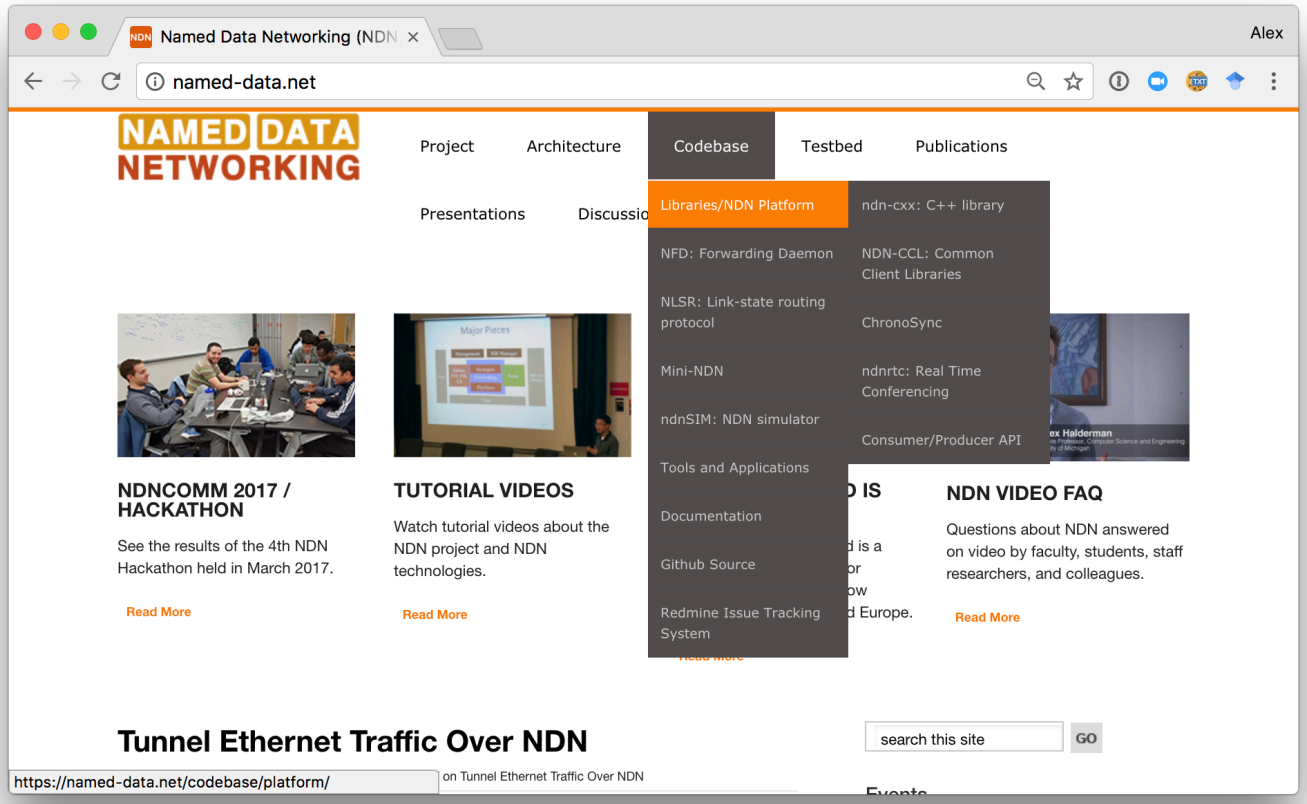
MILCOM 2017
MILITARY COMMUNICATIONS AND INNOVATION - PRIORITIES FOR THE MODERN WARFIGHT

