Colorado State University

NAMED DATA NETWORKING IN SCIENTIFIC APPLICATIONS

Christos Papadopoulos

Colorado State University

NIST NDN Workshop, June1, 2016

Work supported by NSF #1345236 and #13410999

Today's Internet Names Hosts

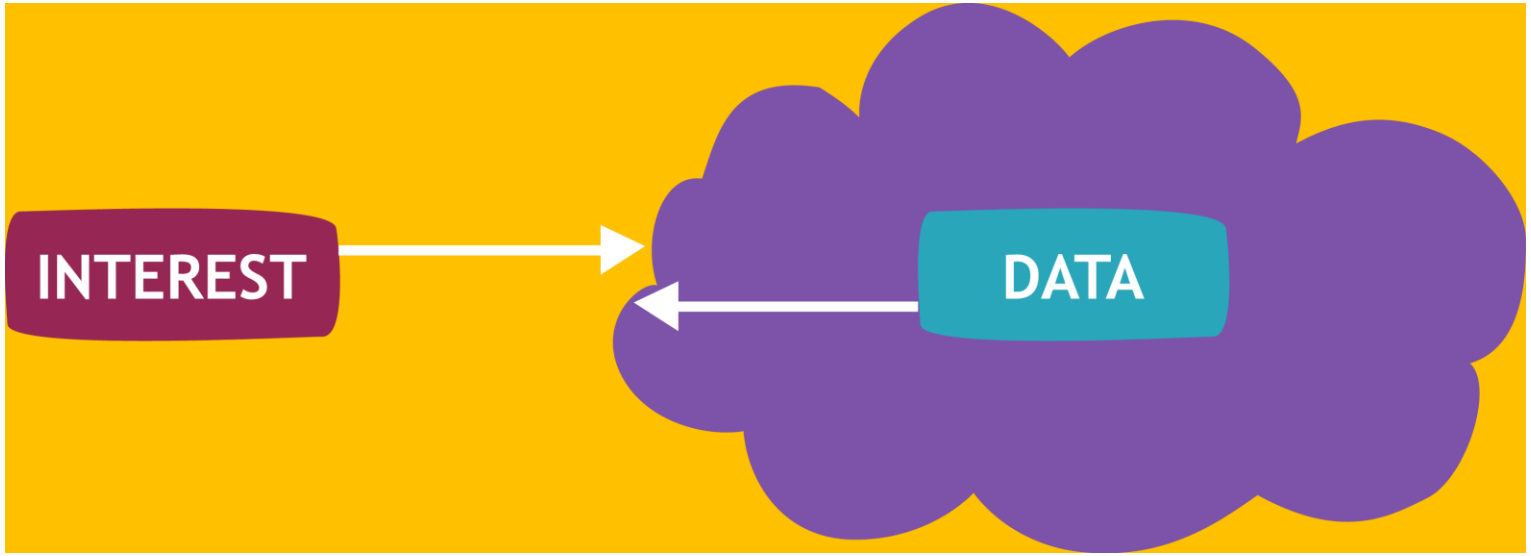
- To find content in the network
- ..you have to learn where the content is
- ..and then ask the network to take you there
- ..so you can tell the server what you want

- But no-one cares about the servers anymore..
- ..we care about the Data!

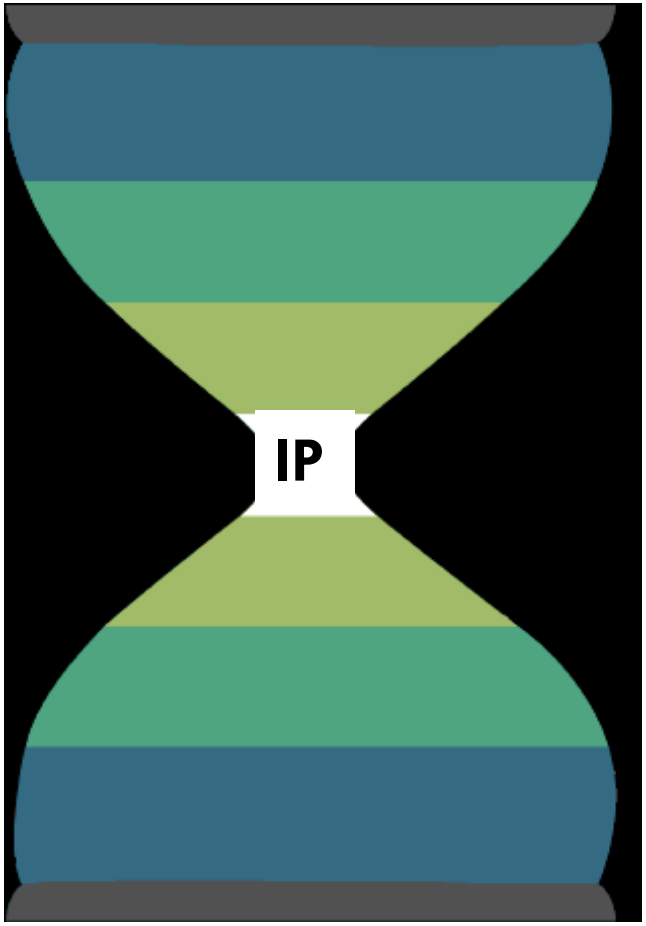
- **Service model mismatch**

Named Data Network (NDN)

