Lecture 5:
Active & Overlay Networks

CSE 222A: Computer Communication Networks

Alex C. Snoeren

UCSDCSE

Thanks: Amin Vahdat

Lecture 5 Overview

- Wetherall ‘99 discussion
- Introduction to Software Defined Networking (SDN)

Active Networking Benefits

- Benefits from:
  - Fusion, Fission, Caching, Delegation
- Active Reliable Multicast
  - Problem: NACK implosion
  - Duplicate NACK suppression, cache multicast data, upstream NACK generation
- Online Auctions
  - Process failed bids closer to users, reducing server load
- Mixing Sensor Data
  - Combine multiple input signals to reduce transmitted messages and end host computing burden

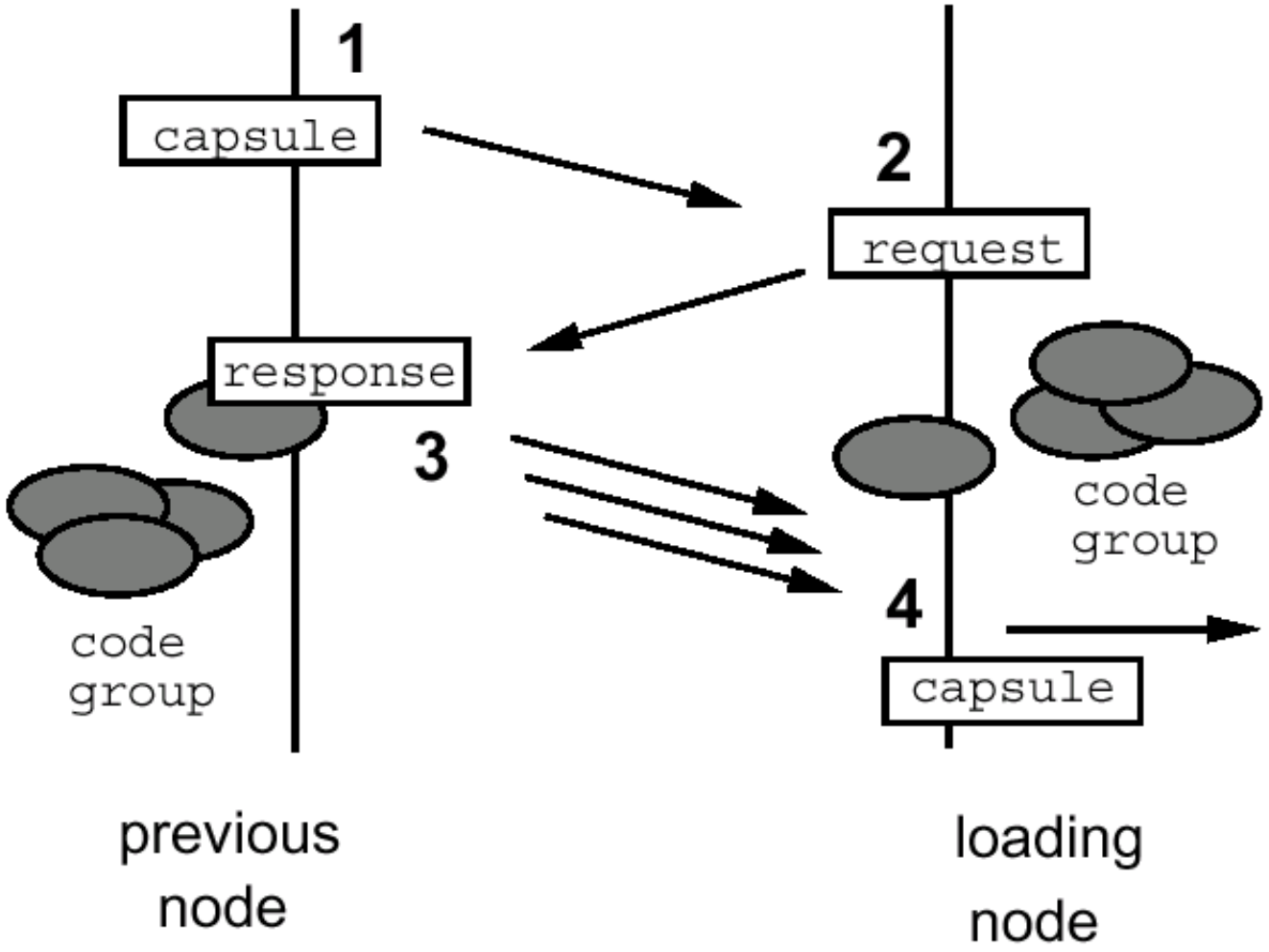
Caching Fast Changing Data

- Service that provides rapidly changing information
  - Military information system, airline flight status, stock quotes
- Web Caching?
  - Yesterday's proxy caches could not cache dynamically generated data – Akamai to the rescue
  - Wrong granularity: pages as opposed to objects
- Active Networks can be customized to provide:
  - Application-specific cache coherence
  - Application-specific object granularity

AN Caching Protocol

- Quotes cached at Active Nodes on client-to-server path
- Subsequent requests intercepted to consult local cache
- Caches automatically lie on the path between client/server
  - Do not redirect requests to caches in wrong direction
- Application-specific cache coherence
  - Different clients have different requirements for “freshness”
- (Potential) Benefits:
  - Decrease client latency
  - Decrease the traffic at routers
  - Decrease server load