NDN Codebase and Tools

Alex Afanasyev
Florida International University

BALTIMORE, MD • OCTOBER 23–25, 2017

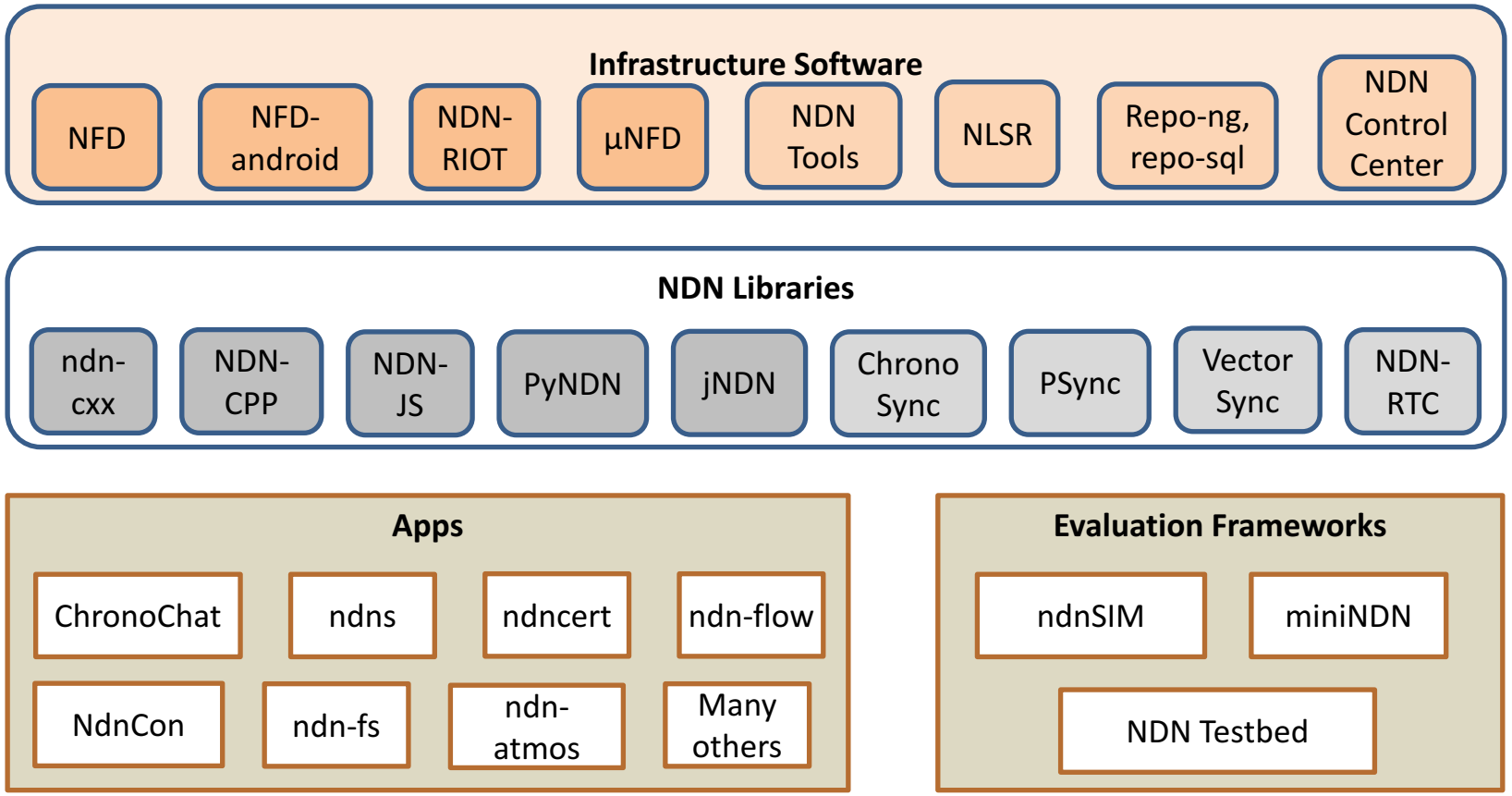
Starting Point: https://named-data.net/ ➠ Codebase



Where to Find Source Code for NDN Codebase

- Most linked from https://named-data.net ➠ Codebase
- Github organizations
  - https://github.com/named-data
    - NFD, core libraries, and other general use software
  - https://github.com/named-data-mobile
    - Android and related software
  - https://github.com/named-data-iot
    - IoT related software
  - https://github.com/named-data-ndnsim
    - ndnSIM core, example and real simulation scenarios

