based pipeline is being refined for deployment in the next series of demonstrations and tests. Additionally, the CCNx repository does not yet support deletion of specific data objects, which can be problematic for long-running live streams; this will be addressed in future versions.

Finally, we plan to enable the consumer to switch codec based on bandwidth or performance. For example, when the consumer detects that it cannot receive data at the desired rate, it could downgrade playback to a lower bit rate by simply changing a component of the Interest name it is requesting. An elegant way to do this would be to provide H.264 Scalable Video Coding (SVC) or a similar solution with enhancement layers expressed directly in the video namespace. We are exploring this solution.

9. Conclusion

We designed, implemented, and tested NDNVideo, a video streaming application that was conceived with NDN architecture in mind and demonstrates some immediate advantages of NDN. The protocol and namespace are equivalent for live and pre-recorded streams, requiring only additional logic at the consumer for live streaming. Reliable and rate-adaptive playback can be provided with no session semantics or negotiation necessary between the consumer and the producer. The approach leverages Content Stores in the network, which makes the video distribution more efficient. Video and audio streaming uses the intrinsic features of the architecture to scale without loading the publisher, and to provide efficient random-access, even to live streams. We believe that NDN-based streaming can enable a better user experience with less strain on the publisher when compared to TCP/IP, and that the approach could be used to enable serverless video publishing from resource-constrained mobile devices.

References

[1] Zhang, L., *et al.*, "Named data networking (ndn) project," http://named-data.net/techreport/TR001ndn-proj.pdf, 2010.

[2] "CCNx vlc plugin," part of the CCNx package. [Online]. Available: https://github.com/ProjectCCNx/

[3] Letourneau, J. "CCNxGST - GStreamer plugin used to transport media traffic over a CCNx network." [Online]. Available: https://github.com/johnlet/gstreamer-ccnx

[4] "CCNx." [Online]. Available: http://www.ccnx.org

[5] CCNx Technical documentation. [Online]. http://www.ccnx.org/releases/latest/doc/index.html

[6] "GStreamer - open source multimedia framework." [Online]. Available: http://gstreamer.freedesktop.org

[7] "PyCCN - python bindings for CCNx." [Online]. Available: https://github.com/remap/PyCCN

[8] Xu, H., et al. "Live Streaming with Content Centric Networking." *IEEE 3rd Intl. Conf. on Networking and Distributed Computing (ICNDC), 2012*.

[9] Detti, A., et al. "Offloading cellular networks with Information-Centric Networking: The case of video streaming," *IEEE WOMOM 2012*, June, 2012.

[10] Han, B. et al. "AMVS-NDN: Adaptive Mobile Video Streaming and Sharing in Wireless Named Data Networking," *IEEE NOMEN 2013*, April 19, 2013.

Derek Kulinski is a Systems Engineer at Edmunds.com. He received his MS in Computer Science from UCLA, where he participated in research at the Center for Embedded Networked Sensing and on Named Data Networking.

Jeff Burke is the Director of Technology Research Initiatives for the UCLA School of Theater, Film and Television and the Executive Director of the UCLA Center for Research in Engineering, Media and Performance. Jeff is the application team lead for NDN.

Lixia Zhang is a professor in the UCLA Computer Science Department. She previously served as vice chair of ACM SIGCOMM, member of the editorial board for the IEEE/ACM Transactions on Networking, member of the Internet Architecture Board, and co-chair of the Routing Research Group under IRTF. She is a fellow of ACM and IEEE, and holds the UCLA Postel Chair in Computer Science.

Experimentation with Large Scale ICN Multimedia Services on the Internet Made Easy

Alina Quereilhac, Daniel Camara, Thierry Turletti and Walid Dabbous
INRIA Sophia Antipolis, France
{*alina.quereilhac,daniel.camara,thierry.turletti,walid.dabbous*}*@inria.fr*

1. Introduction

The Information Centric Networking (ICN) paradigm is gaining attention among network researchers and operators as an alternative to share content more efficiently on the Internet. In the ICN paradigm, content naming is independent from physical server location, and since content can be cached at intermediate hops in the network, it can be retrieved from the closest available cache, lowering delay and reducing redundant traffic.

A shift towards an ICN Internet architecture can particularly impact current multimedia traffic patterns, such as video traffic, which represents an increasing fraction of the traffic on the Internet today [1]. Since it is not feasible to envision a clean slate replacement of the current Internet architecture, ICN solutions deployed by Internet operators will coexist with TCP/IP technologies at least in the short to medium term. For this reason, it is crucial to evaluate and understand the behavior of ICN solutions in realistic Internet environments through prior experimentation.

In this paper we present a framework for evaluating ICN solutions in general, and multimedia solutions in particular. This framework simplifies the challenges of conducting large scale experiments on the wild Internet. We leverage on the existing PlanetLab [2] testbed to provide worldwide distributed access to the Internet at minimum cost, and propose the NEPI [3] tool to simplify the design and deployment of experiments. As a means of illustrating the capabilities of the framework, we consider an example experiment in which we evaluate the performance of broadcasting video to over 100 consumers using CCNx [4], against a classical client-server solution.

2. Evaluation framework

The difficulty to realistically simulate the Internet traffic [5] makes simulation environments not sufficient to evaluate the behavior of ICN technologies to be deployed on the Internet. Furthermore, the simplifications linked to simulation environments can hide important issues related to the performance and feasibility of the solution under study. While probably more realistic to reflect performance issues, dedicated or private testbeds can hardly mimic the complexity and diversity of traffic found on the Internet.

For these reasons, one interesting approach is to conduct experiments on the Internet itself. However, for large scale experiments this requires having access to a large number of nodes. The cost of using Amazon EC2 [6] clouds or other paying services might limit the extension to which ICN solutions can be evaluated and affect the transparency of the experiments, since clouds present restricted node management interfaces.

Taking into account these considerations, PlanetLab [2] presents itself as a good alternative for large scale evaluation of ICN technologies to be deployed on the Internet for two main reasons. The first one is that access to the testbed is free for members of PlanetLab partner institutions. The second one is that PlanetLab nodes can be chosen from locations all around the world to best suit the needs of each experiment scenario. However these features have a tradeoff cost: PlanetLab provides a best effort service which means that nodes can be down or irresponsive for long periods of time, and seemly healthy nodes may have broken configurations. These issues add up to the already difficult task of synchronizing large experiment deployments on highly distributed and unpredictable environments such as the Internet.

To alleviate this complexity we developed NEPI [3], network experimentation programming interface. NEPI is an open source tool which provides an experiment description language to design network experiments, describing both topology and applications, and a controller entity to automatically deploy those experiments on target experimentation environments, such as PlanetLab. The controller entity is capable of collecting result files during the experiment execution to a local directory. NEPI also allows to specify node selection filters while designing the experiment, which permits to automatically discover and provision PlanetLab nodes during experiment deployment, without the user having to hand-pick them. NEPI takes into account node health metrics, exposed through the PlanetLab API, to choose the best suited available nodes, and discards unresponsive nodes, blacklisting them to be ignored for future experiments.

For applications, NEPI provides the possibility to upload arbitrary sources, including user modified source code and input files (e.g. video files), to PlanetLab nodes, and gives the possibility to specify custom compilation and installation instructions in the experiment description. It automates installation of package dependencies for application compilation and execution through the yum package manager [7]. This ability to run arbitrary applications, in a very customizable way on PlanetLab nodes, makes it an ideal tool to experiment with ICN technologies on the Internet. NEPI can be used to conduct experiments in two ways: (i) through its graphical user interface, which allows designing experiments by dragging and dropping components, and connecting them on a canvas, or (ii) through a Python script.

3. Example experiment scenario

In this section we exemplify how our framework can be used to conduct large video broadcasting experiments on the Internet, to evaluate ICN technologies. There are currently several active projects working to develop efficient ICN architectures for the future Internet, such as PSIRP, NETInf and CCN [8], to mention a few. For the purpose of this example we chose to evaluate the network performance of the CCNx software, an implementation of CCN [9], against the popular VLC media player [10], when broadcasting a video from a single source to 100 consumers distributed over 12 countries in Europe. VLC supports video broadcasting over the Internet using point-to-point transmissions. We used RTP to stream a 20 minutes long, 320x240, 658kbps, H.264 encoded video to the clients. Figure 1 shows the two-level topology used for this experiment.

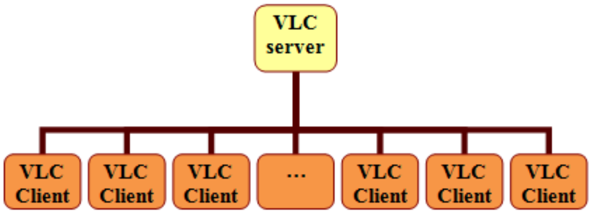


Figure 1. VLC experiment topology

We then used the same video to conduct another experiment with the CCNx software (version 0.7.1). To better exploit the advantages of the content cache, we designed a three-level tree topology where leaf nodes were connected to an intermediate node in the same country, through unicast UDP FIB entries. In turn, intermediate nodes were directly connected to the root node in the same way. All nodes ran a CCNx daemon (*ccnd*), and on the root node the *ccnseqwritter* application was used to publish the video in a local CCNx repository. The 100 leaf nodes retrieved the video using the *ccncat* application. Figure 2 shows the topology chosen for this experiment.

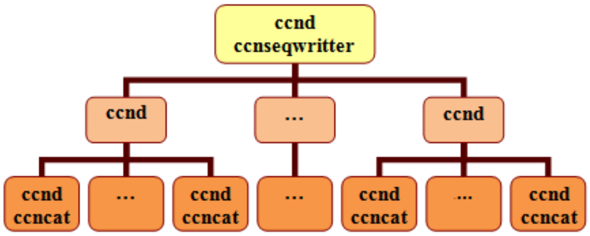


Figure 2. CCNx experiment scenario three-level tree topology on the bottom

In both experiments we made all clients start retrieving the stream simultaneously, and used *tcpdump* to measure the amount of traffic sent from the root node, and received on the leaf nodes. We hand-picked the root and the leaf nodes, and used the same ones for both experiments to ensure comparable results.

Designing and running these experiments took only days with NEPI, while implementing from scratch a script or program to perform the same experiment would have taken several weeks. The scripts used to run the experiments are publicly available on the NEPI source code directory in the "examples/streaming" folder. Instructions on how to download the source code are available at NEPI web page at *http://nepi.inria.fr* .

4. Results

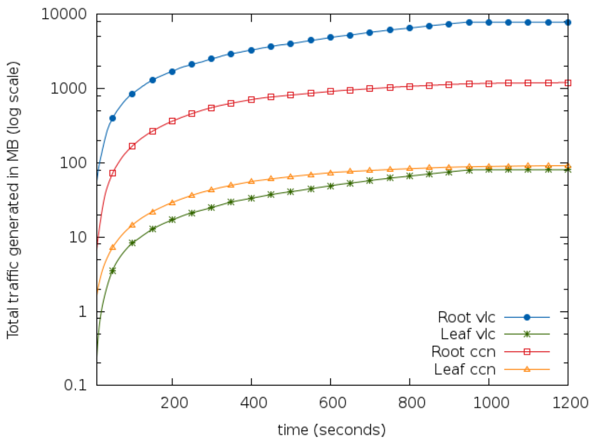


Figure 3. Total traffic generated on root and leaf nodes.

Figure 3 shows the results obtained from this experiment. As expected, we see that the VLC root node generates more traffic (7 times more) than the CCNx root node. The VLC root sends the entire video one time per each client, while the CCNx root node sends the video only one time per intermediate node.

However, what strikes as interesting is that the total traffic received in average per leaf node is bigger for CCNx than for VLC, which can be explained by the CCNx protocol overhead.

NEPI was able to retrieve more than 300 results files from the remote nodes automatically. We had to run the NEPI scripts several times in order to get the experiments running, since many times nodes would fail during installation, or SSH connections to the nodes will not respond. However, NEPI managed to detect problems during the deployment face and rapidly finish the experiment providing an error log.

5. Conclusion

In this paper we presented a framework, favorable for the ICN research community, to simplify the task of conducting large ICN experiments on the Internet. The framework is based on two core components, the PlanetLab testbed and the NEPI experiment management tool. We have provided an example showing the usage of the framework to conduct experiments involving over 100 nodes, deploying complex applications and collecting results in a highly customizable way.

References

[1] C. Labovitz, S. Iekel-Johnson, D. McPherson, J. Oberheide, and F. Jahanian. Internet inter-domain traffic. In *Proceedings of ACM SIGCOMM 2010*.

[2] A. Bavier, M. Bowman, B. Chun, D. Culler, S. Karlin, S. Muir, L. Peterson, T. Roscoe, T. Spalink, and M. Wawrzoniak, Operating System Support for Planetary-Scale Network Services, In *Proceedings of NSDI 2004*.

[3] C. Freire, A. Quereilhac, T. Turletti, and W. Dabbous. Automated deployment and customization of routing overlays on planetlab. In *Testbeds and Research Infrastructure. Development of Networks and Communities.* Spinger vol 44, 2012

[4] CCNx http://www.ccnx.org/

[5] Floyd, Sally, and Vern Paxson. Difficulties in simulating the Internet. *IEEE/ACM Transactions on Networking TON*, volume 9, issue 4, 2001

[6] Amazon EC2. http://aws.amazon.com/ec2/

[7] YUM Package Manager. http://yum.baseurl.org/

[8] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher, and B. Ohlman. A survey of information-centric networking (draft). IEEE Communications Magazine, volume 50, issue 7, 2012

[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 CoNEXT 2009*

[10] VLC media player. http://www.videolan.org/

Alina Quereilhac holds a degree in informatics engineering from the University of Buenos Aires, Argentina, received in 2008. She is currently a Ph.D student at the University of Nice and research engineer at the Diana team at INRIA, Sophia Antipolis. The main axes of her research include simulated and live networking evaluation environments, testbed architecture, and networking experiment control.

Dr. Daniel Camara holds a BSc in Computer Science from the Federal University of Paraná, Brazil an MSc and a PhD in Computer Science from the Federal University of Minas Gerais, Brazil. Currently Daniel is working as a research engineer at INRIA Sophia Antipolis. His research interests include wireless networks, distributed systems, artificial intelligence and quality of software.

Dr. Thierry Turletti is a senior research scientist at the Diana team at INRIA Sophia Antipolis. He received the M.S. (1990) and the Ph.D. (1995) degrees in computer science from the University of Nice - Sophia Antipolis, France. He has done his PhD studies in the RODEO group at INRIA where he designed the Inria Videoconferencing System (IVS). During the year 1995-96, he was a postdoctoral fellow in the Telemedia, Networks and Systems group at LCS, MIT and worked in the area of Software Defined Radio (SDR). His current research interests include information centric network architectures, trustable network evaluation platforms and wireless networking.

Dr. Walid Dabbous is a senior researcher at INRIA where he leads the DIANA research team on networking. He received his Doctorat d'Université from the University of Paris XI in 1991. He participated to several FP7 research projects such as: Muse, E-NEXT, OneLab and OpenLab. He is also involved in the French Equipment of Excellence platform project FIT(Future Internet of the Things). His current research interests include: Information Centric Networking Protocols, Networking Experimental Platforms and Simulators and P2P systems performance and privacy.

.

Exploiting Topology Knowledge in Information Centric Networks: Guidelines and Challenges

Giuseppe Rossini and Dario Rossi
Telecom ParisTech, Paris, France
{name.surname}@telecom-paristech.fr

1. Introduction

The key observation laying behind the advent of Information Centric Networking (ICN)[2] is that nowadays Internet users are more and more interested in multimedia contents (e.g., VoD, or UGC), while the interest for host to host communication tends to disappear. Indeed, applications like YouTube, Netflix, Daily Motion, require smarter mechanisms for distributing contents over the network than actual infrastructures (e.g., CDN), and find in ICN the best fit for their needs. Notwithstanding, this new design comes up with new and interesting challenges.

Part of these challenges lay in the design, planning and operation of ICN. While all architectures are unique in some aspects [2], however all ICN proposals agree on the central role that *caching* has in the design--which makes thus sense to consider as primary aspect of ICN. Furthermore, the majority of proposals agree in considering an ICN network as *a receiver-driven network of caches*--from which it follows that practical guidelines concerning the caching aspect that we address in this paper will be useful irrespectively of the underlying ICN technology.

At the same time, we also point out that among the many ICN proposals, the Content Centric Networking (CCN) [7] approach has raised significant interest from the scientific community. As such, and since a conceptual unifying framework for ICN is still a work in progress at ICN Research Group (ICNRG) of the Internet Research Task Force (IRTF), in the following we place our work in the context of CCN, and adopt the CCN terminology for the sake of readability. Briefly, as a receiver-oriented network, CCN clients request data sending *interests* for named contents. Request are *forwarded* hop-by-hop toward a permanent copy of the requested data: at each interest CCN nodes perform lookup for content names in a *Forwarding Information Base* (FIB), that stores the set of interfaces through which any given content can be reached. Depending on the routing protocol running over the network, FIB can store multiple paths to a given content: in this case, a *Strategy Layer* is responsible for the selection of one (or more) next hop interfaces among the set of possible ones. CCN routes along this path may possess cached copies of the content of interest within their own *Content Store*: in this case, interests need not to reach the permanent copy stored at the repository, and data can be generated in reply directly from the temporarily cached copy. Data travels back toward the requester following a trail of bread-crumbs, that are stored in a *Pending Interest Table* (PIT) in CCN jargon.

