

# Named Data Networking: Lessons Learned and Open Issues

Beichuan Zhang

The University Of Arizona

# Highlights of Achievement

**Protocol specification, packet format, implementation.**

## **Application Development**

- E.g., NDNFit, Building Management Systems (BMS), climate science application, NDN-RTC, smart homes.

## **Mechanisms**

- E.g., routing protocols, scalable forwarding, forwarding strategies, sync, repo, schematized trust, access control, auto configuration and resource discovery.

## **Evaluation Tools**

- Testbed (27 sites and growing), mini-NDN, ndnSIM

## **Research Community**

- NDNComm: 2014 (87 attendees from 31 institutions), 2015 (99 attendees from 49 institutions).
- ACM ICN conference and ICN-related workshops.

# Lessons learned and open issues

**Named data enables a new communication model.**

**Applications drive the architecture development.**

**Naming design is the most fundamental piece.**

**Open-source development**

**Tussles in the real world**

# A new communication model

**Move from point-to-point conversation to distributed content production, retrieval, and consumption.**

- Naming data itself instead of the endpoint.
- A powerful generalization of the IP architecture.
- Also brings up many open issues and invites more research.

# Ex. 1: Forwarding Strategy

**IP forwards packets to the destination: stateless.**

**NDN forwards interests to retrieve data: stateful and closed-loop feedback.**

- Native multicast and in-network caching
- Forwarding plane can detect and adapt to network problems.
  - E.g., choose a different nexthop when the original one doesn't bring back data.
- Relax requirements on routing protocols; enable novel routing protocols.

## **Open Issues**

- Support flexible strategy design and implementation to suit different network environments.

# Ex. 2: Transport

**TCP/IP was designed for point-to-point connections.**

**In NDN, data can be retrieved from anywhere via any path. To applications, the transport problem is generalized to a multi-point synchronization problem.**

- Sync
- Aim to get the desired named data, independent from whether connectivity changes or not; no need to maintain path/connection.

## **Open issues**

- Efficient multi-path, multi-source content retrieval.

# Ex. 3: Security

**Naming data enables securing the data directly.**

**Hierarchical names reflect relationship among data and provide context for trust decisions.**

- Schematized trust
- Name-based access control

**Open issues:**

- Name confidentiality
- Cache poisoning

# Ex. 4: Performance

## **Scalable name lookup and forwarding is feasible.**

- Variable-length names, longest prefix lookup, unbounded ruleset.
- 10Gbps with 256-byte packets and 1 billion forwarding rules, using 8 cores, roughly 4.5MPPS.

## **Open issues:**

- Meet the demands of the very big and the very small:
  - high performance for big-data science applications.
  - optimize for resource-constrained devices.

# Application-driven Arch Development

**Test the architecture with a diverse set of applications.**

**Make applications easy to use**

- Auto-configuration, resource discovery

**Focus on emerging network environments**

- IoT, BMS, big-data science.

**Identify design patterns**

**An ongoing iterative process between application development and architecture design/implementation.**

# Naming Design

## **Both apps and networks use NDN names.**

- Tighter integration of apps and networks, as opposed to two insulated layers.

## **Develop naming conventions to facilitate future apps.**

- E.g., temporal evolution of data is reflected by version/timestamp/seq number component in the name.

## **Open issues:**

- Applications construct multiple views based on the stream of immutable data objects supplied from the network.

# Open-Source Development

**Have developed a large codebase for the forwarder, libraries, applications, and various tools.**

**A collaborative effort of the NDN team and also researchers outside of the NDN team.**

- **Issue tracking, code review, continuous integration.**

**Important to involving the community in the design, implementation and use of NDN software.**

**Management and coordination are challenging.**

# Tussles in the real world

## **Intellectual property rights**

- Our goal: keep the core open, leave the rest flexible.
- Companies: aggressively patent everything.

## **Standardization vs. research**

- Standardization facilitates interoperability, but premature standardization is harmful to the architecture.

## **Short-term performance optimization vs. long-term architectural functionality**

- E.g., Fixed-length header vs. variable-length, exact name match vs. longest prefix match

# Summary

**The future of networking lies in recognizing the right communication abstraction.**

**Confident in the fundamental idea, the application-driven architecture development, and the open-source model to address the research issues and challenges.**