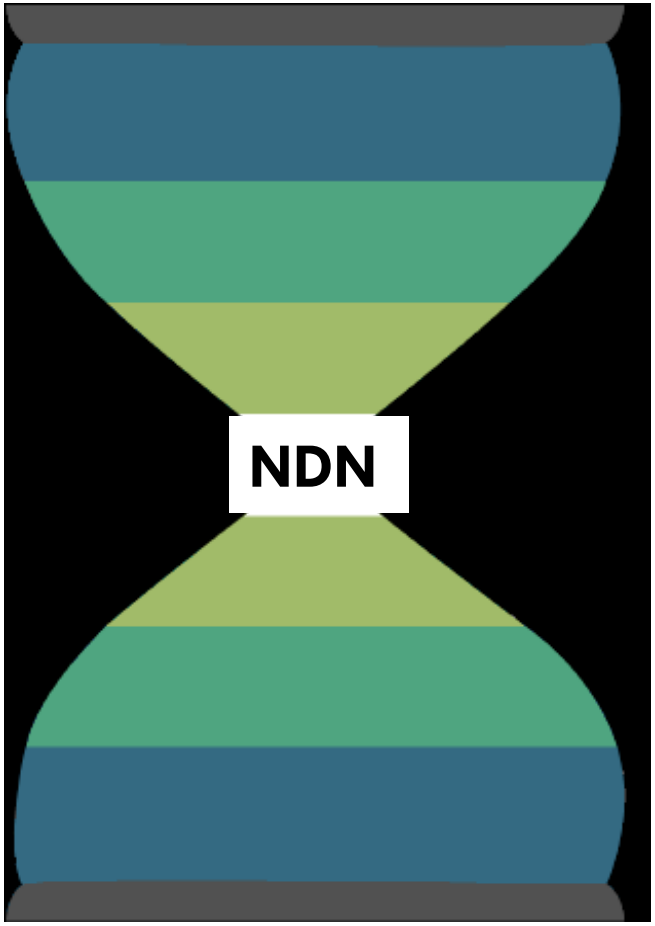
- The main idea: **Name the data, not the hosts!**
- ..so you just tell the network what you want..
- ..and let the network find it for you



