Measurement Techniques: Part 2B

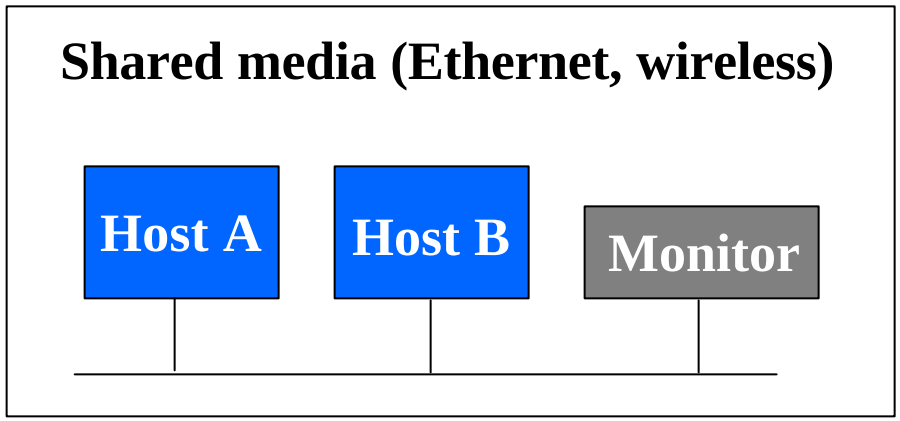
- Packet monitoring
- Flow measurement
- Data interpretation

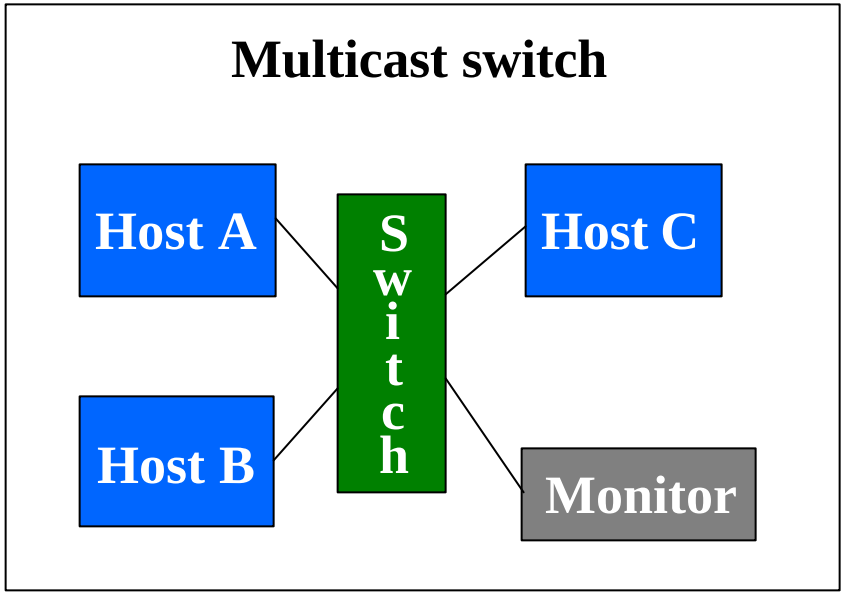
Packet Monitoring

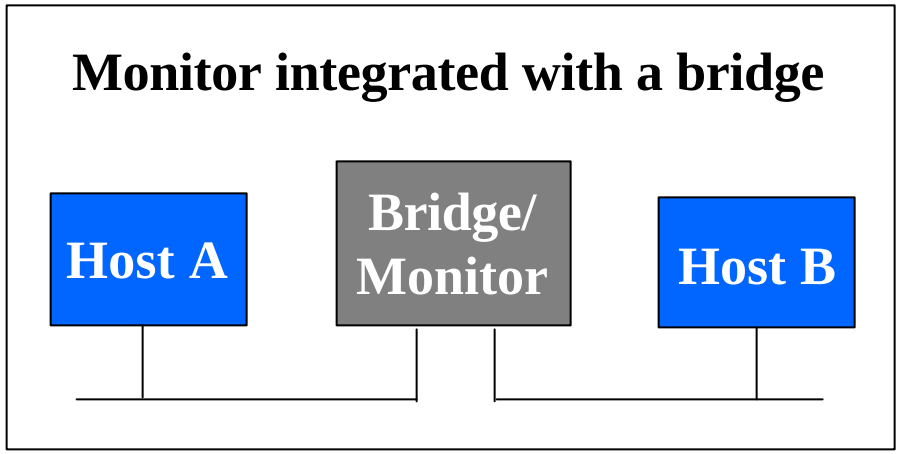
Packet Monitoring: Outline

- Definition
  - Passively collecting IP packets on one or more links
  - Recording IP, TCP/UDP, or application-layer traces
- Scope
  - Fine-grain information about user behavior (e.g., URLs)
  - Passively monitoring parts of the network infrastructure
  - Helpful in characterizing traffic and diagnosing problems
- Outline
  - Tapping a link and capturing packets
  - Information available in packet traces
  - Mechanics of collecting the packet traces
  - Practical challenges in collecting and analyzing the data
  - Existing packet monitoring platforms

