UCLA ENGINEERING
Computer Science

ndnSIM: a modular NDN simulator

Overview and near future plan

http://ndnsim.net
ndnSIM@lists.cs.ucla.edu

ALEX AFANASYEV, ILYA MOISEENKO, LIXIA ZHANG
alexander.afanasyev@ucla.edu

University of California, Los Angeles, Computer Science Department

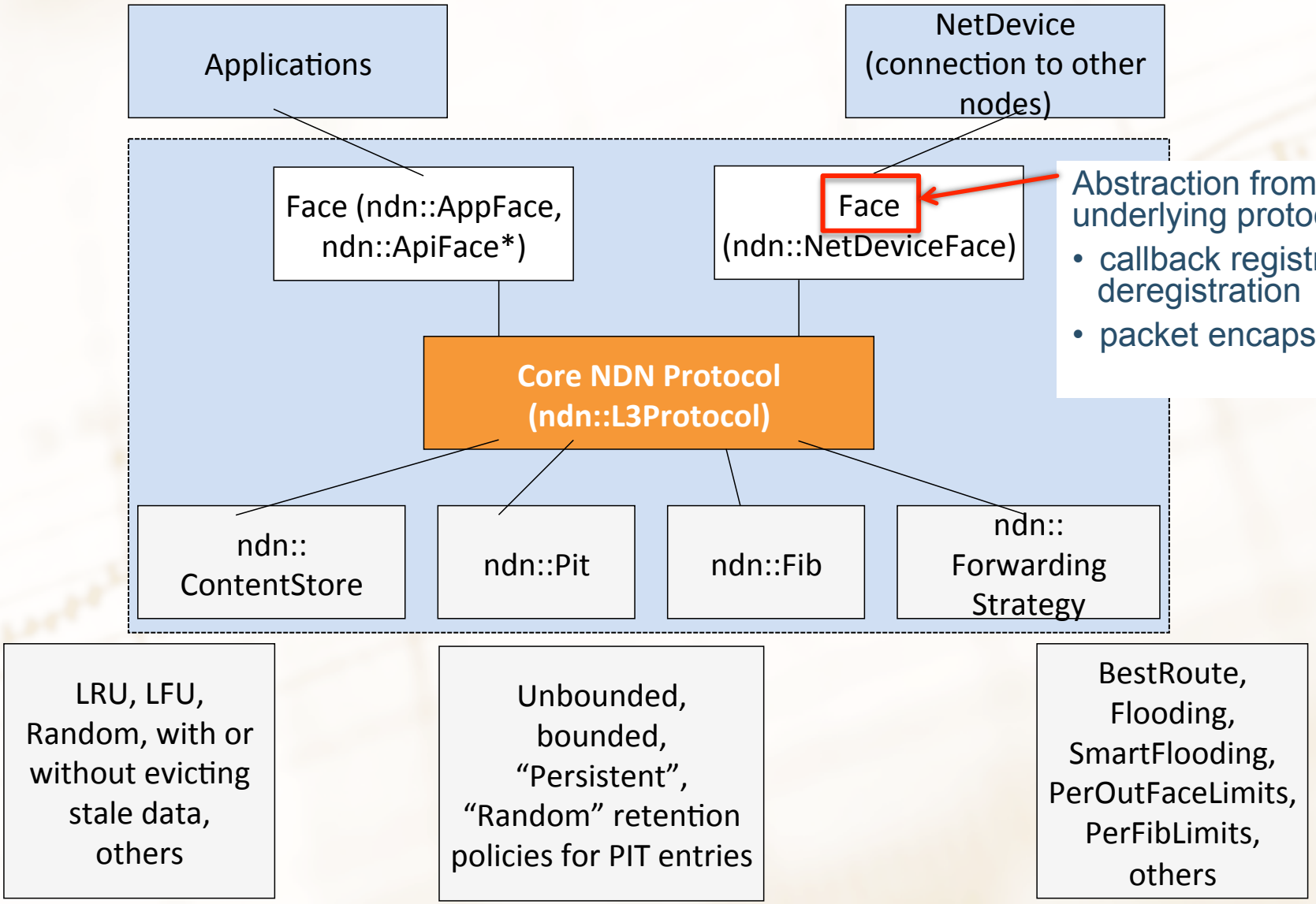
Fundamental Goal

- Establishing a *common* platform shared by the community for all NDN simulation experimentation
  - Making it easy to replicate experiments and compare results