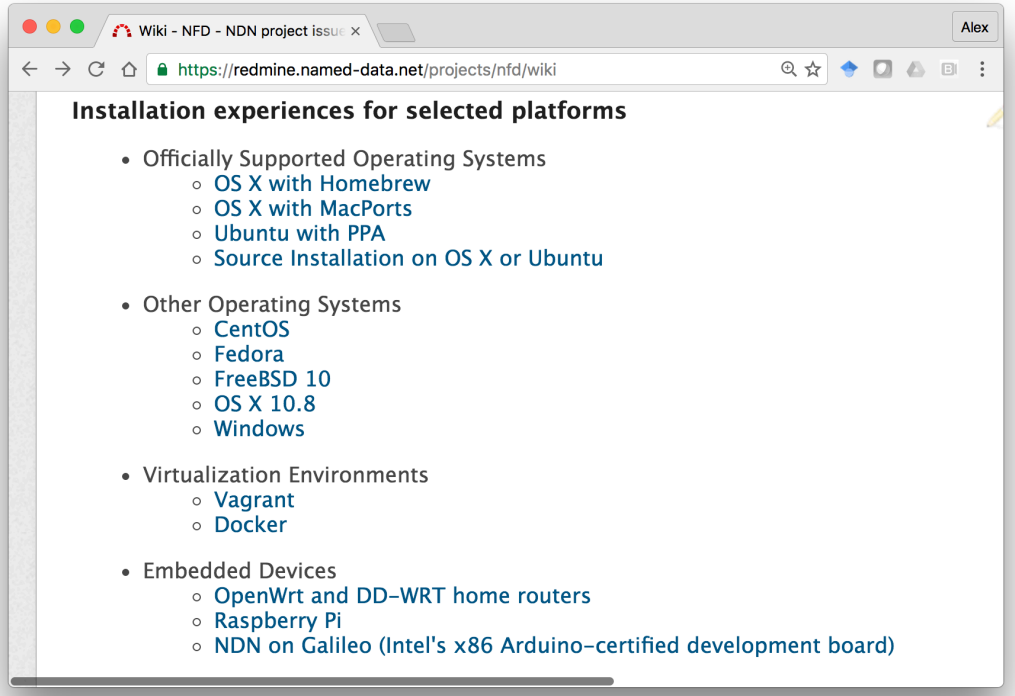
NDN Codebase Overview



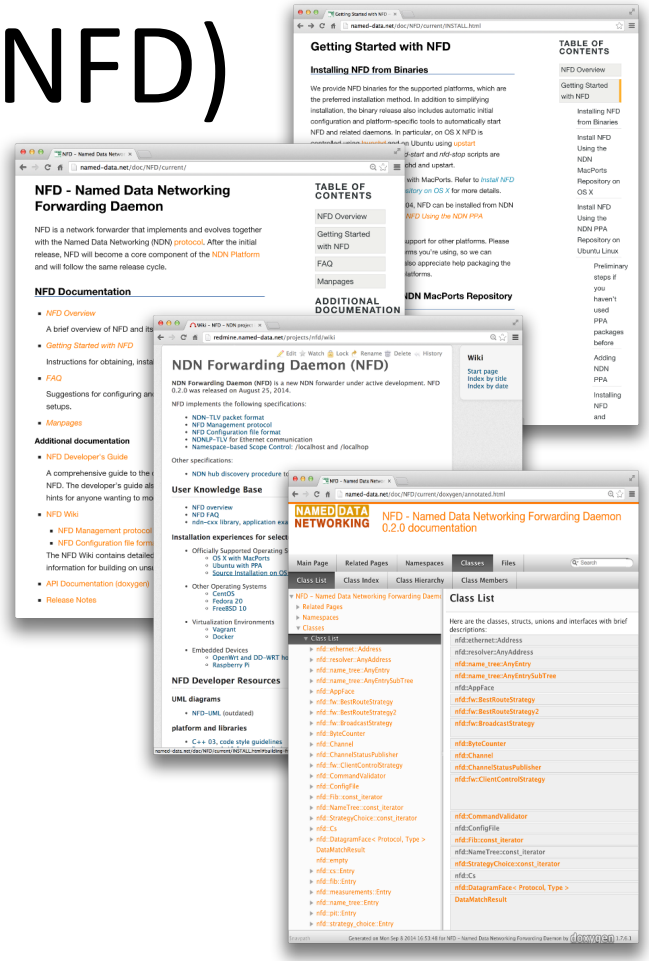
Supported Platforms

- Desktop Systems
  - Ubuntu, OSX, FreeBSD and other Linux distributions
- Home routers
  - OpenWRT, DD-WRT
- Mobile:
  - Android, iOS (library only)
- IoT:
  - Arduino, ESP8266, RIOT-OS
  - Raspberry Pi (runs NFD, available binary packages)
- Web browser
  - NDN-JS library + microforwarder