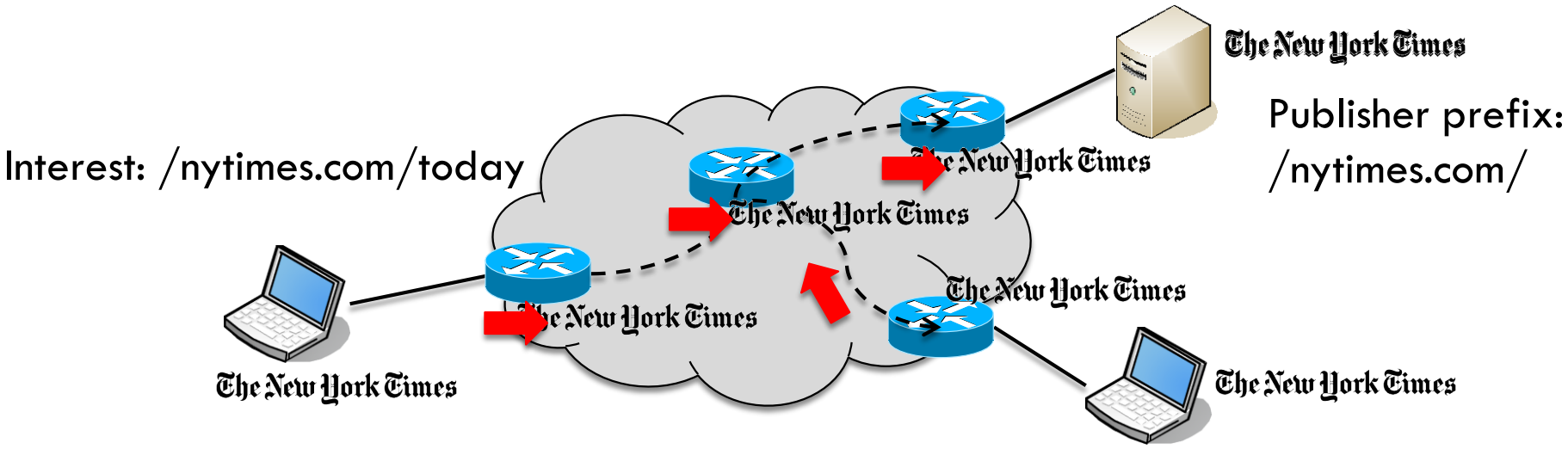
Host-centric addressing



Data-centric addressing

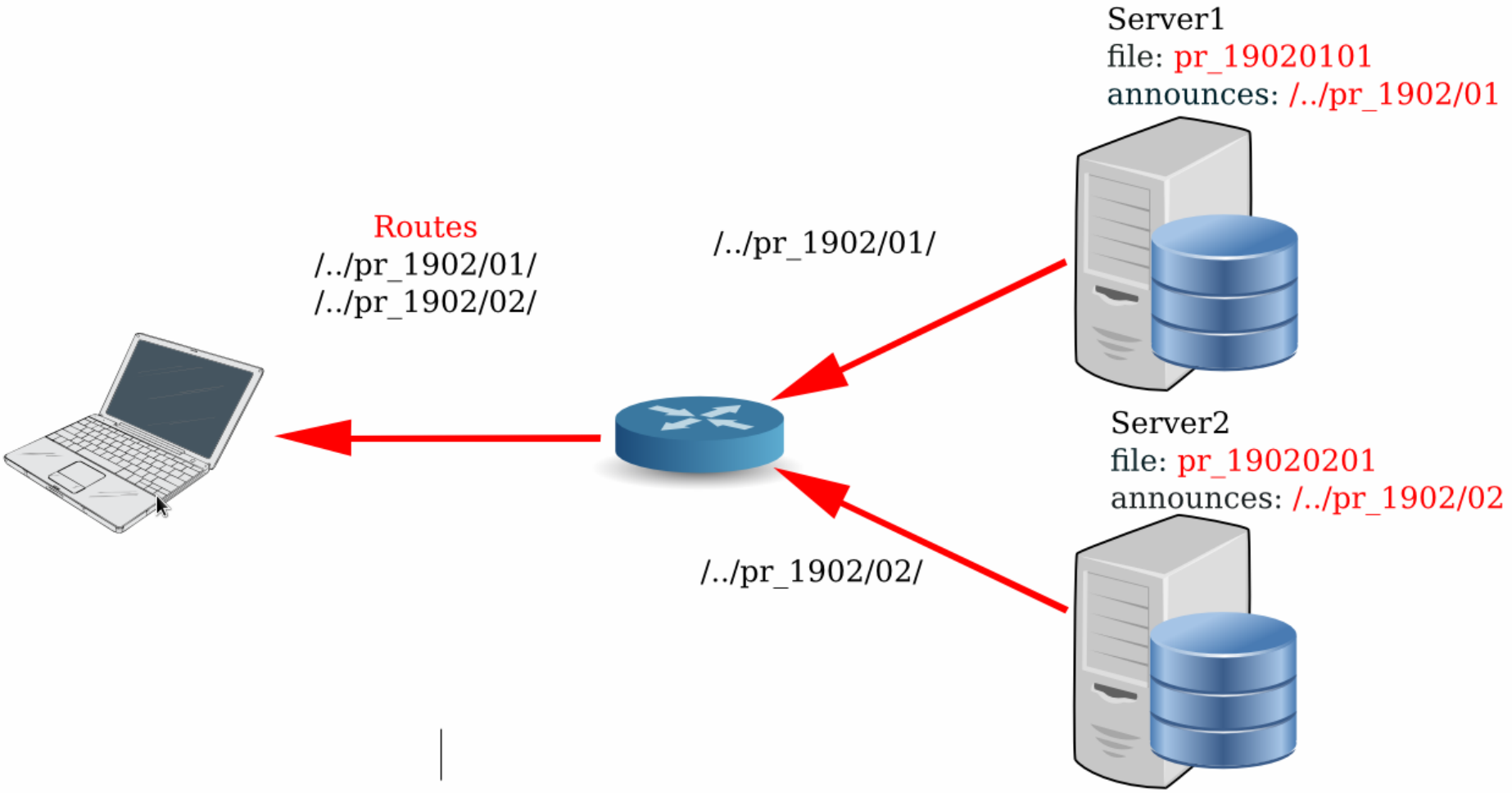


NDN Operation



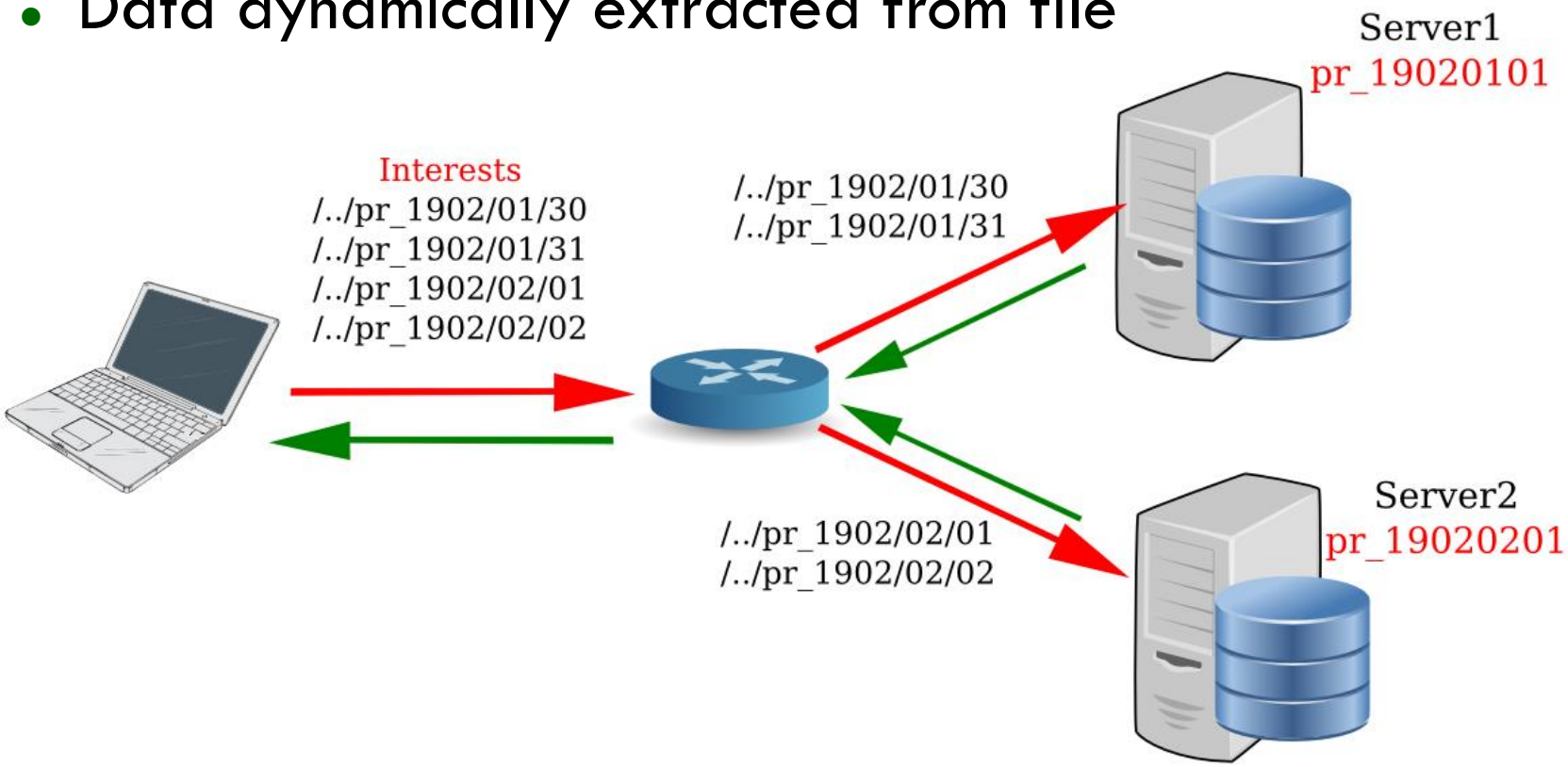
- Publishers push **hierarchical** name prefixes into the network
- Users send **Interests** that follow path to published prefix
- "Breadcrumbs" direct **data** back to the user
- Data is **cached** into the network

Content Publishing



Data Request

- Interests for Jan 30-31 go to server1
- Interests for Feb 01-02 go to server2
- Data dynamically extracted from file



This Sounds Awfully Complex..

