

# **Named Data Networking (NDN)**

**Introduction to NDN**

# Named Data Networking (NDN)



Northeastern University



<http://named-data.net> • <http://github.com/named-data>

# IP

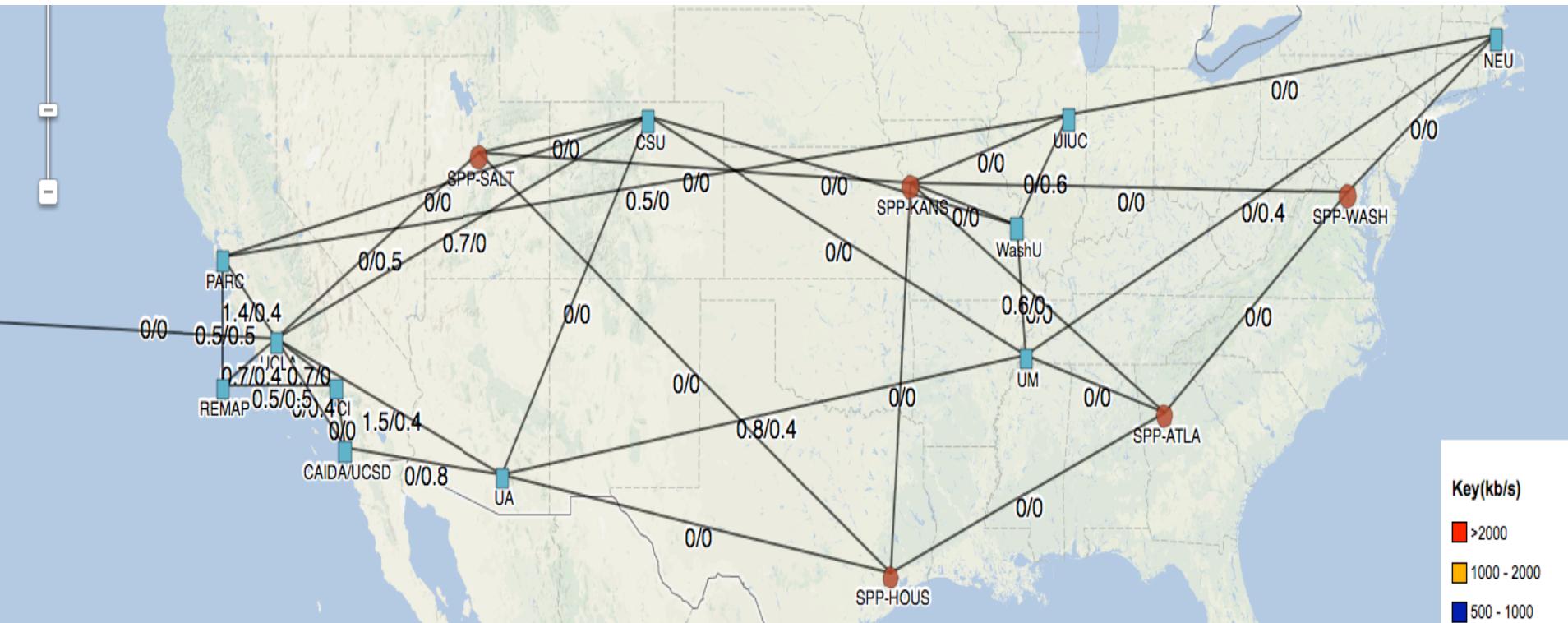


# NDN

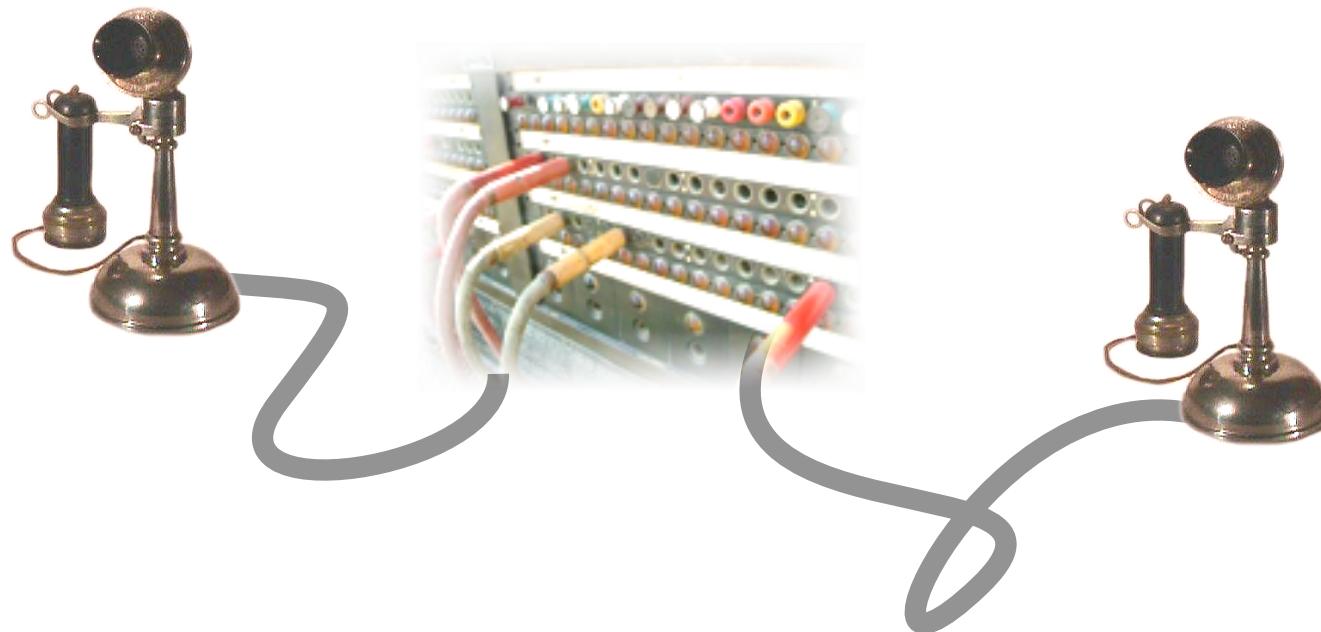


# Is It Deployed?

- Deployable now as an overlay (TCP, UDP) or on Layer 2 transport
  - C, C++, Java, Python, and Javascript libraries
  - <http://github.com/named-data>
  - **Testbed of software routers**

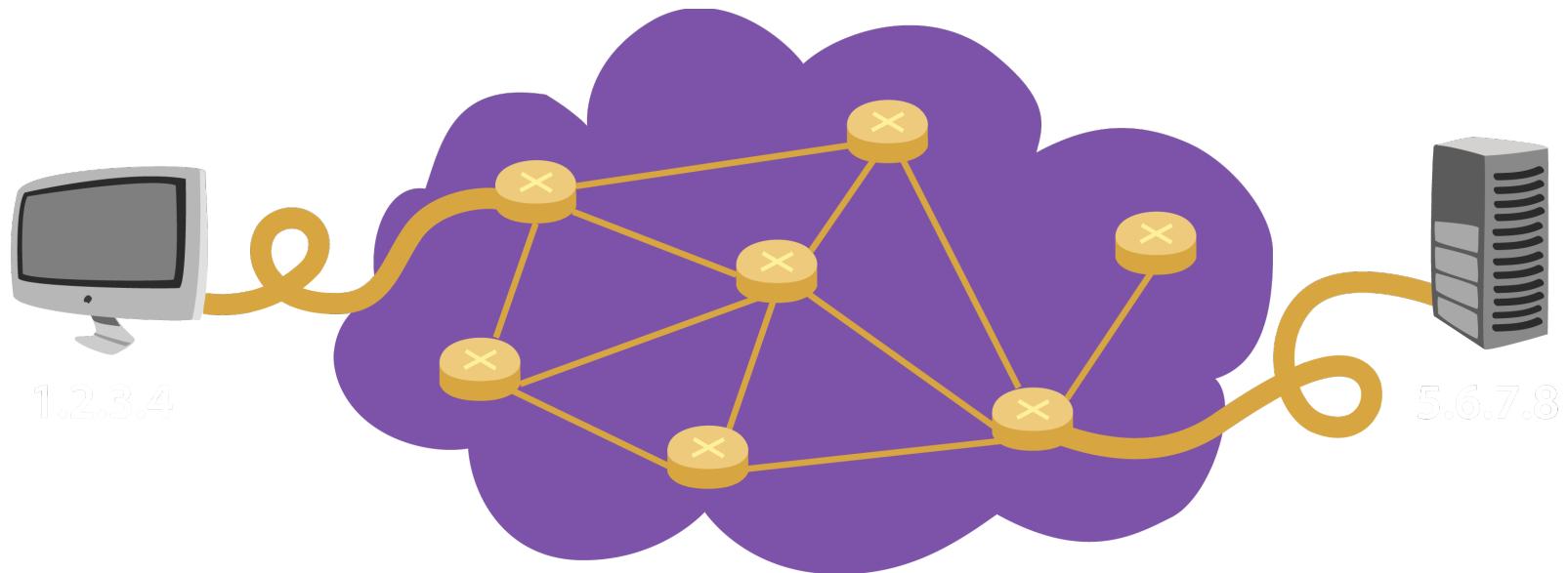


# Telephone Network was the 1<sup>st</sup> Communication System



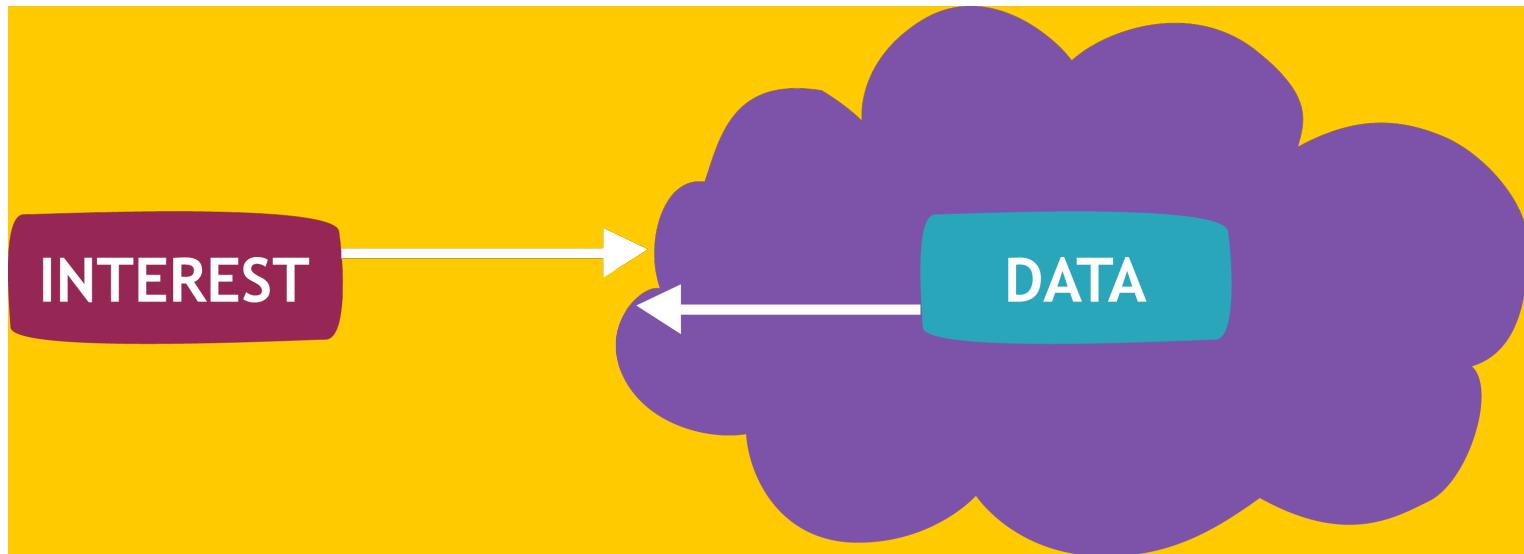
Focus on building and connecting the wires

# IP Revolutionized the Communication System



**Internet Protocol (RFC791):** Focused on delivering packets to destination *host*

# NDN: Focus on Data



Abstracting away the notion of “host”

*Superset of host-to-host communication model*

# Two Problems with Current Internet

- Focus is on end-point communication
  - Artifact of original thinking: share resources, not content
  - Login to fast machine, access to the tape drive, the printer, etc.
- Security
  - To get data, you build a secure path
  - Once you authenticate with the server, you trust the content

# New Communication Paradigm

- Users today care about *content*, not the servers
- Accessing the server is a by-product of the need to retrieve the desired content
  - If the server is down, no access to the content
- But what if the content was available from other places (e.g., my neighbor)?
- We do a lot of this already with HTTP
  - URLs, CDNs, caches, etc.