https://redmine.named-data.net/projects/nfd/wiki

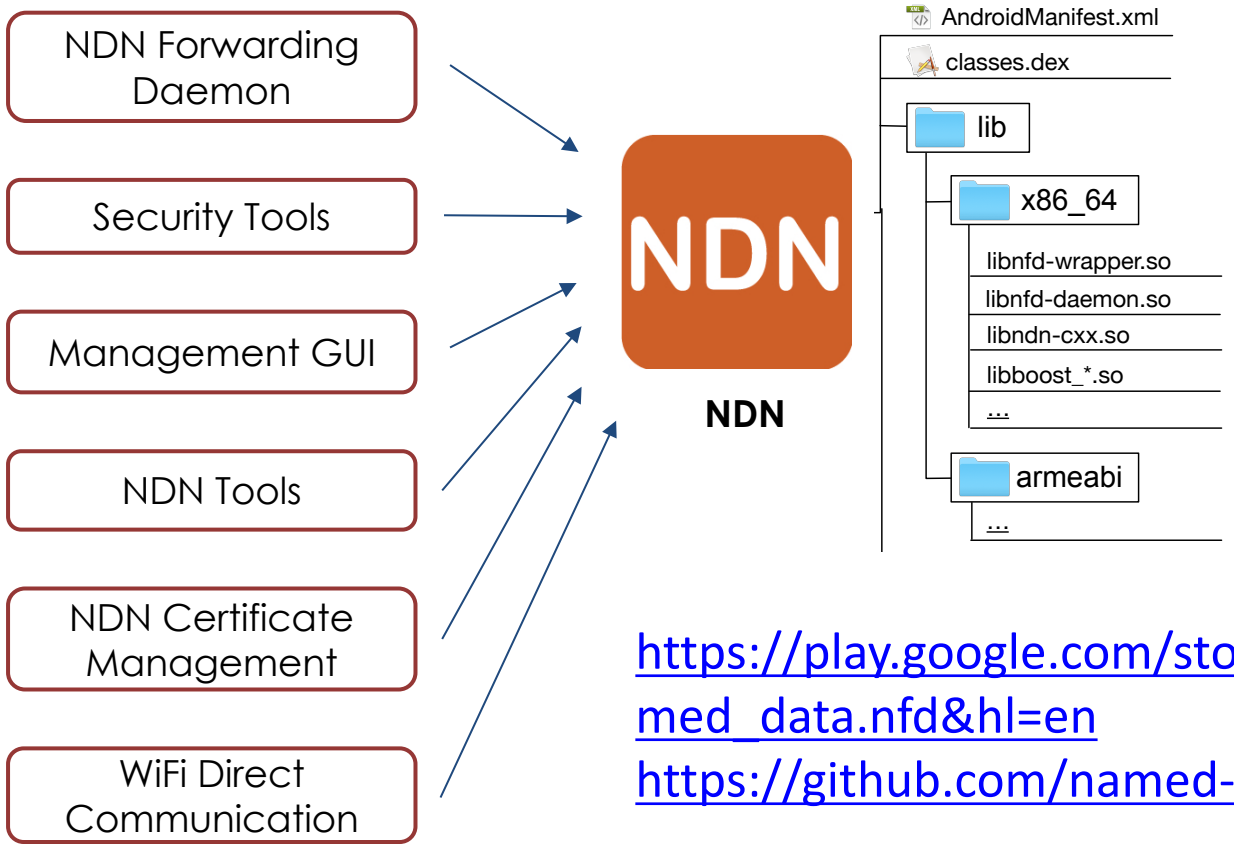


NDN Forwarding Daemon (NFD)



- The reference implementation of NDN forwarder
- https://named-data.net/doc/NFD/current/
  - Overview
  - Getting started
  - NFD Developer's Guide
  - Manpages
  - Wiki
  - API documentation (doxygen)
- Feedback, suggestions, and contributions are welcome.