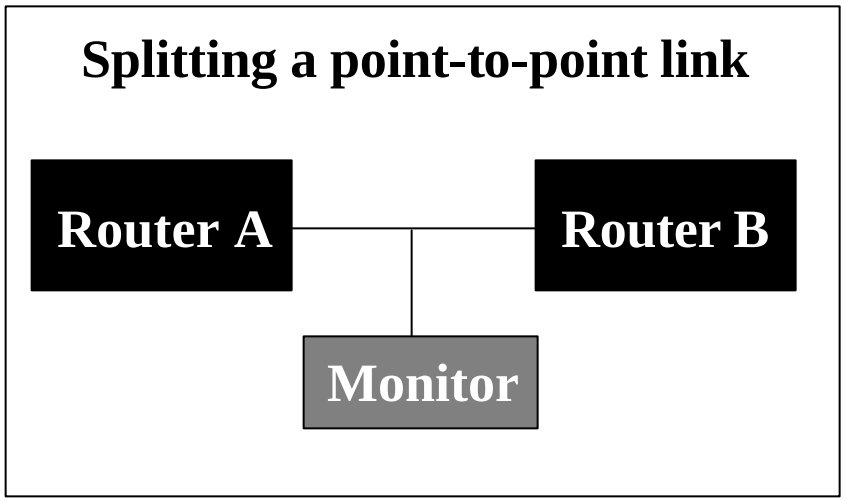
Monitoring a LAN Link

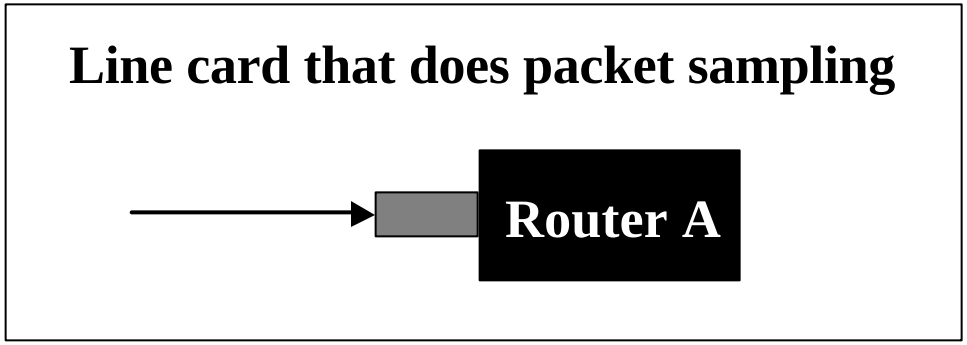






Monitoring a WAN Link

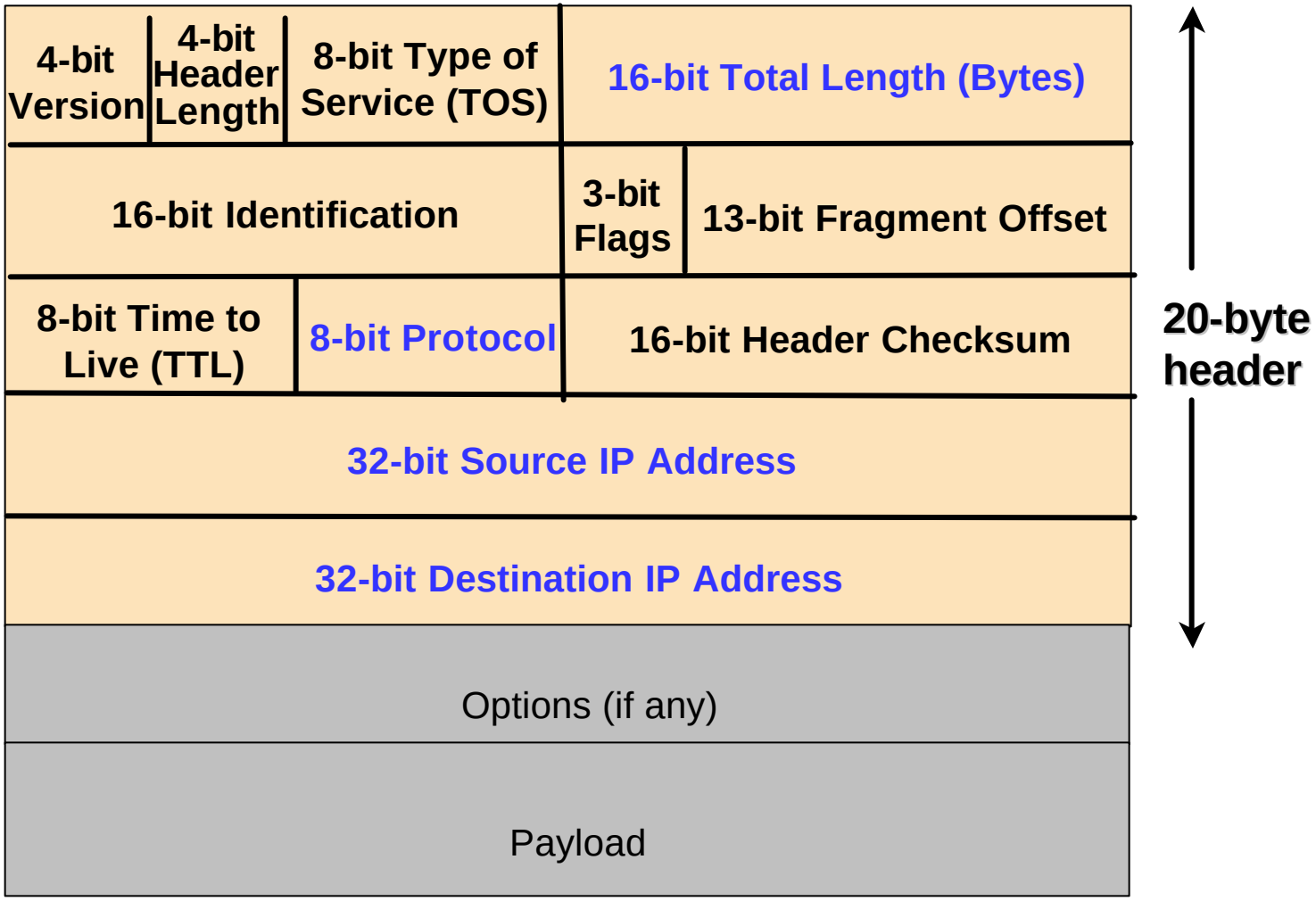




Selecting the Traffic

- Filter to focus on a subset of the packets
  - IP addresses/prefixes (e.g., to/from specific Web sites, client machines, DNS servers, mail servers)
  - Protocol (e.g., TCP, UDP, or ICMP)
  - Port numbers (e.g., HTTP, DNS, BGP, Napster)
- Collect first *n* bytes of packet (snap length)
  - Medium access control header (if present)
  - IP header (typically 20 bytes)
  - IP+UDP header (typically 28 bytes)
  - IP+TCP header (typically 40 bytes)
  - Application-layer message (entire packet)