Overview

- Based on NS-3 network simulator
- Implemented all basic NDN operations
- A modular architecture
  - C++ classes for every NDN component
    - Face, PIT, FIB, Content store, and Forwarding strategy
- Easy to combine different implementations of the components
  - Different management schemes for PIT
  - Different replacement policies for content store
  - Different forwarding strategies
- Easily extendable

Modular ndnSIM structure overview

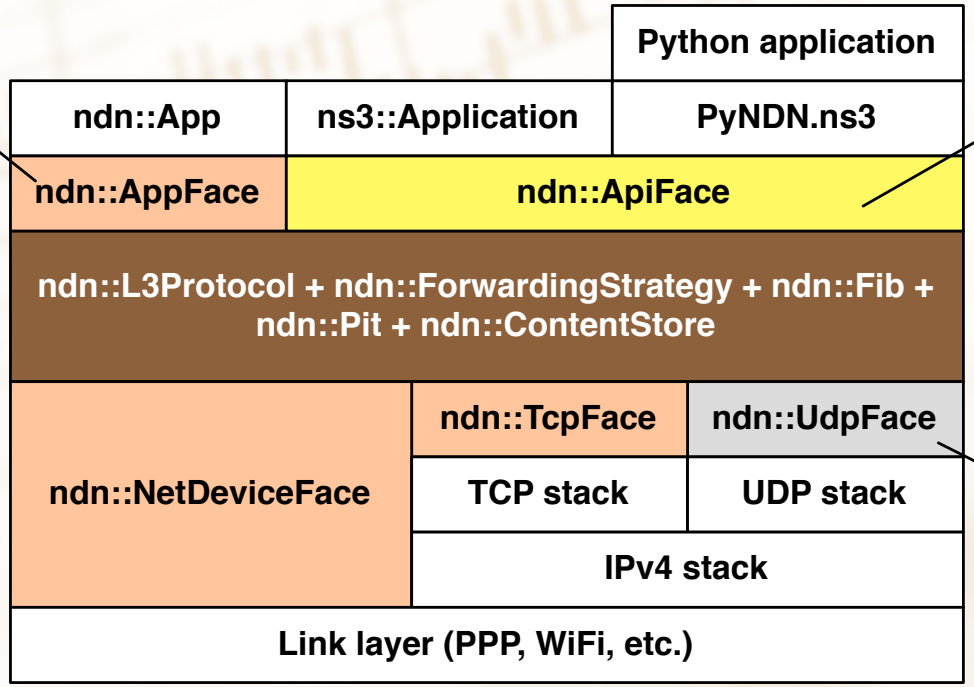


New additions in version 0.5 (soon to be released)

- Advanced application API
  - now it is possible to write full featured applications
    - compatible C++ API with NDN.cxx https://github.com/named-data/ndn.cxx
    - compatible Python API with PyNDN https://github.com/cawka/PyNDN
  - ultimately: compile (just run in case of Python) real applications in ndnSIM