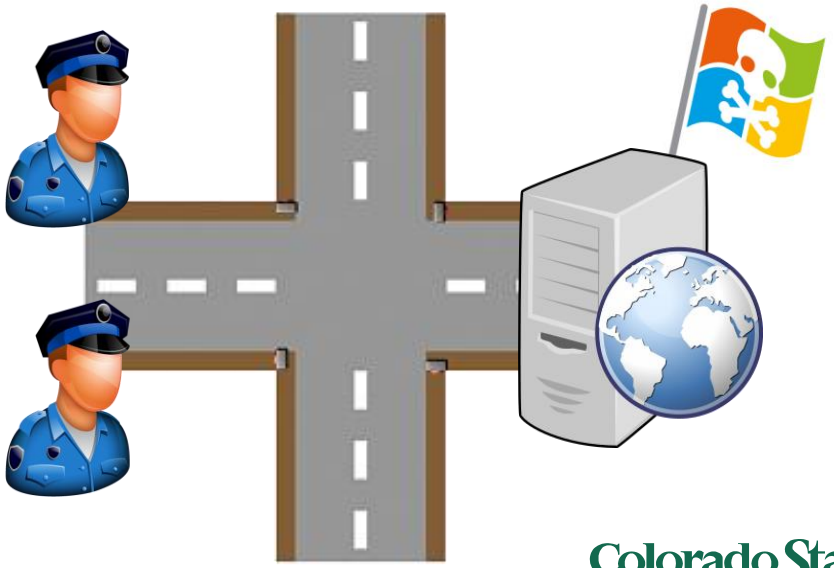
But it's actually quite simple:

- First, name your datasets with a hierarchical, community-agreed name structure:
  - /store/mc/fall13/BprimeBprime_M_3000/GEN-SIM/POSTLS162_v1-v2/10000 /<UUID.root>
- Then, advertise a *prefix* to the network:
  - I can answer any questions starting with:
  - /store/mc/fall13/BprimeBprime_M_3000/GEN-SIM/POSTLS162_v1-v2/*
- Finally, let users issue interests with the appropriate name or name prefix

Named Data is Easy to Secure

- In the Internet you secure your path..
- ..but the server may still be hacked!

- In NDN you **sign** the data with a **digital signature..**
- ..so the users know when they get bad data!

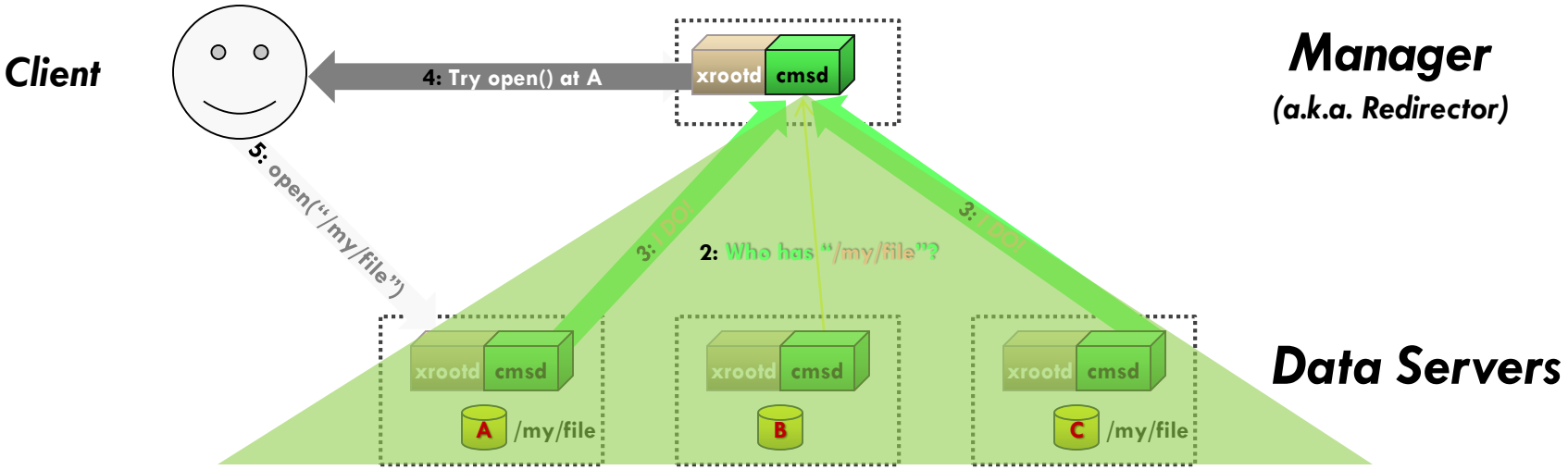


Security, Access Control, Integrity

- Signatures also verify integrity of the data – no need for separate checksums
- Data is signed as soon as it is produced, signatures are for life – much less opportunity for data tampering
- Data is immutable – if you change the data you change the name
- Data name can convey access rights – today, often data inherits the access controls of the resource that hosts it

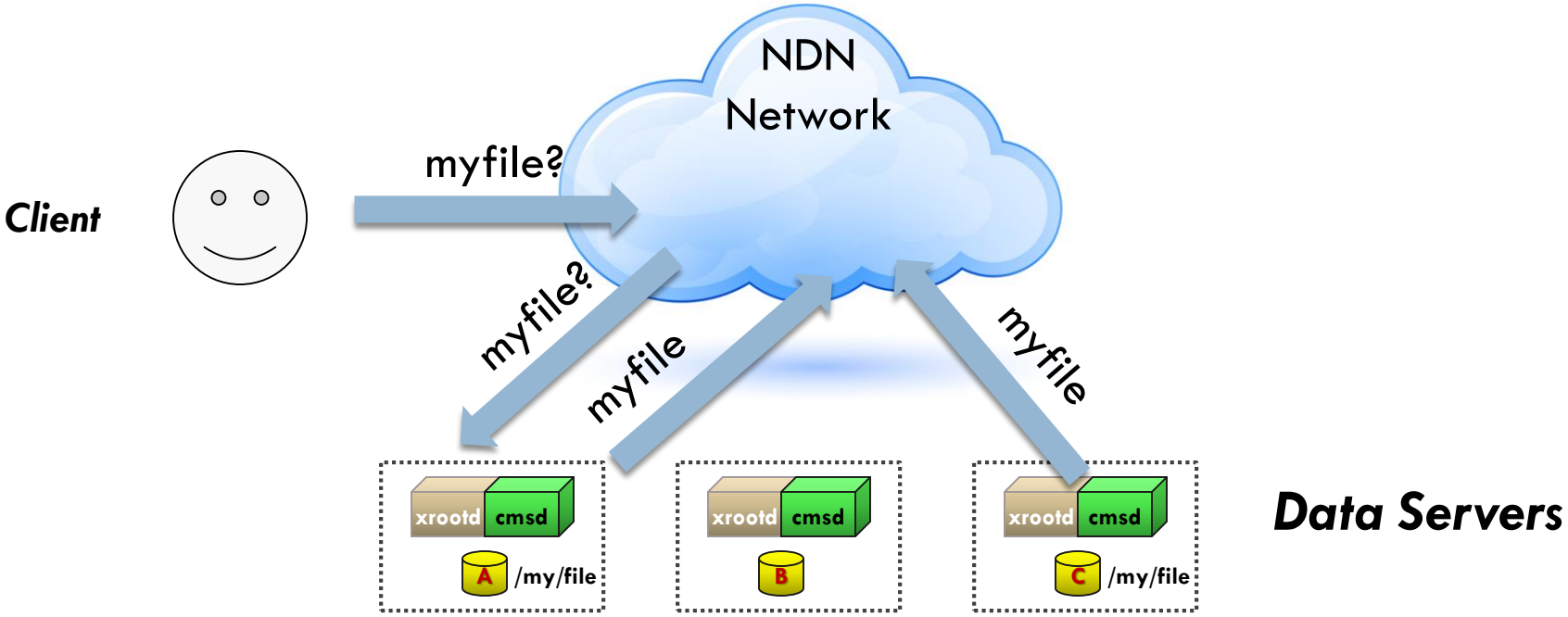
Simplifying a Complex System: xrootd Cluster