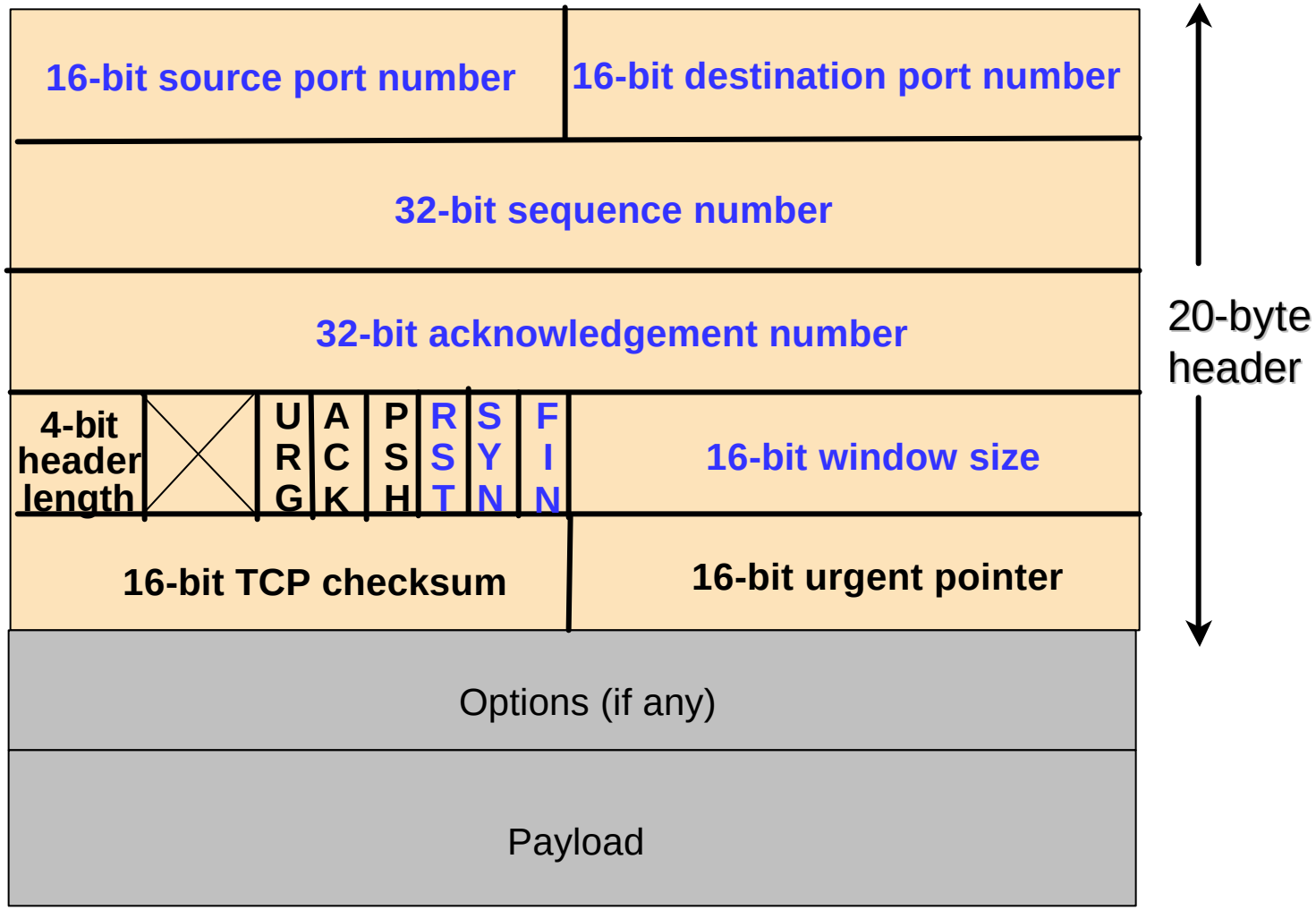
IP Header Format



Analysis of IP Header Traces

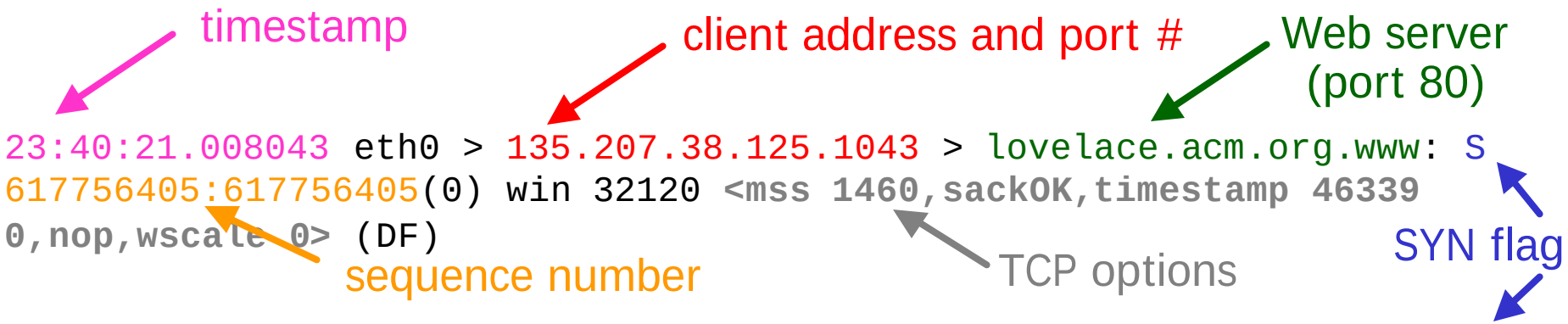
- Source/destination addresses for traffic
  - Identity of popular Web servers & heavy customers
- Traffic breakdown by protocol (TCP/UDP/ICMP)
  - Amount of traffic not using congestion control
- Distribution of packet delay through the router
  - Identification of typical delays and anomalies
- Distribution of packet sizes
  - Workload models for routers and measurement devices
- Burstiness of the traffic on the link over time
  - Provisioning rules for allocating link capacity
- Throughput between each pair of src/dest addresses
  - Detection and diagnosis of performance problems

TCP Header Format



Tcpdump Output

(three-way TCP handshake and HTTP request message)




```
23:40:21.008043 eth0 > 135.207.38.125.1043 > lovelace.acm.org.www: S
617756405:617756405(0) win 32120 <mss 1460,sackOK,timestamp 46339
0,nop,wscale 0> (DF)

23:40:21.036758 eth0 < lovelace.acm.org.www > 135.207.38.125.1043: S
2598794605:2598794605(0) ack 617756406 win 16384 <mss 512>

23:40:21.036789 eth0 > 135.207.38.125.1043 > lovelace.acm.org.www: .
1:1(0) ack 1 win 32120 (DF)

23:40:21.037372 eth0 > 135.207.38.125.1043 > lovelace.acm.org.www: P
1:513(512) ack 1 win 32256 (DF)

23:40:21.085106 eth0 < lovelace.acm.org.www > 135.207.38.125.1043: .
1:1(0) ack 513 win 16384

23:40:21.085140 eth0 > 135.207.38.125.1043 > lovelace.acm.org.www: P