ICN task	Example task	Timescale
Cache planning	Dimensioning	Off-line
Control plane	Routing	Minutes to hours
Data plane	Forwarding	nsec to msec
Cache management	Replacement policies	nsec to msec

Table 1. ICN design space

From a high level viewpoint, CCN design and operational choices pertains to the different categories listed in Tab. 1. At the longest timescale (offline), planning decisions involves tasks such as defining the ICN topology, and dimensioning the network equipment. In CCN, additionally to the link bandwidth, the amount of storage space allocated to each node in the topology is expected to play a paramount role. At shorter timescale (minutes to hours), the control plane will periodically advertise name prefixes so to update FIB information: the choice and configuration of a routing protocol will in this case be of primary importance. Other CCN operations happen at even shorter timescales (nsec to msec, depending on the link capacity) like forwarding decisions taken by the strategy layer upon handling of every interest packet, or cache decision and replacement policies taken in the content store upon handling of every data packet.

Clearly, the overall performance of the CCN architecture will depend on the above critical decisions.

2. CCN design space

Network planning.

Network planning and device design is an offline task. The goal of the planning activity is dimensioning links, load, storage, and so forth. As previously outlined, the main resource to distribute among nodes in an CCN network, is the content store space. This is quite different with respect to a traditional IP network, where the principal resource is typically the link bandwidth.

Another important aspect concerns the design of hardware devices, as technology limits impose a tradeoff between caches size and lookup speed. Indeed, having large caches increases the hit rate of the single cache, but poses severe constraints in terms of memory access time [9]. As rule of thumb, for core routers [3],

[9] points out than tens of GB are about nowadays technological limit due to line speed requirement.

In other words, CCN routers must be able to service each interest packet by doing a lookup for content in real time (similarly to IP longest prefix matching lookup for addresses). Hence, memory access speed (and cost) limits the practically achievable content store size, unless some hierarchical scheme is used to move data back and forth between slow and fast memory (as it is done in L2 and L3 caches of nowadays CPUs). This is an important point, on which we will get back in Sec. 4.

Control Plane.

As in traditional host-centric architectures, routing represents one of the main control plane tasks. In contrast with traditional architectures, CCN provides native support for multipath routing. As a result, FIB entries may contain one or multiple interfaces for each permanent copies--in which case the routing protocol should advertise all the different permanent copies of the same content.

Distributed control plane protocols can use two kind of information: on the one hand, they can exploit topological information, (e.g., as in OSPF); on the other hand, they can leverage administrative information and policies (e.g., as in BGP). To date, a single routing protocol has been proposed for CCN, namely OSPFn [14].

Notice also that, with few exceptions [15], control plane disseminates only information concerning permanent copies--conversely, the task of discovering temporary copies is usually implemented in the data plane that we discuss next.

Data Plane.

Data plane handles forwarding of interest and data packets. Strategy layer can either exploit FIB information pertaining permanent copies advertised in the control plane [13], or assume no FIB knowledge and explore the network to find unadvertised temporary copies [4],[16].

In the latter case, both [4], [16], show that a degree of exploration in the data plane is beneficial in terms of system performance, and topological information can assist the distributed decision process.

Interest and data forwarding tasks also imply fast lookup on FIB and PIT structures--which is a critical component of the infrastructure, that cannot however be enhanced via topological information. As the whole catalog may be composed by billions of objects, further divided in multiple chunks each of which need to be individually looked up, efficient data structures (e.g., bloom filters) should be used to “implement fast, space-efficient and cost-friendly PIT tables” [9].

Cache management.

Finally, another critical piece of the CCN architecture is constituted by the content store. Furthermore, in the case of a network of caches, the central issue ceases to be the cache *replacement* policy, and rather becomes the cache *decision* policy [3], [8], [10], [11]. Notice indeed that the latter may represent an *implicit coordination* mechanism between multiple caches--which is important since explicit coordination may not be feasible in reason of the overhead associated.

Simple randomized criterion to avoid content to be replicated everywhere are proposed in [3], while [5], [10], [11] more explicitly bias the cache decision process based on some network properties. In more details, WAVE [5] is inspired by Leave a Copy Down (LCD) [8], where the content ``moves" only one hop down the hierarchy at each new requests. Yet, [10], [11] even more explicitly take into account network topology properties, such as the Betweenness Centrality (BC), to explicitly take into account nodes position in the network. While [5],[10],[11] address an interesting area of the CCN design space, doubts are raised in [9] so as to whether such complex decision policies can be supported in hardware at high speed.

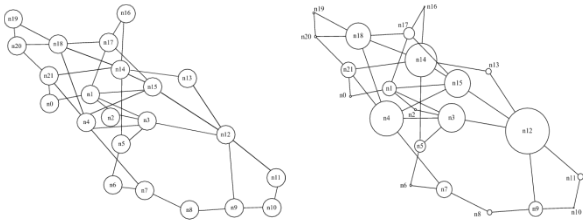
3. Topology-aware ICN design.

With reference to Tab. 1 , topological information can be exploited at multiple layers, and by multiple planes. Clearly, the main task of routing protocols in the control plane is to disseminate topological information to build FIB, that are later exploited by the strategy layer. At the same time, also offline ICN planning decision [12] and online cache management operations [5],[10],[11] can possibly exploit topological information. In the reminder of this section, we illustrate the case of topology-aware ICN planning[12].

Problem at a glance.

Suppose that an ISP deploying CCN, invests in a given amount of Content Store memory. Irrespectively of the exact amount, the network planning team has now the option to equally distribut the memory among all nodes (homogeneous sizing). Alternatively, the team can identify some *central* nodes in the topology, and decide to allocate them *more memory* with respect to the others. This planning activity may be run off-line, for instance each time the physical topology changes.

For the sake of example, Fig. 1 shows the



(a) (b)

Figure 1. Pictorial representation of cache sizing. The picture reports (a) homogeneous and (b) heterogeneous cache size (proportional to the betweenness centrality index).

aforementioned idea on the Geant topology: in Fig. 1(a), content store memory is equally split among all nodes; in Fig. 2(b), instead, memory is distributed proportionally to the node Betweenness Centrality (BC) index. In [12], we evaluate several centrality metrics, the simplest of which being the Degree Centrality (DC), defined as the number of links of any given node. Interestingly, we find that this simple metric proves to be a sufficiently good allocation criterion, even compared to more complex metrics.

This is a positive finding, as it implies a rather simple rule of thumb for sizing the content store: i.d., “if you add a line card to a CCN router, add some content store memory as well”. In the following, we report a subset of interest findings from [12], to which we refer the reader for further technical details.

Problem formulation.

More formally, let us define C_{tot} as the overall size of cache in the network. In the case of homogeneous network, we fix the size of the individual caches to $C_i=10$ GB [3]. In the case of homogeneous networks, $C_{tot} = |V|C_i$ with $|V|$ the number of nodes in the network. In the case of heterogeneous networks, we exploit the centrality scores as follows. Consider a generic graph centrality $X \in \{CC,GC,DC,EC,SC,BC\}$, where we denote by $X(i)$ the value of X for node $i \in V$ for the considered topology.

We then adopt two criteria for cache sizing:

$$C^{P_{X(i)}} = C_{tot} \frac{X(i)}{\sum_{j \in V} X(j)}, \forall\ i \tag{1}$$

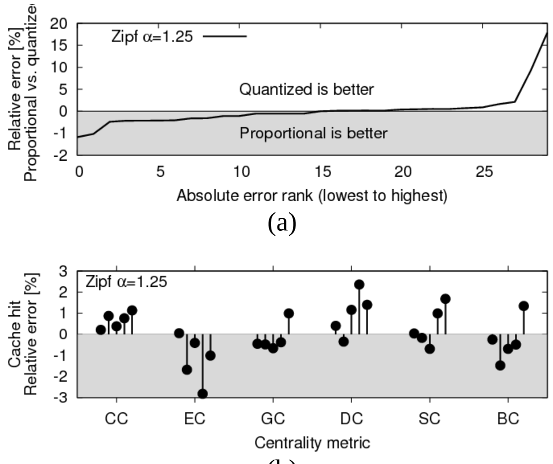
$$C^{Q_{X(i)}} = max\left(c, \left(C^{P_{X(i)}}\right)\right) \tag{2}$$

Notice that Eq. (1) corresponds to a perfectly *proportional* criterion, where the cache size $C^{P_{X(i)}}$ is distributed to the i-th node proportionally to the metric $X(i)$ normalized over the sum of the $X(i)$ score over the whole graph.

While (1) is an ideal strategy that allows to gauge the relative importance of the centrality score, we acknowledge that it may be hardly feasible in practice: indeed, CCN content store modules will be quantized in multiples of a unit module c, as it happens for nowadays RAM memory. As such, we also consider a *quantized* strategy (2), where the size of individual caches $C^{Q_{X(i)}}$ is multiple of c = 1 GB units.

The model assumed by (2) is that ISPs will invest in a fixed number of memory modules C_{tot}/c that they can then arbitrarily deploy in the network.

The viewpoint we adopt in this work is that an ISP may wish to reallocate the C_{tot}/c cache modules at its disposal (e.g., moving from an homogeneous setup to an heterogeneous one), so to optimize the achievable performance without incurring in further costs.



(a)

(b)

Figure 2. Proportional vs quantization error (top) and cache hit for different centralities and topologies (bottom).

4. Results.

We simulate the system with ccn*S*im, an open source simulator available at [1]. First, we observe that *quantization* actually plays a beneficial role.

Let us consider the cache hit metric H and as before denote by $H_X^P(H_X^Q)$ the cache hit metric achieved for a given topology and popularity settings by using a proportional (quantized) allocation according to centrality metric X.

We then define the relative error induced by quantization on cache hit as $(H_X^Q - H_X^P)/H_X^P$, which is depicted for all topologies and metrics in a monotonously increasing fashion in Fig. 2(a). In the picture, a negative value (gray shaded zone) corresponds to cases (i.e., metric and topology combinations) where proportional allocation would yield better cache hit results. Interestingly, Fig. 2(a) shows that though proportional allocation yields better performance in some cases (left part), however the performance difference with the corresponding quantized allocation is minimal (below 2%). Conversely, there are cases in which quantization can bring almost up to 20% gain with respect to a proportional allocation. Essentially, this is due to the fact that some metrics (especially, BC and SC [12]) may allocate a very low amount of cache space to some nodes, which is “corrected” by having a minimum amount of cache in the quantization process. We thus gather that quantized allocations is both robust (as we avoid outliers due to skew in the centrality metrics) and realistic (as a perfectly proportional allocation is not directly applicable).

Second, we observe that only degree-based $C_{DC}^{Q(i)}$ allocation proves to be a simple yet robust criterion, yielding consistent gain over all networks and popularity settings. Indeed, notice from Fig. 2(b) that for the other centrality metrics, gains are not consistent across topologies (nor across popularity settings as pointed out in [12], not reported here for lack of space). On the one hand, this is a positive finding, since very simple operational rules of thumb can be defined (i.d. "if you add a line card, add a content store module as well"). On the other hand, we notice that performance gain is upper-bounded by a modest 2.5% in the best case (Level3 topology, α =1.25 in Fig. (b)), so that there may be no real incentive in using heterogeneous cache allocation policies altogether.

4. Summary.

We have presented challenges for ICN design and operation, and offered guidelines to some of its settings. We point out that topological information can be exploited at multiple planes (e.g., control vs. data) and layers (e.g., offline design vs. online cache management). In this context, we have seen that while topology information is easy to exploit in the offline design case (e.g., devising simple memory allocation schemes proportional to the CCN router degree), however performance gain are limited [12]. Conversely, exploiting topology-awareness for online cache management (e.g., biasing the cache decision of each node via their centrality index in the topology) seem to bring higher performance benefits [5], [10], [11], though doubts remain so as to whether such complex decision policies can be supported in hardware at high speed [9].

Furthermore we remark that such diverse schemes [5],[10]-[12] have been tested under rather diverse settings (e.g., in terms of catalog and content store size, network topology, content popularity, etc.) so that the above conclusions appear to be preliminary, and need to be assessed by further study.

Finally, we point out that yet other design directions may have a far more important impact on CCN performance and, consequently, on its adoption.

Notably, one of this direction is the design of a hierarchical cache structure, that would move data across fast and slow memory, exploiting the temporal correlation of CCN requests. As a hierarchical content store design may improve the size of the cache by some orders of magnitude, this would set the catalog to cache size ration toward a more interesting operational point for CCN [6],[13].

Acknowledgements

The research leading to these results has received funding from the European Union under the EIT ICT Labs project on *Smart Ubiquitous Content* (SmartUC).

References

[1] ccnSim http://www.infres.enst.fr/~drossi/ ccnSim.

[2] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher, and B. Ohlman. A survey of information-centric networking. Communications Magazine, 50(7):26 –36, july 2012.

[3] S. Arianfar and P. Nikander. Packet-level Caching for ICN. In ReArch Workshop, 2010.

[4] R. Chiocchetti, D. Rossi, G. Rossini, G. Carofiglio, and D. Perino. Exploit the known or explore the unknown?: hamlet-like doubts in icn. In ACM SIGCOMM ICN, Helsinki, Finland, August 2012.

[5] K. Cho, M. Lee, K. Park, T.T. Kwon, Y. Choi, and S. Pack. Wave:Popularity-based and collaborative in-network caching for content-oriented networks. In IEEE NOMEN, 2012.

[6] C. Fricker, P. Robert, J. Roberts, and N. Sbihi. Impact of traffic mix on caching performance in a content-centric network. In NOMEN, 2012.

[7] V. Jacobson, D. K Smetters, N. H Briggs, J. D Thornton, M. F Plass, and R. L Braynard. Networking Named Content. In CoNEXT, 2009.

[8] N Laoutaris, H Che, and I Stavrakakis. The LCD interconnection of LRU caches and its analysis. Performance Evaluation, 63(7), 2006.

[9] D. Perino and M. Varvello. A reality check for content centric networking. In ACM ICN, 2011.

[10] I. Psaras, W. K. Chai, and G. Pavlou. Probabilistic in-network caching for information-centric networks. In ACM ICN, 2012.

[11] I. Psaras, R. G Clegg, R. Landa, W. K. Chai, and G. Pavlou. Modelling and Evaluation of CCN-Caching Trees. IFIP Networking, 2011.

[12] D. Rossi and G. Rossini. On sizing ccn content stores by exploiting topological information. In IEEE NOMEN, 2012.

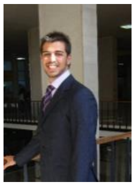
[13] G. Rossini and D. Rossi. Evaluating ccn multi-path interest forwarding strategies Computer Communications, 2013. To appear.

[14] L. Wang, A. K. M. Hoque, Cheng Yi, A. Alyyan, and B. Zhang. OSPFn:An OSPF based routing protocol for named data networking. Technical report, March 2012.

[15] W. Yaogong, L. Kyunghan, B. Venkataraman, R.L. Shamanna, R. Injong,and Y. Sunhee. Advertising cached contents in the control plane: Necessity and feasibility. In IEEE NOMEN, 2012.

[16] C. Yi, A. Afanasyev, L. Wang, B. Zhang, and L. Zhang. Adaptive forwarding in named data networking. SIGCOMM Comput. Commun. Rev., 42(3):6267, Jun 2012.

Dario Rossi (M'02,S'13) is a Professor at the Computer Science and Networking department of Telecom ParisTech, in Paris, France. He received his MSc and PhD degrees from Politecnico di Torino in 2001 and 2005 respectively, and his HDR degree from Universite Pierre et Marie Curie (UPMC) in 2010. He is responsible for several European research projects, such as FP7 mPlane, NAPA-WINE, Celtic TIGER, TIGER2 and TRANS, ANR Connect.



Giuseppe Rossini received the M.Sc. degree in computer engineering from the University of Naples Federico II, Italy, discussing his Master Thesis in the field of routing protocols and algorithms. Since October 2010, he is a Ph.D. student at the Computer Science and Networking department of Telecom ParisTech, Paris, France. His research interests are in the field of routing algorithms, network architecture, and distributed caching.

Enabling Real-time TV Services in CCN Networks

Giuseppe Piro[1] *and Vincenzo Ciancaglini*[2]
[1]*Politecnico di Bari, Italy,* g.piro@poliba.it
[2]*INRIA Sophia Antipolis Méditerranée, France,* vincenzo.ciancaglini@inria.fr

1. Introduction

Due to the growing importance that content sharing applications are experiencing in our everyday life [1], the emerging Content Centric Networking (CCN) paradigm [2] represents the most attractive solution for driving the current *host-based* Internet system towards a novel architecture focused around the *content-centric* concept. Basically, the communication in CCN requires the adoption of only two types of messages, namely *Interest* and *Data* [2]. A user may ask for a specific content by issuing an *Interest*, which is routed across the network towards nodes in posses of the required information, thus triggering them to reply with *Data* packets. While routing operations are performed by the strategy layer only for *Interest* packets, *Data* messages are sent back to requesting users just following the reverse path of the *Interest*, allowing every intermediate node to cache the forwarded content.

