Tutorial: Exploring NDN Research through Real World Problem Solving

Alex Afanasyev
UCLA
aa@cs.ucla.edu

Jeff Burke
UCLA
jburke@remap.ucla.edu

Davide Pesavento
UPMC
davide.pesavento@lip6.fr

Beichuan Zhang
U. Arizona
bzhang@cs.arizona.edu

Lixia Zhang
UCLA
lixia@cs.ucla.edu

ABSTRACT

This full-day tutorial on application development over Named Data Networking (NDN) aims to help the audience get a quick start with NDN research. We plan to first provide a brief overview of the NDN protocol architecture and then illustrate how one can apply the basic architectural concepts to simple application scenarios. In particular, we will introduce the existing open source NDN codebase and show how one can explore new research questions via application development and running them in emulated and simulated network environments.

Keywords

Named Data Networking (NDN), Information Centric Networking (ICN), applications.

1. INTRODUCTION

As the interest in ICN research continues to grow, many people become interested in getting involved, but do not see clearly where is an easy entry point into this exciting new area. Based on our experience from the application-driven NDN architecture development over the last six years, we believe that an easy and productive way to get started in ICN research is to build simple applications over NDN and run them in real, emulated, and simulated environments. Such first-hand experience with the architecture, its available open source libraries, frameworks, tools, and environments, can help one clarify and deepen the conceptual understanding about this new networking paradigm, see its advantages, appreciate the differences with the existing TCP/IP protocol stack, as well as identify new research challenges.

The tutorial would be most beneficial for students and other researches who want to gain further understanding of ICN beyond reading the literature and are looking for interesting ICN research topics to work on. The main goal of the tutorial is to offer a booster to get one started through interactive and fun mental exercises together with hands-on experiences.

ICN'16, September 26-28, 2016, Kyoto, Japan

DOI: http://dx.doi.org/10.1145/2984356.2984379

2. TUTORIAL DESCRIPTION

This tutorial is expected to run 8 hours, including a 1-hour working lunch and two 30-min breaks in the morning and afternoon. We expect that the breaks will provide valuable time gaps for people to socialize, clarify missed points, catch up in their coding exercise, and to exchange their experience.

2.1 NDN Architecture Overview (0.5 hour)

We plan to illustrate the basic concepts of the NDN architecture with specific examples to help both create an interactive learning atmosphere and demonstrate the key differences between the existing TCP/IP and the NDN protocol architectures.

2.2 A Day in NDN Development Wonderland (4.5 hours)

2.2.1 A Quick NDN Codebase Overview (1 hour)

We plan to provide an overview of the existing codebase, including NDN Forwarding daemon (NFD), NDN C++ library with eXperimental eXtensions (ndn-cxx), NDN Common Client libraries (NDN-CCL), and port of NFD on Android platform (NFD Android). We will use the existing NDN testbed and application examples to illustrate how the code is being used today.

2.2.2 Contributing to NDN Open Codebase (0.5 hour)

Here we plan to illustrate the overall development process of the core NDN components that the NDN team has been using for codebase development over the last several years. We will describe how to discover known missing pieces in the software, how to start with code development, how to submit code to be reviewed by NDN team, and how to respond to NDN team comments so your contributions will be quickly merged. Information covered in this section will be particularly useful for those who may later join NDN's open source development effort. A new architecture needs the broader community's effort, and you would be proud of your contributions 20 years down the road!

2.2.3 Developing Simple Simulations with ndnSIM (1 hour)

Sometimes it may be difficult to start with a real NDN environment, or test your idea in a non-trivial setting. Simulations can then overcome these constraints. ndnSIM is a pretty popular package that have been used over the last several years and helped produce over 270 publications according to Google Scholar.

In this section of the tutorial we will use a set of pre-planned and/or participant-suggested scenarios and show what needs to be

done to run the simulation-based evaluation.

2.2.4 *Developing Simple NDN Applications (1.5 hours)*

Everyone must have had the experience that they hear something that seems very simple, but when they sit down to do it, they could not figure out how. Here is your opportunity of taking what you have learned so far during the day and do an exercise to enhance the learning. Depending on the participants background, we consider splitting into several groups where participants together with the mentors will go through a complete process of developing a new NDN applications: what code can be used as a bootstrapping point, what are the next steps, and how to run the application. Given the importance of the security in NDN, we will also show what is expected from the developer, user, and operator to secure the applications.

2.2.5 *Experimenting with NDN Apps using Mini-NDN (0.5 hour)*

Although, the developed applications can be run in real environments, such as NDN Testbed, it may be needed to evaluate applications in more controlled environments. In this section of the tutorial we will demonstrate how one can use an NDN network emulator, Mini-NDN, to emulate multiple NDN nodes to run and evaluate the applications on a single system. With Mini-NDN, users do not need to modify their application code that may be necessary to for simulation purposes or reserve servers on emulation services like Emulab.

2.3 Q & A (1 hour)

During the tutorial, we will set up an online board to collect questions about the basic ICN/NDN concepts and the programming exercises. During this last section of the tutorial, we plan to break into smaller groups, with each addressing a specific area of questions that people may bring up.

3. EXPECTED OUTCOME

We hope this tutorial would help open the door for people to get into NDN research. We plan to put the tutorial recording online in case anyone wants to refresh their memory, and even better, introduce their colleagues and friends to watch and learn.