513:676(163) ack 1 win 32256 (DF)

23:40:21.124835 eth0 < lovelace.acm.org.www > 135.207.38.125.1043: P
1:179(178) ack 676 win 16384
```

TCP Header Analysis

- Source and destination port numbers
  - Popular applications (HTTP, Napster, SMTP, DNS)
  - Number of parallel connections between source-dest pairs
- Sequence/ACK numbers and packet timestamps
  - Out-of-order/lost packets; violations of congestion control
  - Estimates of throughput and delay of Web downloads
- Number of packets/bytes per connection
  - Size of typical Web transfers; frequency of bulk transfers
- SYN flags from client machines
  - Unsuccessful connection requests; denial-of-service attacks
- FIN/RST flags from client machines
  - Frequency of Web transfers aborted by clients

Packet Contents

- Application-layer header
  - HTTP and RTSP request and response headers
  - FTP, NNTP, and SMTP commands and replies
  - DNS queries and responses; OSPF/BGP messages
- Application-layer body
  - HTTP resources (or checksums of the contents)
  - User keystrokes in Telnet/Rlogin sessions
- Security/privacy
  - Significant risk of violating user privacy
  - More sensitive for information from higher-level protocols
  - Traffic analysis thwarted by use of end-to-end encryption

HTTP Request and Response Message

```
GET /tutorial.html HTTP/1.1
Date: Mon, 27 Aug 2001 08:09:01 GMT
From: jrex@research.att.com
User-Agent: Mozilla/4.03
CRLF
```