Code Propagation

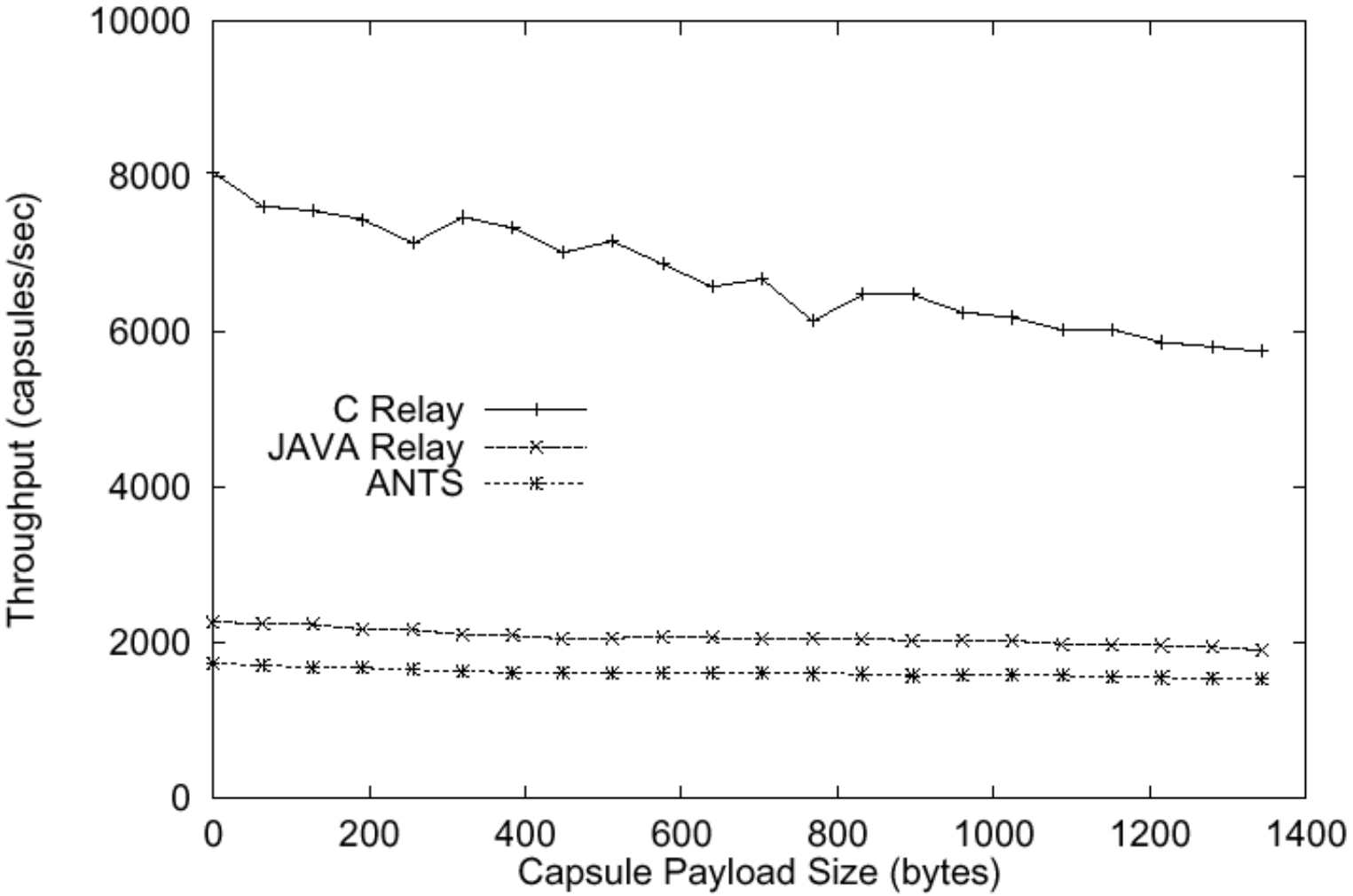


Performance Discussion

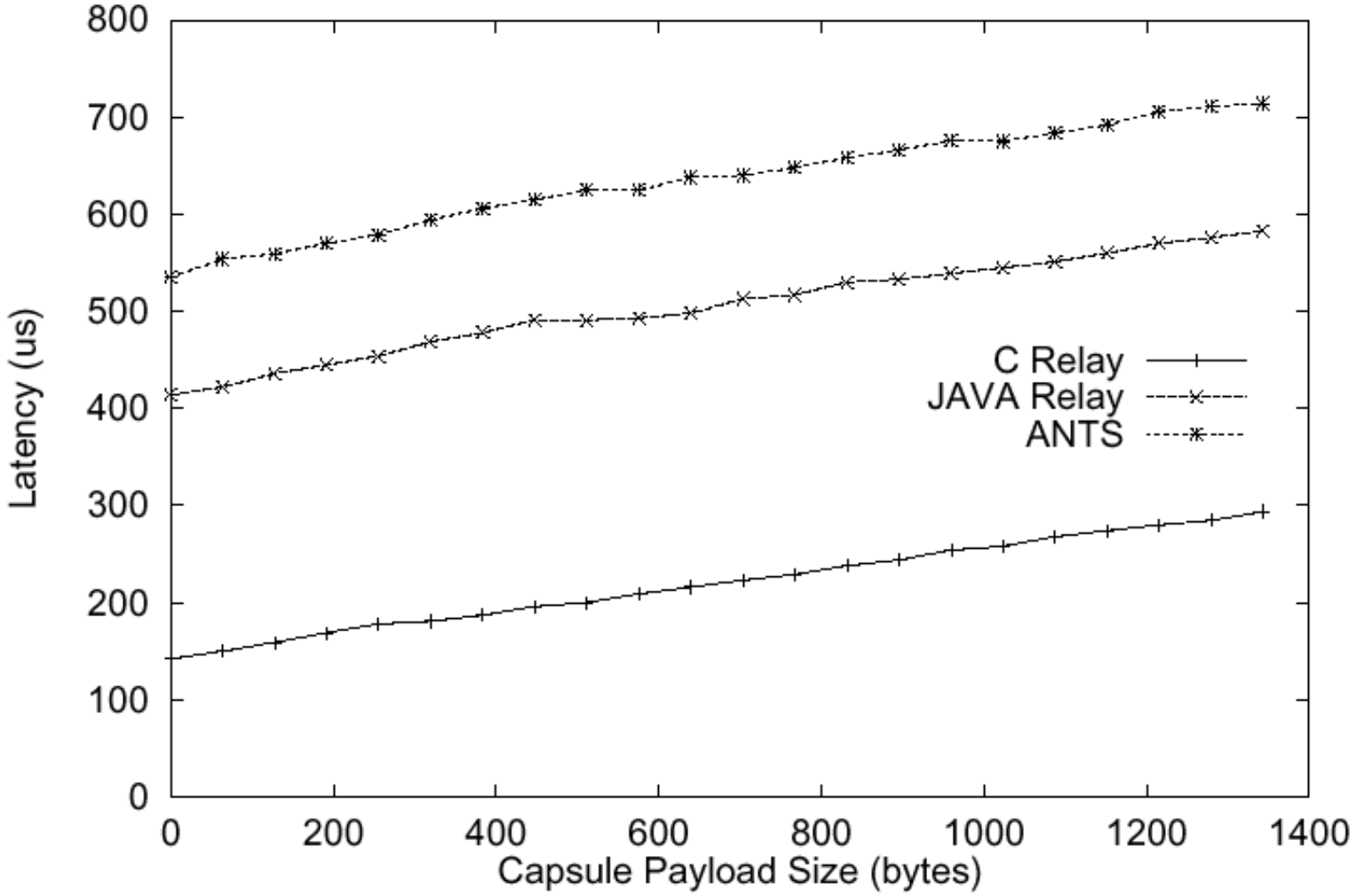
- 1999 PC-based routers able to forward 200,000 pkt/sec
  - Reaching 1 Gbps at typical packet sizes
- Modern commodity routers forward 16,000,000 pkt/sec/port
  - At least OC-192 rates (10 Gbps), maybe higher
  - Factor of 10 better than PC's per port