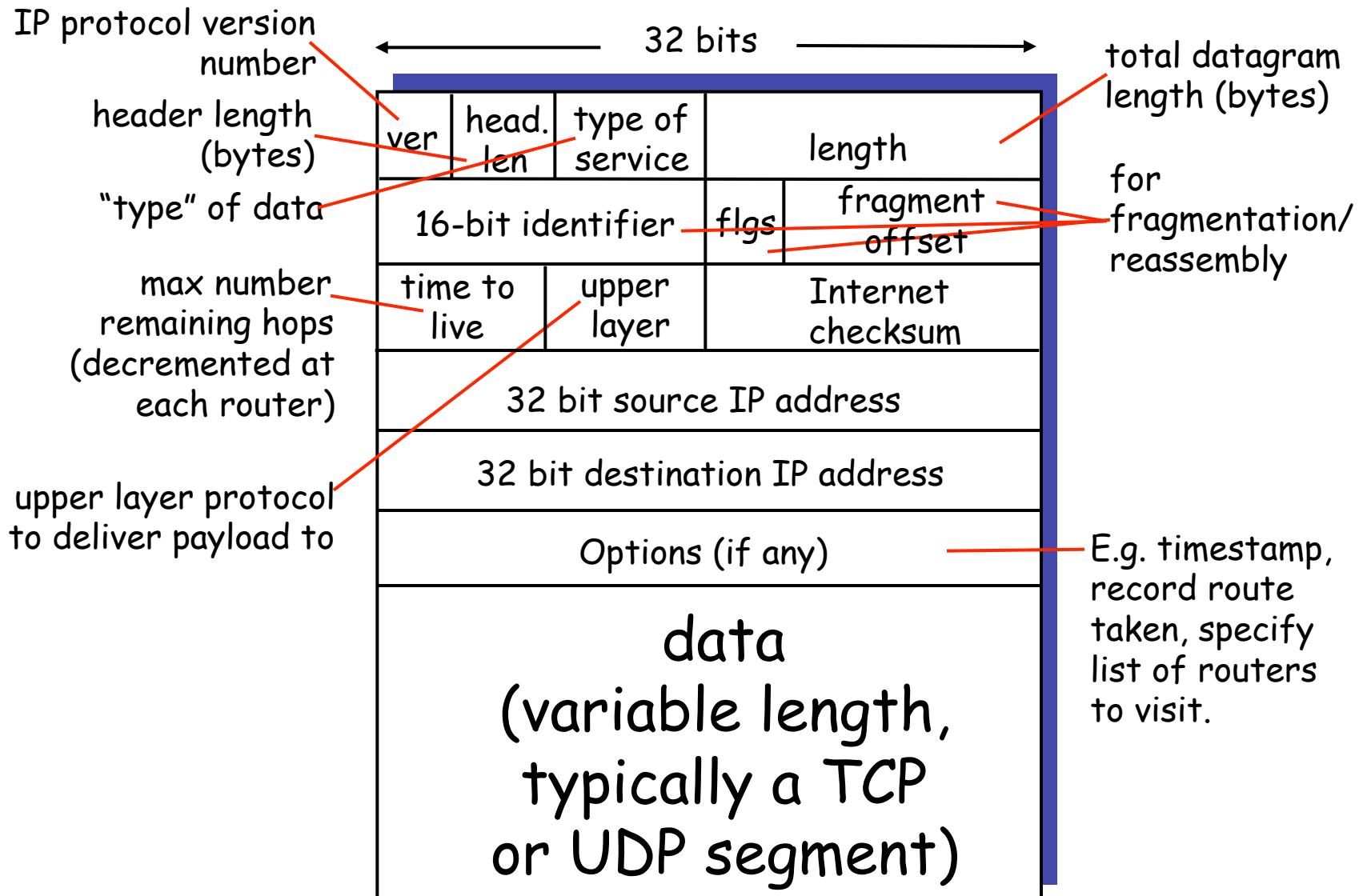
# Two Focal Points in NDN

- Focus on the **what** not the **where**
- Secure the **data** not the **container**

# NDN Operation

- Interest packets
- Data packets
- Enhanced Forwarding
  - Pending Interest Table (PIT)- **new!**
  - Content Store (CS) – **new!**
  - Forwarding Information Base (FIB) –  
similar to IP

# The IPv4 Datagram Format



# NDN Packets

## Interest Packet

**Content Name:**  
Identifies the data I want to receive

**Selector:** identifier publisher, etc

**Nonce**

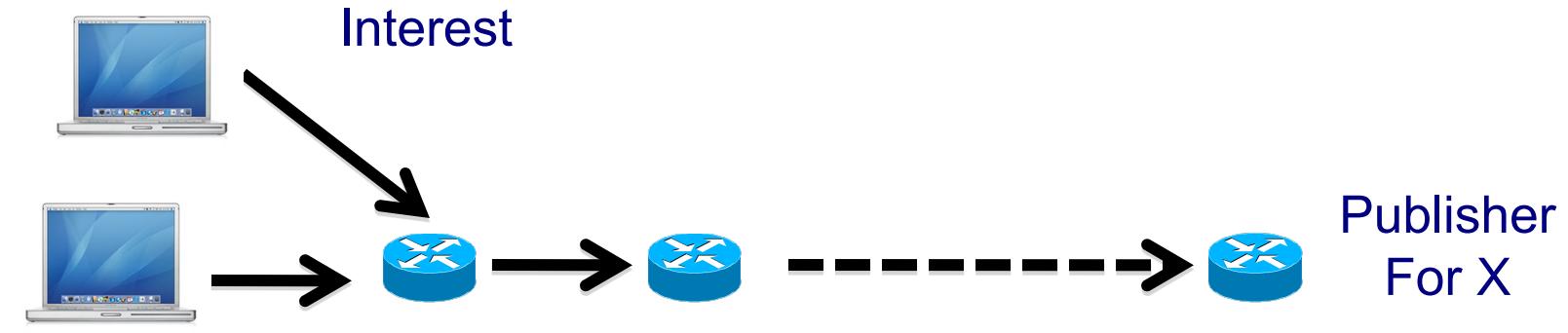
## Data Packet

**Content Name:**  
Identifies the data in this packet

**Signature:** Required for all packets

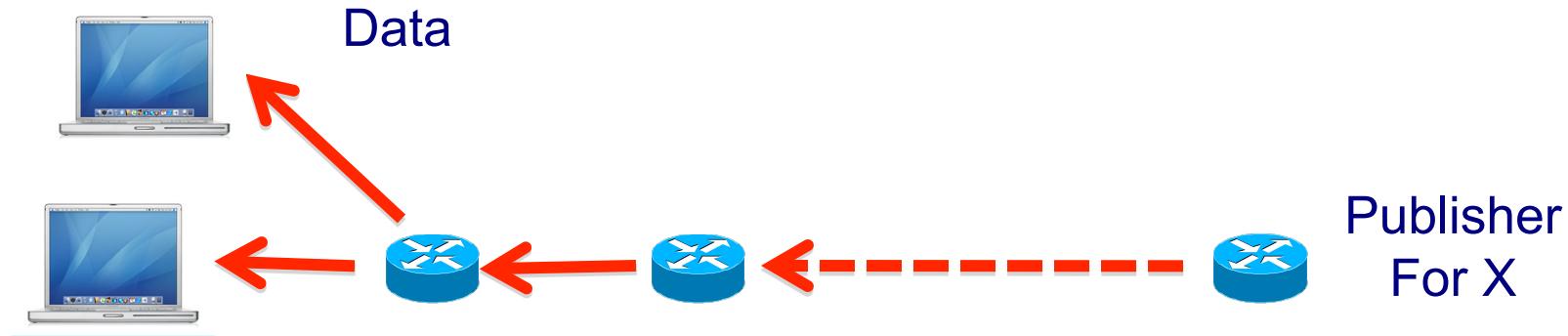
**Data**

# NDN Forwarding: Interests



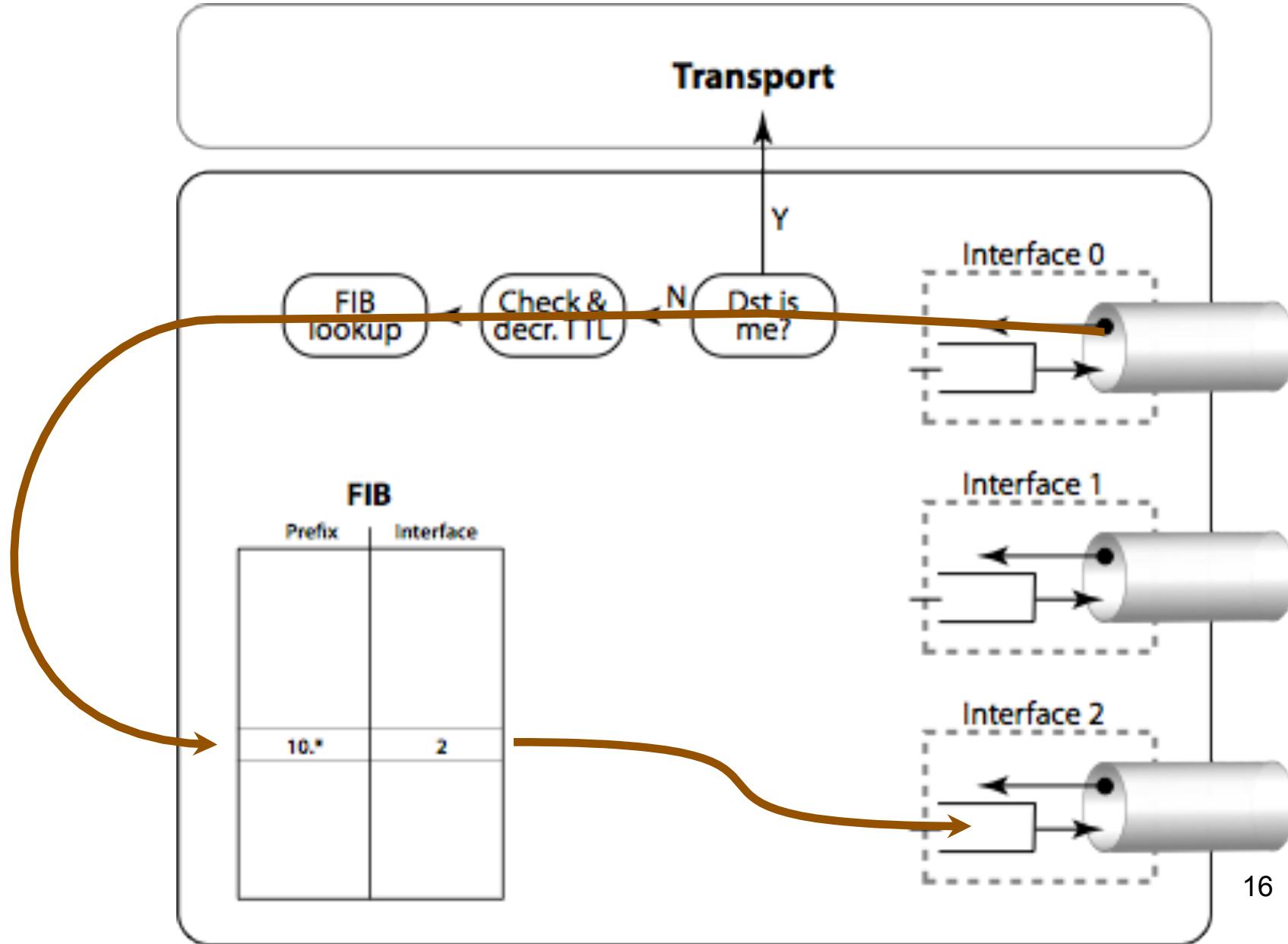
- Interest: Content Name (CN) = X
- Forward interest towards Publisher (X)
- Mark incoming faces as wanting X (lay down breadcrumbs)
- Merge same interests for X