Simple NDN API for network-layer (strategy, cache, etc.) evaluations

ndn::ConsumerCbr,
ndn::ConsumerWindow,
ndn::ConsumerZipfMandelbrot
ndn::Producer



Newest addition in version 0.5

Rich (full) NDN API for application level evaluations

ChronoSync / ChonoChat

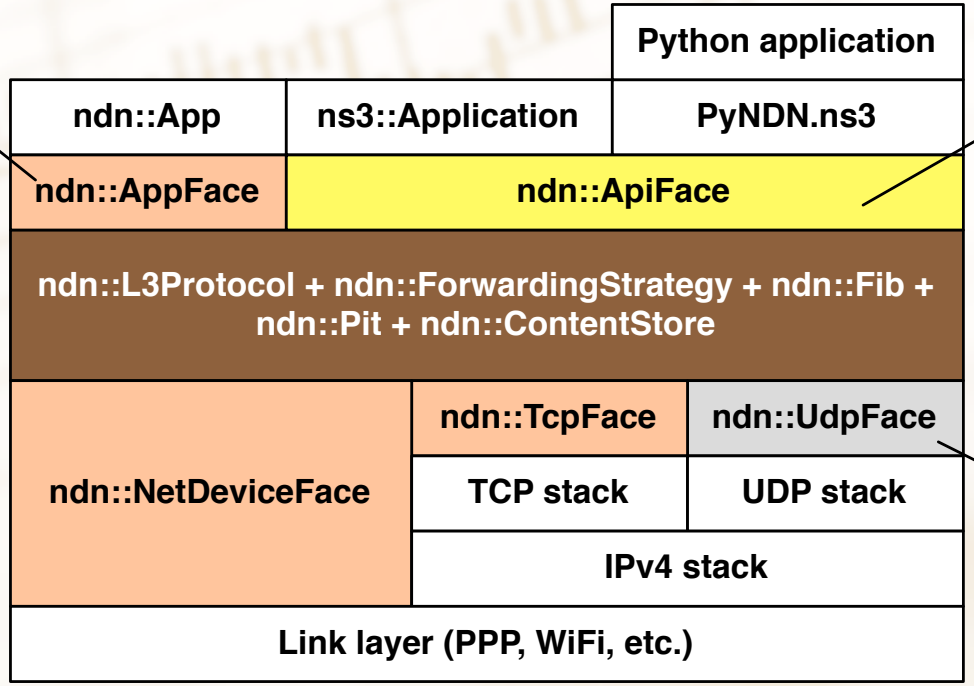
Work in progress

New additions in version 0.5 (II)

- Redesigned/simplified/unified API for Interest and Data packets, Forwarding strategy
  - not fully backward compatible, but easy to adapt
- Exclude filter support
  - other interest selectors may be coming, if requested