During past few years, CCN obtained a very warm attention in the scientific community. This is testified by the presence of several studies that have already investigated caching policies and data-transfer performances [3][4][5], congestion control issues [6], and routing strategies [7][8]. On the other hand, the growing demand for multimedia services has driven some researchers to focus their attention also to the design of sophisticated techniques enabling the transmission of video and voice contents over CCN networks. In this context, the management of real-time video transmissions has pointed out some challenges, which have been difficult to face. Unlike Video-On-Demand, in fact, the real-time video distribution has to deal with a specific class of problems to ensure the timely delivery of an ordered stream of chunks. Moreover, video chunks have to be received within a given time interval (the *playout delay*), before being actually played. A chunk not delivered before such time deadline will result in degradation of the rendered video. To overcome such issues, very promising works have been proposed in literature. The architecture presented in [9] has been designed for mapping HTTP-based streaming applications in a CCN. A novel cooperative caching strategy enabling time-shifted TV services has been discussed in [10]. A time-based Interest protocol is proposed in [11] in which a user sends a specific *Interest* message asking for a group of contents generated by the server during a specific time interval. Similarly to the previous paper, also in [12] is proposed a mechanism through which a user may request for multiple *Data* packets by issuing one *Interest* message.

To complement such interesting contributions, in our very recent paper [13] we designed a novel architecture, called CCN-TV, supporting data-centric real-time streaming services. In this paper, we will evaluate the performance of the CCN-TV system in a more complex network scenario, thus demonstrating its effectiveness in more realistic and higher loaded network conditions. In addition, an in depth analysis of the role that some of the main components of a CCN node, i.e., the cache and the Pending Interest Table (PIT) table, have in the presence of real-time services, as well as the comparison with respect to a *baseline* scenario where these features are not implemented, will be provided too.

2. The CCN-TV Architecture

In CCN-TV we consider a network of nodes requesting different real-time video streams, identified by a *channelID*, served by one or more servers. Unlike canonical UDP/TCP-based streaming, in CCN-TV each video is divided in consecutive chunks, identified by a progressive *chunk number*, that have to be requested individually, via a dedicated *Interest*. This fundamental aspect naturally supports the implementation of a flow control mechanism through which each user can explicitly request for new chunks just when the old ones have been received (or in the case they are not more useful because out of delay). In line with these premises, a *channel bootstrap phase*, a *flow control strategy*, and an efficient mechanism for retransmitting *Interest* packets have been designed within the CCN-TV architecture. For enabling these functionalities we need to extend the basic structure of the *Interest* packet by introducing an additional *Status* field marking if the *Interest* is related to the *channel bootstrap phase* or to a *retransmission*. In the case it is necessary to be conformed to classical CCN messages, this field can be easily replaced by an additional entry in the content name.

The channel bootstrap phase

Due to video codec requirements, a video stream can be visualized at the user side only once a specific I-Frame has been received. Therefore, to bootstrap a TV channel, a client has to find the closest server and gather from it the chunk (and the corresponding *chunkID*) of the last generated I-Frame. To this end, it sends an *Interest* packet for the URI: [domain]/[channelID], with the *Status* field set to *BOOTSTRAP* and a *Nonce* field containing a uniquely generated value. In this way, the message will travel unblocked until the first good stream repository, that will answer with a *Data* packet providing information about the first chunk of the last generated I-Frame. Once the user received this *Data* packet, it will request subsequent chunks, using a sliding window mechanism detailed in the following.

The flow control mechanism

A sliding window mechanism has been properly designed for enabling the user to request subsequent chunks of a video content. First, let us define *Pending Chunk* and *Pending Window* as the chunk whose *Interest* has been sent by the node and the window containing W different pending chunks not yet received, respectively. In details, together with the *chunkID*, we store in the *Pending Window* the timestamp of the first request and the timestamp of the last retransmission. Hence, whenever a new data message is received, or if the node does not receive any data for at least *windowTimeout* seconds, the following operations are performed: (1) purge the *Pending Window* from all the chunks who are expired, i.e., who have already been played; (2) retransmit all chunks that have not been received within the *windowTimeout*; (3) transmit, for each slot that got freed by the received or expired chunks, the *Interest* for a new one.

Interest routing

Normally, a CCN node does not propagate *Interest* packets related to contents already requested by other users in the past but not yet satisfied with corresponding *Data* packets [2]. The PIT table is used to keep track of *Interest* packets that have been forwarded upstream towards content sources, combining them with the respective arrival faces, thus allowing the properly delivery of backward Data packets sent in response to *Interests*. It is important to note that this mechanism prevents the propagation of retransmitted *Interest* packets, thus compromising the right behavior of CCN-TV. In order to force the propagation of retransmitted *Interests*, the *Status* field is set to *Retransmission*: this configuration would impose nodes along the routing path to propagate it versus the router that can satisfy this request (i.e., by skipping the usual CCN mechanism).

3. Performance evaluation of CCN-TV

We evaluated performances of CCN-TV architecture, through computer simulations carried out with *ccnSim*, an open source and scalable chunk level simulator of CCN, built on top of the Omnet++ framework [14].

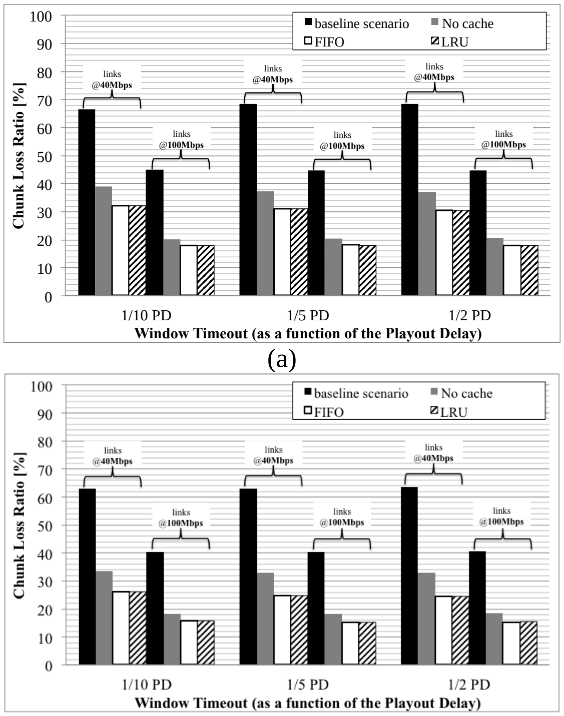
Network configuration and system parameters

Differently from [13], we considered a more complex network architecture composed by 68 routers connected among them according to the Deutsche Telekom topology [15]. A CCN node is directly installed to each router and no TCP or UDP encapsulation has been implemented. We assume the presence of only one small video-streaming provider that offers 5 parallel real-time transmissions to remote clients, each one connected to one router of the Deutsche Telekom network. In every simulation round, each video content is mapped to a video stream compressed using H.264 at an average coding rate randomly chosen in the range [250, 2000] kbps. On the other hand, every client chooses to watch one specific TV channel based on its popularity, which has been modeled through the Zipf distribution (in line with [13] we set α=1). In our tests, we adopted the optimal routing strategy, already available within the ccnSim framework [14]. According to it, *Interest* packets are routed towards the video server along the shortest path. Moreover, three caching strategies have been considered in our study: no-cache, LRU, and FIFO. When well-known LRU or FIFO policies are adopted, we set the size of the cache to 210 Mbits, i.e., a typical value for SRAM memories already available in the commerce [16]. The no-cache policy is intended to evaluate the performance of the CCN without using any caching mechanism. Furthermore, a *baseline* scenario, in which the no-cache policy is enabled and the PIT table is totally disabled (this means that each user establishes with the service provider a unicast communication and the server should generate a dedicated *Data* packet for each generated *Interest*), has been considered as reference configuration. Regarding the flow control mechanism, the window size W has been set to 10, ensuring that faces of the server are almost fully loaded in all considered scenarios. The transmission queue length associated to each face, Q, has been set in order to be larger than $Q = L_C \times \tau$, where L_C and τ represent the link capacity and the maximum propagation delay in the considered network topology, respectively.

In order to evaluate performances of CCN-TV under various system configurations, we considered different settings of the bandwidth dedicated to real-time services (set in the range [40-100] Mbps), the *playout delay* (chosen in the range [10-20] s), and the *windowTimeout* (chosen in the range [1/10-1/2] of the *playout delay*). To conclude, each simulation lasts 300s and all results have been averaged over 15 simulations.

Simulation results

The chunk loss ratio, which represents the percentage of chunks that have not been received in time (i.e., before the expiration of the *playout delay*) by clients, is the first important parameter that we reported in Fig. 1 which describes how CCN-TV settings affect the quality of service offered to end users. We note that the amount of discarded chunks is very influenced by the *playout delay:* the highest *playout delay* allows the client to receive more *Data* packets before the expiration of the time deadline, thus reducing the amount of discarded chunks. In addition, the reduction of link capacities leads to a higher number of lost chunks, due to increased latencies induced by network congestion. By handling unicast communications, the *baseline* scenario generates the highest network congestion level, thus registering the worst performances. From this finding emerges the important role that both cache and PIT table have on network performances.

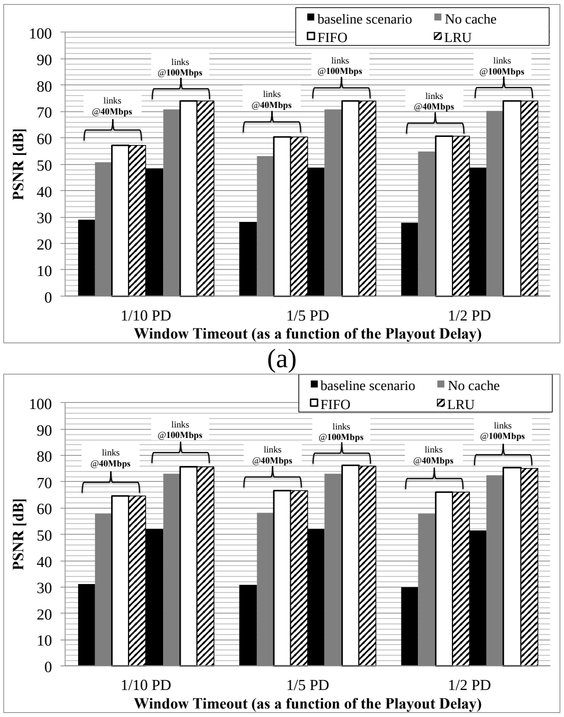


(a)

(b)

Figure 1. Chunk Loss Ratio when the *playout delay* has been set to (a) 10s and (b) 20s.

In order to estimate the Quality of Experience perceived by end users, we have also computed Peak Signal to Noise Ratio (PSNR) of received video flows (results are shown in Fig. 2). In line with previous results, the PNSR is higher in the same case in which the chunk loss ratio is lower. This means that the quality of TV services improves when we increase the *playout delay* and the link capacity. Also in this case, we remark that no-cache, LRU, and FIFO caching policies outperform always the *baseline* scenario.



(a)

(b)

Figure 2. PSNR of received video flows when the *playout delay* has been set to (a) 10s and (b) 20s.

To provide a further insight, we also reported in Fig. 3 the percentage of *Interest* packets sent by users and directly received by the service provider. In the *baseline* scenario, the total amount of generated *Interests* reach the remote server, thus excessively overloading its faces. By enabling the PIT table, even without implementing any caching mechanism, the system is able to halve the traffic load at the server side, thus improving significantly network performances. Finally, the traffic load handled by the server further reduces when a cache policy is activated. Anyway, it is evident that, in the presence of real-time flows, the cache does not represent an important CCN feature because it is not able to guarantee a notable improvement of system performances with respect to the case it is not used. On the other hand, we noticed that the PIT plays a more relevant role. In fact, in presence of live video streaming services, clients that are connected to a channel request same chunks simultaneously. In this case, a CCN router has to handle multiple Interest messages that, even though sent by different users, are related to the same content. According to the CCN paradigm, such a node will store all of these requests into the PIT, waiting for the corresponding *Data* packet. As soon as the packet is received, the router will forward it to all users that have requested the chunk in the past. According to these considerations, the use of the cache will not produce a relevant gain of network performances. Indeed, the PIT helps reducing the burden at the server side by avoiding that many *Interest* packets for the same chunk are routed to the server.

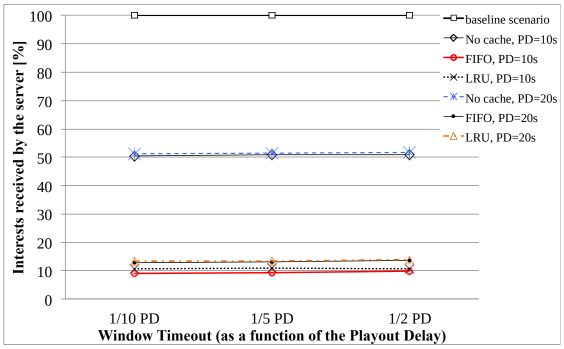


Figure 3. Percentage of *Interest* packets received by the remote server.

4. Conclusion

In this work, we investigate the performance of the CCN-TV architecture, which has been properly designed to offer real-time TV services in CCN networks, under different system settings. Besides having shown the effectiveness of the discussed architecture, presented results have highlighted that, differently from any caching policies, the PIT table has a fundamental role in reducing the burden at the server side in the presence of real-time streaming services.

References

[1] B. Ahlgren, P. A. Aranda, P. Chemouil, S. Oueslati, L. M. Correia, H. Karl, M. Sollner, and A. Welin, “Content, connectivity, and cloud: ingredients for the network of the future,” *IEEE Commun. Mag.*, vol. 49, no. 7, Jul. 2011.

[2] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard, “Networking named content,” in *ACM CoNEXT*, 2009.

[3] H G. Carofiglio, M. Gallo, L. Muscariello, and D. Perino, “Modeling data transfer in content-centric networking,” in *Int. Teletraffic Congress, (ITC)*, 2011.

[4] M. Tortelli, I. Cianci, L. A. Grieco, G. Boggia, and P. Camarda, “A fairness analysis of content centric networks,” in Proc. Of Int. Conf. on Network of the Future, (NOF) Paris, France, 2011.

[5] D. Rossi and G. Rossini, “On sizing CCN content stores by exploiting topological information,” in *IEEE INFOCOM, NOMEN Worshop*, 2012.

[6] L. A. Grieco, D. Saucez, and C. Barakat, “AIMD and CCN: past and novel acronyms working together in the Future Internet,” in Proc. of *Capacity Sharing Workshop (CSWS) co- located with ACM SIGCOMM CoNEXT*, Dec. 2012.

[7] M. Tortelli, L. A. Grieco, and G. Boggia, “CCN forwarding engine based on bloom filters,” in Proc. of ACM Int. Conf. on Future Internet Technologies (CFI), Seoul, Korea, Sep. 2012.

[8] W. You, B. Mathieu, P. Truong, J. Peltier, and G. Simon, “Dipit: A distributed bloom-filter based pit table for CCN nodes,” in Proc. of Int. Conf. on Computer Communications and Networks (ICCCN), 2012.

[9] “Z. Li and G. Simon, “Time-shifted TV in content centric networks: the case for cooperative in-network caching,” in Proc. of IEEE ICC, Jun. 2011.

[10] H. Xu, Z. Chen, R. Chen, and J. Cao, “Live streaming with content centric networking,” in Proc. 3rd Int. Conf. on Networking and Distributed Computing, Hangzhou, China, 2012.

[11] J. Park, J. Kim, M. Jang, and B. Lee, "Time-based interest protocol for real-time content streaming in content-centric networking (CCN)", in Proc. of IEEE Int. Conf. on Consumer Electronics (ICCE), 2013.

[12] H. Li, Y. Li, T. Lin, Z. Zhao, H. Tang, and X. Zhang, "MERTS: A more efficient real-time traffic support scheme for Content Centric Networking", in Proc. of IEEE Int. Conf. on Computer Sciences and Convergence Information Technology (ICCIT), 2011.

[13] V. Ciancaglini, G. Piro, R. Loti, L. A. Grieco, and L. Liquori," CCN-TV: a data-centric approach to real-time video services", in Proc. of IEEE Int. Conf. on Advanced Information Networking and Applications (AINA), Barcelona, Spain, Mar., 2013.

[14] G. Rossini and D. Rossi, “Large scale simulation of CCN networks,” in *In Algotel* 2012.

[15] http://www.telekom.com/ .

[16] D. Perino and M. Varvello. “A reality check for content centric networking”, in Proc. of the ACM SIGCOMM workshop on Information-centric networking (ICN). New York, USA, 2011.

Giuseppe Piro is a postdoctoral researcher at Politecnico di Bari, Italy. His main research interests include quality of service in wireless networks, network simulation tools, 4G cellular systems, nanonetwork communications, and future Internet architectures. Piro has a PhD in electronics engineering from Politecnico di Bari. He founded the LTE-Sim project and is a developer of Network Simulator 3.