NDN-Android: NDN Stack for Android

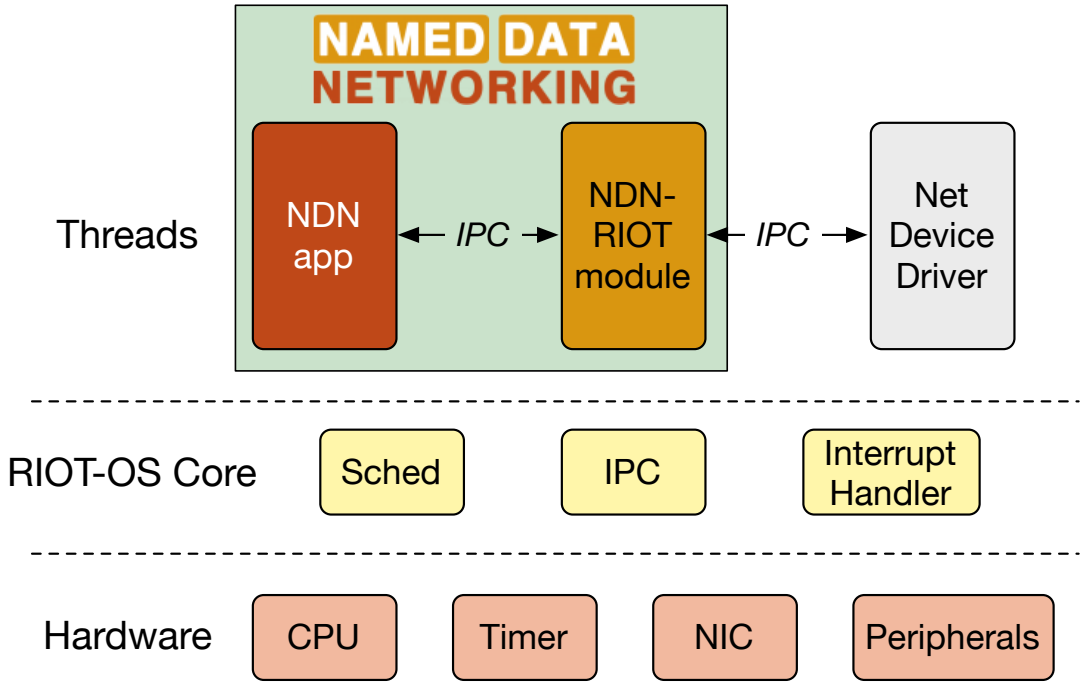


- Embeds actual NFD, compiled using NDK
- Works with all (non-rooted) Android devices

https://play.google.com/store/apps/details?id=net.named_data.nfd&hl=en
https://github.com/named-data-mobile

NDN-RIOT: NDN for RIOT-OS

- Optimized for IoT apps
- Support
  - Data-centric security
  - Stateful NDN packet forwarding
  - Replaceable forwarding strategies
  - 802.15.4 and Ethernet
- Simple application APIs
- Several simple examples to get started



https://github.com/named-data-iot

Getting Started with NDN-RIOT Examples

- Downloading
  - `mkdir riot`
  - `cd riot`
  - `git clone https://github.com/named-data-iot/RIOT`
  - `git clone https://github.com/named-data-iot/ndn-riot`
  - `git clone https://github.com/named-data-iot/ndn-riot-examples`
- Compiling an example
  - cd ndn-riot-examples/<APP>
  - For host architecture (for debugging)
    - make
  - For a specific RIOT board
    - make BOARD=samr21-xpro
    - make flash BOARD=samr21-xpro # to flash firmware
    - make term BOARD=samr21-xpro # to access board via serial interface