- Active Networks (1999 numbers) limited by Java, user-level implementation, safety, general-purpose routines
  - 1700 capsules/sec, 16 Mbps on Sun Ultra 1
  - Latencies range from 500-700 us
  - Related effort (PAN) forwards 100 Mbps w/kernel support

Active Node Throughput



Active Node Latency



Overheads

Operation	IP?	Time (μs)	(%)
1. Packet Receive	no	180	29
2. Header Processing	yes	30	5
3. Type Demultiplex	no	20	3
4. Capsule Decode	no	110	18
5. Capsule Evaluate	no	10	2
6. Route Lookup	yes	30	5
7. Capsule Encode	no	90	14
8. Header Processing	yes	40	7
9. Packet Transmit	no	80	13
Other	n/a	25	4
Total	n/a	615	100

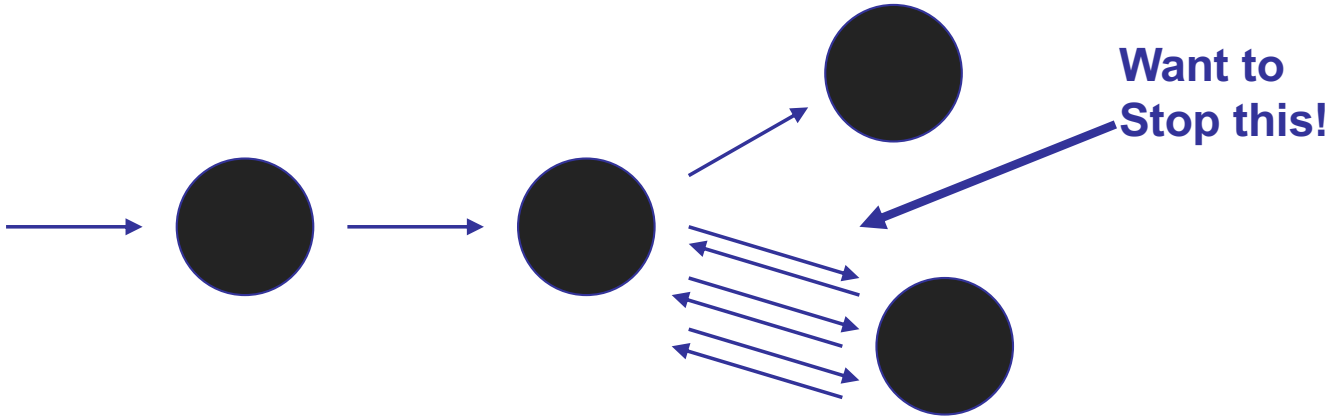
Rethinking Performance

- Traditional networking metrics:
  - Bandwidth, latency on a packet level
- What really matters is end-to-end performance