Vincenzo Ciancaglini is a PhD candidate at INRIA Sophia Antipolis Méditerranée, France. His research topics include the interoperability of content-cased systems, future internet architectures and media streaming systems. Ciancaglini has a MSc in Electrical Engineering, from Stockholm’s Royal Institute of Technology, Sweden and a MSc in Telecommunications Engineering form Politecnico di Torino, Italy.

The Role of Virtualization in Information-centric Network Deployment

Jing Ren, Kostas Pentikousis†, Cedric Westphal†, Will Liu†, and Jianping Wang**
** Department of Computer Science, City University of Hong Kong*
† Huawei Technologies
{jingren, jianwang}@cityu.edu.hk,
{k.pentikousis, cedric.westphal, liushucheng}@huawei.com

1. Introduction

In recent years, the dissemination of information, including audio/video (A/V) and multimedia content, has comprised a significant portion of the total traffic on the Internet. Popular multimedia dissemination technologies include services based on peer-to-peer as well as centralized mechanisms. As these distribution services are defined on top of the traditional host-centric paradigm, they cannot leverage advanced network-layer techniques, such as in-network caching and request aggregation, to optimize the network traffic load [1]. What is more, the current use of the core TCP/IP network stack does not capitalize on the native broadcast nature of wireless communication, effectively shoehorning all information exchanges into a remote access paradigm. Dannewitz et el. [2] motivate the need for a new look at problems that are solved today through a patchwork of solutions. Thus, there is an increasing need to redesign the Internet architecture.

Information-Centric Networking (ICN), a new communication paradigm, has attracted significant attention in the Future Internet research areal; see the recent Feature Topic on ICN in *IEEE Communications* [3] and the references therein. A key ingredient of this new paradigm is the ability to capitalize on all information bits regardless of whether they are bits on the wire, on the ether or on storage devices. This would allow the future Internet to take advantage of in-network storage natively and in combination with other modern IT concepts that are now applied in networking such as virtualization.

So far, many solutions are proposed to address ICN design issues pertaining to naming and name resolution, routing, and security, just to name a few. Each proposal has its individual features and tackles problems in a different manner. To date there is no sufficient theoretical and empirical evidence that one of the proposals clearly outperforms all other alternatives by a wide margin. This may be great news for researchers, as still many hard problems need to be resolved, but operators and vendors alike may opt to adopt a waiting stance on the matter, thus delaying the deployment of ICN to well beyond 2020 in practice.

A possible way out of this conundrum is to capitalize on the recently growing interest in deploying virtualization technologies in infrastructure networks. Virtualization is well established in the IT world, but has yet to obtain a significant foothold in carrier networks. In this letter, we contribute to the discussion on ways forward for ICN deployment by arguing that as modern virtualization techniques enter the operator domain, ICN solutions may be able to gain more traction in production networks and reach everyday user well in advance of what typical standardization time frames would imply.

2. ICN Overview

In the current host-centric paradigm, the core networking primitives revolve around naming hosts and network interfaces. Network devices can only forward packets according to the destination addresses. There are few operations performed on the packet based on the information carried in the packet. In other words, the information is transparent to the network devices.

In contrast, information is directly named in information-centric networks. Although network devices cannot interpret the semantics of carried information per se, they can take advantage of the named data content and apply appropriate methods to optimize network load and user experience. For example, routers can respond to information requests using their content cache in Named Data Networking [4] without having to understand explicitly what is the actual content in terms of encoding formatting, bit rates and so on.

To date, the European Framework Programme as well as research agencies in North America and in the Asia Pacific region have invested considerable amounts of funds to foster the development of Future Internet architectures that take into account ICN concepts and can become the foundation for multi-billion industry in the decades to come. Among these architectures some have attracted a larger researcher following, including Named Data Networking (NDN) [4], the publish-subscribe Internet architecture (PSI) [5], and Network of Information (NetInf) [6].

However, we would like to highlight that ICN is not limited to these pioneering approaches. On the contrary we expect the area to grow substantially in the coming years. Indeed already there is a growing corpus of work which involves not only the definition and refinement of new and previously proposed architectures but possible ways forward from a standardization perspective.

Besides addressing ICN design issues (including naming and name resolution, routing, and security, just to name a few), many research efforts are made to evaluate the performance in different scenarios to illustrate the benefit of these ICN designs [7]. Some notable efforts are listed below:

- NDN has been evaluated with respect to its ability to support static content dissemination, timely delivery of real-time data (VoCCN) [8], audio conferencing (ACT) [9], privacy and anonymity (ANDaNA) [10], and ad hoc network support (LFBL) [11].
- The PSI architecture has been shown to demonstrate its capacity to support file retrieval, access control and privacy, context-aware delivery, mobility, voice (VoPSI) [12], Internet of Things, etc.
- Recent work on NetInf shows its potential for efficient multimedia content dissemination, delay tolerant video distribution (DTVideo) [13], mobile communications, social networks and so on.

These evaluations are made in selected scenarios to highlight their advantages [7]. It is hard to say at this stage that a specific ICN design can outperform all others in all scenarios. Thus, there is an emerging requirement to design a generic framework where different ICN architectures can be co-exist and compared. This generic framework can also enable the interoperability of different ICN architectures and provide a transition path of ICN.

3. ICN through Network Virtualization

In this section, we review recent and ongoing efforts with respect to implementing ICN by employing a Software-Defined Networking (SDN) substrate. SDN is characterized by the separation of the control plane and the data plane of the network. SDN enables virtualizing network resources by manipulating the flow entries of switches/routers and provides an abstracted view of the network to the upper control programs [14]. However, network virtualization can be implemented based on other techniques which are used in datacenters such as hardware virtualization techniques, although we will not explore this issue in more detail in this letter.

Benefits

As discussed above, there is an emerging requirement to establish a generic foundation for deploying, and experimenting with we may add, different ICN proposals. With such a framework in place, we can implement different ICN designs over generic hardware and compare their performance or allow them to co-exist as the case may be. Without virtualization, vendors and operators would have to commit to a specific flavor early on, and then perhaps tie in capital in promising and eventually not widely deployed ICN architectures.

Below we consider how virtualization can foster ICN deployment in practice.

- Network virtualization will enable the co-existence of various ICN architectures running on the same physical network substrate while new network devices can be implemented using a software-defined approach. With network virtualization techniques, network resources can be partitioned into different slices which are separated from each other. Each slice can then be assigned to implement different network architectures which use different routing, forwarding and transport mechanisms, as shown Figure 1. NFV can guarantee the experimental traffic which belongs to different ICN proposals is isolated from each other, which can allow us to evaluate the performance of different ICN architectures.

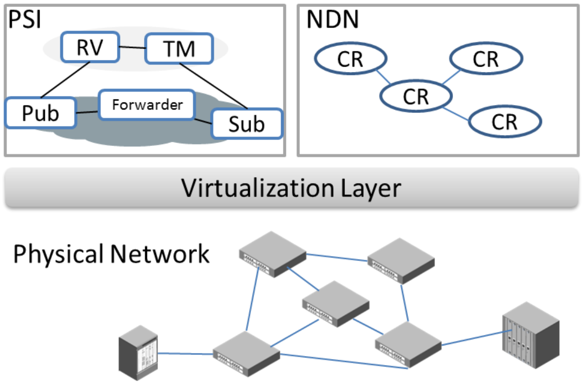


Figure 1. Different ICN architectures using the same physical network and virtualized infrastructure

- The ability of programming the network in SDN can facilitate ICN traffic optimization, for instance, by taking advantage of in-network cache management and traffic engineering. With SDN, it is easier to monitor the utilization of virtualized resources as well. With this utilization information, traffic optimization programs can determine whether a piece of content should be cached and how to route a flow. Meanwhile, the final decisions can be deployed in the virtualized devices.

Ongoing work – Software-defined ICN

Recent research efforts have already made some inroads in relate ICN with SDN. For example, L.Veltri *et al* [15] discuss how ICN could be implemented based on SDN and how SDN should be extended to suit ICN requirements. D. Syrivelis *et al* [16] implement PSIRP's crucial forwarding function using SDN. A. Chanda *et al* [17] present a content-centric network architecture which is based on SDN principles and implements metadata driven services, such as metadata driven traffic engineering and firewalling, with the ability to parse content metadata at the network layer.

However, there are some limitations for implementing ICN on SDN. SDN provides flow-level abstractions as it virtualizes network resources by manipulating flow entries of switches/routers. In particular, the switches/routers in SDN operate on lower-layer protocol headers. However, what ICN needs is content-level abstractions. It requires switches/routers to operate directly on content. For example, information requests may be satisfied by the copies in switches/routers' cache where there is no end-to-end flow.

4. Summary and Outlook

Information-Centric Networking (ICN) has become an active research topic in the Future Internet research area for some time now and several designs that have been put forth have reached the maturity level that is sufficient for operators to start experimenting with this new paradigm. It becomes timely and important to study the deployment and measurement of various ICN designs. We briefly discussed the first-order benefits in implementing ICN though network virtualization. Future work in this area will need to take into consideration not only the practical aspects of combining information-centric concepts with network virtualization but also a range of ICN-specific challenges as outlined in [18].

References

[1] V. Jacobson, "A new way to look at networking", video available online at *Google Tech Talks*, 2006.
[2] D. Dannewitz, et al., "Scenarios and research issues for a network of information", in *Proc. MobiMedia*, Oulu, Finland, July 2008.
[3] K. Pentikousis, et al., "Information-centric networking", in *IEEE Commun. Mag.*, vol. 50, no. 7, July 2012.
[4] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs and R. L. Braynard, "Networking named content," in *Proc. CoNEXT*, 2009.
[5] Trossen, D. and G. Parisis, "Designing and realizing an information-centric internet", in *IEEE Commun. Mag.*, vol. 50, no. 7, 60-67, 2012.
[6] Ahlgren, B. et al., "Design considerations for a network of information", in *Proc. CoNEXT Re-Arch Workshop*, 2008
[7] K. Pentikousis, et al., "ICN Baseline Scenarios", Internet Draft, draft-pentikousis-icn-scenarios (work in progress), 2013.
[8] V.Jacobson, et al., "VoCCN: voice-over content-centric networks", in *Proc. CoNEXT Re-Arch Workshop*, 2009.
[9] Z. Zhu, et al., "ACT: Audio Conference Tool Over Named Data Networks" in *Proc. SIGCOMM ICN Workshop, 2011*.
[10] S. DiBenedetto, et al., "ANDaNA: Anonymous Named Data Networking Application", in *Proc. 19th Annual Network & Distributed System Security Symposium*, 2012.
[11] M. Meisel, V. Pappas and L. Zhang, "Ad Hoc Networking via Named Data", in *Proc. ACM MobiArch*, 2010.
[12] C. Stais, et al., "VoPSI: Voice over a Publish-Subscribe Internetwork", in *Proc. Future Network & Mobile Summit*, 2011.
[13] D. Kutscher, et.al, "The Network of Information: Architecture and Applications", FP7 SAIL *Public Report D-B.1*, 2011.
[14] "Software-Defined Networking: The New Norm for Networks", in *ONF*, 2012.
[15] L. Veltri, et al., "Supporting Information-Centric Functionality in software defined networks", in *Proc. IEEE ICC*, pp. 6645-6650, 2012.
[16] D. Syrivelis, et al., "Pursuing a Software Defined Information-centric Network", in *Proc. EWSDN*, pp. 103-108, 2012.
[17] A. Chanda, C. Westphal, and D. Raychaudhuri, "Content Based Traffic Engineering in Software Defined Information Centric Networks", in *Proc. IEEE NOMEN Workshop*, 2013.
[18] D. Kutscher, et al., "ICN Research Challenges", Internet Draft, draft-kutscher-icnrg-challenges (work in progress), 2013.

Jing Ren is a research assistant in the Department of Computer Science at City University of Hong Kong. She received the B.S. degree in Communication Engineering from University of electronic science and technology of China, Chengdu, China in 2007 and now is a Ph.D student in Communication and Information System from University of electronic science and technology of China, Chengdu, China. Her research interests include network architecture and protocol design, information-centric networking and software-defined networking.

Kostas Pentikousis is a senior research engineer with Huawei Technologies, in Berlin, Germany. From 2005 to 2009 he was a senior research scientist with VTT Technical Research Centre of Finland. He earned his Bachelor's degree in informatics (1996) from Aristotle University of Thessaloniki, Greece, and his Master's (2000) and doctoral degrees (2004) in computer science from Stony Brook University. His current research interests include network architecture and protocol design, network virtualization, and information-centric networking.

Cedric Westphal is a Principal Research Architect with Huawei Innovations working on future network architectures, both for wired and wireless networks. His current focus is on Information Centric Networks. He also has been an adjunct assistant professor with the University of California, Santa Cruz since 2009. Prior to Huawei, he was with DOCOMO Innovations from 2007-2011 in the Networking Architecture Group. His work at DOCOMO has covered several topics, all related to next generation network architectures: scalable routing, network virtualization and reliability, using social networks for traffic offloading, etc. Prior to that, he was at Nokia Research Center from 2000 to 2006. He received a MSEE in 1995 from Ecole Centrale Paris, and a MS (1995) and Ph.D. (2000) in EE from the University of California, Los Angeles. Cedric Westphal has co-authored over fifty journal and conference papers, including several best paper awards; and been awarded twenty patents. He has been an area editor for the ACM/IEEE Transactions on Netoworking since 2009, an assistant editor for (Elsevier) Computer Networks journal, and a guest editor for Ad Hoc Networks journal. He has served as a reviewer for the NSF, GENI, the EU FP7, and other funding agencies; he has co-chaired the program committee of several conferences, including IEEE ICC (NGN symposium). He is a senior member of the IEEE.

Will Liu is currently a Research Engineer at Huawei Technologies. He received the B.S. degree and Ph.D. degree in Computer Science from University of Science and Technology of China (USTC) in 2006 and 2011, respectively. He also received a Joint Ph.D. degree in Computer Science from City University of Hong Kong (CityU) in 2011. His research interests include wireless sensor Network, IPv6, software-defined networking, Information-centric networking and network functions virtualization.

Jianping Wang is an associate professor in the Department of Computer Science at City University of Hong Kong. She received the B.S. and the M.S. degrees in computer science from Nankai University, Tianjin, China in 1996 and 1999, respectively, and the Ph.D. degree in computer science from the University of Texas at Dallas in 2003. Jianping’s research interests include dependable networking, optical networks, cloud computing, service oriented networking and data center networks.

The Potential of Information Centric Networking in Two Illustrative Use Scenarios: Mobile Video Delivery and Network Management in Disaster Situations

N. Blefari Melazzi, L. Chiariglione
Dpt. of Electronic Engineering, University of Rome, Tor Vergata, Italy, blefari@uniroma2.it
CEDEO.net, Villar Dora (TO), Italy, leonardo@chiariglione.org

1. Introduction

Information Centric Networking (ICN) is a new paradigm in which the network layer provides users with content exposed as names, instead of providing communication channels between hosts (see e.g. the papers [1][2] and the projects [3][4][5][6][7]). In ICN, the network transfers individual, identifiable content chunks, instead of unidentifiable data containers (i.e., IP packets). The basic functions of an ICN infrastructure are to: i) address contents, adopting an addressing scheme based on names (identifiers), which do not include references to their location; ii) route a user request, which includes a "destination" content-name, toward the "closest" copy of the content with such a name; this copy could be stored in the original server, in a cache contained in a network node or even in another user's device; iii) deliver the content back to the requesting host (see Figure 3).

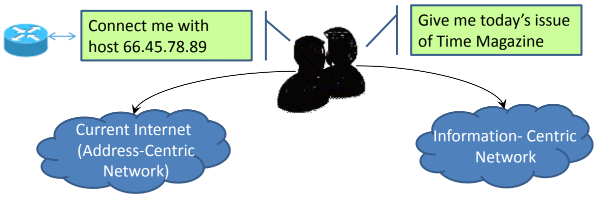


Figure 3: ICN concept

In our opinion, expected advantages of ICN include:

1) *Efficient content-routing*: ICN would enable Internet Service Providers (ISPs) to perform native content routing. This would be a built-in facility of the network, which would transform the Internet to a native content distribution network, as opposed to Content Delivery Networks (CDN) overlays, which are the current practice.
2) *In-network caching*: caching enabled today by HTTP proxies requires performing complex stateful operations. ICN would significantly improve efficiency and scalability of caching by judiciously using caches at critical points in the network, exploiting the knowledge of the name of content.
3) *Simplified handling of mobile and multicast communication*: with ICN, unlike current mobility architectures, when a user changes point of attachment to the network, she will simply ask for the next chunk of the content she is interested in, without the need for maintaining tunnels or re-routing from an anchor point and maintaining state in the network; the next chunk may be provided by a different node than the one that it would have been used before the handover. Furthermore, multicast is an inherent capability in ICN, with content requested from a node being delivered to all interested receivers, without using overlays.
4) *Simplified support for time/space-decoupled communications*: allowing fragmented networks, or sets of devices to operate even when disconnected from the rest of the 'network' (e.g. sensors networks, ad-hoc networks, vehicular networks, social gatherings, mobile networks on board trains, planes, or networks stricken by disaster).
5) *Simplified support for peer-to-peer communications*: ICN inherently supports communications between peers, without the need for application-layer overlays, as today.
6) *Content-oriented security model*: securing the content itself, instead of securing the communication channels, allows for a more flexible and customizable protection of content and user privacy and protects in-network caches from fake content.
7) *Content-oriented access control*: ICN can provide access to content as a function of time, place (e.g., country), or profile of user requesting the item. Likewise, this functionality allows implementing: i) access revocation (also known as digital forgetting), so that content may be removed from the system by its creator, ii) garbage collection, deleting from the network 'expired'/obsolete contents.
8) *Content-oriented quality of service differentiation (and possibly pricing):* ICN would enable ISPs to differentiate the quality perceived by users from different services without complex, high-layer procedures.
9) *Create, deliver and consume contents in a modular and personalized way*: ICN provides opportunities for better customization of the interests of users and the content that is published by providers. This will enable more efficient consumption of content because of better "granularity" in how content is described and identified.
10) *Network awareness of transferred content*, allowing network operators to better control information and related revenue flows, favoring competition between operators in the inter-domain market and better balancing the equilibrium of power across the entire eco-system, including over-the-top players.