Here is how xrootd works today:



xrootd under NDN

No manager, fewer steps, more robust



Supporting Science Applications

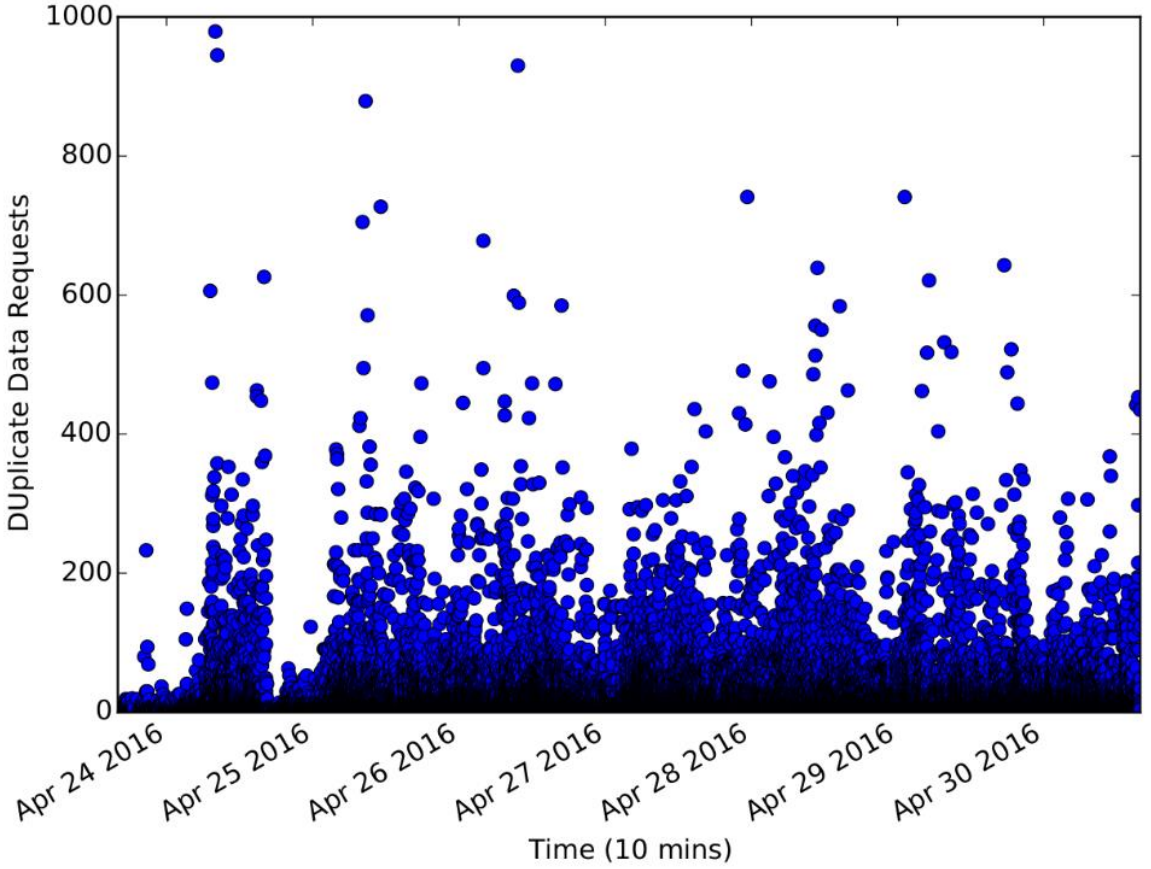
- Scientific apps generate tremendous amounts of data and face challenging management issues
  - Climate science CMIP5 dataset: 3.5 PB, 10x expected for CMIP6
  - High Energy Physics (HEP): 1 PB/s raw data, ATLAS project filters to 4 PB/yr
  - Data distributed to various local repositories
  - Variety of data naming schemes
    - E.g. different units and user defined parameters
- Existing, mature, software for dataset discovery, publishing, and retrieval
  - E.g. ESGF, xrootd, etc.
  - Lots of effort to overcome fragility of IP's host-centric paradigm

Xrootd Access Patterns

Seven day log of xrootd data access

- 115K unique records
- 10 min granularity
- Avg file size: 2GB
- Hits at dataset level