ndn-benchmark
ndn-consumer
ndn-ping
ndn-producer
ndn-rtt
ndn-template

NDN Tools

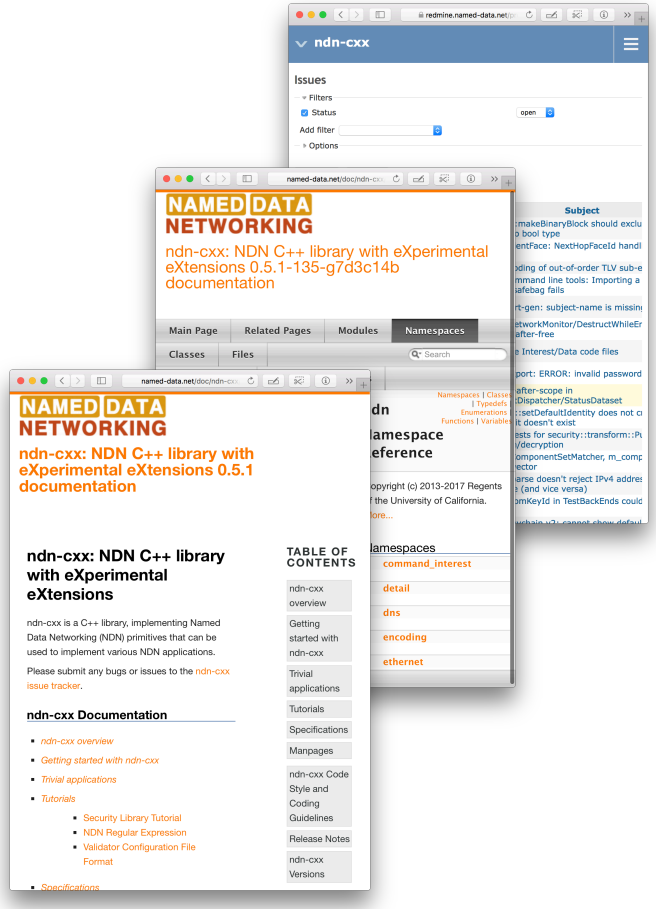
- ndnping, ndnpingserver
  - Rechability testing tools
- ndncatchunks, ndnputchunks
  - Segmented file transfer between a consumer and producer
- ndnpeek, ndnpoke
  - Transmit a single packet between a consumer and a producer
- ndndump, dissect, wireshark-dissect
  - Debug NDN packet flow

- repo-ng, repo-sql: NDN repositories providing managed persistent s

LIBRARIES

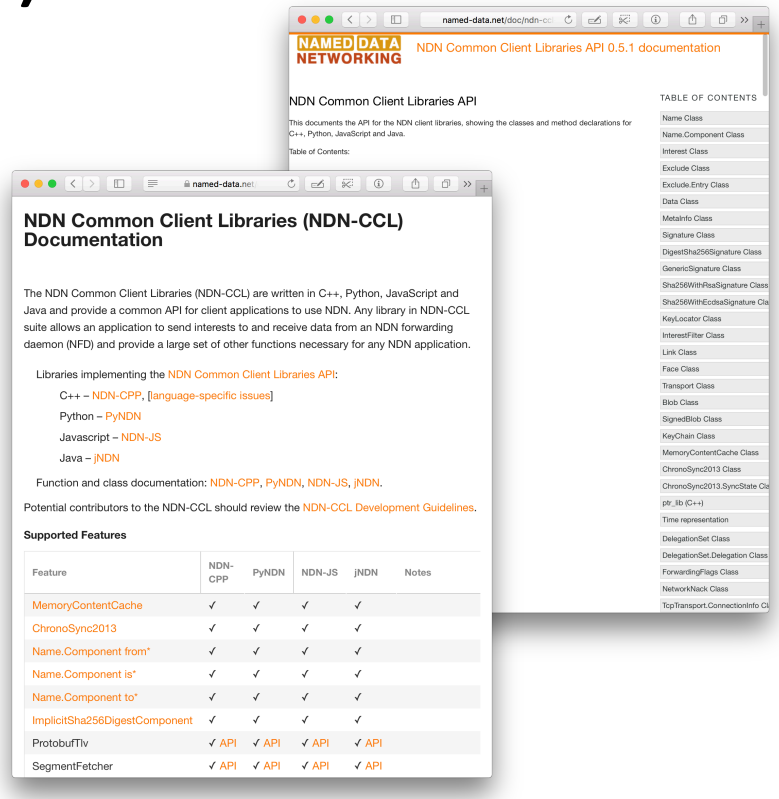
ndn-cxx: NDN C++ library with eXperimental eXtensions

- C++11
- The reference library and security library implementation
- Used in: NFD, NLSR, ndn-tools, ChronoChat, etc.
- https://named-data.net/doc/ndn-cxx/current/
  - Overview
  - Getting started
  - Trivial applications
  - Tutorials
  - Specifications
  - Manpages
  - API documentation (doxygen)
- Feedback, suggestions, and contributions are welcome.