A final overall advantage of ICN, which in a way comprehends the specific advantages listed above, is a simplification of network design, operation and management. Currently, content and service providers have to “patch” shortcomings and deficiencies of IP data delivery by using several “extra-IP” functionalities, such as HTTP proxies, CDNs, multi-homing and intra-domain multicast delivery, to name a few. This implies the involvement of several parties, the use of several specific protocols, the deployment of ad hoc devices and the interplay of different functionalities, often offered and managed by different companies and businesses. Apart from technical complexity, such operations also add management and administrative complexity. In an ICN environment, such diverse functions can be integrated in the network in a smooth and seamless way, e.g. by supporting inherently data replication, caching, multi-homing and multicast delivery.

The research community is working to achieve the advantages promised by ICN while addressing its two biggest drawbacks, namely backward compatibility with current networks and scalability of the routing functionality. In this paper, we briefly describe an ICN architecture proposed by the EU FP7 project CONVERGENCE [7], which ends in May 2013, and we show how this work can be a possible starting point to support two appealing use scenarios of the EU-Japan joint project GreenICN [8], which is about to start.

2. The Convergence approach and architecture

The goal of CONVERGENCE is to enhance the Internet with an information-centric, publish-subscribe service model, based on a common container for any kind of digital data. We call this container the Versatile Digital Item (VDI), which is the basic unit of distribution and transaction in our system. VDIs can incorporate every possible kind of information, including signaling and control, and therefore minimize the need to store external information and states outside it.

VDIs bind meta-information, describing the content and structure of the item, and resources (other VDIs, audio, images, video, text, etc.). The meta-data describing the VDI include structural information, describing the content of the VDI; cryptographic keys allowing authentication and protection of the VDI; rights information defining rights to use the item; event reporting; and an expiry date, supporting “digital forgetting”. VDIs bear a unique identifier, which is translated (or which is identical) to the network-level ICN name used to route the VDI.

The CONVERGENCE architecture comprises three levels (see Figure 4 and [9]):

1. *Application level*. Applications bundle resources and metadata into VDIs, and consume VDIs and their components, by extending middleware functionality.

2. *Middleware level*. The CONVERGENCE Middleware (CoMid) is the level responsible for manipulating and processing VDIs. CoMid allows users to publish and subscribe to VDIs and to search for them using semantic subscriptions. It builds on and extends the MPEG-M standard [10], which provides a distributed eco-system of Protocol Engines (PE), Technology Engines (TE) and Aggregated Services. The middleware executes all functions that are too complex to be performed at line speed by network forwarding devices.

3. *Computing Platform level*. The Computing Platform level provides novel Information-Centric Networking functions (CoNet [11]) and secure handling (CoSec) of data, and interfaces to access local hardware of network and user devices. CoMid transfers data (i.e., VDIs) for its own purposes and at the request of applications. While this could be done with standard TCP/IP means, CONVERGENCE has taken an alternative approach, which is more consistent with the use of a common and self-contained data unit at the application and middleware level: all communications are handled through the CoNet information-centric network.

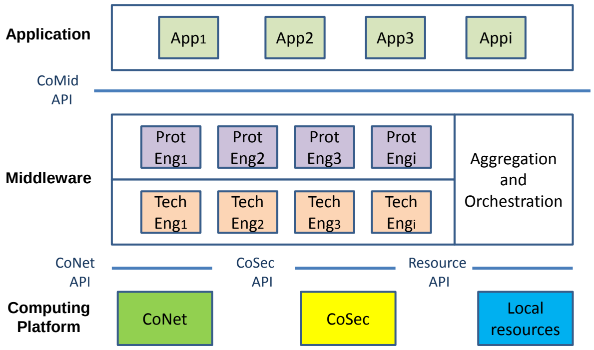


Figure 4. Architecture of CONVERGENCE

The VDIs, which are data units of the middleware, are transported in the network by CoNet data units whose identifier is a name, according to ICN principles. Such ICN names can be chosen independently from VDI identifiers, or coincide with them, according to the naming scheme chosen for the CoNet.

Not all CONVERGENCE devices implement the same functionalities and levels; we have peers (including user devices and selected network devices), which implement all levels, and nodes, which only include a CoNet networking component and/or a CoSec component. Therefore we can have CoNet nodes and CoSec nodes and peers (see Figure 5). The CoNet comprises transport functionalities [12].

In the following section we focus on the CoNet; details about applications, middleware and CoSec can be found in [9] and in project deliverables [7].

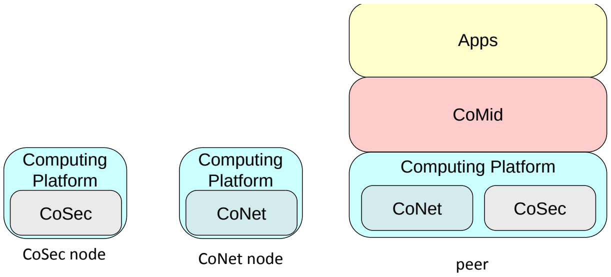


Figure 5: CONVERGENCE nodes and peers

The Convergence network

Figure 6 shows the CoNet architecture [11]. CoNet works pretty much like CCN [1]. The end-nodes are called *ICN clients* and *ICN servers*. As in CCN [1], content requests are called "interests" and are forwarded taking into account the requested content-name. When a node that contains the content receives the request, it replies with a "data" packet that is sent back towards the requesting node. The key entities involved in these operations are the *Border Nodes*, which: i) forward content-requests from ICN clients to ICN servers, implementing the ICN "forward-by-name" mechanism; ii) deliver content data back from ICN servers to ICN clients, implementing the ICN "data forwarding" mechanism; iii) may cache content and therefore provide it to ICN clients without forwarding the requests to ICN servers, in this case the Border Nodes perform security checks in order not to store and redistribute fake content.

Unlike CCN, CoNet is an inter-net, made up of CoNet Sub-System (CSS). Nodes belonging to the same CSS can directly exchange interest and data packets; nodes belonging to different CSSs will be interconnected by Border Nodes. A CSS exploits an underlying technology to transfer requests and data across its nodes. A CCS can be for example: two nodes connected by a point-to-point link, a layer 2 network like Ethernet, an overlay UDP link (socket) among two nodes, a layer 3 network e.g. a private IPv4 or IPv6 network or a whole Autonomous System.

An IP CSS (e.g. CSS n.2 in Figure 6) can be composed by an arbitrary number of plain IP Routers that do not perform ICN operations. Moreover we defined the Internal Nodes (IN), which are "enhanced" IP routers that can perform content caching but are not able to perform "forward-by-name" (only plain IP routing).

An Information Centric Autonomous System (IC-AS) includes a set of CSSs under the same administrative domain. The *Name Routing System (NRS) Nodes* assist the Border Nodes in performing the forward-by-name operation: they are responsible for the content-routing mechanisms, both at the intra-domain and inter-domain level. The NRS functionality can be seen as logically centralized within one IC-AS. The introduction of NRS Nodes is fully aligned with the Software-Defined Networking (SDN) approach of using "controllers" to drive the forwarding behavior of switches/routers and to enforce an explicit separation between a data forwarding plane and a control plane [13]. Though we do not necessarily rely on OpenFlow, we have experimented a CoNet implementation based on it [14].

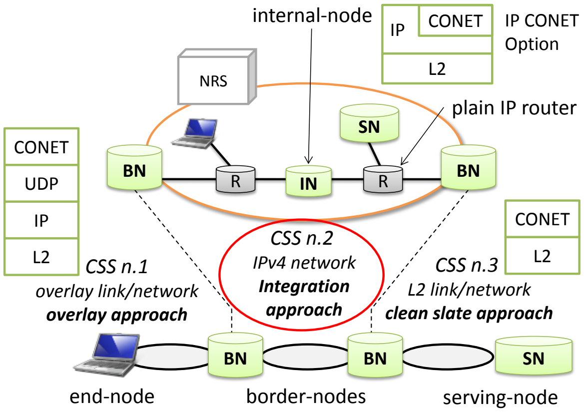


Figure 6- CONET Architecture

Other differences between CCN and CoNet are:

1) *Stateless delivery*: CoNet inserts the reverse-path routing information used to deliver data back to user within its data unit, avoiding the temporary storing of routing states in network nodes (i.e. the Pending Interest Table, PIT, of CCN [1]).

2) *Transport*: CoNet implements segmentation and transport level mechanisms that improve the performance of current CCN transport by defining a smaller transport data units (aka carrier-packet), contained in single IP packets [12].

3) *Integration with IP*. The "clean slate" approach for ICN aims at fully replacing the IP layer. On the other hand, we see two main drivers for an "integration" approach, in which ICN interworks with IP: i) conversational services still remain very important and ICN does not provide apparent benefits for such services; in our "integration" approach conversational services are supported by IP, while ICN is used for content-oriented communications; ii) the only way for a real deployment of ICN is an evolutionary path from existing IP networks. The conceptual difference between other ICN "overlay" solutions and our "integration" approach is that we introduce ICN functionality in a backward compatible way in IP routers, rather than "tunneling" ICN information over UDP or TCP. We have defined and experimented two ways of transporting name based information. The first one introduces CoNet IP options in IPv4 and IPv6 headers [15]. The second proposal includes it in the headers of our proposed CoNet transport protocol [12]. In both cases, each IP packet supporting content-based communication carries the content-name and further content-related information so that it can be efficiently

processed by CoNet nodes (while CoNet unaware nodes can still process it as a regular IP packet).

4) *Routing scalability*. CoNet includes a *route caching* approach, named Lookup and Cache, which inserts and removes on-demand the entries of the Forwarding Information Base (FIB) of the forwarding engines, and uses centralized routing servers for maintaining the Routing Information Base (RIB) (à-la SDN), and synchronizes the RIBs with a BGP-like distribution of name-prefixes [16].

Use scenarios

Having presented the core of our work on ICN, with pointers to specific results and papers, we introduce two use scenarios of a forthcoming project, and how they can benefit from ICN concepts and functionalities.

1) Mobile video delivery. Video traffic consumes about 60% of the total bandwidth in the Internet and increasingly consumed through mobile devices. The multicast nature of ICN and the inherent support of in-network caching provided by ICN can significantly reduce the amount of resources needed to transport video traffic. In addition, ICN handles mobility much more naturally and without having to store states: when a user changes point of attachment to the network, she will simply ask for the next chunk of the content she is interested in.

2) Network management in disaster situations. ICN can be very helpful in disaster situation by providing at least two key functionalities: i) support of group communications, content-oriented access control and content-oriented security control, so that access to specific information items is allowed only to authorized personnel; ii) ability to operate also in fragmented networks, thanks to its time/space-decoupled model of communications.

Acknowledgements

This paper contains a brief survey of ICN-related work performed in the CONVERGENCE project. We thank all partners of CONVERGENCE and the co-authors of papers cited in this work.

References

[1] V. Jacobson, et al., ”Networking named content”, in Proc. of ACM CoNEXT 2009

[2] D. Smetters, V. Jacobson: “Securing Network Content”, PARC technical report, October 2009

[3] SAIL project website, http://www.sail-project.eu/

[4] PURSUIT project website: http://www.fp7-pursuit.eu

[5] COMET project website: http://www.comet-project.org

[6] NDN project website, http://named-data.org/

[7] CONVERGENCE website: www.ict-convergence.eu

[8] GreenICN: Architecture and Applications of Green Information Centric Networking, a joint EU-Japan research project. http://www.greenicn.org

[9] L. Chiariglione et al.: “Publish/Subscribe over Information Centric Networks: a Standardized Approach in CONVERGENCE”, Future Network & Mobile Summit 2012, 4 – 6 July 2012, Berlin, Germany

[10] ISO/IEC 23006 – Information Technology – Multimedia Service Platform Technologies (MPEG-M)

[11] Detti, N. Blefari-Melazzi, S. Salsano, M. Pomposini, “CONET: A Content Centric Inter-Networking Architecture”, ACM SIGCOMM Workshop ICN 2011, August 19, 2011, Toronto, Canada

[12] S. Salsano, A. Detti, M. Cancellieri, M. Pomposini, N. Blefari-Melazzi, “Transport-layer issues in Information Centric Networks”, ACM SIGCOMM Workshop ICN 2012, August 17, 2012, Helsinki, Finland

[13] N. McKeown et al. “OpenFlow: Enabling Innovation in Campus Networks”, ACM SIGCOMM CCR, Volume 38, Number 2, April 2008. See also http://www.openflow.org

[14] N. Blefari-Melazzi, A. Detti, G. Morabito, S. Salsano, L. Veltri: “Supporting Information-Centric Functionality in Software Defined Networks”, IEEE ICC 2012, June 10-15, 2012 Ottawa, Canada

[15] Detti, S. Salsano, N. Blefari-Melazzi, “IPv4 and IPv6 Options to support Information Centric Networking”, Internet Draft, draft-detti-conet-ip-option-04, Work in progress, November 30, 2012

[16] Detti, M. Pomposini, N. Blefari-Melazzi, S. Salsano: “Supporting the Web with an Information Centric Network that Routes by Name”, Elsevier Computer Networks, vol. 56, Issue 17, 2012.

Nicola Blefari-Melazzi is a Full Professor of Telecommunications and the Director of the Department of Electronic Engineering at the University of Rome “Tor Vergata”, Italy. Dr. Blefari-Melazzi has participated in over 20 international projects and coordinated three EU-funded projects. He is author/co-author of about 170 papers, in international journals and conference proceedings. His research interests lie in the performance evaluation, design and control of communications networks.

Leonardo Chiariglione obtained his Ph. D. degree from the University of Tokyo in 1973. During his career he has launched several standardisation initiatives, such as MPEG in 1988, DAVIC in 1994 and the Digital Media Project in 2003. He is currently CEO of CEDEO.net a technology company developing a technology portfolio in the area of digital media. Dr. Chiariglione is the recipient of several awards: among these the Eduard Rhein Foundation Award, the IBC John Tucker Award, the IEEE Masaru Ibuka Consumer Electronics Award and the Kilby Foundation Award.

INDUSTRIAL COLUMN: SPECIAL ISSUE ON “Content Distribution in Future Mobile Networks”

Guest Editor: Joerg Widmer, IMDEA Networks, Spain

joerg.widmer@imdea.org

The proliferation of powerful mobile devices such as smart-phones, tablets and notebooks with integrated 3G and LTE interfaces have led to an exponential increase in mobile traffic over the past years, and this level of growth is likely to continue over the foreseeable future. Mobile network operators are urgently searching for solutions that allow to transport this traffic without substantially increasing the cost of the network and at the same time provide better network support for content delivery. Video content accounts for the vast majority of the traffic, and it is particularly amenable to traffic optimizations that take into account the characteristics of the data rather than just transporting it is as any other traffic.

This special issue of E-Letter highlights some of the challenges and potential solutions related to content distribution in current and future mobile networks. It is an honor to have four outstanding invited papers from the research laboratories of leading mobile operators and manufacturers that provide an industry point of view on this highly relevant research topic.

Ali Gouta and Yannick Le Louédec of Orange Labs, France report their findings of a “Large scale analysis of HTTP adaptive streaming (HAS) in mobile networks” in the first article of this special issue. With HAS, videos are split into chunks that are made available at different quality levels, allowing video clients to adapt to changing network conditions. The authors analyze traces of video chunk requests from a large mobile network operator. The analysis shows that chunk popularity is biased and first chunks in a video sequence are viewed much more often than later ones. The authors further look into statistics how clients switch between different quality levels. These findings can then be used to improve cache update mechanisms to cache the most relevant chunks. As an example, the authors show that a simple chunk-position threshold beyond which chunks are not cached can significantly reduce the cache update ratio without affecting the cache hit-ratio.