# NDN Forwarding: Data

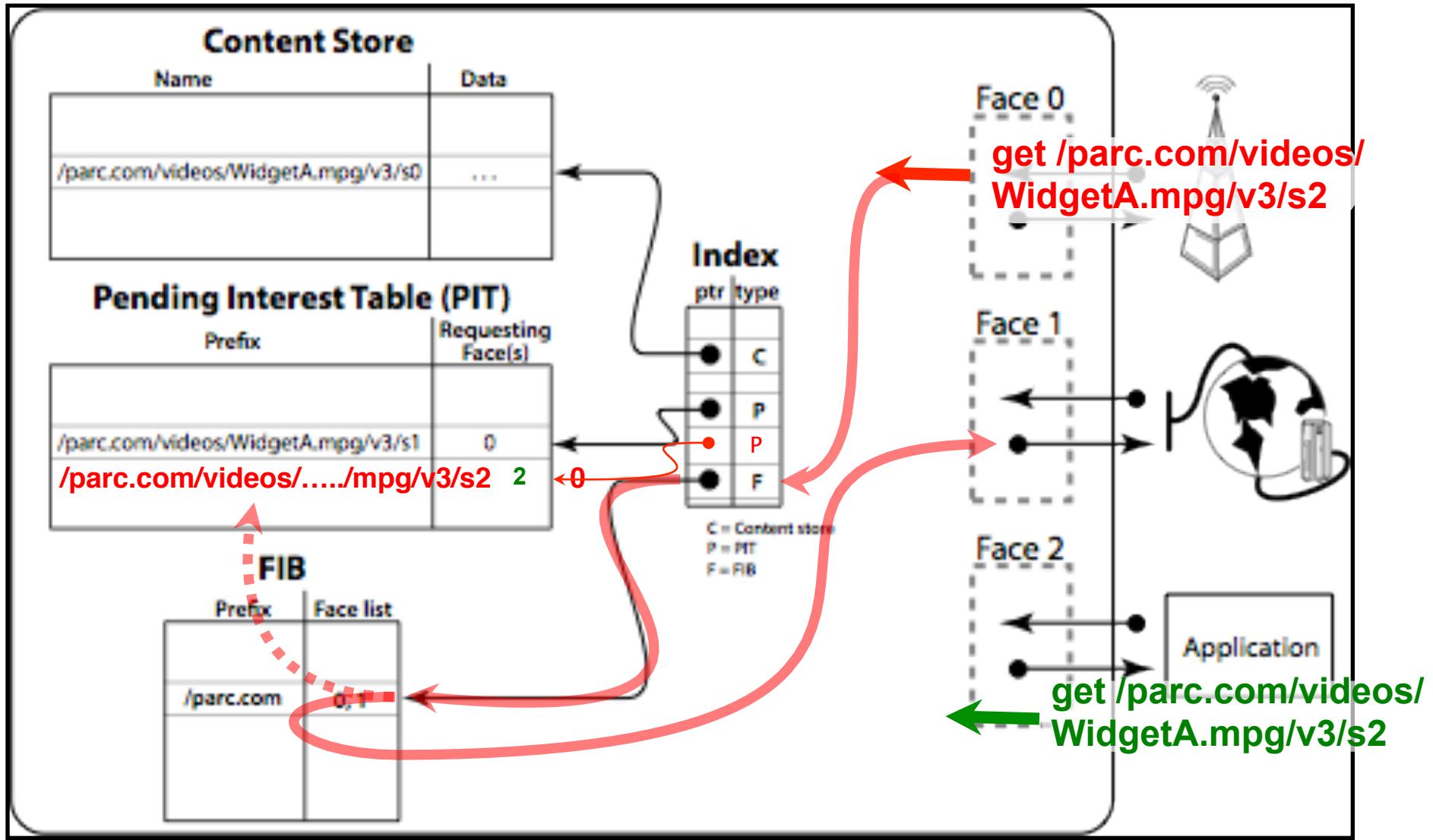


- Data: Content Name (CN) = X  
Forward Data back to where interests came from
  - Follow the breadcrumbs back to requestors
  - Delete breadcrumbs
- Duplicate at appropriate routers
- Cache data at each router