Request

Response

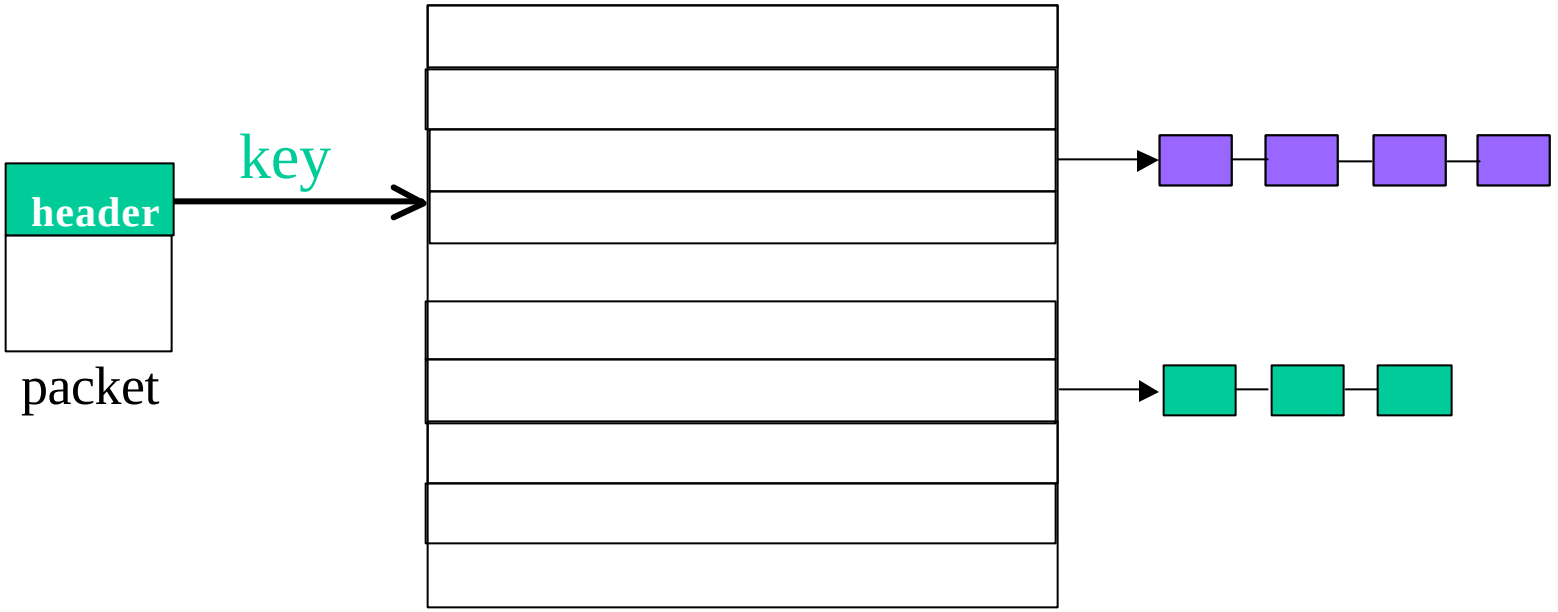
```
HTTP/1.1 200 OK
Date: Thu, 12 Jul 2001 10:09:03 GMT
Server: Netscape-Enterprise/3.5.1
Last-Modified: Sun, 12 Mar  2000 11:12:23 GMT
Content-Length: 23
CRLF
Traffic measurement talk
```

Application-Layer Analysis

- URLs from HTTP request messages
  - Popular resources/sites; potential benefits of caching
- Meta-data in HTTP request/response messages
  - Content type, cacheability, change frequency, etc.
  - Browsers, protocol versions, protocol features, etc.
- Contents of DNS messages
  - Common queries, frequency of errors, query latency
- Contents of Telnet/Rlogin sessions
  - Intrusion detection (break-ins, stepping stones)
- Routing protocol messages
  - Workload for routers; detection of routing anomalies
  - Tracking the current topology/routes in the backbone

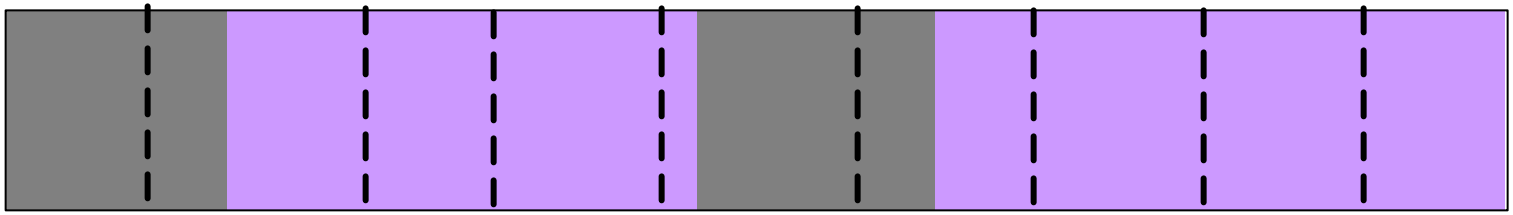
Mechanics: Application-Level Messages

- Application-level transfer may span multiple packets
  - Demultiplex packets into separate “flows”
  - Key of source/dest IP addresses, port #s, and protocol
  - Hash table to store packets from different flows