NDN Common Client Libraries (NDN-CPP, NDN-JS, jNDN, PyNDN)

- C++, Java, Python, JavaScript, C#, Squirrel
- Used in: NDN-RTC, NdnCon, NFD-Android, etc.
- https://named-data.net/codebase/platform/ndn-ccl/
  - NDN Common Client Libraries API
  - NDN-CPP API
  - PyNDN API
  - NDN-JS API
  - jNDN API



EVALUATION TOOLS AT DIFFERENT SCALES

NDN Testbed

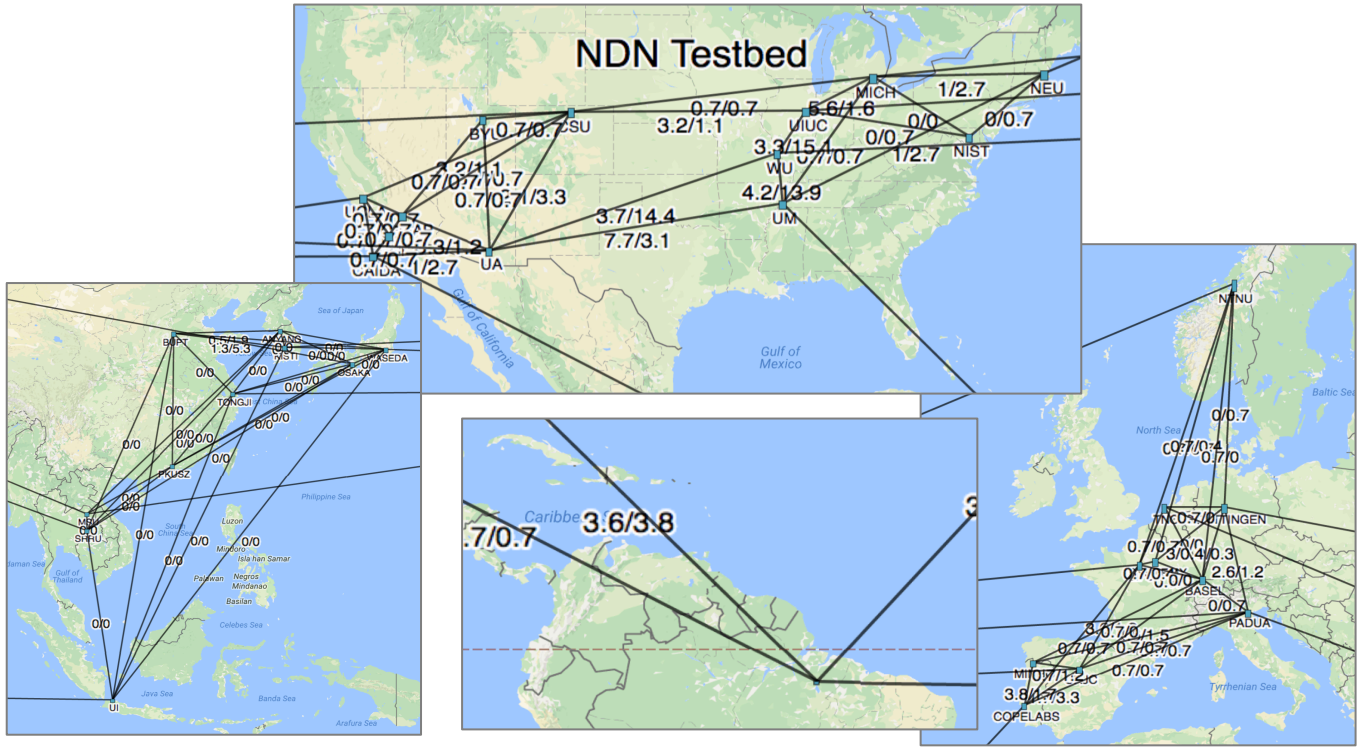
- Network of 37 sites across 4 continents, 14 countries

Open to join and use

https://named-data.net/ndn-testbed/policies-connecting-nodes-ndn-testbed/

- Examples applications and experiments: videoconferencing, network management, virtual machine migration, strategies, nTorrent, etc.