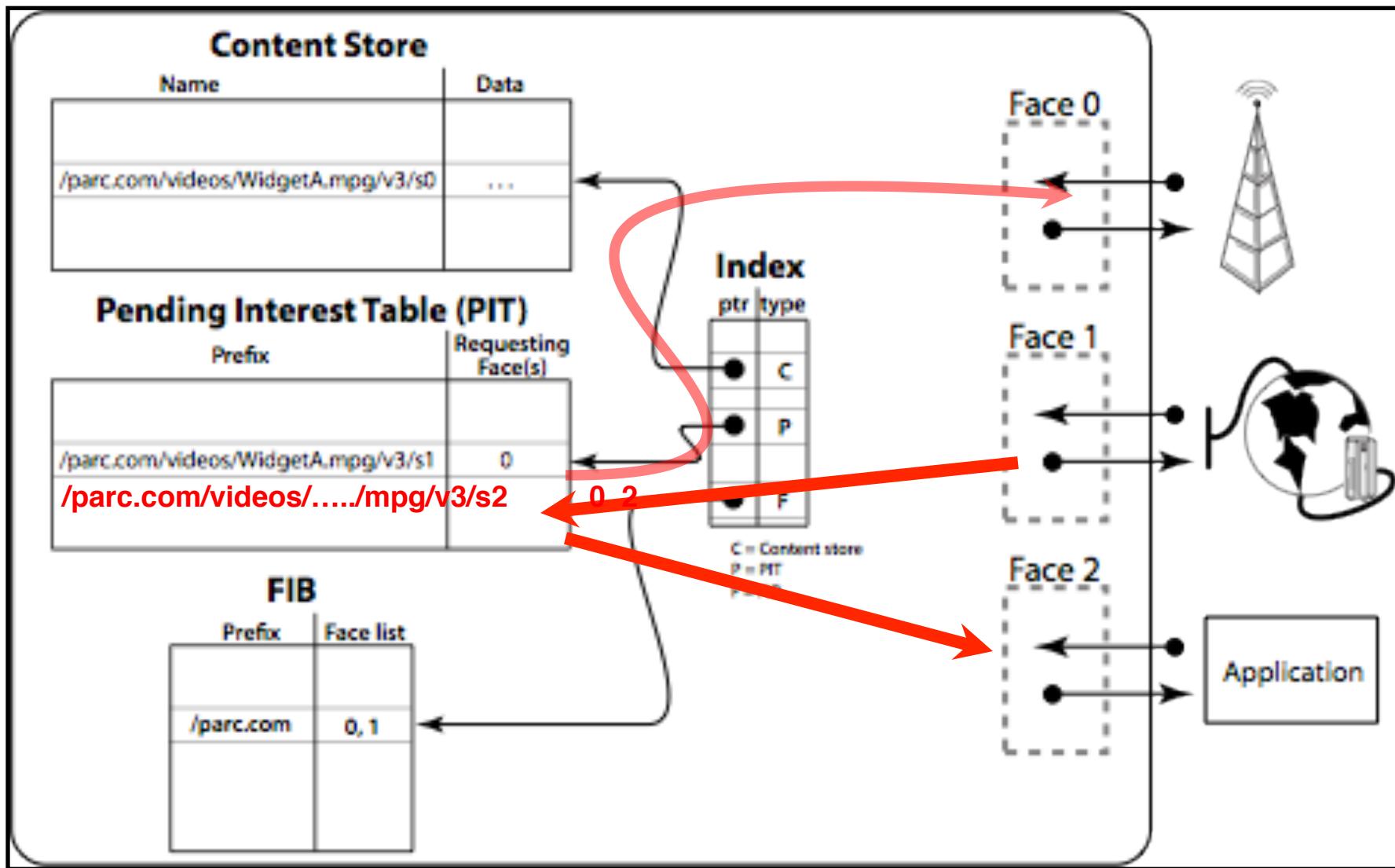
# IP Packet Forwarding



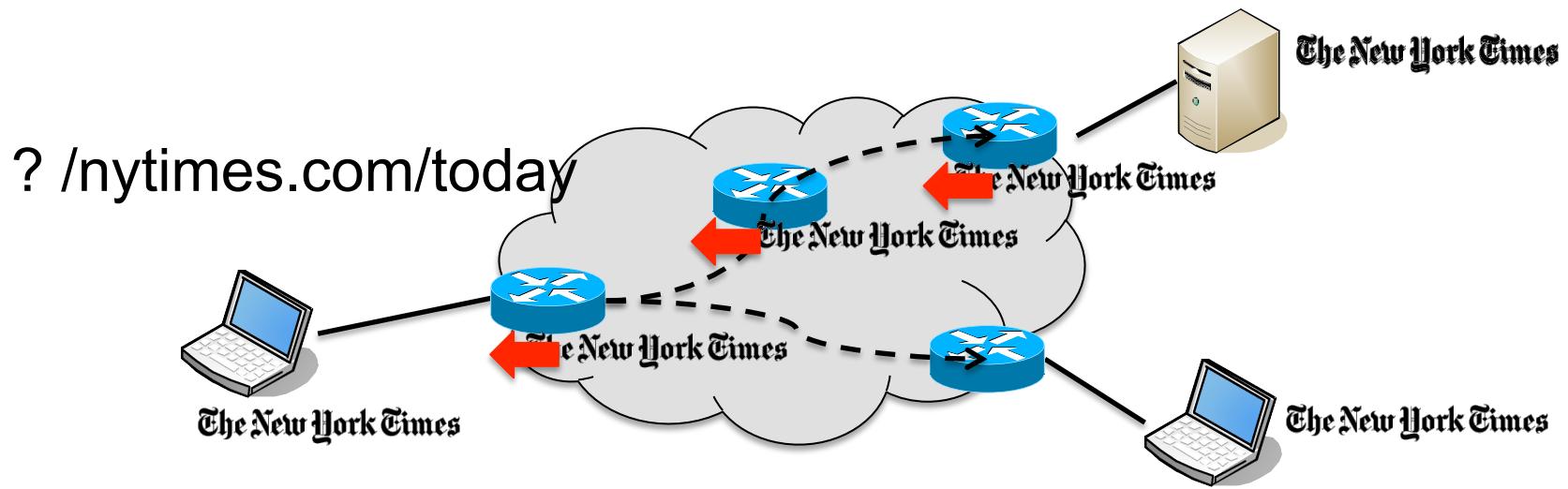
# NDN Interest Forwarding



# NDN Data Forwarding

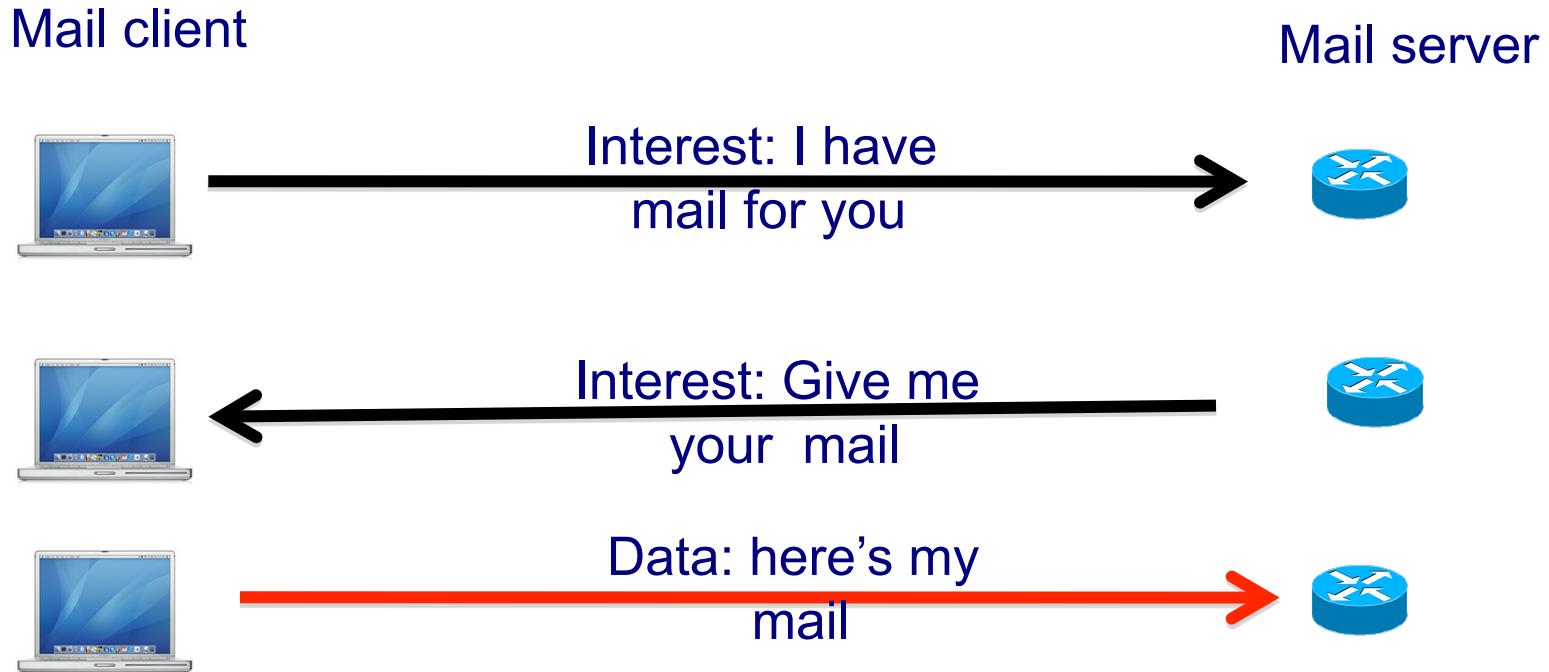


# Summary



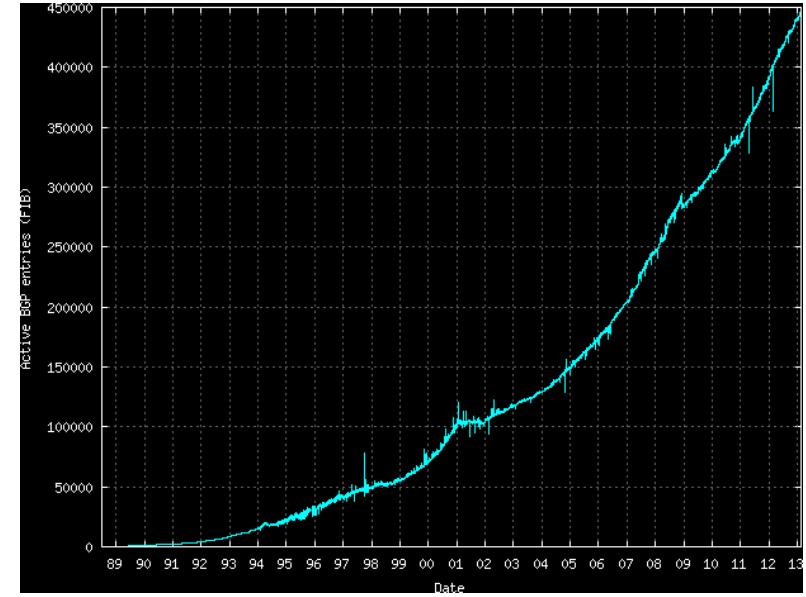
- Route on content names
- “Breadcrumbs” & de-duplication of requests
- Cache retrieved data
- Content from anywhere: not just the producer

# Example: Sending Mail



# Can it Scale?

- WUSTL Results for NDN Forwarding
  - (in submission)
- Software router prototype
- Preliminary hardware design
- Multi-gigabit forwarding rates for:
  - Name-based FIBs, based on real world URLs, of 1-3M entries;
  - Synthetic FIBs, based on model of future namespace, of up to 1B entries.



<http://www.cidr-report.org>

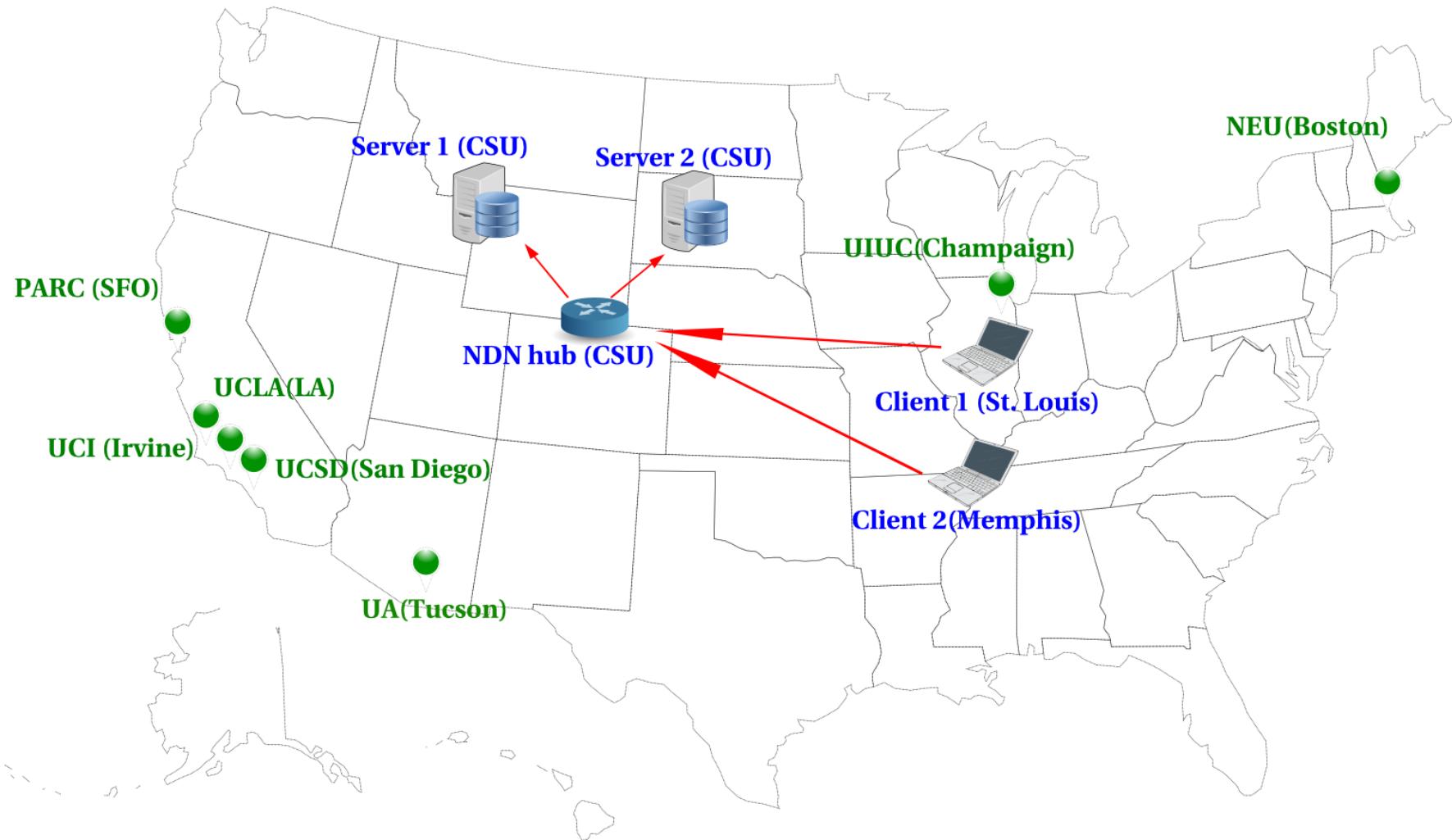
All	New	Deleted	Transferred	TLD
144,040,087	155,151	136,956	239,097	All TLDs
107,508,504	115,331	100,189	194,269	.COM
15,033,351	16,353	13,437	20,649	.NET
10,204,641	9,829	7,165	10,224	.ORG
7,185,246	8,227	12,882	8,916	.INFO
2,305,965	3,715	1,875	2,895	.BIZ
1,802,380	1,696	1,408	2,144	.US