Mechanics: Application-Level Messages

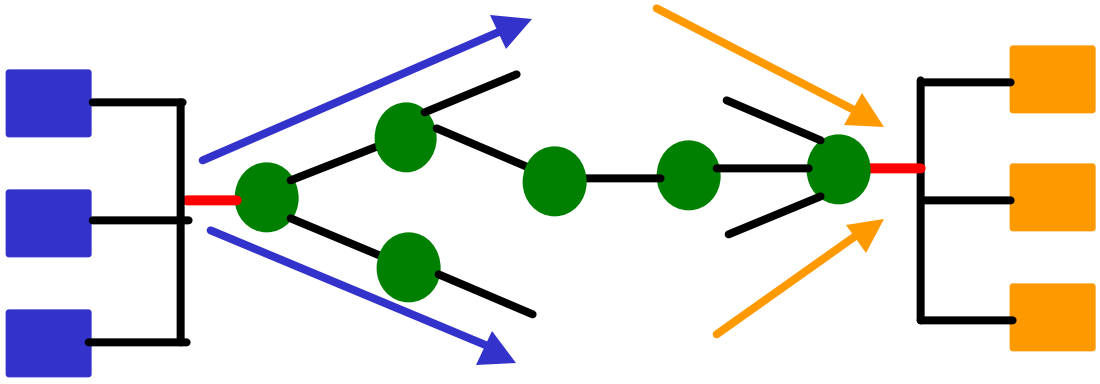
- Reconstructing ordered, reliable byte stream
  - Sequence number and segment length in TCP header
  - Heap to store packets in correct order & discard duplicates
- Extraction of application-level messages
  - Parsing the syntax of the application-level header
  - Identifying the the start of the next message (if any)



- Logging or online analysis of message
  - Record URL, header, body, checksum, timestamps, etc.
  - Copy traces or analysis result to separate machine

Placement of the Monitor (Edge)

- Client edge
  - Capture all traffic to/from a single group of clients
  - Useful for evaluating effectiveness of a proxy
  - May not be representative of other clients
- Server edge
  - Capture all traffic to/from a set of Web sites
  - Useful for detailed characterization of access patterns
  - May not be representative of accesses to other sites

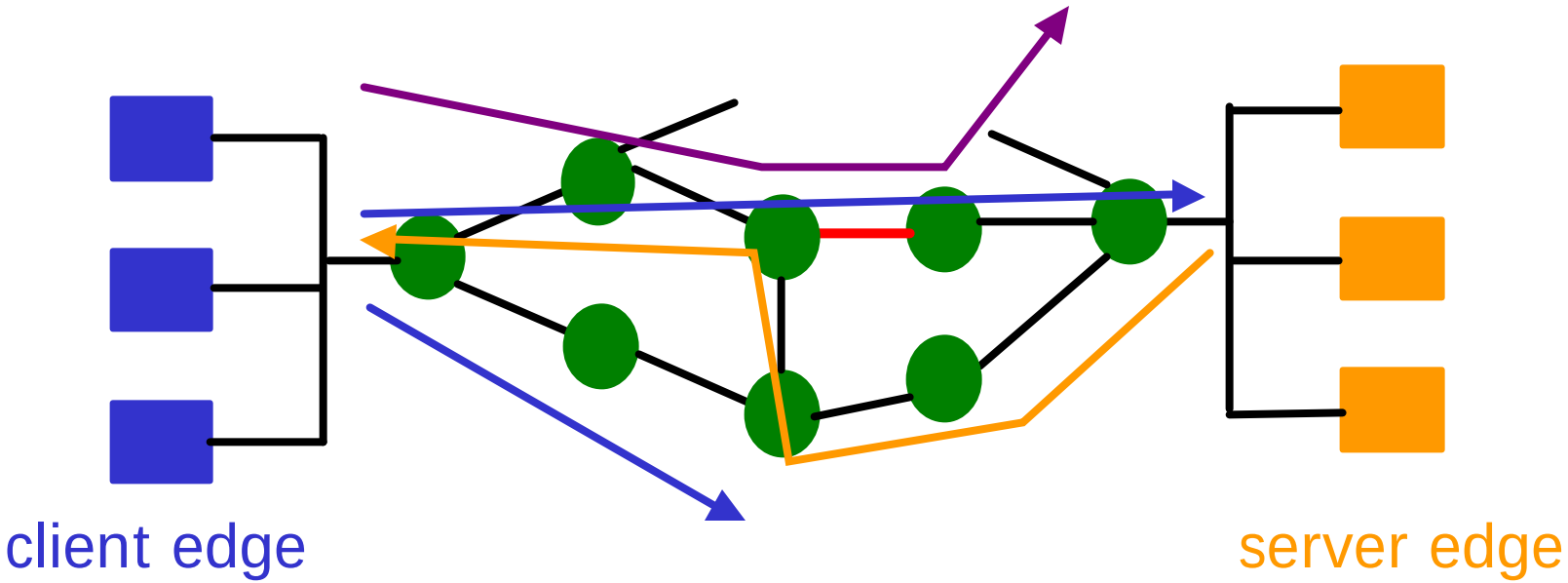


client edge

server edge

Placement of the Monitor (Core)

- Middle of network
  - Capture all traffic traversing a particular link
  - Useful for capturing a diverse mix of traffic
  - May not see all traffic traveling from host A to host B
  - May not see the reverse traffic from host B to host A