  - Application throughput
  - Client-perceived latency
- Active Networks may slow routing down
  - But improve end-to-end application performance
  - Use application-specific notions of throughput/latency

Resource Allocation Issues

- Difficulties with allocating resources in active nets:
  - Single capsule consumes too much resources at active node
  - Capsule and other capsules it creates consume unbounded resources across wide area
  - End application introduces large number of capsules
- How to address these problems?

DJW's Buggy Multicast

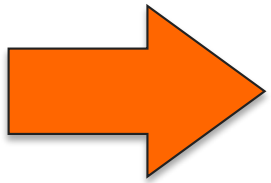


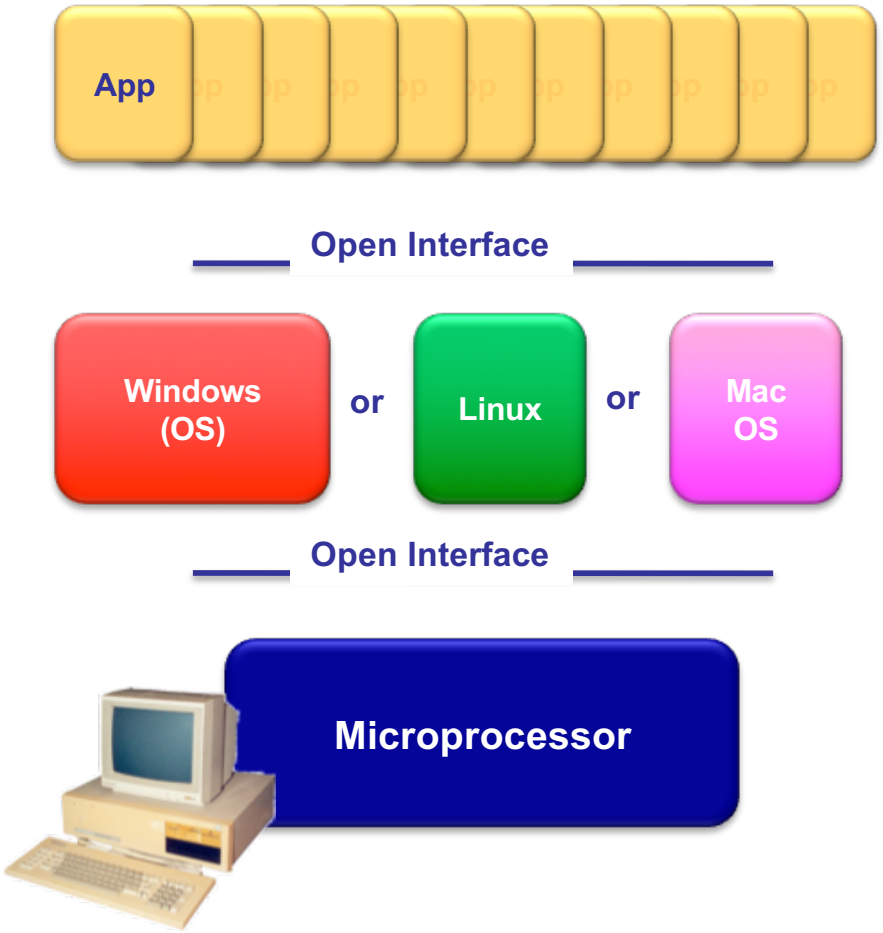
- How should we prevent this?
  - TTLs are a weak solution; not related to topology
  - Fairness mechanisms mitigate, but not enough
  - ANTS falls back on certification of programs …