Simple NDN API for network-layer (strategy, cache, etc.) evaluations

ndn::ConsumerCbr,
ndn::ConsumerWindow,
ndn::ConsumerZipfMandelbrot
ndn::Producer



Newest addition in version 0.5

Rich (full) NDN API for application level evaluations

ChronoSync / ChonoChat

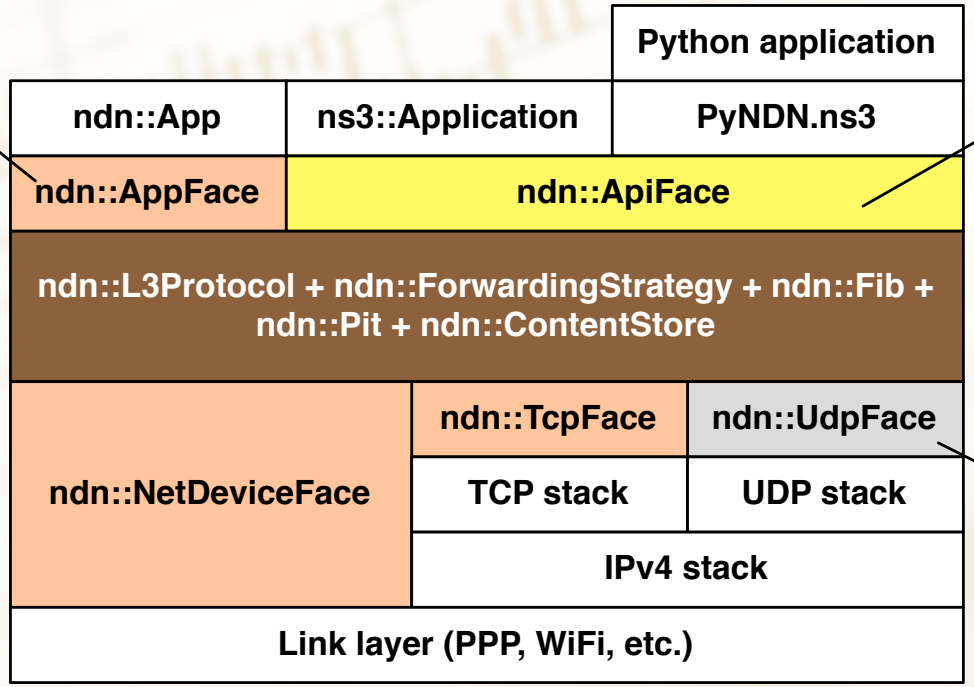
Work in progress

New additions in version 0.5 (III)

- Support for overlay-based simulations
  - using ndn::TcpFace and ndn::UdpFace
- Support for multiple wire format, selectable at runtime
  - simplified ndnSIM format http://ndnsim.net/ndnsim-packet-formats.html
  - full-featured (but not too optimal) CCNb format
  - other experimental formats

Simple NDN API for network-layer (strategy, cache, etc.) evaluations

ndn::ConsumerCbr,
ndn::ConsumerWindow,
ndn::ConsumerZipfMandelbrot
ndn::Producer



Newest addition in version 0.5

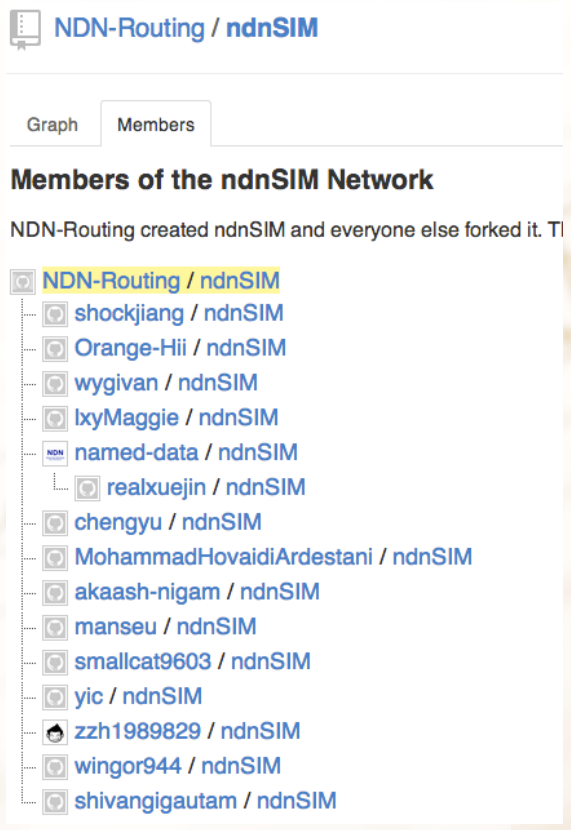
Rich (full) NDN API for application level evaluations