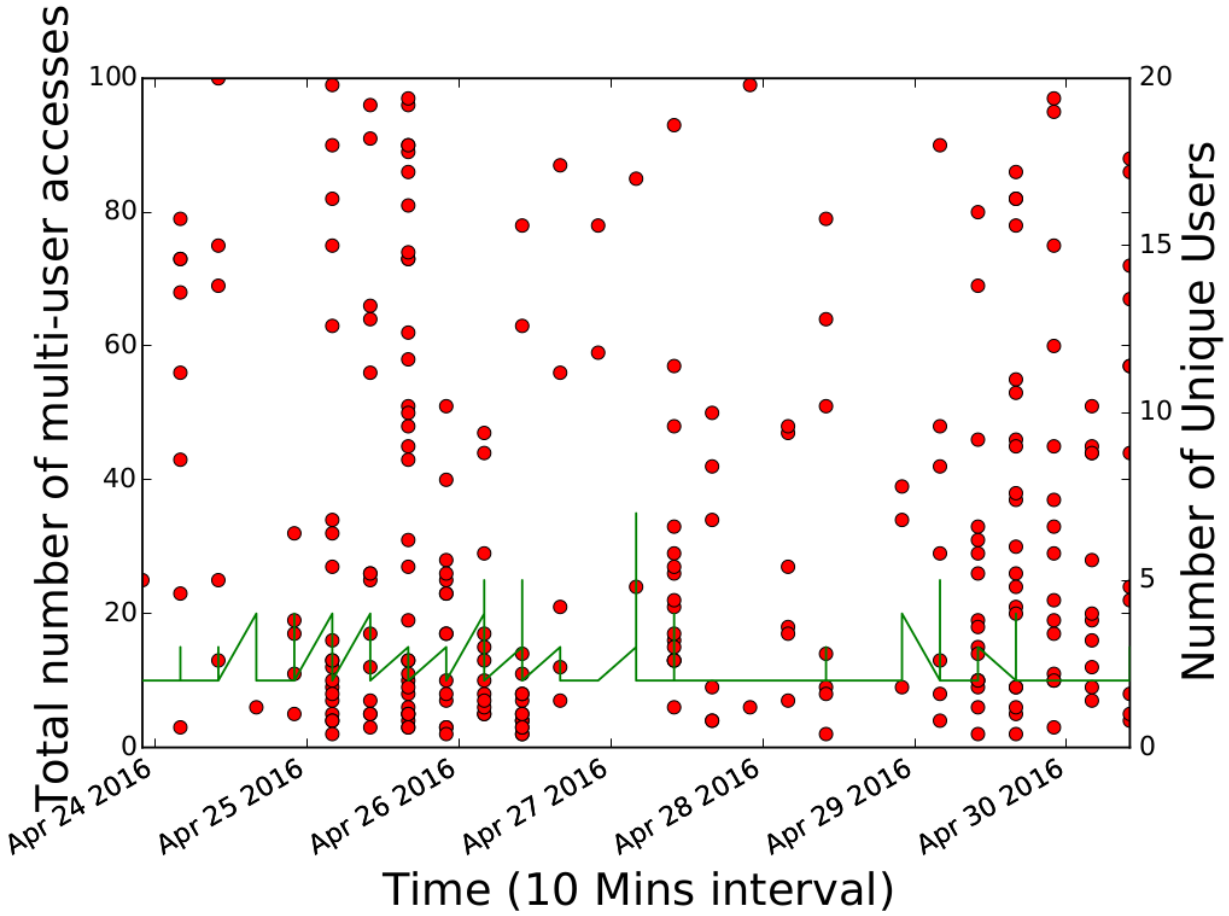
Up to 1000 duplicate hits!