4. ADDITIONAL INFORMATION

4.1 Type of the Tutorial

After a short introduction of the architectural concepts, the lecture part of the tutorial will be focused on description of the available open source prototype frameworks, their structure, and how to get started to use them. In parallel, we will run hands-on tasks implementing simple applications using the frameworks. We are hoping for an active engagement with the tutorial participants, answering questions from the participants on how NDN architecture helps with their research and applications needs, showing and writing of real examples of code in real-time.

4.2 Previous Tutorials

The NDN team has provided tutorials, including hands-on workshops, at venues including:

- ACM 2nd Information-Centric Networking Conference (ICN), September 30, 2015, San Francisco (≈30 attendees)
- GENI Engineering Conference 21, October 20-23, 2014, Indiana University (≈30 attendees);
- ACM 1st Information-Centric Networking (ICN) Conference, September 24, 2014, Paris, France (≈60 attendees);
- NDN Community Meeting 2014, September 3, UCLA (≈50 attendees);
- AsiaFI NDN Hands-on Workshop, March 19-21, 2012, Seoul National University, Korea (≈30 attendees).

The tutorials covered a number of topics, including overall description of the NDN architecture, information on how to implement applications using the basic architectural primitives, and "intermediate" topics such as multi-party synchronization and security. In this tutorial we would like to focus on describing and real-time writing of a set of simple applications that demonstrate unique advantages of NDN architecture and may expose research challenges. The primary goal of the tutorial is to deepen the engagement of the community with the ICN effort though concrete examples of running code.

4.3 Requirements for the Tutorial Room

The tutorial room must provide:

- at least one data projector with XGA resolution or better for showing code examples that can be clearly viewed by all participants
- sound reinforcement (microphone) for presenters
- wired internet access, preferably unfiltered with a static IP
- power outlets for participants computers

The room layout, if possible, should allow easy movement of participants around the room, e.g., should have relatively wide aisles between seating rows, etc.

4.4 Requirements for the Participants

We assume that everyone will bring a laptop or smartphone; some people will want to follow up the coding exercise; some may not. We hope that those who do not code will still be able to post questions for Q&A section of the tutorial.

Those who code, should bring a laptop capable of running Docker platform. The NDN team will prepare a set of docker environments (with online and the hand out instructions) to create unified development environment. Participants will also be able to use local or cloud-based VM or directly install tools on their laptops (only for those running Linux and OS X), but no time will be allocated for troubleshooting participants' installations. In the weeks leading up to the tutorial, we will provide a limited email support to participants willing to prepared the development environment before the tutorial starts.

The developed examples will use a variety of programming languages, including C++ (C++11), C (C99), JavaScript, Python. Participants are expected to be proficient at least in one of the listed languages and general understanding of today's TCP/IP protocol architecture.

4.5 Limitations on Participation

The maximum number of participants may be limited by the capacity of the tutorial room. We do not see other limitations, as long as participants can work comfortably on their own laptop, with power and network access.

5. BIOGRAPHIES OF TUTORIAL SPEAKERS

Alexander Afanasyev is an Adjunct Assistant Professor in University of California, Los Angeles. He received his Ph.D. degree in computer science from UCLA in 2013. His research focus is on the next-generation Internet architecture as part of the Named Data Networking (NDN) project. His research interests include a variety of topics that are vital for the success of NDN, including scalability of name-based routing, auto-configuration, distributed data

synchronization, application and network security. Dr. Afanasyev is also leading the development effort of the overall NDN codebase.

Jeff Burke is Assistant Dean for Technology and Innovation at the UCLA School of Theater, Film and Television (TFT), where he has been a faculty member since 2001. His research explores the intersections of the built environment, computer networks, and storytelling. He has produced, managed, programmed, and designed performances, short films, new genre art installations and new facility construction internationally for over fifteen years, incorporating emerging technologies as part of these projects and creative works. Burke co-founded REMAP, a joint center of TFT and the Henry Samueli School of Engineering and Applied Science, which uses a mixture of research, artistic production, and community engagement to investigate the interrelationships among culture, community, and technology. From 2006-2012, Burke was the area lead for participatory sensing at the NSF Center for Embedded Networked Sensing (CENS). He is Co-PI and application team lead for the Named Data Networking research project.

Davide Pesavento is currently a Ph.D. candidate at the University Pierre & Marie Curie, Sorbonne University, in Paris (France), under the supervision of Prof. Giovanni Pau. He received his B.S. and M.S. in computer science from the University of Padua, Italy. His main research interests are opportunistic mobile connectivity, vehicular networks, and the application of the NDN approach to vehicular networks. In 2012 he designed and developed the first version of VNDN, a complete NDN protocol stack for vehicular ad-hoc networks. Since 2014 he is one of the main developers for the core NDN software, including the NDN Forwarding Daemon (NFD).

Beichuan Zhang is an Associate Professor at the Department of Computer Science, the University of Arizona. His research interest is in Internet routing architectures and protocols. He has been working on Named Data Networking, green networking, and interdomain routing. He received the Applied Networking Research Prize in 2011 by ISOC and IRTF, and best paper awards at IEEE ICDCS in 2005 and IWQoS in 2014. Dr. Zhang received Ph.D. from UCLA and B.S. from Peking University.

Lixia Zhang is a Professor in the Computer Science Department of UCLA. She received her Ph.D in computer science from MIT and was a member of the research staff at Xerox PARC before joining UCLA. She is a fellow of ACM and IEEE, the recipient of IEEE Internet Award, and the holder of UCLA Postel Chair in Computer Science. Since 2010 she has been leading the effort on the design and development of the NDN architecture.

Acknowledgments

This work is supported by the US National Science Foundation through award CNS-1345318, CNS-1345142, and others.