- Small scale evaluations



Open Network Lab (ONL)

- Remotely accessible network testbed
  - Operated and maintained by Applied Research Lab in Department of Computer Science and Engineering at Washington University in St. Louis
  - Real Hardware for running repeatable network experiments with trusted results. (NOT simulations)
- Use for NDN
  - NDN installed on each host/VM
  - NFD performance study
  - NDN Testbed Emulation to test new releases
- How to join?
  - https://onl.wustl.edu/
    - And “Get an account”

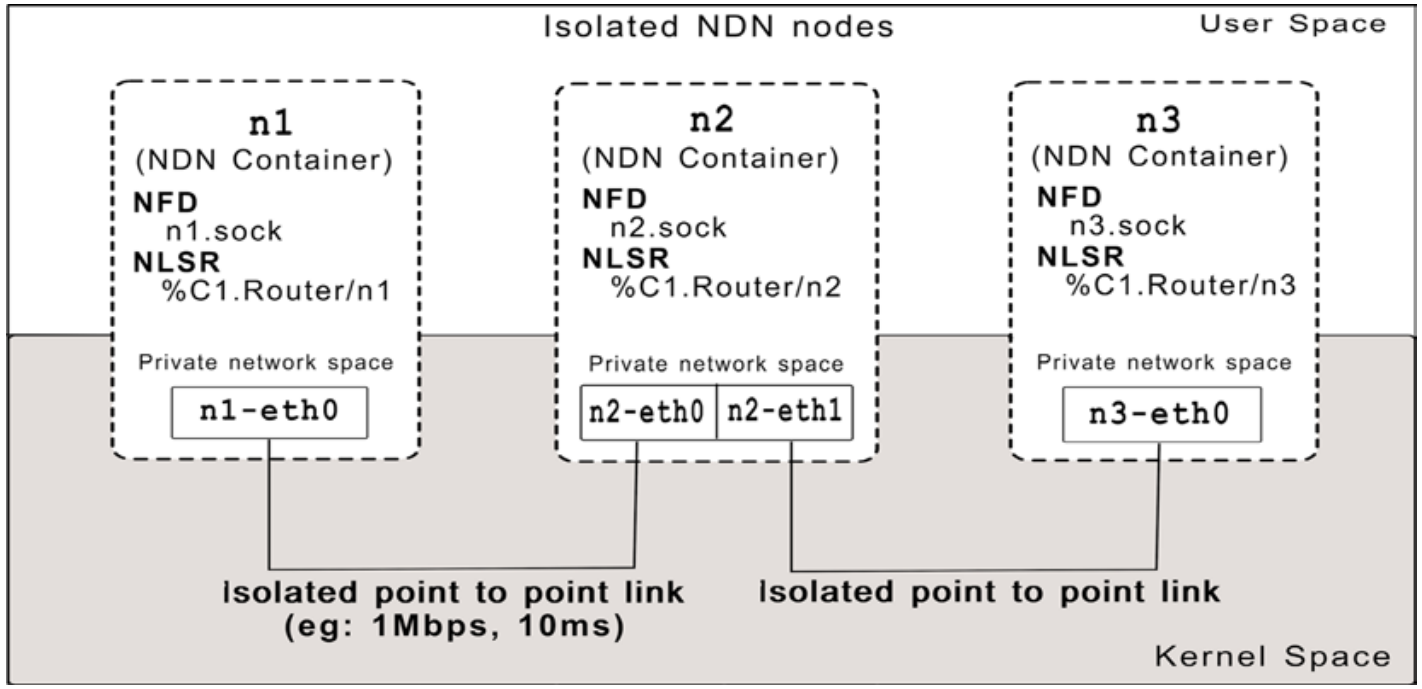
http://minindn.memphis.edu/

MiniNDN: NDN Emulation Framework (Based on MiniNet)

Runs actual instances of NFD, NLSR

Medium-scale evaluations

- Easy to configure network emulation
- Runs any real application
- Number of emulated nodes ∝ CPU power
- Cluster edition can be used to scale emulations



ndnSIM: NDN Simulation Framework (Based on NS-3)

Fully integrated with NDN prototype implementations: NFD & ndn-cxx

Large scale evaluations

- Provide interoperability between simulation and prototyping
- Enable a two-way of experimentation and evaluation
- Enable high-fidelity NDN simulations

- 1500+ nodes with WiFi channels in the evaluation of NDN for vehicular networking