User Access Patterns

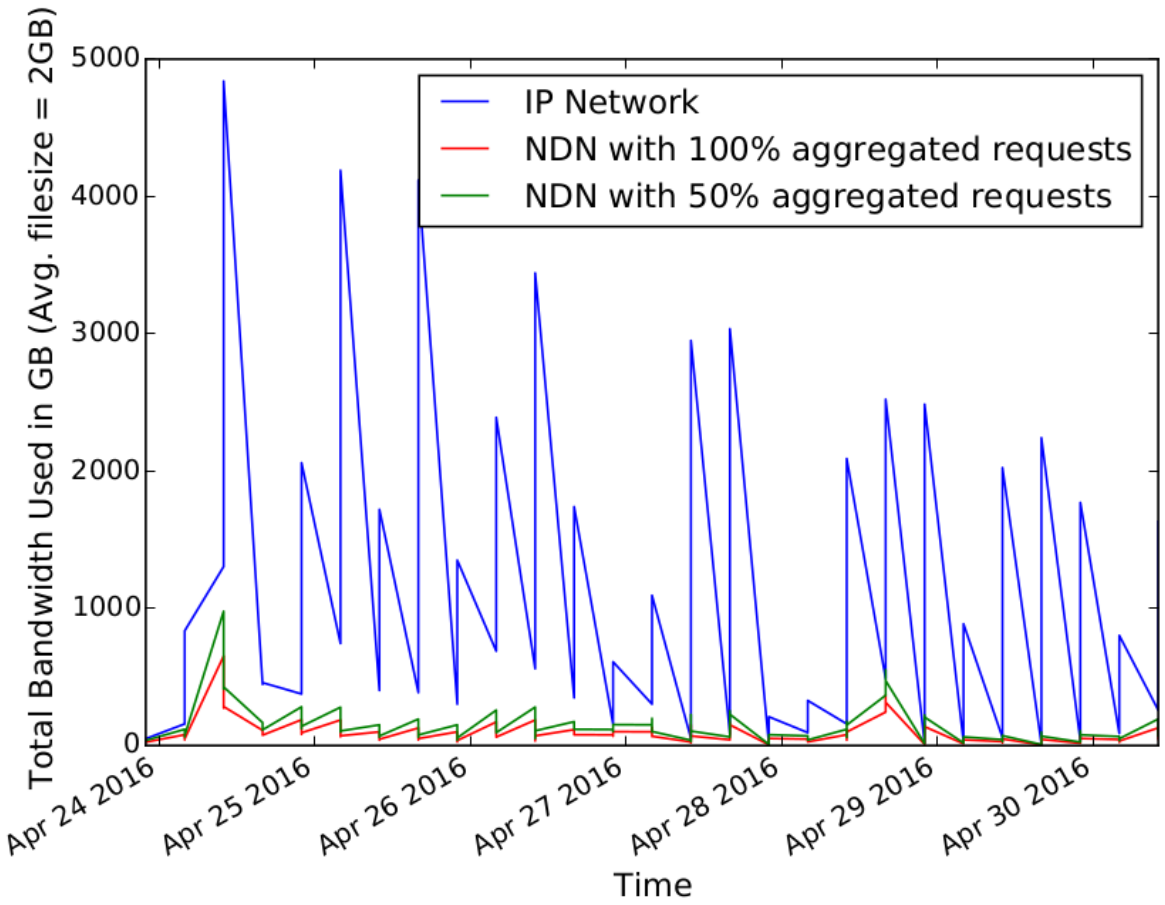
Request aggregation: 6hours

Up to eight simultaneous users request the same dataset



Bandwidth Reduction with NDN

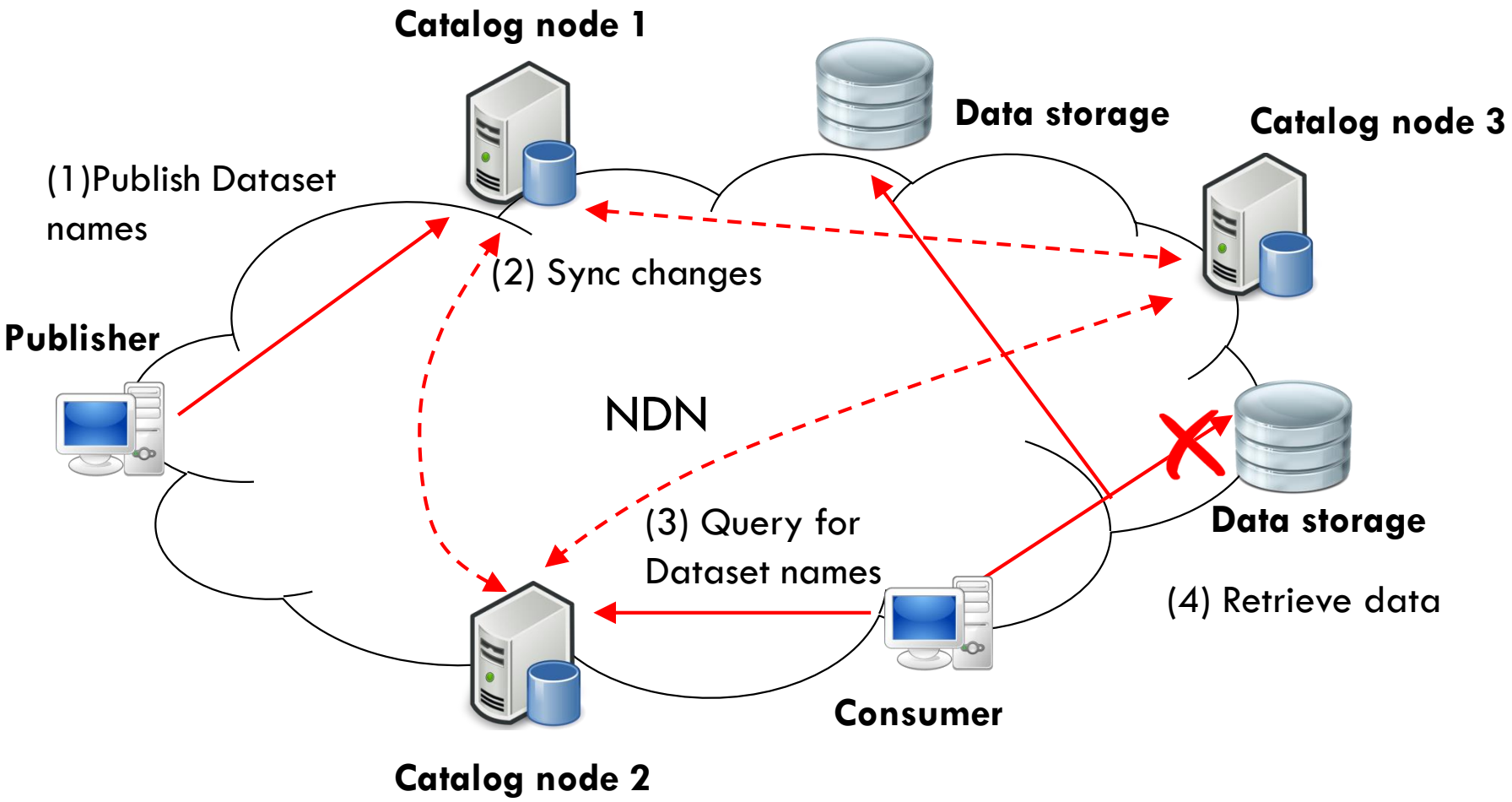
- Bandwidth peaks to 5000GB/10minutes (64Gbps)
- With 100% aggregation bandwidth drops to 8.2Gbps
- With 50% aggregation bandwidth drops to 13.2Gbps