<http://www.whois.sc/internet-statistics/>

# The Power of Naming

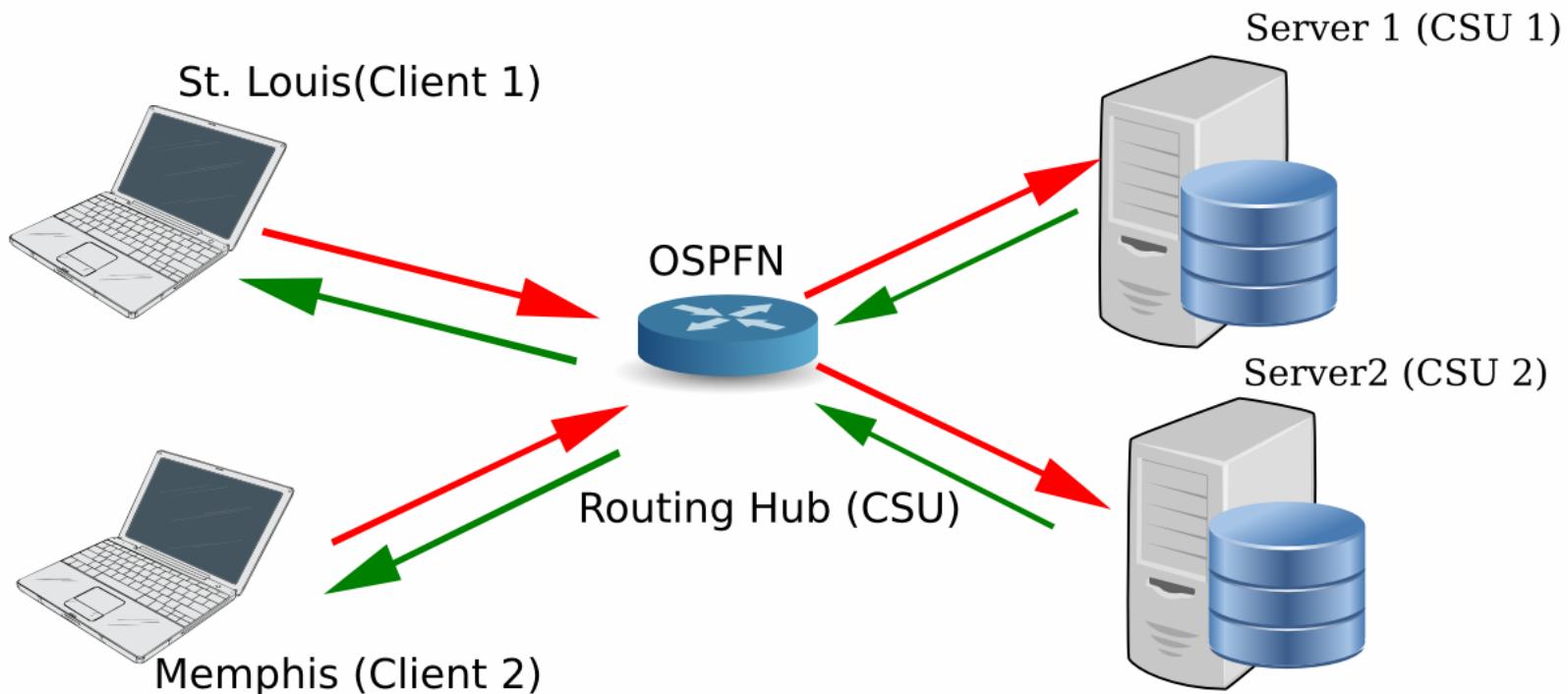
- Fetch the data transparently regardless of location
  - Requests can go to appropriate place
- Result in generation of new data
  - Can ask for data that does not yet exist!
- Naming structure allows us to learn about what's available

# Experiment Topology



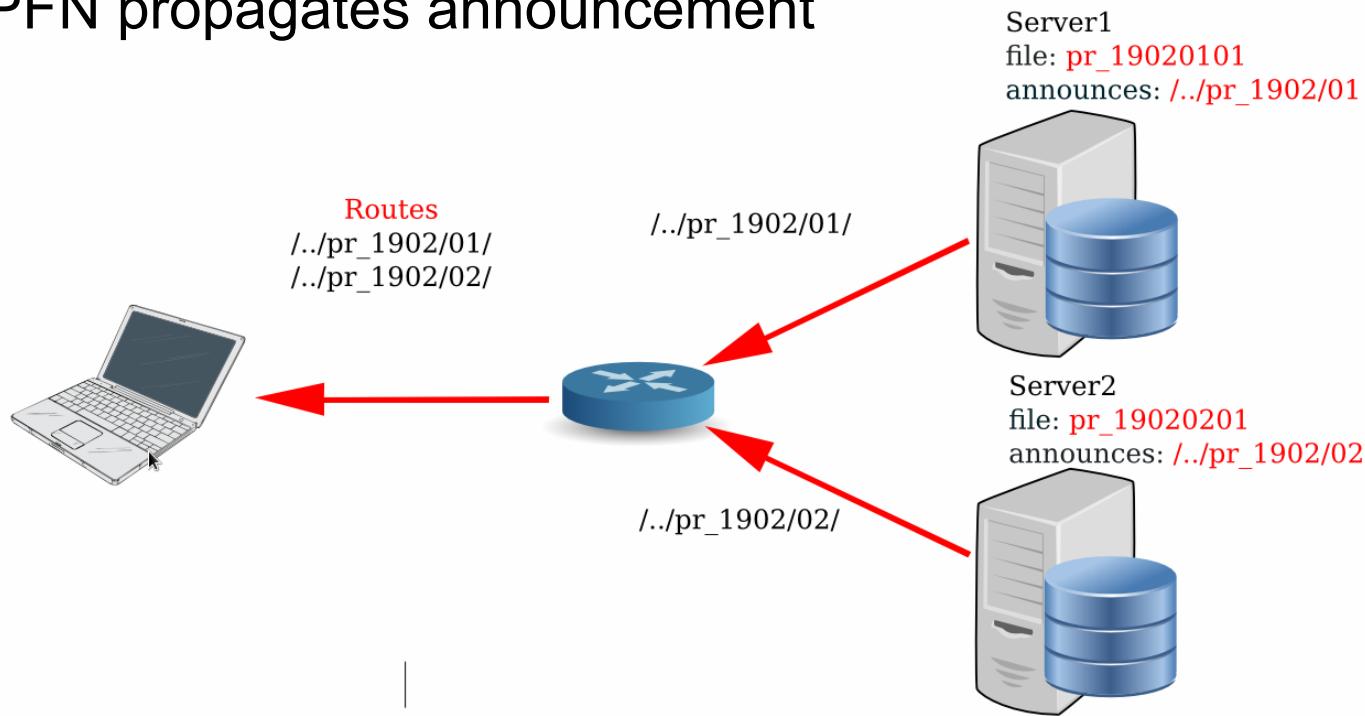
# Experiment Setup

- Two servers and two clients
- Servers at CSU, clients at Memphis and St. Louis
- Nodes exchange routes using OSPFN



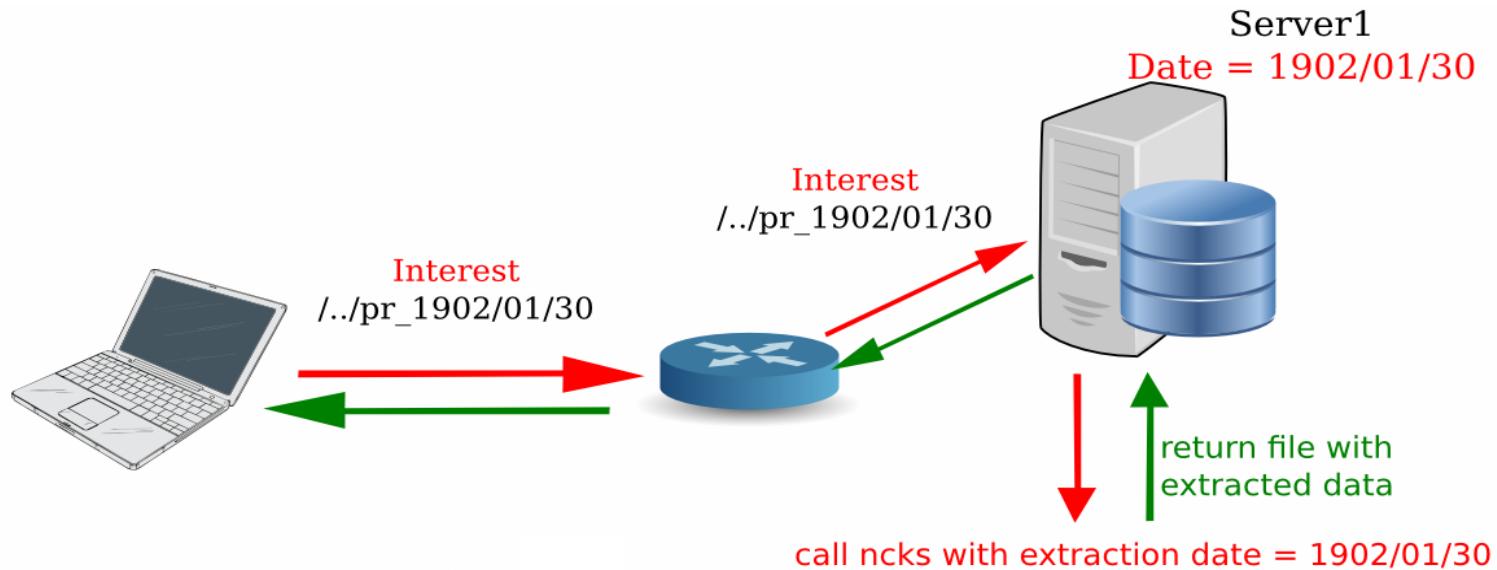
# Announcements

- Servers have .nc files, each .nc file have one month's data
- Route announcements in network are based on filename
- Each server advertises one prefix for a file
  - Server having file pr\_190201.nc announces ../../pr\_1902/01/
- OSPFN propagates announcement



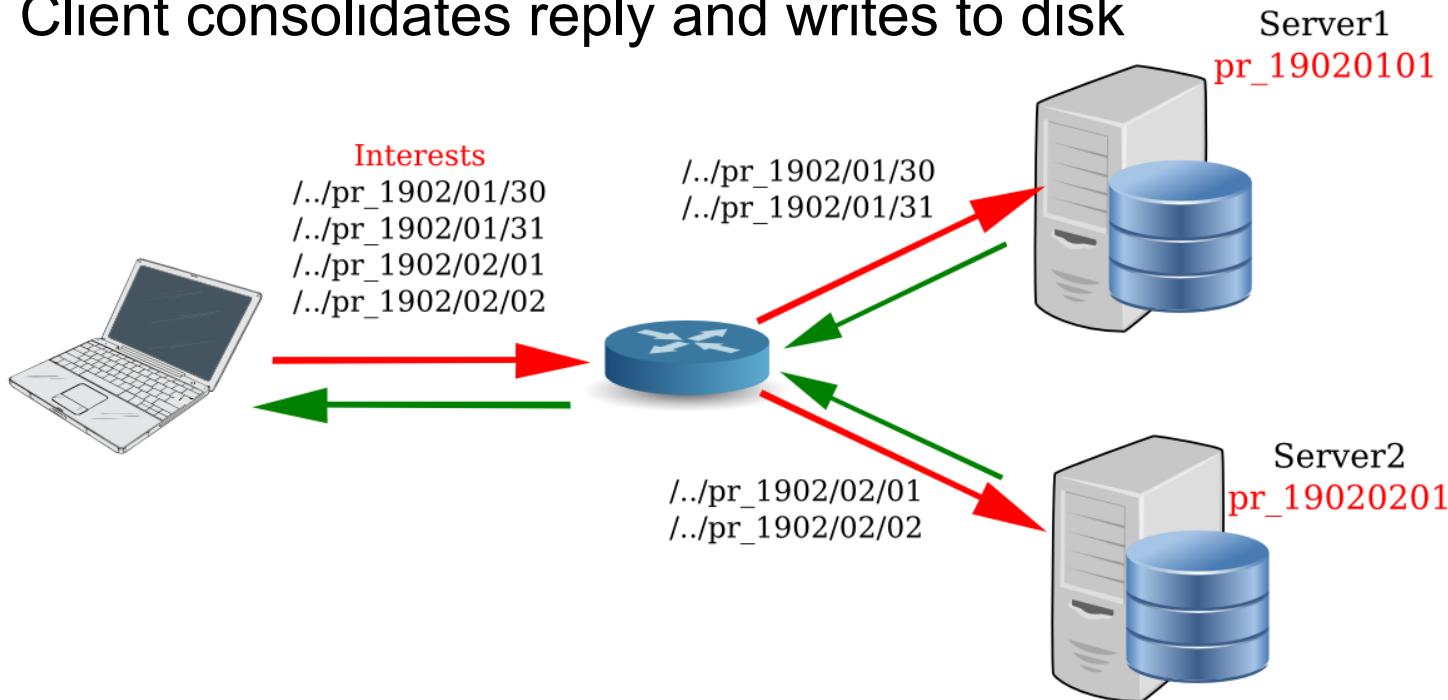
# Dynamic Data Generation

- Servers parse interest names and find the date range
- Pass date range to ncks tool.
- ncks tool extracts data, writes to file and returns the filename to server
- Server sends back file