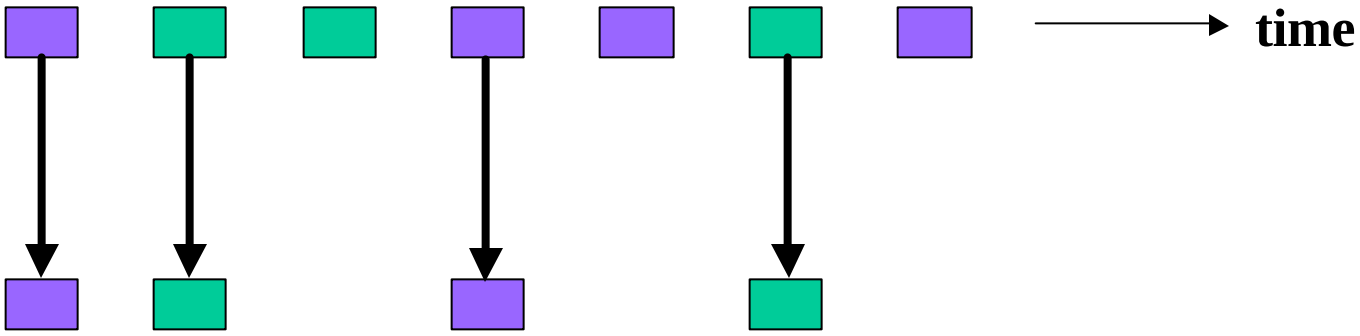


System Constraints

- High data rate
  - Bandwidth limits on CPU, I/O, memory, and disk/tape
  - Could monitor lower-speed links (near the edge of network)
- High data volume
  - Space limitations in main memory and on disk/tape
  - Could do online analysis to sample, filter, & aggregate
- High processing load
  - CPU/memory limits for extracting, counting, & analyzing
  - Could do offline processing for time-consuming analysis
- General solutions to system constraints
  - Sub-select the traffic (addresses/ports, first *n* bytes)
  - Kernel and interface card support for measurement
  - Efficient/robust software and hardware for the monitor

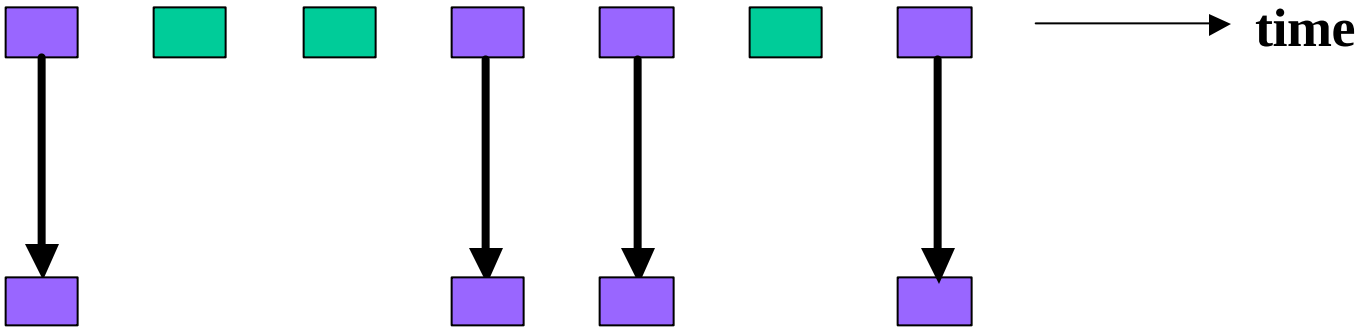
Packet Sampling

- Random sampling of individual packets
  - Per-packet: select 1-out-of-m packets
  - Per-byte: sample in proportion to packet length
- Appropriate for certain types of analysis
  - Traffic mix by port #, src/dest IP address, protocol, etc.
  - Packet size distribution, frequency of IP/TCP header values
  - Not appropriate for gleaning application-level information



Application-Layer Sampling

- Sample application-layer messages
  - Select 1-out-of-m flows (all packets or no packets)
  - Cannot randomly sample at the packet level
- Sample based on hash of the packet header
  - TCP/UDP session: addresses and port numbers
  - End-host pair: source and destination address



Packet Monitoring Platforms: Research

- Tcpdump software
  - Publically-available, easy to use, widely used
  - UNIX PC with Ethernet/FDDI interface and running tcpdump
- OCxMon (CoralReef) from CAIDA
  - Monitoring OC3/OC12 links with ATM and SONET interfaces
  - First ATM cell in a packet, all ATM headers, or all cells
- AT&T Labs PacketScope
  - Tcpdump for capture/filtering; fine-grain timestamps
  - Software for appl-level measurement (HTTP, multimedia)
- SprintLabs monitor
  - Special-purpose interface (to capture 44 bytes per packet)
  - Fine-grain timestamps with GPS and clock synchronization

Packet Monitors: Commercial

- Characteristics of commercial monitors
  - Interfaces for tapping a variety of types of links
  - Collection at IP, TCP/UDP, and application layer
  - Exporting of data via SNMP/RMON and using Netflow format
  - Software for generating standard and customized reports
- Example products
  - Narus Semantic Traffic Analyzers
    - http://www.narus.com/solutions/analyzers.html
  - Niksun NetVCR
    - http://www.niksun.com/products/netvcr.html
  - Sniffer.com
    - http://www.sniffer.com/products/default.asp
  - Agilent NetMetrix
    - http://www.agilent.com/cm/product/netmetrix/nmx_products_datcol.shtml