In the second article *“When Mobile Networks meet Big Data and SDN”*, Ulaş Kozat from DOCOMO Innovations, USA argues that future mobile networks need to have much more agile resource allocation mechanisms to deliver content to consumers. Large scale data analysis and software defined networking are identified as the two key ingredients for such a change. Using information about 1) the network, such as utilization and channel quality, and 2) the user, such as location, direct user context and the user’s social context, allow predicting future traffic demands and capacity requirements at a much finer granularity than is possible in today’s mobile networks. With the added flexibility brought about by software defined networking, the network control plane can make decisions based on a timely and global view of the current network state. It also allows using this real-time data to better tailor mobility management, scheduling, and MAC and PHY layer parameters to the requirements of specific traffic flows, resulting in a more scalable, cost effective and faster network.

The third article, “Addressing the Wireless Content Challenge” by Ivica Rimac and Volker Hilt of Alcatel-Lucent Bell Labs, Germany highlights the difficulty of satisfying the demand for high-volume content with current mobile networks and discusses potential improvements to the content delivery. The authors propose to place caches in cellular access nodes and use local breakout to alleviate backhaul capacity bottlenecks. In addition, using local device-to-device communication it is possible to fetch popular content directly from nearby mobile terminals that happen to have requested that content earlier and stored it locally. This reduces the load on the radio access network, as local device-to-device communication usually consumes less wireless resources than a direct link to a base station. It is particularly useful for scenarios that involve a high user density and high correlation in requested content, such as sport events. The authors report first results of the benefits of such content offloading based on measurement data collected in Manhattan.

In the last article, Cedric Westphal of Huawei Technologies and the University of California, Santa Cruz, USA proposes “A Unifying Framework for Resource Allocation in Mobile Information Centric Networks”. Information Centric Networking (ICN) offers some unique features to address the challenges identified in the other papers. Since content is identified by name rather than an address that specifies a network location, not only does this facilitate caching of content in multiple locations, but it also allows to make resource allocation decisions at the granularity of content. Content semantics and meta-data about the content can easily be extracted at the network layer and used for more intelligent resource allocation. The architecture proposed in the paper provides network-level support for advanced content distribution mechanisms that, for example, transmit files only when network conditions are above a threshold or use the predicted user mobility for the decision where to store streaming content.

All of the articles of the special issue emphasize the enormous challenge that mobile networks and network operators are currently facing and propose various solutions that range from near-term optimizations of caching mechanisms to longer-term changes of the overall network architecture as with ICN. This special issue can only provide a brief overview of some novel and exciting research directions, given the substantial amount of work currently being done in industry and academia to address this important and pressing problem.

We would like to thank all the authors for their great contributions and the E-Letter Board for making this special issue possible.

Joerg Widmer is a Chief Researcher at Institute IMDEA Networks in Madrid, Spain. His research expertise covers computer networks and distributed systems, ranging from MAC layer design, sensor networking, and network coding to transport protocols and Future Internet architectures. From June 2005 to July 2010, he was manager of the Ubiquitous Networking Research Group at DOCOMO Euro-Labs in Munich, Germany, leading several projects in the area of mobile and cellular networks. Before joining DOCOMO Euro-Labs, he worked as post-doctoral researcher at EPFL, Switzerland on ultra-wide band communication and network coding.

Joerg Widmer received his M.S. and PhD degrees in computer science from the University of Mannheim, Germany in 2000 and 2003, respectively. In 1999 and 2000 he was a visiting researcher at the International Computer Science Institute in Berkeley, CA, USA. He authored more than 100 conference and journal papers and three IETF RFCs, holds several patents, serves on the editorial board of IEEE Transactions on Communications, and regularly participates in program committees of several major conferences. He is senior member of IEEE and ACM.

Large Scale Analysis of HTTP Adaptive Streaming in Mobile Networks

Ali Gouta and Yannick Le Louédec
Orange Labs, France
{ali.gouta, yannick.lelouedec}@orange.com

1. Introduction

HTTP adaptive streaming (HAS) is getting widely used for video content delivery over Internet and mobile networks. In HAS, several representations of the video content are made available in the network and each of these representations is segmented into chunks that usually ranges from 2 to 10 second length, so that the most adequate chunks can be selected by the client to best fit the network's and terminal's status.

We report here an in-depth analysis at large scale of the behavior of the mobile clients when requesting such HAS-based contents. We also study through simulations the opportunity to enhance the caching logics of proxy caches for such HAS-based contents.

2. Data collection and processing

Our dataset has been collected from five Measurement Points (MPs) spread over the networks of a major French mobile operator. More precisely, the measurement points are located in the mobile backbone on 5 Gi interfaces just above the Gateway GPRS Support Nodes (GGSNs). That way, we collect all HAS sessions of all mobile subscribers for the considered operator in France. The data collection is based on capturing all packet headers of HAS streams that contain useful information, such as the packet size and sequence number. We aggregate all information relative to each persistent-TCP connection and export it to a database, from where it is analyzed. Each persistent-TCP connection corresponds to one downloaded video chunk. This means that the number of downloaded chunks during one HAS session is equal to the number of persistent TCP connections established between the server and the client over a period of time.

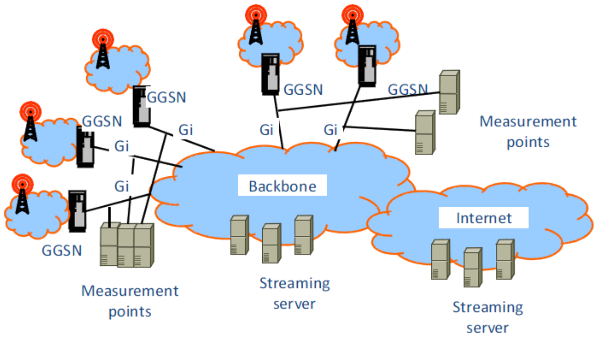


Figure 1. Streaming infrastructure and data collection

The measurements were conducted over a 6 week and one day period, from February 28th to April 10th 2012, involving g 485,544 unique active clients and 8,131,747 HAS sessions. Our dataset mainly contains live TV sessions (where clients watch live TV) and catch-up TV sessions (where TV content providers allow clients to replay a set of videos previously broadcasted in live). Around 70% of the HAS traffic corresponds to live TV sessions, and 30% to catch-up TV sessions. The dataset encompasses 8,127,762 Apple HTTP Live Streaming (HLS) sessions and 3,985 Microsoft Smooth Streaming (HSS) sessions. As it represents the very large majority of the HAS traffic the next sections focus on Apple HLS.

3. Chunk based analysis

Figure 2 presents the distribution of the number of requested chunks per HAS session, for both live TV and catch-up TV sessions from the collected data. The log-normal distribution laws best match (as per the Maximum Likelihood Estimation) the respective Cumulative Distribution Functions (CDF) of the live TV and catch-up TV sessions. Using the Kolmogorov Serminov goodness-of-fit (KS) to assess the accuracy of the estimated parameters characterizing these log-normal laws and setting the confidence level to up to 95%, the estimated parameters of the log-normal laws and the results of the KS test are provided in Table I.

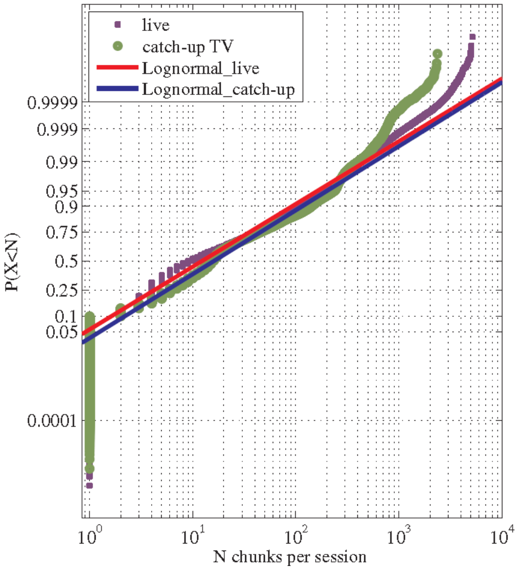


Figure 2. CDF of number of chunks per HAS session for Live and Catch-up TV sessions

Table I. Estimated parameters

	σ	μ	test statistic	p_value
Lognormal_live	1.57033	2.50498	0.1019	0.2368
Lognormal_catch-up TV	1.54	2.76728	0.0839	0.4648

Figure 2 shows that the first chunks in the temporal sequence of a given content are much more requested than the last ones. About 90% of video sessions do not exceed 100 chunks, i.e. 16 min (each chunk containing a 10 second of the video). Moreover, the cumulative distribution functions diverge from the log-normal laws when the HAS sessions have more than 1000 chunks. A minority of clients keeps on requesting chunks over a much longer period than the others, and some until the end of the videos. In the case where the chunks are delivered by edge servers (belonging for example to a CDN or a set of transparent caching servers) such a tailed profile could degrade the caching efficiency of these edge servers, even more if their cache size is limited and their caching replacement algorithm is highly reactive to the end user requests (e.g. Least Recently Used (LRU) algorithm). Therefore it could be wise to investigate the interest of adapting these caching algorithms so that they do not cache the last and very infrequently requested chunks of the videos. Figures 3 and 4 report the results of a trace-driven simulation achieved with such an algorithm. The considered algorithm extends the LRU algorithm with a "chunk-position threshold" beyond which the edge servers do not cache any chunk. The simulation uses a seven-day trace containing 29,921,935 HTTP-client requests from catch-up TV sessions of the collected dataset. For the sake of simplicity the simulation relies on the following assumptions:

- All HAS traffic gets forwarded to a proxy-cache deployed just above the 5 GGSNs.
- The videos are encoded at 500kbps.
- All chunks are 10 second length.
- There is no trick mode during the video session (pause/jump forward/backward).

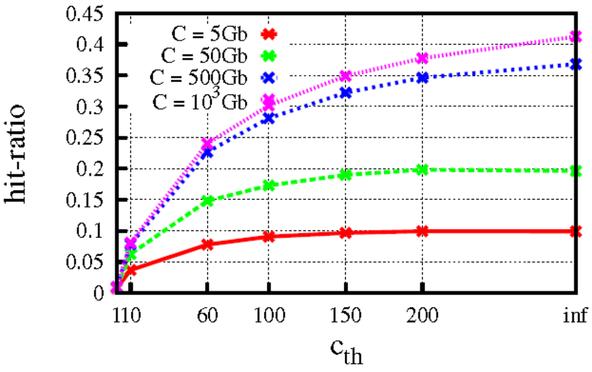


Figure 3. Cache Hit-ratio with respect to Cth

Figure 3 represents the cache hit ratio obtained with this simulation scenario for different cache sizes (C) and for different values of the chunk-position threshold (Cth). Figure 3 shows that for small cache sizes (C<50Gb) setting Cth to 100 leads to approximately the same cache hit ratio as if there were no chunk-position threshold (i.e. Cth = 1).

On Figure 4 we observe that this allows to save up to 50% on the cache-update ratio (i.e. the ratio of requests making the cache updating the list of cached objects due to a cache-miss). For larger cache sizes (C>50Gb), setting Cth to 200 leads to approximately the same cache hit ratio as if there were no chunk position threshold (i.e. Cth = 1), while this allows to gain 20% on the cache update-ratio. This can contribute to significantly reduce the cache replacement processing time, especially in large caches where the object lookup time is critical.

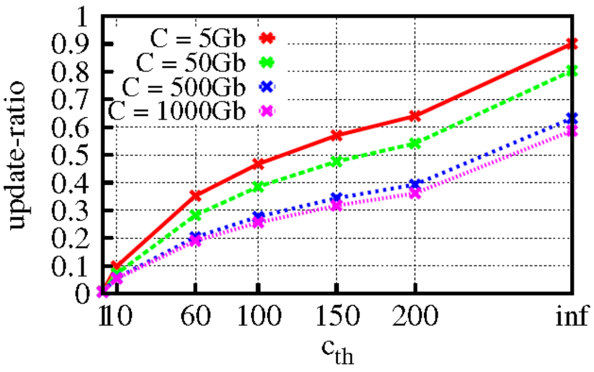


Figure 4. Cache Update-ratio with respect to cth

4. Profiles in HAS

We classify the requested chunks from the collected dataset according to their encoding bitrates (profiles). The encoding schema of the different profiles of video contents may differ slightly from one TV content provider to the other. Hence, we proceed by setting a scale of 8 ranges (see Table II) that closely fit the users' profiles recommended by Apple [1].

Table II. Profiles

Profile i	Encoding bitrate (kbps)
Profile 0	< 50
Profile 1	[50-150[
Profile 2	[150-280[
Profile 3	[280-420[
Profile 4	[420-600[
Profile 5	[600-1000[
Profile 6	[1000-2000[
Profile 7	$\geq$ 2000

To map a given requested chunk to the associated profile, we assume that its size (in bytes) is approximately equal to the volume of information contained in the HTTP payloads transporting it, and that its encoding bitrate (video bitrate + audio bitrate) is equal to its size divided by its duration. The measurement of the size of the chunk starts when the HTTP-200 response to the chunk request is detected. And the duration of the chunk is given in the manifest file.

Figure 5 presents the distribution of the number of visited profiles during the HAS sessions from the collected dataset. Almost 80% of catch-up TV HAS sessions visited only two different profiles, and 40% stay within a single profile during the whole session. This is largely due to the significant proportion of very short catch-up TV sessions as shown in Figure 2: the HAS session is ended by the user before any HAS adaptation happens. This phenomenon is even more pronounced in the case of the live TV sessions. There is no HAS adaptation for almost 60% of the live TV sessions. This is most probably due to the specific behavior of the users watching live TV: they access to a TV channel without necessarily knowing what is currently broadcasted, and they need a few seconds to decide whether to keep or stop watching it.

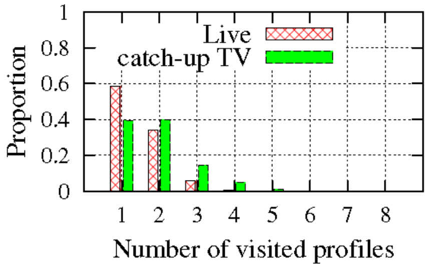


Figure 5. Breakdown of number of visited profiles during HAS sessions

Figure 6 depicts the requests' distribution per profile respectively for the live TV sessions and for the catch-up TV sessions. The sum of the fractions of the requests as per the 8 profiles is higher than one for both the live TV and catch-up TV sessions, since, as shown on Figure 5, a significant proportion of the HAS sessions visit more than one profile. We observe that the clients' requests are more concentrated on profiles 2, 3 and 4 for catchup TV sessions, while the proportion of live TV HAS sessions visiting profiles above profile 3 is marginal. This calls for deeper investigations on how to best manage the specificities of live TV HAS sessions in a mobile context, in both the content preparation and delivery processes, for example with encoding and caching strategies optimized for live and/or for catch-up TV contents. We also observe that profiles 5, 6 and 7, which correspond to encoding bitrates higher than 600 kbps, are rarely visited by catch-up TV sessions. The most probable reason for this behavior is that most of TV broadcasters set a range of encoding bitrates that is below 600 kbps. However with an HSPA or HSPA+ connection, clients could be eligible to request such profiles. We believe that adding higher profiles within the manifest file with higher encoding bitrates would not necessarily cause performance degradation at the client side especially with the deployment of the LTE technology and with the on-going evolution of the capabilities of the client devices.

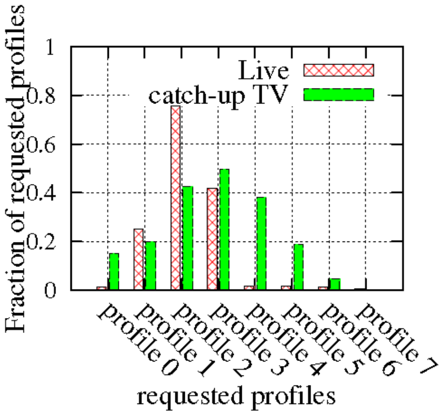


Figure 6. Proportion of requested profiles during HAS sessions

5. Related works and conclusion

Capturing and looking into the real behavior of mobile clients when requesting HAS-based content is based applications. As shown, our findings provide insights that can be leveraged to optimize content caching strategies for HAS-based contents as well as to model users' behaviors. In future works, we will concentrate on the adaptation process by tracking the exact timestamp of clients when switching between profiles in each session. We will also study the implication of such transitions on the cache replacement behavior and propose novel caching policies tailored for HAS adaptation.

Acknowledgments

The research leading to these results has received funding from the European Union's Seventh Framework Programme ([FP7/2007-2013]) under grant agreement n °318398, project eCOUSIN.

References

[1] http://developer.apple.com/library/ios/technotes/tn2224/index.html
[1] L. D. Cicco and S. Mascolo, An Experimental Investigation of the Akamai Adaptive Video Streaming,

2010.

Ali Gouta is a PhD student at Orange Labs and Inria. He holds an engineering degree in telecommunication from Sup'Com "école supérieure des communications de Tunis". His research interests include queuing models, TCP modeling, content delivery networks, caching algorithms and social networks.

Yannick Le Louédec holds an engineering degree in telecommunications from the "Ecole Nationale Supérieure des Telecommunications de Bretagne". He joined France Télécom/Orange in 2001. Since then he has been with its R&D Division, and he is currently the Project Coordinator of the European Projects FP7 OCEAN (Open ContEnt Aware Networks) and eCOUSIN (enhanced COntent distribUtion with Social INformation). His current research interests include video coding, Content Distribution Networks and social enhanced content delivery.