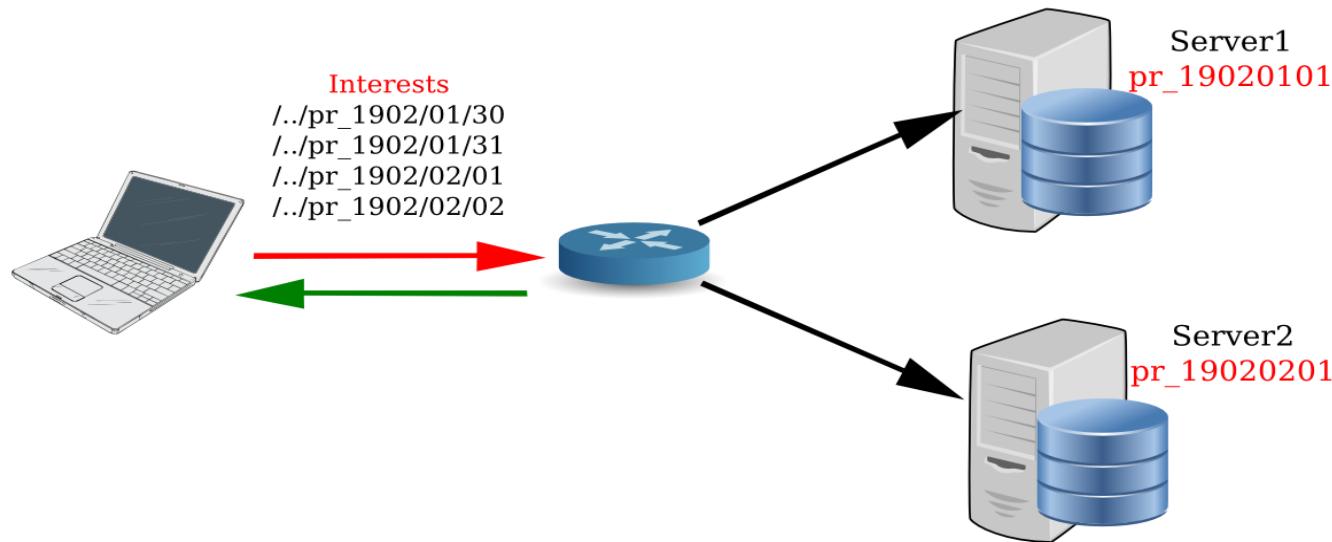
# An Example Data Request

- Want data for Jan 30 – Feb 02
- Client expresses interests, one for each day
- Interests for Jan 30-31 go to server1
- Interests for Feb 01-02 go to server2
- Data is dynamically generated and sent back
- Client consolidates reply and writes to disk



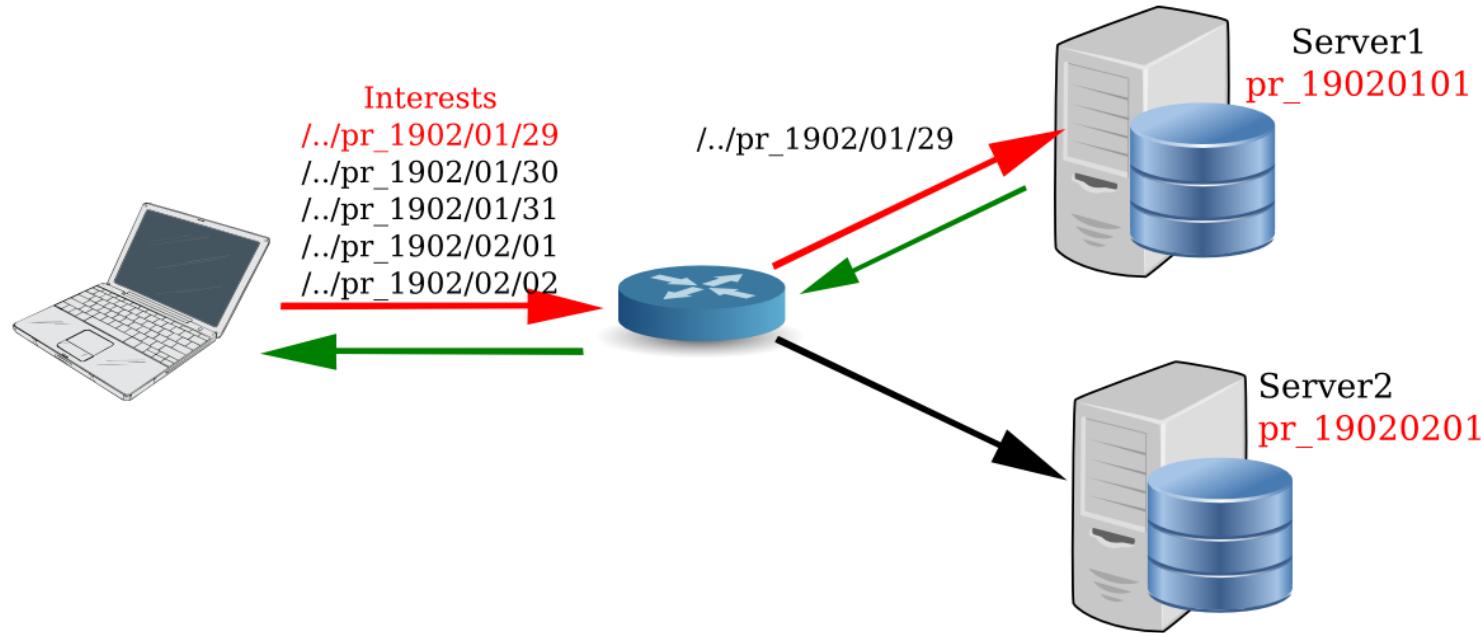
# Repeat Requests and Cache

- If asked for same data, requests are answered from cache
- Saves transmission time, extraction time and transfer time
- Remember: security is on the data, not the channel!



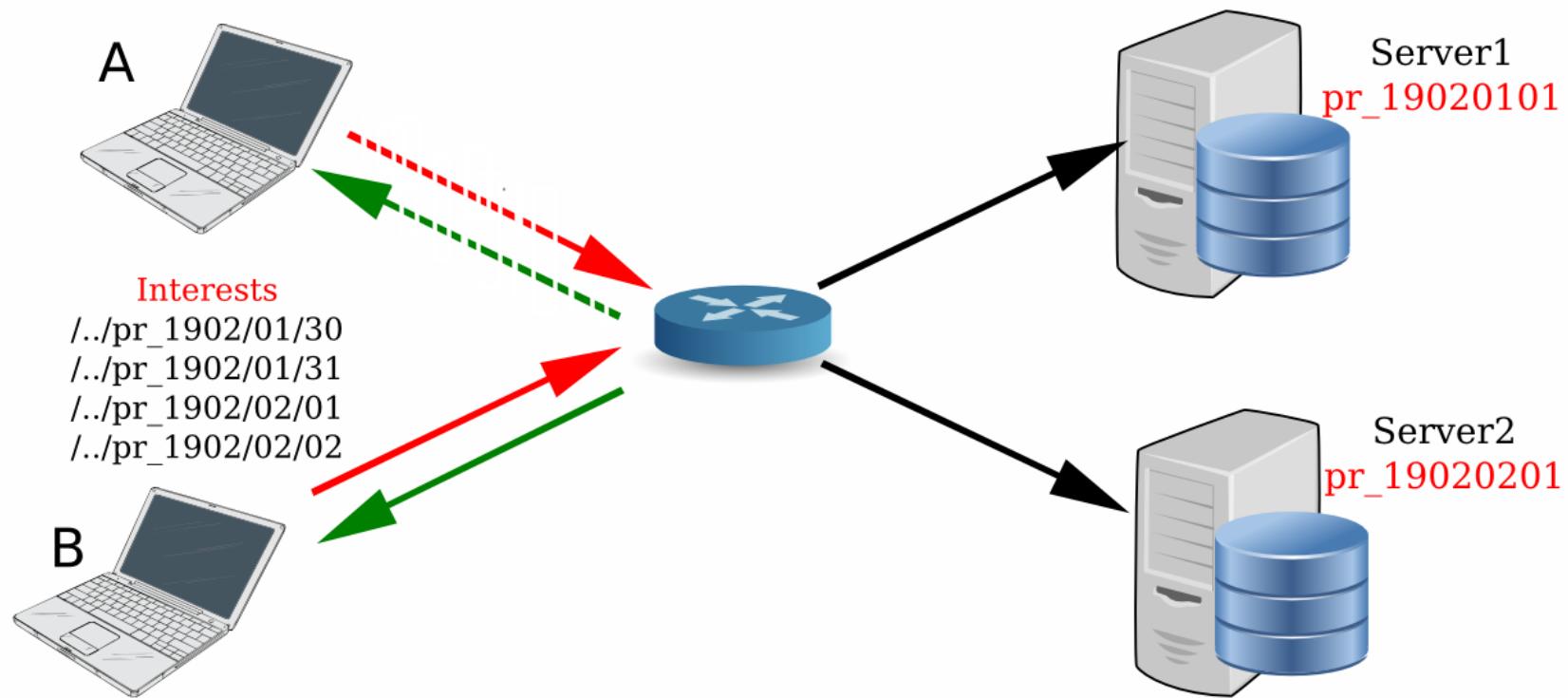
# Partially Cached Data

- What happens if we ask for Jan 29 – Feb 2 ?
- Request for data not cached goes to server
- Rest is answered from cache

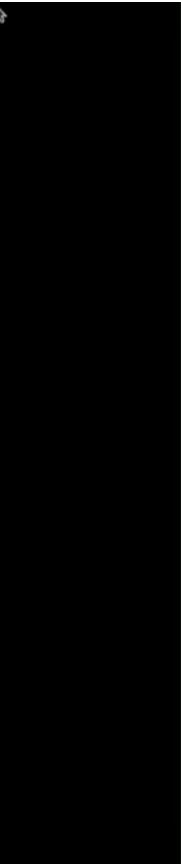


# Collaborations

- A asked for data for Jan 30- Feb 2.
- B later asks for same data.
- B receives data from cache.