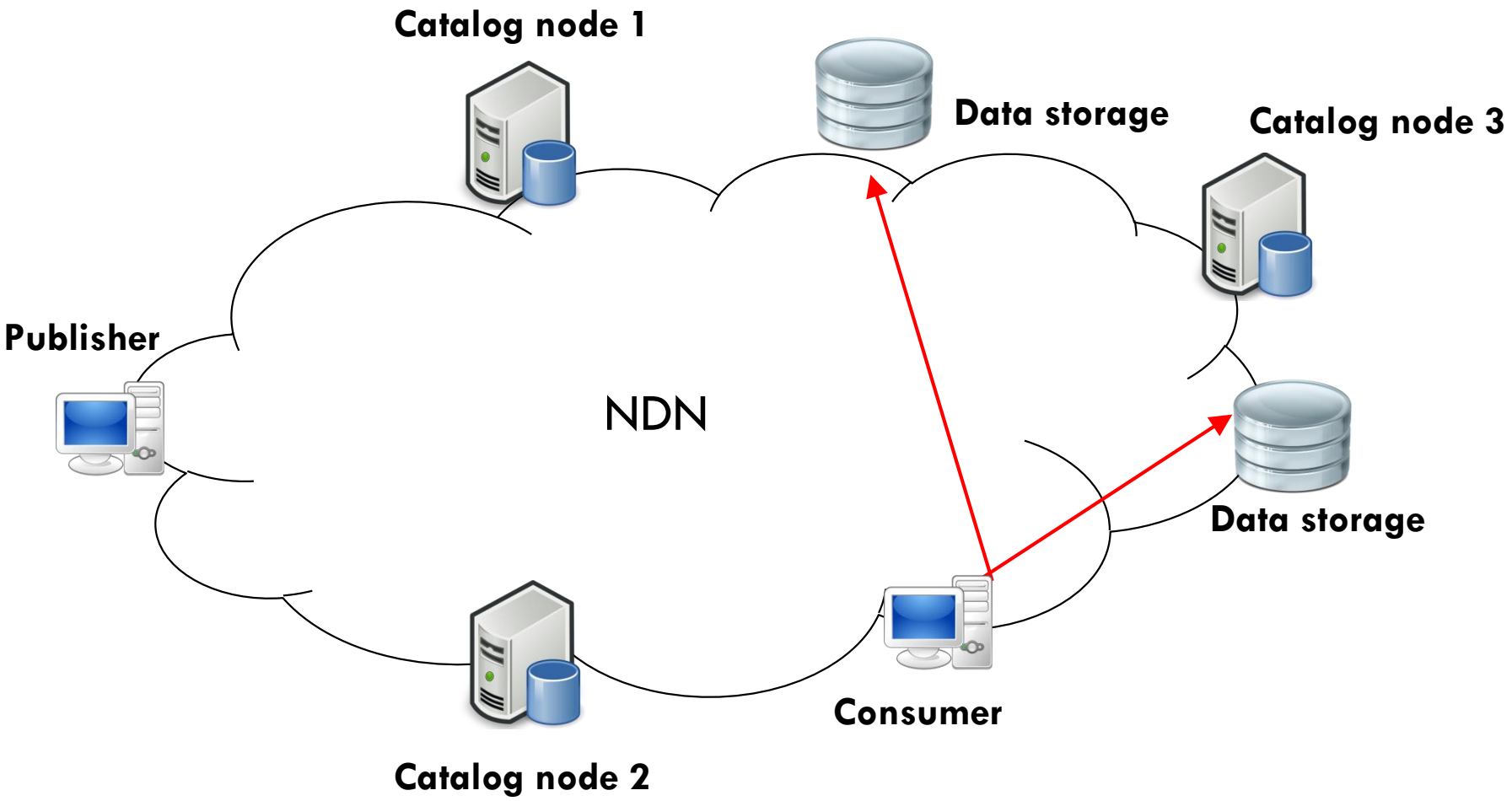
First Step – Build a Catalog

- Create a **shared resource** – a distributed, synchronized **catalog of names** over NDN
  - Provide common operations such as publishing, discovery, access control
  - Catalog only deals with name management, not dataset retrieval
  - Platform for further research and experimentation
- Research questions:
  - Namespace construction, distributed publishing, key management, UI design, failover, etc.
  - Functional services such as subsetting
  - Mapping of name-based routing to tunneling services (VPN, OSCARS, MPLS)

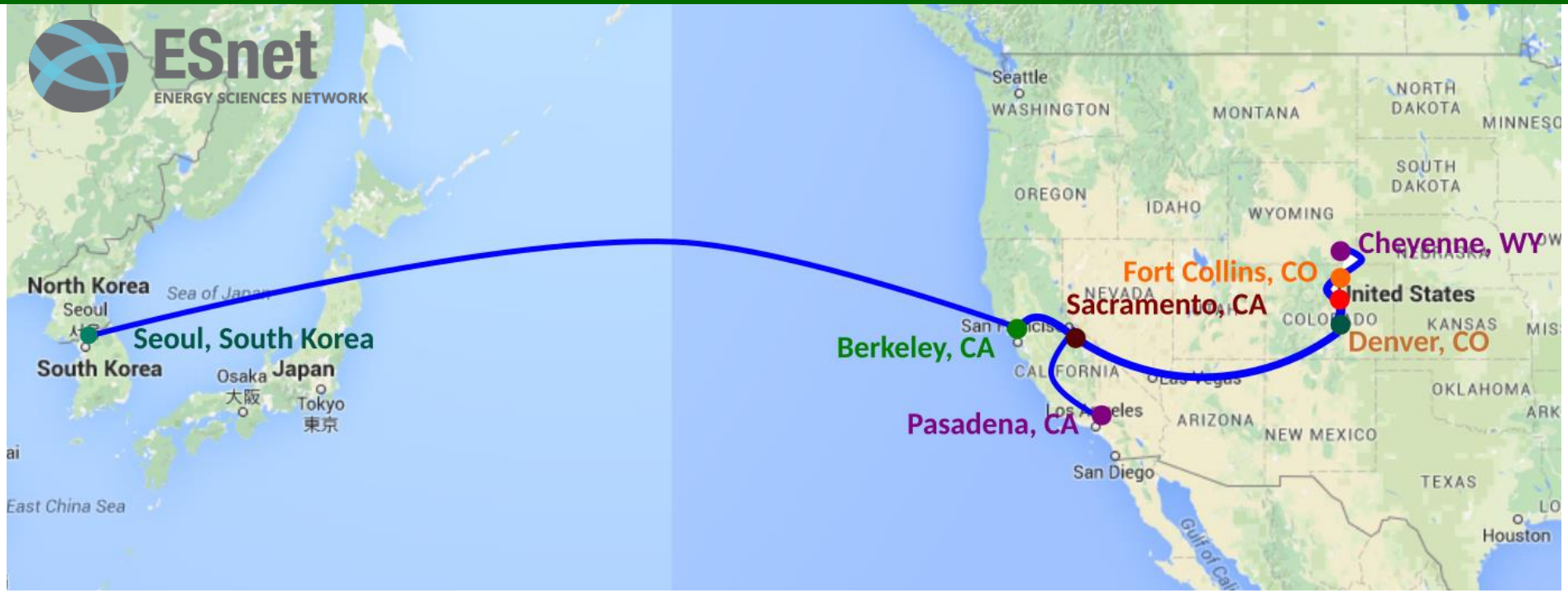
NDN Catalog


Catalog node 1
Data storage
Catalog node 3
(1)Publish Dataset names
(2) Sync changes
Publisher
NDN
(3) Query for Dataset names
Data storage
(4) Retrieve data
Consumer
Catalog node 2

Forwarding Strategies



Science NDN Testbed



- NSF CC-NIE campus infrastructure award
  - 10G testbed (courtesy of ESnet, UCAR, and CSU Research LAN)
- Currently ~50TB of CMIP5, ~20TB of HEP data

Conclusions

- NDN encourages common **data** access methods where IP encourages common **host** access methods
  - NDN encourages interoperability at the content level
- NDN unifies scientific data access methods
  - Eliminates repetition of functionality
  - Adds significant security leverage
  - Strongly encourages and rewards structured naming

For More Info

christos@colostate.edu

http://named-data.net

http://github.com/named-data