When Mobile Networks Meet Big Data and SDN

Ulaş C. Kozat
DOCOMO Innovations, Inc., Palo Alto CA, USA
kozat@docomoinnovations.com

1. Introduction

LTE enjoys a much faster global adoption than 3G systems and is rapidly changing mobile broadband from a marketing term into a reality. Hence, video has emerged as the main payload being consumed in various forms (live TV, video on demand, user generated content, social content, ads, music clips, games, etc.). In parallel, the combination of open smart terminals, public clouds and fat pipes has been quite successful in creating a rich market space for a plethora of new applications and services. Despite these positive developments, consumers still see a wildly fluctuating experience depending on where they are, what device they use, what content they consume and when they consume it.

To deliver quality of experience, today's mobile networks are designed to react locally without considering larger system implications. For instance, all the best effort data flows (including Skype sessions, YouTube videos, Netflix videos, etc.) converging at the same base station go through the same scheduling policy, such as proportional fair scheduling (PFS) that serves the user with the highest value for the metric $R_i(t)/T_i(t)$, where $R_i(t)$ is the current transmission rate for user i and T_i is the average throughput for the same user until current time t. Even if different flows are differentiated by adding a weighting factor, the same weight is used all the time and in all cells. The same short-sightedness is also exhibited by QoE based traffic optimization techniques (e.g., scalable video) [3], utility based scheduling [4] as well as transport layer rate control [5] that all react to the recent past using closed loop feedback control.

In academic literature, a broader view is taken on content distribution that uses predictions based on historical data such as user location, network utilization, user demand, and social networks. The bulk of these works try to predict the future content to be consumed by a particular mobile user and push this future content when mobile is connected to a hot spot [6-9] or when network utilization is low [1][2].

Moving forward, mobile networks can be designed in a more cost effective way by being more agile in the decision making process for allocating wireless resources to individual traffic flows by incorporating more global network state and trends. In Figure 1, a simple scenario is depicted. Suppose User-1 starts streaming video at point A and throughout the streaming stays in high speed region. At the same time, User-2 starts streaming another video at point C, but unlike User-1, she roams around areas with lower data rate. Serving User-2 more than User-1 while she remains in the blue region has two very appealing benefits: (i) User-2 gets faster service while User-1 has no user-perceived quality degradation. (ii) Possible congestion in green and orange regions is prevented as traffic is offloaded to blue region.

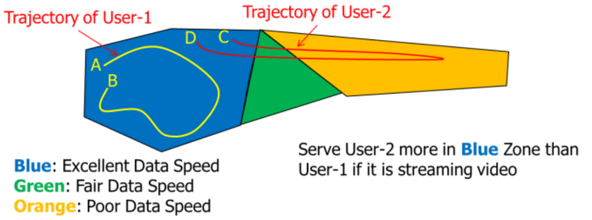


Figure 1: Decision making for wireless resource allocation based on the bigger picture.

In theory these and similar optimization ideas in content distribution bring great benefits in improving user experience and capacity usage, but in practice they require a radical change in how we build mobile networks. Luckily, industry trends around big data, cloud computing and storage, and software defined networks (SDN) finally seem to all come together and make this shift happen. Next, we describe the main ingredients for realizing such forward looking ideas.

2. Big Data Driven Content Distribution using SDN Approach

What are SDNs and what should we expect from Software Defined Mobile Networks (SDMNs)?

SDNs decouple the forwarding plane from the control plane. The forwarding plane is responsible for matching packets to network flows and taking a set of actions on a particular match such as manipulating the packet bits, switching to an output port, buffering, rate limiting, scheduling, and carrying out the transmission. The control plane orchestrates how network flows are defined and what actions should be taken on each network flow. In the special case of OpenFlow [10], one can define flows using L1 through L4 header fields and specify outgoing port(s) for a flow match.

In mobile networks, to realize the agility highlighted in Section 1, the forwarding plane should and can expose the control of the following functions:

Mobility Management. In current cellular networks, the mobility management protocol is built into the forwarding elements. For instance, in LTE, mobility between base stations (i.e., eNBs) is handled with a tight coordination and signaling between eNBs. The current eNB signals the target eNB to set up the radio bearers for the mobile device, export link layer context, and transfer buffered packets. Although not all flows are equal, they are treated as one by using the same handover delay, outage, and packet loss rate targets. Considering that video is already the main payload and small cells get more widespread, a more opportunistic strategy that risks more packet losses at substantially higher rates can be adopted on a per flow basis. This can be enabled if base stations report end user signal qualities to a network controller, which then notifies the corresponding mobility application to orchestrate with eNBs in how to handle mobility of each flow.

Wireless Scheduling. To realize the scenario in Figure 1, we must expose scheduling function at base stations so that a network controller can dynamically program the scheduling priorities of different flows based on a higher degree of information about users (e.g., their trajectory, current activity, preferences) and the network (demand forecasts and capacity utilization) that is not readily available in a base station. It is important to provide a scheduling abstraction where each base station is only concerned about the current priorities of queued flows and runs a simple enough scheduling algorithm (e.g., weighted PFS) that utilizes short-term opportunities while having the controller guess long-term opportunities and adapt priorities.

Radio Link Layer (RLL). Retransmissions at the link layer constitute an important mechanism for increasing link reliability. As a mechanism to cope with poor TCP performance due to wireless losses, it is not an uncommon approach to use aggressive link layer retransmissions. Controllability of this parameter on a per flow basis enables low delay and high throughput (when a more capable transport layer protocol is used) or low delay and low throughput operation regimes. Another important factor that can be controlled is to enable/disable CRC checks. If a video flow uses a decoder with good error concealment techniques, even if a few bits are falsely decoded, it can improve the quality. Note that the RLL has two ends: base station and mobile device. Thus, programmability of RLL parameters might affect both ends.

Physical Layer. Although it might be too ambitious to suggest migrating the adaptive coding modulation, precoding, and power adaptation decisions from base stations to a network controller (due to the time scales involved), physical layer targets such as block error rate or peak transmission power can be instructed by an external controller. Moreover, if there are fixed transmission profiles/templates or adaptation (rate or power) is not desired due to a global consideration, these can be also set by an external controller. For instance if a network operator wants to use small cells always at peak rate and at a particular transmit power, it should be able to do so.

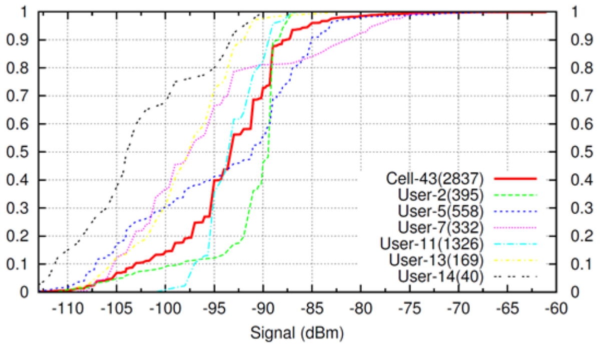


Figure 2: Signal strength distribution for various users served by the same cell ID.

What is Big Data for Mobile Content Distribution?

We can narrow down the big data in the context of content distribution to any data that would help in estimating the current snapshot and predicting the future snapshot of capacity coverage and user demand. Below we list some of them.

User Location: This information alone, even at a coarse granularity (e.g., cell ID, paging area.), conveys a vast amount of information on the trajectories of subscribers as individuals and as groups. Further, it reveals correlations across subscribers, time, and space. Combined with side information on maps, roads, public transport routes and schedules, the accuracy of trajectory estimation and prediction can be boosted significantly without burdening the mobile devices. Trajectory information can tell us at what time scales, what kind of transmission opportunities may arise for mobile content distribution. However, this information must be considered jointly with other data sets to be useful for content distribution.

Channel Quality: When combined with location information, this data set can reveal information on which locations tend to perform better for which terminals. Figure 2 shows a snippet of data collected as part of ongoing research collaboration between the University of Michigan, Ann Arbor and DOCOMO Innovations. The figure plots the cumulative distribution of received signal strength at different 3G terminals under a popular cell ID (i.e., visited most).

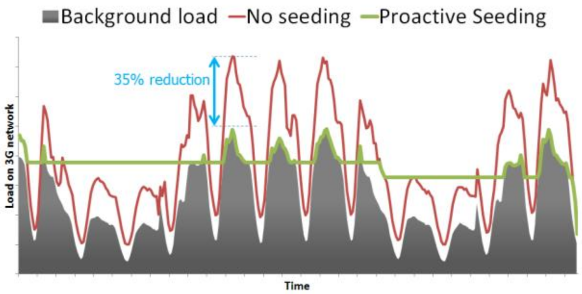


Figure 3: Traffic offloading by online social network analytics.

Results with LTE terminals also exhibit similar behavior. Clearly the same location has different channel quality meaning for different users. For instance, user-14 always operates below the median quality of user-2. Although not shown, when signal strength distributions for the same user are plotted for different locations, one can also see a clear separation between good and bad locations for many users from the signal coverage point of view.

User Context: Whether a user is moving or stationary; if moving, whether she is running, biking, driving or using a particular public transport; if stationary, whether in a stable (e.g., home, office, etc.) or a transient location (e.g., train stop) can all be used as inputs to schedule content distribution and modify network flow priorities.

Network Utilization: Both backhaul link and wireless capacity utilization data are of critical importance to determine where the bottlenecks are occurring (if any). This data can serve as an input to (re)route content distribution around the bottleneck nodes and links.

Social Context: People influence each other’s behavior and content consumption is not an exception. A content consumption model based on social network analysis can provide a very useful tool in identification and efficient distribution of viral content [1][2]. When combined with network utilization trends, one can move content that is likely to be consumed in peak utilization hours to lower utilization hours or from busy cells to less busy cells. Figure 3 shows using real network usage data and superimposing social network driven data usage, that a 35% reduction in peak utilization is possible. The details can be found in [1].

3. Final Remarks

Future mobile networks should not be only about capacity increase, but also about agility in utilizing the existing capacity. SDN and Big Data have the potential to pave the way for this next frontier in mobile networks. Today we are still missing a solution that streamlines real-time (or near real-time) data collection in mobile networks and carries the data over to the control plane in a scalable, cost effective and fast manner. We are also yet to see a control plane for mobile networks that can make decisions with a more global view to evenly utilize network resources, improve user perceived quality, and increase network efficiency as new flows join and existing flows move around.

References

[1] F. Malandrino, M. Kurant, A. Markopoulou, C. Westphal, and U.C. Kozat, “Proactive seeding for information cascades in cellular networks,” in IEEE INFOCOM’12, pp. 1719–1727, March 2012.

[2] H. Sharara, C. Westphal, S. Radosavac, and U.C. Kozat, “Utilizing social influence in content distribution networks,” in IEEE ICC’11, pp. 1–6, June 2011.

[3] S. Thakolsri, W. Kellerer, E. Steinbach, "QoE-Based Cross-Layer Optimization of Wireless Video with Unperceivable Temporal Video Quality Fluctuation," in IEEE ICC’11, pp. 1–6, June 2011.

[4] F. Fu and U.C. Kozat, "Stochastic Game for Wireless Network Virtualization," in IEEE/ACM Transactions on Networking, vol.21, no.1, pp.84,97, Feb. 2013.

[5] K. Igarashi, M. Yabusaki, "Mobility aware TCP congestion control," in IEEE ISWPMC’02, vol.2, pp.338-342, Oct. 2002.

[6] D. Goodman; J. Borras; N. Mandayam, B. Narayan, R.D. Yates, "INFOSTATIONS: a new system model for data and messaging services," in IEEE VTC’97, vol.2, pp.969-973, May 1997.

[7] T. Ye, H.-A. Jacobsen, R. Katz, “Mobile awareness in a wide area wireless network of info-stations,” in ACM/IEEE MobiCom'98, pp. 109-120, Oct. 1998.

[8] A.L. Iacono and C. Rose. "Infostations: New perspectives on wireless data networks." In Next Generation Wireless Networks, pp. 3-63. Springer US, 2002.

[9] G. M. Lee, S. Rallapalli, W. Dong, Y.-C. Chen, L. Qiu, Y. Zhang, “Mobile Video Delivery via Human Movement,” in IEEE SECON 2013, June 2013.

[10] OpenFlow Switch Specification, Version 1.3.1, Open Networking Foundation, September 2012.

Ulaş C. Kozat works at DOCOMO Innovations as principal researcher. He also serves as adjunct faculty at Özyeğin University, Istanbul. Dr. Kozat received his B.S. in Electrical and Electronics Engineering at Bilkent University, Ankara in 1997 and his M.S. in Electrical Engineering at the George Washington University, Washington DC in 1999. He received his Ph.D. in Electrical Engineering from the University of Maryland, College Park in 2004. He is a Senior Member of IEEE. Dr. Kozat conducts research in the broad areas of computer and communication networks.

Addressing the Wireless Content Challenge

Ivica Rimac and Volker Hilt
Bell Laboratories, Alcatel-Lucent, Germany
{*rimac, volker.hilt*}*@bell-labs.com*

Abstract—The ever increasing demand for high-volume content in the Internet is imposing heavy demands on mobile data networks. The growth in demand is forecasted to far outpace the increase in capacity through advances in wireless technologies. To alleviate this problem, we propose to serve user requests for content from caches in cellular access nodes and from other mobile devices located geographically close to the requesting user.

1. Introduction

With the arrival of smartphones and tablets, mobile broadband usage has been increasing at an alarming pace. The demand from mobile users listening to music, browsing pictures, and watching videos is predicted to overload cellular networks soon.

Today's solutions to mitigate the problem are mostly focused on the wireless infrastructure; e.g., LTE network upgrade and WiFi offloading to increase downlink and uplink peak rates, deployment of small cells to ease congestion from macro cells. The increase in wireless capacity in turn also requires upgrades to the mobile backhaul network, which has become a bottleneck in many wireless network deployments. However, simply increasing network capacity is not sustainable. The capacity increase cannot keep up with the pace of the accelerating demand, and flat *average revenues per user* **(**ARPU) challenge the justification for the capital expenditure required for the above solutions.

Evolving mobile network architectures and standardization efforts might enable complementary and less expensive solutions by leveraging concepts from content delivery networks and peer-to-peer communication. More specifically, local breakout allows deploying caches in small cell sites that can reduce backhaul capacity requirements and direct device-to-device exchange can ease congestion on the last mile to the user.

In this paper, we discuss the aforementioned alternatives to wireless infrastructure solutions for increasing the effective capacity and the quality of experience of the mobile users. In Section 2, we first provide an overview of the scenarios and the use cases in this paper. Section 3 focuses on content delivery using caches in small cell sites to ease congestion on the wireless backhaul, followed by Section 4 on mediated device-to-device delivery of content to increase effective data rate to the mobile users. We conclude the paper in Section 5.

2. Scenarios and use case

This paper focuses on solutions for content delivery over operator-deployed mobile networks. The objective is to present generic concepts that are applicable to a broader class of network architectures independent of access technologies. We assume certain mechanisms provided by mobile network, such as local breakout and direct communication, which might not yet be widely deployed but are well established in literature and proposed for inclusion in standardizations.

We investigate the following two problem domains:

- Backhaul capacity bottleneck: capacity hotspot in dense urban areas with small cell deployment, for which the wireless backhaul to the macro station represents the capacity bottleneck [1].
- Last-mile bottleneck: capacity hotspot in dense urban areas (potentially with small cell deployment), for which the last-mile radio link represents the capacity bottleneck.

To address the backhaul bottleneck, we leverage local breakout and caching in small cells, while for the last-mile bottleneck we leverage direct device communication opportunities (e.g., WiFi or LTE Direct).

We concentrate the discussion on solutions for reactive content delivery, i.e., the immediate delivery of content in response to a user request. These techniques are general enough to cover other use cases with relaxed timing requirements, such as predication-based or subscription-based services; the latter can be further enhanced by content scheduling algorithms such as proposed in [2].

3. Backhaul capacity bottleneck

We assume that content popularity follows power law, for which the golden standard is to deploy content caches behind the capacity bottlenecks. This leaves the small cell sites, the UEs, or both as potential locations for placing the caches. From the perspective of caching efficiency, the latter generally increases with a larger user set served. Thus, the preferred location for caching is in the small cell sites using local breakout, as depicted in Figure 1.

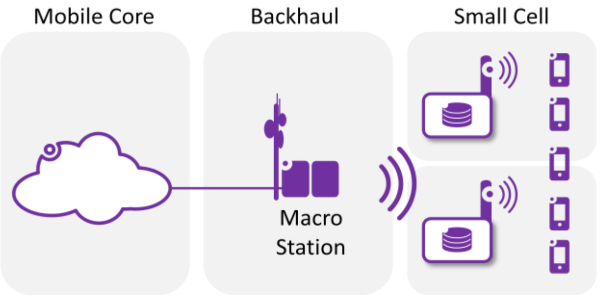


Figure 1: Caching in small cells to ease backhaul congestion.