Active Networks Discussion

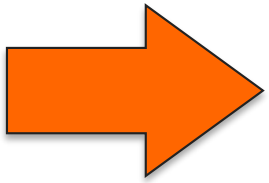
- Introduce programmability for
  - Rapid introduction of new protocols
  - Increased end-to-end performance
- Rethink network performance in terms of app performance
- Issues:
  - Speed, Resource allocation, Safety/Security
- Active Networks can make explicit “transparent” network caching, network address translation, etc.

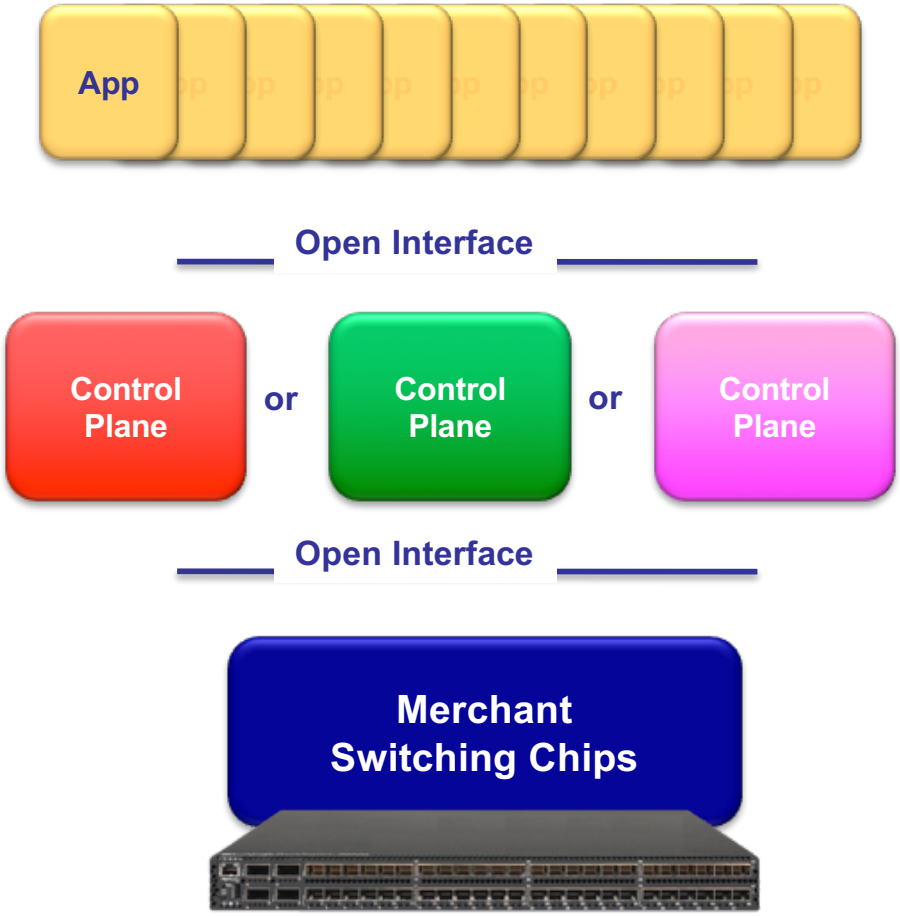
Evolution of Computing



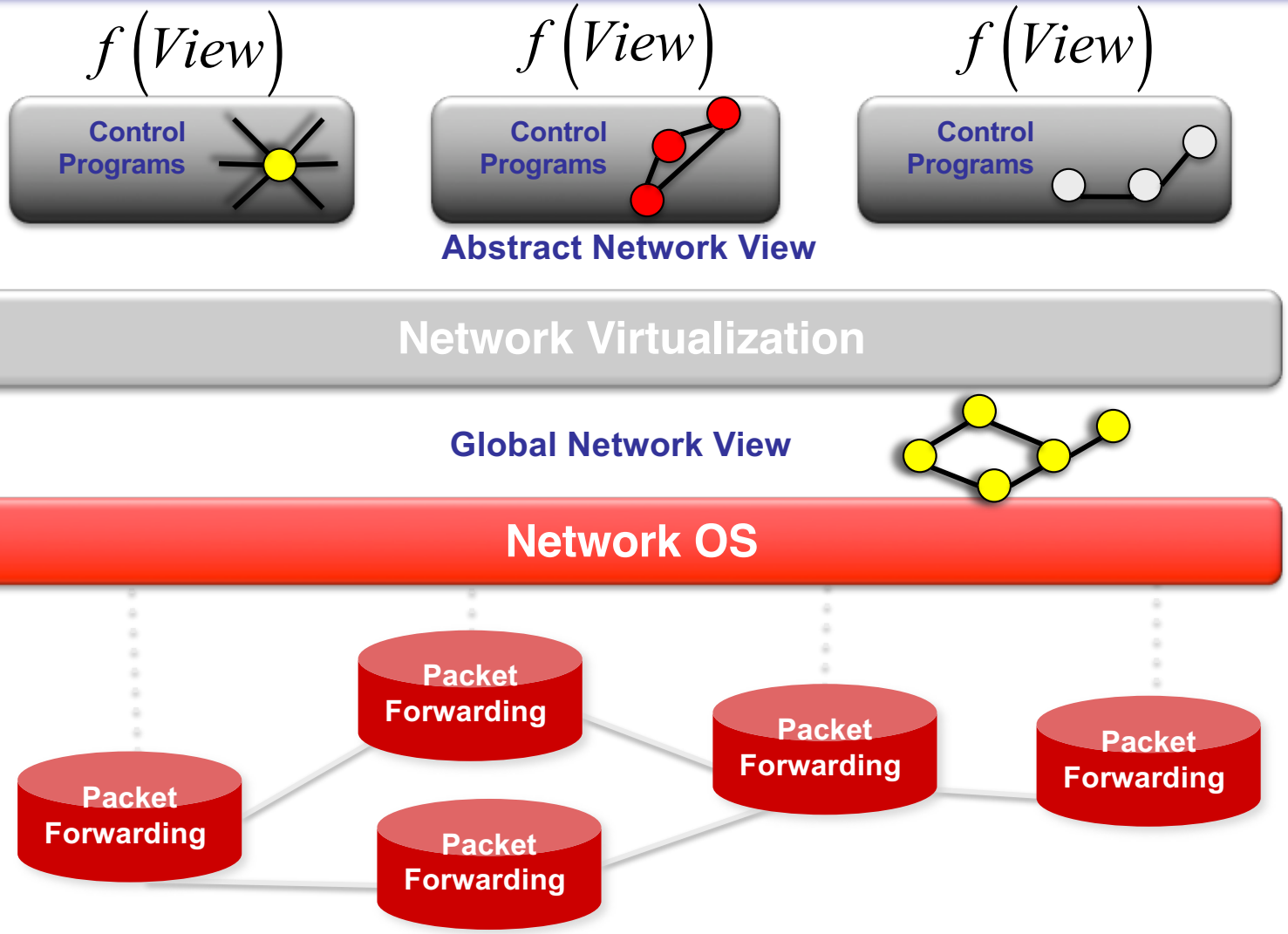


Evolution of Networking?