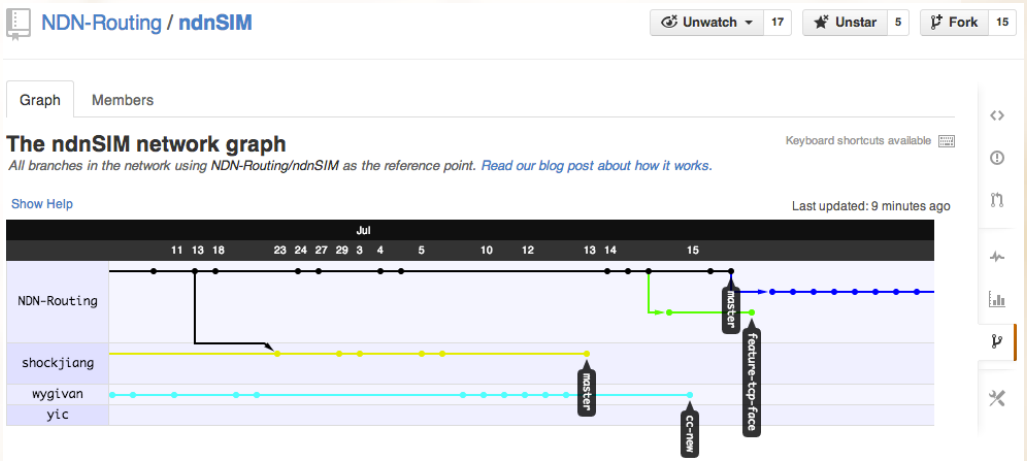
ChronoSync / ChonoChat

Work in progress

Current status

- 15 public forks on github
- Active development
  - new features
  - extended API
  - usage examples and documentation
- A lot of activity on the mailing list





ndnSIM usage scope trends (based on published papers and mailing list data)

- http://ndnsim.net/ndnsim-research-papers.html#research-papers-that-use-ndnsim
  - at least 5 published papers (by early adopters, excluding us) using ndnSIM
- Caching-related evaluation
  - various caching replacement policies, collaborative caching
- Congestion control related
  - TCP-like transfers (end-to-end, host-by-host)
  - queuing
- Mobile and vehicular environment evaluations
- DDoS-related evaluations
  - interest flooding (us)
  - content poisoning
- Forwarding strategy experimentation (us)
  - behavior in the presence of link failures, prefix black-holing
- Application-level evaluations (us)
  - exploration of ChronoSync protocol design

NDN-Routing / ndnSIM

branch: master

ndnSIM / AUTHORS

file | 15 lines (11 sloc) | 0.571 kb

Edit

```
1  The PRIMARY AUTHORS are (and/or have been):
2
3    Alex Afanasyev <alexander.afanasyev@ucla.edu>
4    Ilya Moiseenko <iliamo@ucla.edu>
5
6  The following is an inevitably incomplete list of MUCH-APPRECIATED CONTRIBUTORS,
7  people who have reported bugs, submitted patches,and  implemented new features of ndnSIM:
8
9    Jiangzhe Wang (Lucas) <lucas@cs.ucla.edu>
10   Cheng Yi <yic@email.arizona.edu>
11   Saeid Montazeri <saeid.montazeri@gmail.com>
12   Xiaoke Jiang <shock.jiang@gmail.com>
13   Saran Tarnoi <sarantarnoi@gmail.com>
14   Hovaidi Ardestani Mohammad <mohammad.hovaidi.ardestani@aalto.fi>
```

Getting started

- http://ndnsim.net/getting-started.html
- Works in OSX, Linux, FreeBSD
  - requires boost libraries >= 1.46
  - visualizer module need python and various python bindings
- Download
  - mkdir ndnSIM
  - cd ndnSIM
  - git clone git://github.com/cawka/ns-3-dev-ndnSIM.git ns-3
  - git clone git://github.com/cawka/pybindgen.git pybindgen
  - git clone git://github.com/NDN-Routing/ndnSIM.git ns-3/src/ndnSIM
- Build
  - ./waf configure --enable-examples
- Run examples
  - ./waf --run=ndn-grid
  - ./waf --run=ndn-grid --vis
  - other examples: http://ndnsim.net/examples.html

ndnSIM Tutorial

- Organized by AsiaFI Summer School
- To be held online Friday August 9th

  13:30 Hong Kong

  7:30 Berlin

  22:30 Los Angeles (Aug 8th)

- Channel: TBD
  - Contact Prof. Jun Bi of Tsinghua

Your feedback most appreciated!

- Try out ndnSIM and let us know your thought/ comments/bug reports/new feature requests
- Join nsnSIM mailing list
  - http://www.lists.cs.ucla.edu/mailman/listinfo/ndnsim
- Contribute
  - issues on Github
    - https://github.com/NDN-Routing/ndnSIM/issues?state=open
  - fork and create pull requests on Github
  - issues in NDN redmine
    - http://redmine.named-data.net/projects/ndnsim

http://ndnsim.net