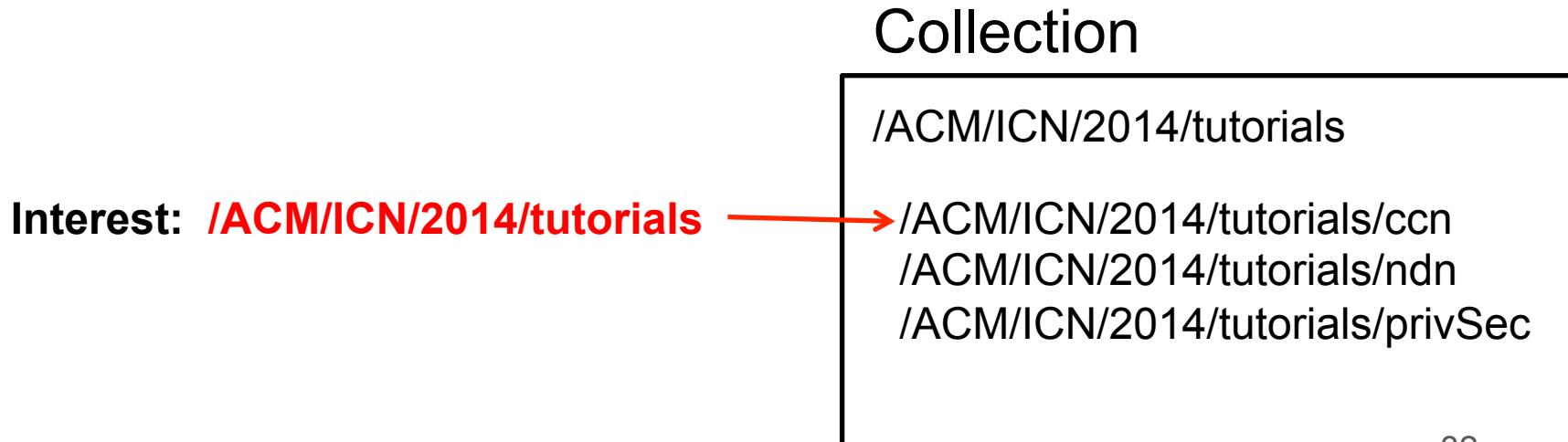
# Demo: Automatic Failover



Failure Recovery

# Interest-Data Matching

- An NDN prefix represents a collection of objects
  - NDN Interests may match multiple Data Objects in a collection
  - In response to an Interest, NDN returns the first (canonically ordered) matching Data Object

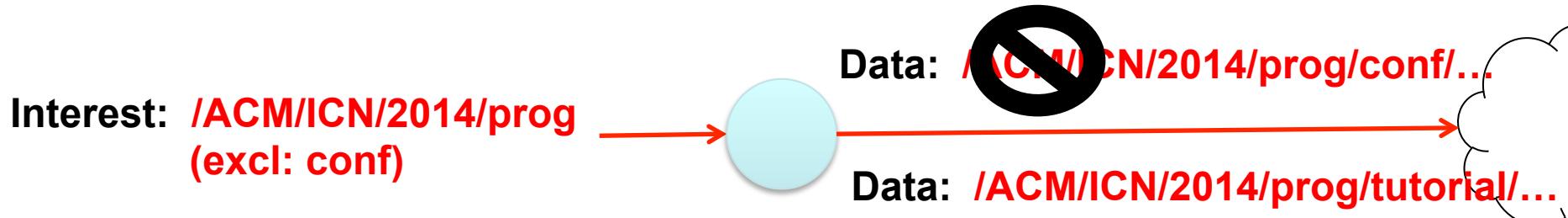


# Enumerating Objects in a Collection

- Suppose I want to discover if a NDN tutorial is available at ICN; all I know is conference name and year
  - Send Interest for **/ACM/ICN/2014**
  - Receive **/ACM/ICN/2014/prog/conf/Friday/paper1**
- Answer reveals information about name structure
  - Send new interest for **/ACM/ICN/2014/prog** excluding “**conf**”
  - Receive **/ACM/ICN/2014/prog/tutorials/ccn**
  - Send new Interest for **/ACM/ICN/2014/prog/tutorials** excluding “**ccn**”
  - Receive **/ACM/ICN/2014/prog/tutorials/ndn**
- Exclude capability provided through **Selectors**

# Selectors

- Selectors are optional elements that further qualify Data that may match the Interest
  - Used for discovering and choosing the Data that best matches what the application wants
  - Achieved by constraining matches against Data's name and meta-info
  - Enforced by each node when attempting to satisfy PIT entries



# Enumerating Objects in a Collection

Interest: /ACM/ICN/2014

Data: /.../prog/conf/Friday/paper1

Interest: /.../prog  
(excl: conf)

Data: /.../prog/tutorials/ccn

Interest: /.../prog/tutorials  
(excl: ccn)

Data: /.../prog/tutorials/ndn

Knowledge

/ACM/ICN/2014

/.../prog  
/.../prog/conf  
/.../prog/conf/Friday  
/.../prog/conf/Friday/paper1

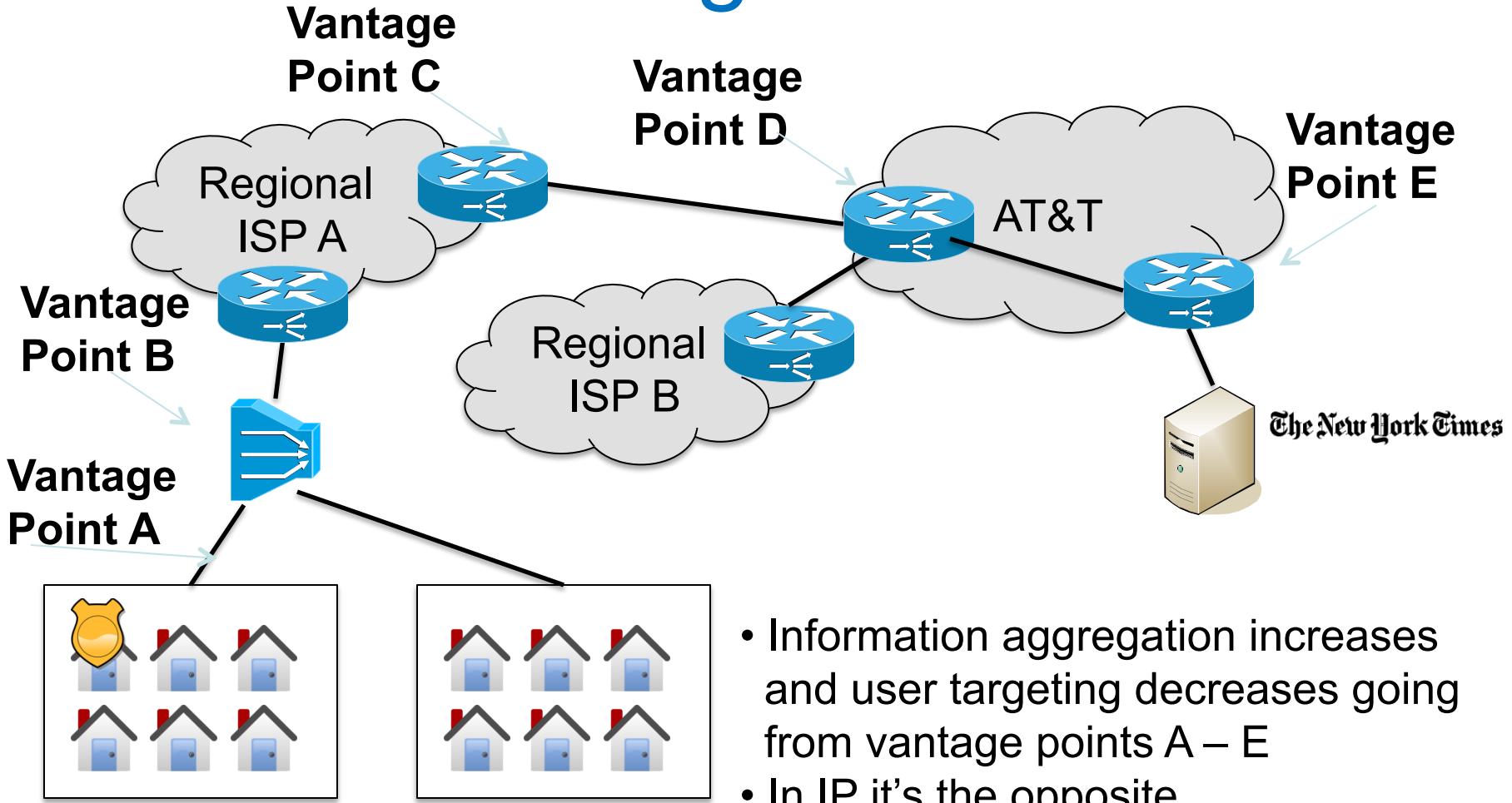
/.../prog/tutorials  
/.../prog/tutorials/ccn

/.../prog/tutorials/ndn

# NDN and Anonymity

- NDN has no endpoint addresses – names in interests and data packets are ephemerally associated with incoming/outgoing faces
- Info retrievable at a router:
  - PIT – interest/data names and associated face information; in-memory, ephemeral ( $\sim$ RTT)
  - Content Store – data names, no face information; in-memory, ephemeral ( $\sim$ nRTT)
  - Attached storage (repo) – data names, no face information; on non-volatile memory, long-lived
- Individual endpoints, however, engaged in a private conversation *can still be identified* by their names

# In NDN Vantage Point Matters



# Recap

- In NDN routers hold ephemeral name-to-interface associations – no e2e associations
- Vantage point matters
- Caching may satisfy interests before they reach your vantage point
- Multipath may divert interests away from your vantage point
- But private parties still visible on the wire

# DDoS Attacks

- Interest packet flooding
  - Standard push-back defenses still possible
  - Smart decisions based on parsing names
- Content poisoning

# Congestion Control

- Use lessons learned from TCP – mechanisms carry over
- Note that receiver window is not needed – receiver pulls what it wants
- Key difference: current thinking is hop-by-hop instead of end-to-end

# Key Distribution

- No single way to distribute keys
  - Key distribution outside the architecture
  - Certificates, consensus, out-of-band, applications are free to implement anything that works
  - Packets tell you how to get the key (or may even carry the key with them)
- Key delegation
  - Example: [www.nytimes.com](http://www.nytimes.com) can delegate keys to editors for [www.nytimes.com/sports](http://www.nytimes.com/sports), [www.nytimes.com/business](http://www.nytimes.com/business), etc.

# Conclusions

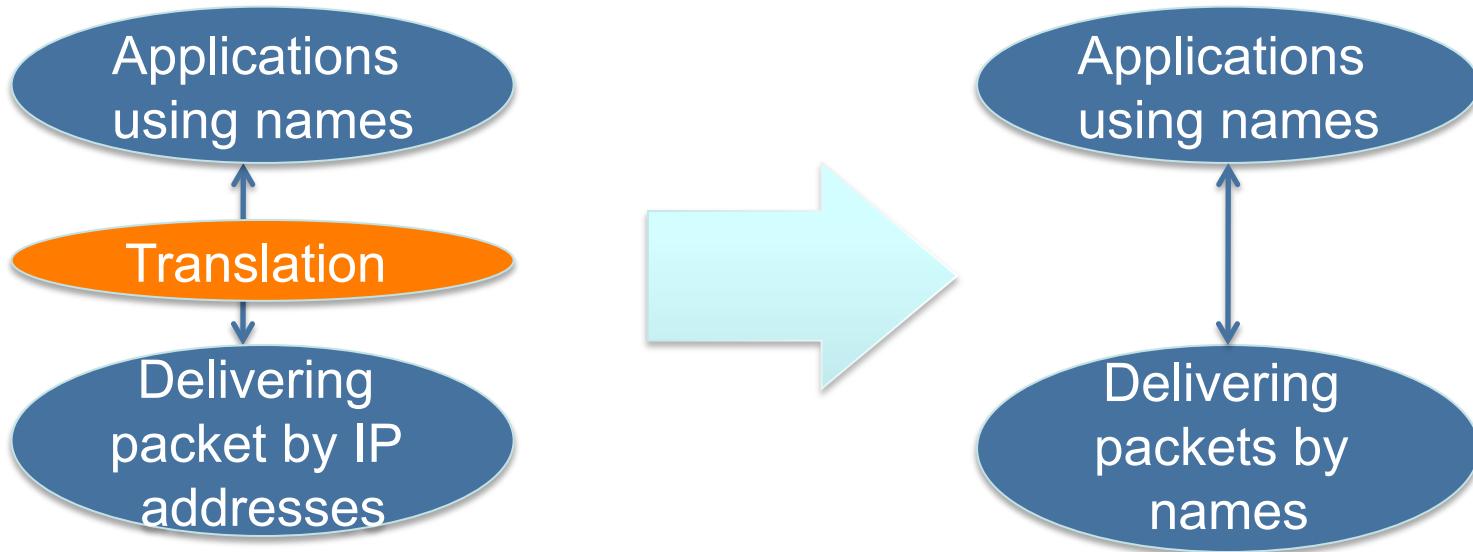
- NDN preserves the hourglass shape of IP but with names at the narrow waist
- Architecture focused on the what, not the where
- New forwarding mechanisms enable multipath, multicast and other group operations
- All content is signed
- More at **<http://www.named-data.net/>**