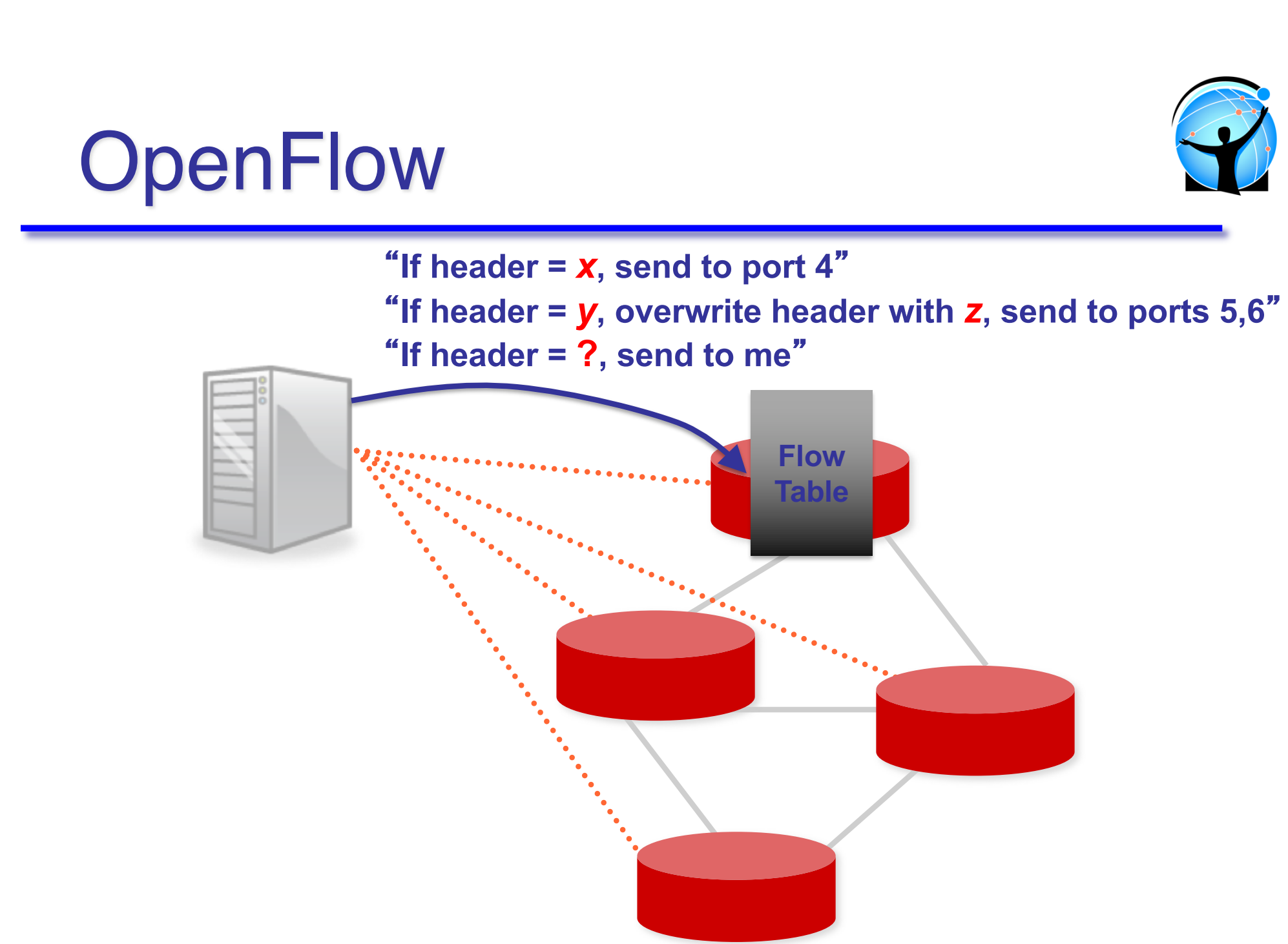




Software Defined Networking



OpenFlow


"If header = x, send to port 4"
"If header = y, overwrite header with z, send to ports 5,6"
"If header = ?, send to me"
Flow Table

Basic OF Operations

- Match arbitrary bits in headers:

Match: 1000x01xx0101001x

  - Match on any header, or new header
  - Allows any flow granularity
- Action
  - Forward to port(s), drop, send to controller
  - Overwrite header with mask, push or pop
  - Forward at specific bit-rate

For Next Class…

- Read and review ONIX
- Submit project ideas by tonight
  - Email to me
  - I'll post ideas/names to help you find groups by tomorrow