Application-layer caching solutions are well suited for the core of the network; however, they are challenging to deploy in the above environment. Transparent caching solutions require the interception of user requests and TCP splicing to not break the end-to-end context. Moreover, they have to process and maintain state for all content requests, whether they are cacheable or not. Adding such functionality in each small cell site will significantly increase their power and processing requirements and thus system cost. As a remedy, we propose to enhance the transparent proxies in the core network with explicit redirection functionality. The message flow is then as follows:

1. The user request is sent toward the origin server.
2. The redirection node in the core intercepts and responds with a redirection message (e.g., HTTP 302) containing the address of the user's small-cell cache; this would potentially be a network local (anycast) address.
3. The user resends the request to the local cache.
4. The local cache receives the connection request, if necessary contacts the origin server, and serves the content to the user.

An issue that needs to be addressed is the increased probability of hand-overs for a user. While for short-lived sessions this is potentially less of an issue, it can become a real problem for longer HTTP streaming sessions using persistent TCP connections. One potential solution is to continue to route the packets of a user session to the cache the user started with, thereby hairpinning traffic over the backhaul network. Since the backhaul network is a bottleneck, it would be preferable to avoid hairpinning and hand over the session to the cache in the new cell. However, this will break existing HTTP/TCP connections and, if not detected and acted upon quickly, impair the quality of experience of the user. Thus, with today's TCP/IP stacks that don't allow dynamic rebinding of TCP sessions, the application has to provide for the session management in the application.

4. Access capacity bottleneck

For the capacity bottleneck in the last mile, traditional caching becomes ineffective: any requested content has to be pulled once over the bottleneck to a user device but the probability of reuse by the user is close to zero for most content. Thus, we propose to leverage direct connectivity (WiFi and LTE Direct) between end devices as depicted in Figure 2.

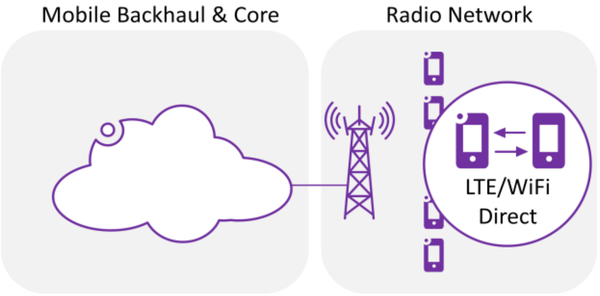


Figure 2: Leveraging direct connectivity to ease congestion in the radio access.

This solution is conceptually enhancing regular service with a peer-to-peer (p2p) component, for which trends are changing in favor of: smartphones and tablets provide plenty and extendible storage, and a small number of web sites and services are becoming extremely popular drawing many mobile users to access content from them (e.g., YouTube, ESPN, Hulu). Furthermore, standards for scalable, low power, and highly reliable wireless device-to-device communication such as WiFi and LTE Direct are emerging, which presents a greater potential for exploiting P2P data exchanges to offload cellular traffic, particularly during peak hours.

We propose to exploit the observation that cellular networks are most strained during high-density events and in capacity hotspots. Importantly, these are also the scenarios where the opportunity of P2P transfers is maximized. One approach is to have the operator track the location of participating devices and build a map that indicates where the device clusters are reasonably dense, so called *data spots* [3]. These data-spot maps are periodically pushed to the phones. If the operator can also maintain a digest of available content in different devices in that area, it can perform a match-making service and notify a requesting device to connect to the appropriate device for the desired content through direct mode. In this process, the wireless interface can be woken up only when the transfer is taking place, which obviates the need for a device to constantly scan for other devices in the vicinity, making it an energy-effective solution. Moreover, the knowledge of cached content in each phone does not spread among peer devices, avoiding privacy concerns.

Finally, operator-mediated transfers are also amenable to accounting—a mobile device that has served content may be appropriately rewarded at the end of the month.

After having schematically described the concept, we now focus on understanding the viability and rough performance gains expected of such a solution. We performed a set of simulations to address the question using the following baseline setting (based on measurement data collected in Manhattan [3]): (1) average device density of 2912 devices/km^2; (2) a realistic human mobility model described in [4]; (3) content popularity following Zipf law with skew parameter $\alpha = 0.66$; (4) content library set of size 10,000; (5) device caches sized to hold 50 objects. Table 1 summarizes some of our simulation results.

Table 1: Expected benefits of device-to-device content delivery for different settings.

Scenarios	Hit Rate	Failures
Crowded Areas (Manhattan)	36.27%	3.5%
Less Crowded (1/10 of Manhattan)	9.57%	1.13%
Less Crowded (larger cache space)	14.28%	1.19%

The results depicted in Table 1 show that naturally clustered mobile devices can offload cellular traffic by up to 36% if all requests are for the same content library. At the same time, following the data spot map and content digest info, devices unsuccessfully trigger direct mode for less than 4% of the requests.

To summarize, while our results are not conclusive, they are strongly indicative of the potential benefits of leveraging device-to-device communication for mobile content delivery.

5. Conclusion

Meeting the increasing demand for content in wireless networks is one of the key challenges for service providers. The key bottlenecks in wireless networks are the access and the backhaul capacity. In this paper, we have introduced two approaches to address these bottlenecks: deploying caches in small cell sites and direct device communication. We have discussed the technical feasibility of both approaches and show that using content distribution techniques provide important tools to augment a wireless capacity build out. These techniques enable wireless service providers to meet the wireless content challenge in an economically sustainable way.

References

[1] Small Cell Forum Release One, "Backhaul technologies for small cells: Use cases, requirements and solutions," Document SCF049, February, 2013.

[2] R. Bhatia, I. Rimac, G. Narlikar, and A. Beck "UNAP: User-centric network-aware push for mobile content delivery," in *Proc. of IEEE INFOCOM*, pp. 2034–2042, 2009.

[3] X. Bao, Y. Lin, U. Lee, I. Rimac, and R. R. Choudhury, "DataSpotting: Exploiting naturally clustered mobile devices to offload cellular traffic," in *Proc. of IEEE INFOCOM*, 2013.

[4] K. Lee, S. Hong, S. Kim, I. Rhee, and S. Chong, "Slaw: A new mobility model for human walks," in *Proc. Of IEEE INFOCOM*, 2009.

Ivica Rimac is a senior research member of Networked Services Research at Bell Labs, the research organization of Alcatel-Lucent. Ivica joined Bell Labs in 2005 after receiving his Ph.D. in electrical engineering and information technology from Darmstadt University of Technology, Germany. His field of research is computer networking and distributed systems where he has co-authored numerous papers and patents, among others in the areas of content distribution and delivery over the Internet and mobile networks.

Volker Hilt is the head of Networked Services Research at Bell Labs/Alcatel-Lucent in Stuttgart, Germany. Volker received his master's degree in Information Systems in 1996 and his Ph.D. in Computer Science in 2001 both from the University of Mannheim in Germany. His field of research is computer networking where he has made contributions in the areas of cloud computing, content distribution networks, peer-to-peer applications, distributed multimedia systems and the Session Initiation Protocol (SIP). Volker is a contributor to the Internet Engineering Task Force (IETF) and has published many papers, Internet drafts and RFCs.

A Unifying Framework for Resource Allocation in Mobile Information Centric Networks

Cedric Westphal
Department of Computer Engineering, University of California, Santa Cruz
& Huawei Technologies
cedric.westphal@huawei.com

1. Introduction

Content distribution has become the predominant form of traffic over the Internet; yet, this shift has happened "over the top" from the point of view of the operator: the content is placed and distributed in content distribution overlays which relegate the operator to the role of a dumb pipe. However, content distribution would benefit from the operator's involvement: the knowledge of the network, its topology, what features it supports, the congestion level of links, etc, would all improve the end-user experience while consuming content.

Information-Centric Networks (ICN)[1-6] have been proposed to help the operator route to content. The network architectures suggest using content name for routing, so as to direct a request for a specific piece of content to the copy which maximizes the utility of the operator. Requesting an item by name, rather than by address, frees the operator to decide which location to use. The operator can then make its decision based upon availability of the content at multiple locations and the network conditions between the user and these locations.

We argue further that Information-Centric Networks offer further benefits to the operator beyond selecting which cache to use. Namely, the use of a single globally unique name for content allows the operator to perform fined-grained resource allocation decision based on the properties of the content associated with this name.

In this letter, we describe an architecture initially presented in [7,8] to construct an Information-Centric Network architecture with a centralized content management plane. This architecture extracts content meta-data from the content going through the network and uses this meta-data for resource allocation. We argue that content is the proper granularity for resource allocation, as it makes the resource allocation problem more deterministic rather than probabilistic. We will see how this framework can be expanded for mobile networks.

2. ICN Overview and Background

Information-Centric Networks associate content with a unique name which is then used to request content. A typical request will have different semantics under different ICN proposals, but will be of the form GET(name).

In an ICN, because the request is for a specific name, it is not attached to any location, unlike IP. This gives the network operator the ability to use caching mechanisms to store the content within the network, and to route content requests to any copy of the content.

The idea to make routing decisions in a centralized manner has taken a new life recently with the emergence of Software-Defined Networks (SDN)[9]; in SDN, the forwarding plane is separated from the control plane, and the control plane has a global view of the network that allows a (logically) centralized controller to make routing decisions based on this view of the network's conditions. The controller then enacts the routing decision by setting flow labels in the flow tables of the switches in the forwarding plane.

A flow is a natural primitive to work with in an IP context, as it is understood by the existing forwarding elements. However, it is the wrong granularity for content routing. In the remainder of this document, we assume that we have an architecture which supports a separation of control and forwarding planes; where the control plane sets up content forwarding rules in the data plane; and where the data plane is able to forward based upon content name.

Such an architecture can be built from scratch as in some ICN architectures (say, [10]), or can be constructed on top of IP (say, as an extension of OpenFlow as in [8]).

3. Resource Allocations in ICN

The conjunction of a centralized content management with uniquely named pieces of content allows for the network to transparently extract content meta-data [7].

The centralized controller includes a content management layer which keeps a database indexed by the content name which keeps track of its popularity, its size (as read by counting bytes through the network or by the memory footprint in a cache server) and its potential locations.

Content has explicit boundaries. Once the size of a piece of content is known, it is straightforward to know how much data is left in transit. This contrasts with flows in IP networks which have no explicit end semantics. Once a flow is allocated to a link, it is as difficult to predict the link utilization as to predict when the flow will terminate. This provides a significant advantage for content-based resource allocation.

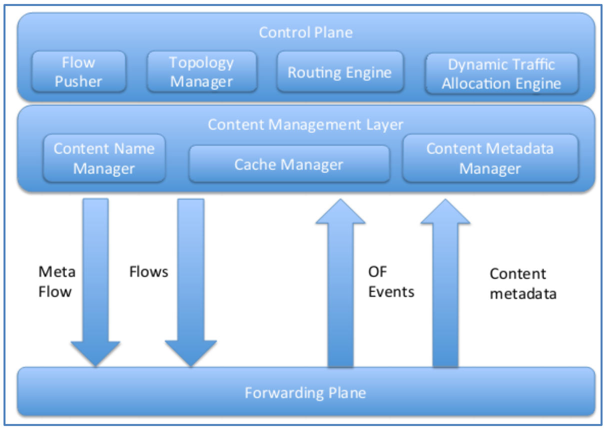


Fig. 1*: Meta-data extraction [7]*

Upon receiving a content request, the forwarding plane invokes the control plane to resolve the content name to a path in the network. The controller then identifies the location of the content and the path that jointly maximize some utility. For instance, a typical traffic engineering goal would be to minimize the maximal link utilization in the network. This is straightforward once the additional load of adding the content to a path is known.

The key insight is that a content-based architecture transforms the resource allocation problem from a *probabilistic* problem to a *deterministic* problem. This in turn strongly diminishes the need for guard bandwidth and other mechanisms to protect the network from burstiness in the traffic, thus reducing significantly the capex of the network.

4. Mobile Networks

We have seen how an information-centric architecture allows for better resource allocation in wired networks, and in particular for traffic engineering.

We now discuss through examples how an information-centric architecture with a (logically) centralized controller can improve resource allocation in mobile networks, and how it would provide a homogenous management framework for multiple mobile optimizations which use different mechanisms.

There are multiple examples of content-based optimizations for mobile networks, but we will focus on two as an illustration of the need for an architectural solution. These two mobile optimizations are Sprinkler [11] and Infinity [12]. Many other optimizations would fit in such a framework, and this underlines the need for the network to natively support such optimizations.

Infinity is an in-network storage system which increases the wireless utilization by scheduling transmission of files when the network conditions are above a threshold. Sprinkler is a streaming system which stores the content in the network by predicting the user's mobility.

These two mechanisms can be viewed as network functions which require a programmatic framework at the content level to be implemented satisfactorily.

Infinity can be implemented in the architecture of [7] by having the mobile device request the content, tagged with a deadline by which to transmit the content. This request is then fulfilled once the link quality between the mobile device and the user exceeds some threshold or when the deadline expires.

Sprinkler can similarly be viewed as an information-centric architecture, where the network schedules the delivery based upon a predicted trajectory.

There are many other such mobile network optimizations which can be expressed using a content-based framework which supports relatively simple network programmability.

The point is thus: resource allocation can gain from being performed at the level of content, using a control plane with a global network view in a network with the ability to store content, in exactly the same way traffic engineering benefitted in a wired network.

5. Summary and Outlook

Information-Centric Networking (ICN) has become an active research topic for content delivery. One of the unsung advantages of an ICN is that it allows resource allocation at the granularity of content. Since content has explicit semantics, and meta-data about the content can be extracted at the network layer, the resource allocation is more accurate than in an IP network where it relies on statistical multiplexing for accuracy.

We have seen that an architecture with the following set of features:

- The content is uniquely identified and requested by name;
- The forwarding plane can store content;
- The forwarding plane can extract content metadata;
- The centralized control plane keeps track of content and its metadata;
- A programmatic interface is provided to implement new network and content management functions in the controller;

would offer significant improvements for traffic engineering. We also argue that such a framework would allow implementing many mobile networking optimizations which have been proposed in the literature using a common underlying architecture.

References

[1] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs and R. L. Braynard, "Networking named content," in *Proc. CoNEXT*, 2009.

[2] D. Dannewitz, et al., “Scenarios and research issues for a network of information”, in *Proc. MobiMedia*, Oulu, Finland, July 2008.

[3] K. Pentikousis, et al., “Information-centric networking”, in *IEEE Commun. Mag.*, vol. 50, no. 7, July 2012.

[4] D. Kutscher, et.al, “The Network of Information: Architecture and Applications”, FP7 SAIL *Public Report D-B.1*, 2011.

[5] Trossen, D. and G. Parisis, "Designing and realizing an information-centric internet", in *IEEE Commun. Mag.*, vol. 50, no. 7, 60-67, 2012.

[6] Ahlgren, B. et al., "Design considerations for a network of information", in *Proc. CoNEXT Re-Arch Workshop*, 2008

[7] Chanda, C. Westphal, R. Raychaudhuri, “Content Based Traffic Engineering in Software Defined Information Centric Networks”, in *IEEE INFOCOM NOMEN workshop*, April 2013.

[8] Chanda, C. Westphal, “ContentFlow: Mapping Content to Flows in Software Defined Networks”, *arXiv:1302.1493*, February 2013.

[9] “Software-Defined Networking: The New Norm for Networks”, in *ONF*, 2012.

[10] PURSUIT project, available at *http://www.fp7-pursuit.eu*, 2011.

[11] S. K. Dandapat, S. Pradhan, N. Gangulyk, R. Choudhury, “Sprinkler : Distributed Content Storage for Just-in-Time Streaming*" in ACM MobiSys workshop CellNets* 2013.

[12] Y. Gao, Z. M. Mao, P. Padala,U. Kozat, “Infinity: In-Network Storage for Mobile Devices,” presentation at *GENI GEC11*, 2011.

Cedric Westphal is a Principal Research Architect with Huawei Innovations working on future network architectures, both for wired and wireless networks. His current focus is on Information Centric Networks. He also has been an adjunct assistant professor with the University of California, Santa Cruz since 2009. Prior to Huawei, he was with DOCOMO Innovations from 2007-2011 in the Networking Architecture Group. His work at DOCOMO has covered several topics, all related to next generation network architectures: scalable routing, network virtualization and reliability, using social networks for traffic offloading, etc. Prior to that, he was at Nokia Research Center from 2000 to 2006. He received a MSEE in 1995 from Ecole Centrale Paris, and a MS (1995) and Ph.D. (2000) in EE from the University of California, Los Angeles. Cedric Westphal has co-authored over fifty journal and conference papers, including several best paper awards; and been awarded twenty patents. He has been an area editor for the ACM/IEEE Transactions on Networking since 2009, an assistant editor for (Elsevier) Computer Networks journal, and a guest editor for Ad Hoc Networks journal. He has served as a reviewer for the NSF, GENI, the EU FP7, and other funding agencies; he has co-chaired the program committee of several conferences, including IEEE ICC (NGN symposium). He is a senior member of the IEEE.

MMTC OFFICERS

E-LETTER BOARD MEMBERS