# **BACKUP SLIDES**

# Naming the Content

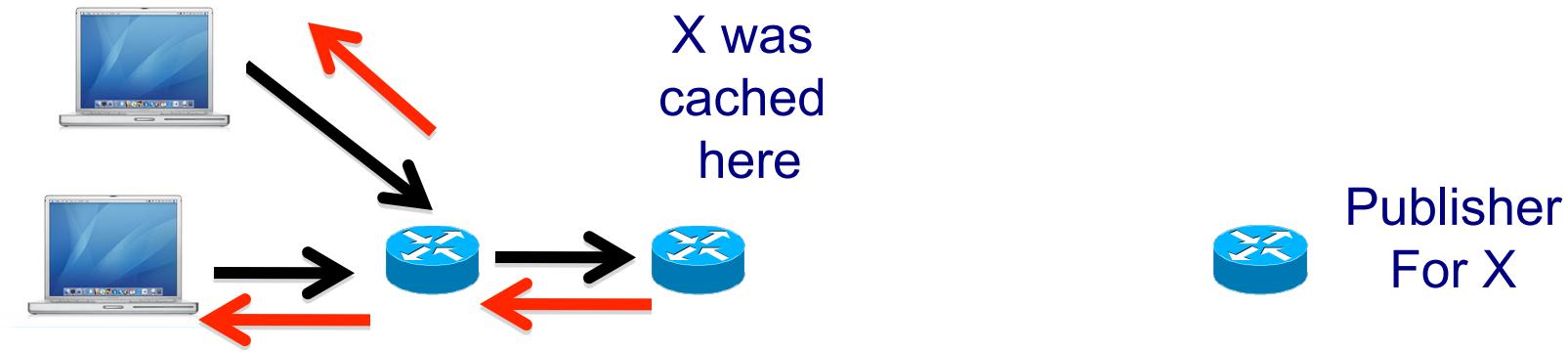
- Client requests [www.nytimes.com/today](http://www.nytimes.com/today)
- Interests go out for each **packet**:
  - [www.nytimes.com/today/packet1](http://www.nytimes.com/today/packet1)
  - [www.nytimes.com/today/packet2](http://www.nytimes.com/today/packet2)
  - ...
- Routers forward based on [www.nytimes.com](http://www.nytimes.com) prefix (longest prefix match, just like IP)
- Data is pulled and cached one packet at a time
- Each packet contains information on how to retrieve the signing key

# Communication by Names



- Producer announces data prefix
  - e.g., [www.nytimes.com/](http://www.nytimes.com/)
- Consumer sends interest
- Producer replies with data

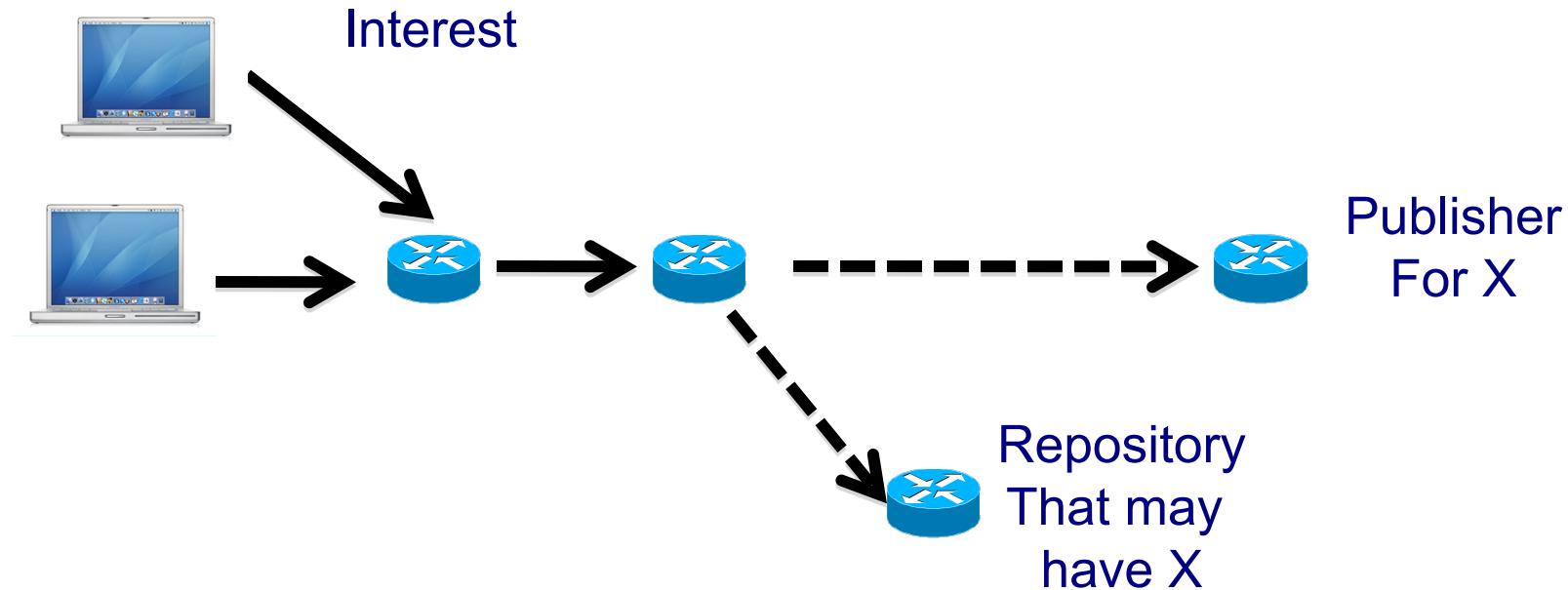
# Hitting Cached Data



Interests only go so far until they find the data

Cached data can satisfy requests efficiently

# Multipath Interest Forwarding



- Interests may be forwarded opportunistically to many destinations
  - Strategy Layer
- Data may be concurrently retrieved from multiple places

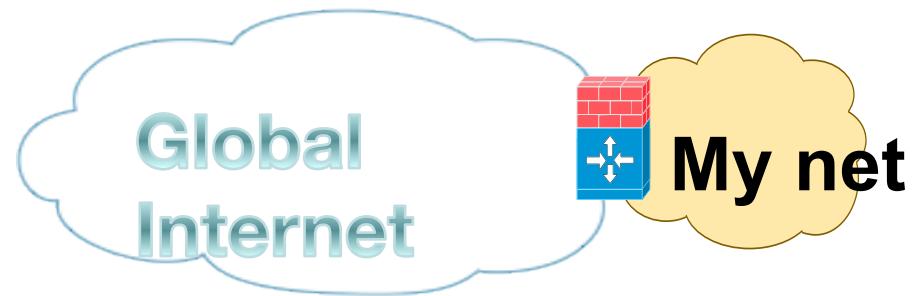
# Transparency in NDN

With a search warrant for a router, what can you discover about an ISP's users?

- *Assumption: warrant covers volatile and non-volatile memory*

# Network Security in IP: why so Hard?

- IP identifies interfaces/hosts
- Current attempts aim at:
  - Securing the box
  - Securing the channel
  - Securing an IP network by firewall
- Securing the perimeter is hard**



# 30 Years Down the Road

## IP changed the world

Interconnections  
of computers

Moore's Law &  
silicon revolution



facebook



A new world of  
applications &  
computing  
devices



Google

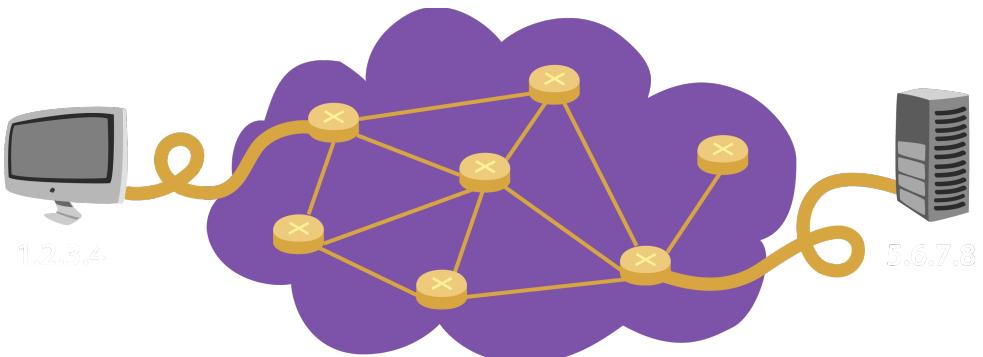
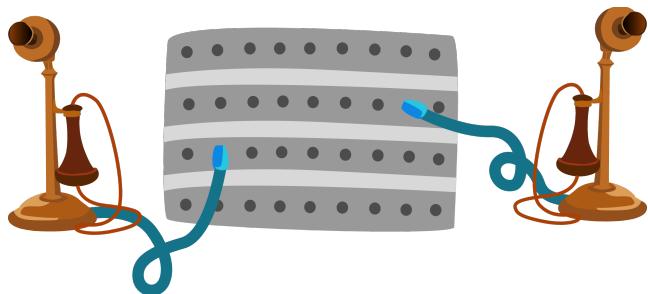
YouTube



# So Why a New Architecture?

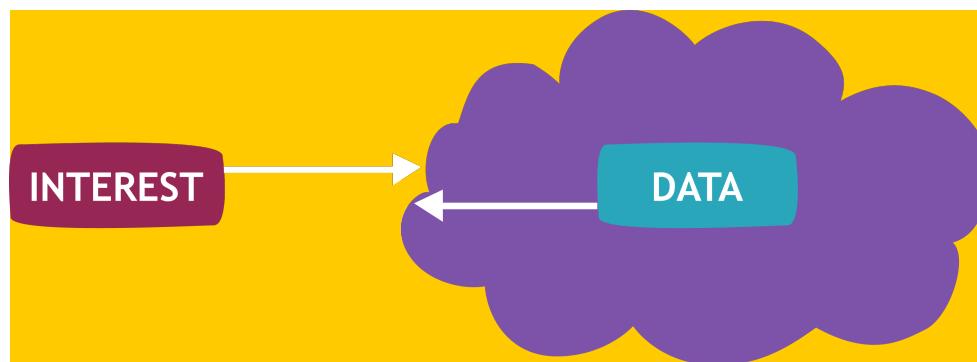
- What are the problems with the current Internet?
- Are they worth re-designing the network?
- With the current architecture being so entrenched, can we even deploy a new one?

# Evolution of Communication Abstraction



**Telephone Network:**  
Focused on building the  
*wires*

**Internet Protocol (RFC791):**  
Focused on delivering packets to  
destination *node*



**NDN: Focusing on retrieving *data***

Abstracting away the notion of “node”

*Superset of node-to-node communication model*

# A New Way to Think About Security

- ***Secure the Content, Not the Channel!***
  - SSL, VPN, ssh tunnel, ToR, etc all focus on providing a secure channel
  - Users don't really care if the channel is secure, focus on the content
- Require Authentication on All Content
  - Security is not an option, its part of architecture
- Encrypt the content if you don't trust the channel
  - Encryption is optional and applied where needed