Conclusions

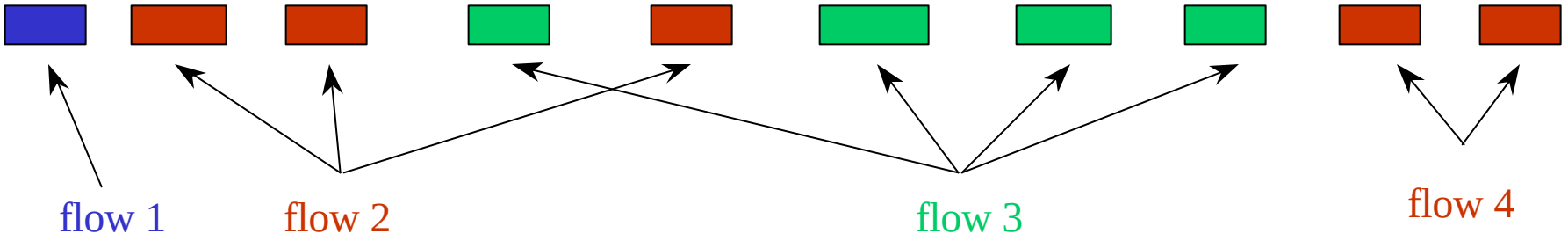
- Packet monitoring
  - Detailed, fine-grain view of individual links
- Advantages
  - Finest level of granularity (individual IP packets)
  - Primary source of application-level information
- Disadvantages
  - Expensive to build and deploy
  - Large volume of measurement data
  - Difficult to deploy widely over a large network
  - Challenges collecting on high-speed links
  - Challenges of reconstructing application-level info

Flow Measurement

Flow Measurement: Outline

- Definition
  - Passively collecting statistics about groups of packets
  - Group packets based on headers and spacing in time
  - Essentially a way to aggregate packet measurement data
- Scope
  - Medium-grain information about user behavior
  - Passively monitoring the link or the interface/router
  - Helpful in characterizing, detecting, diagnosing, and fixing
- Outline
  - Definition of an IP “flow” (sequence of related packets)
  - Flow measurement data and its applications
  - Mechanics of collecting flow-level measurements
  - Reducing the overheads of flow-level measurement

IP Flows



- Set of packets that “belong together”
  - Source/destination IP addresses and port numbers
  - Same protocol, ToS bits, ...
  - Same input/output interfaces at a router (if known)
- Packets that are “close” together in time
  - Maximum spacing between packets (e.g., 15 sec, 30 sec)
  - Example: flows 2 and 4 are different flows due to time

Flow Abstraction

- A flow is not exactly the same as a “session”
  - Sequence of related packets may be multiple flows (due to the “close together in time” requirement)
  - Sequence of related packets may not follow the same links (due to changes in IP routing)
  - A “session” is difficult to measure from inside the network
- Motivation for this abstraction
  - As close to a “session” as possible from inside the network
  - Flow switching paradigm from IP-over-ATM technology
  - Router optimization for forwarding/access-control decisions (cache the result after the first packet in a flow)
  - ... might as well throw in a few counters

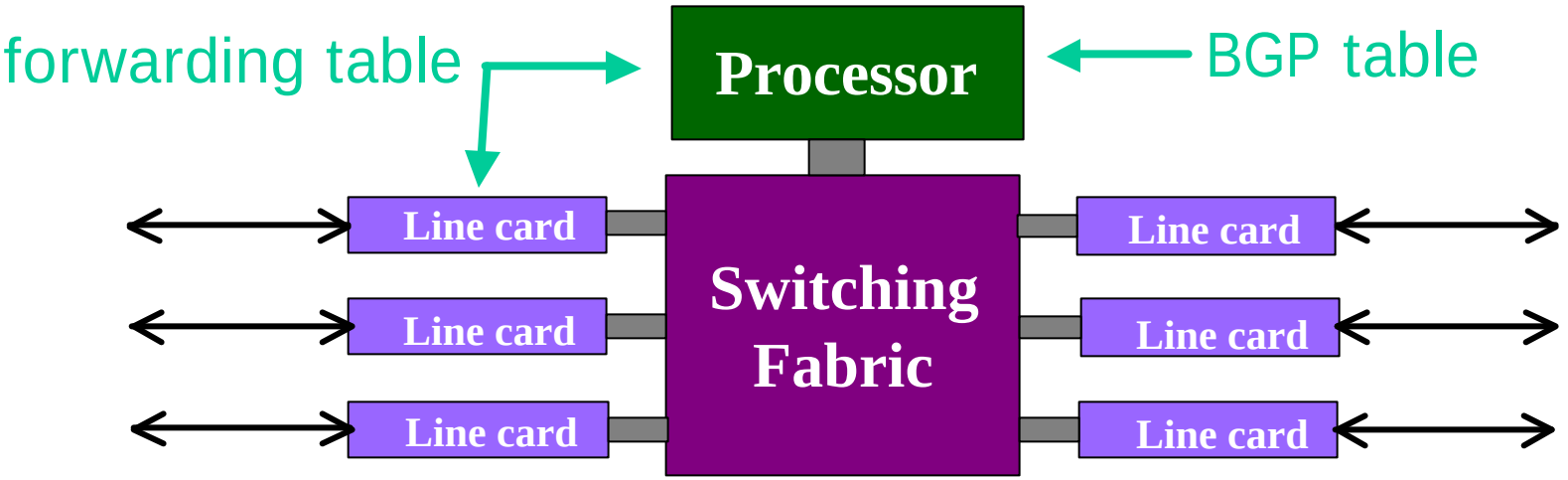
Recording Traffic Statistics (e.g., Netflow)

- Packet header information (same for every packet)
  - Source and destination IP addresses
  - Source and destination TCP/UDP port numbers
  - Other IP & TCP/UDP header fields (protocol, ToS bits, etc.)
- Aggregate traffic information (summary of the traffic)
  - Start and finish time of the flow (time of first & last packet)
  - Total number of bytes and number of packets in the flow
  - TCP flags (e.g., logical OR over the sequence of packets)

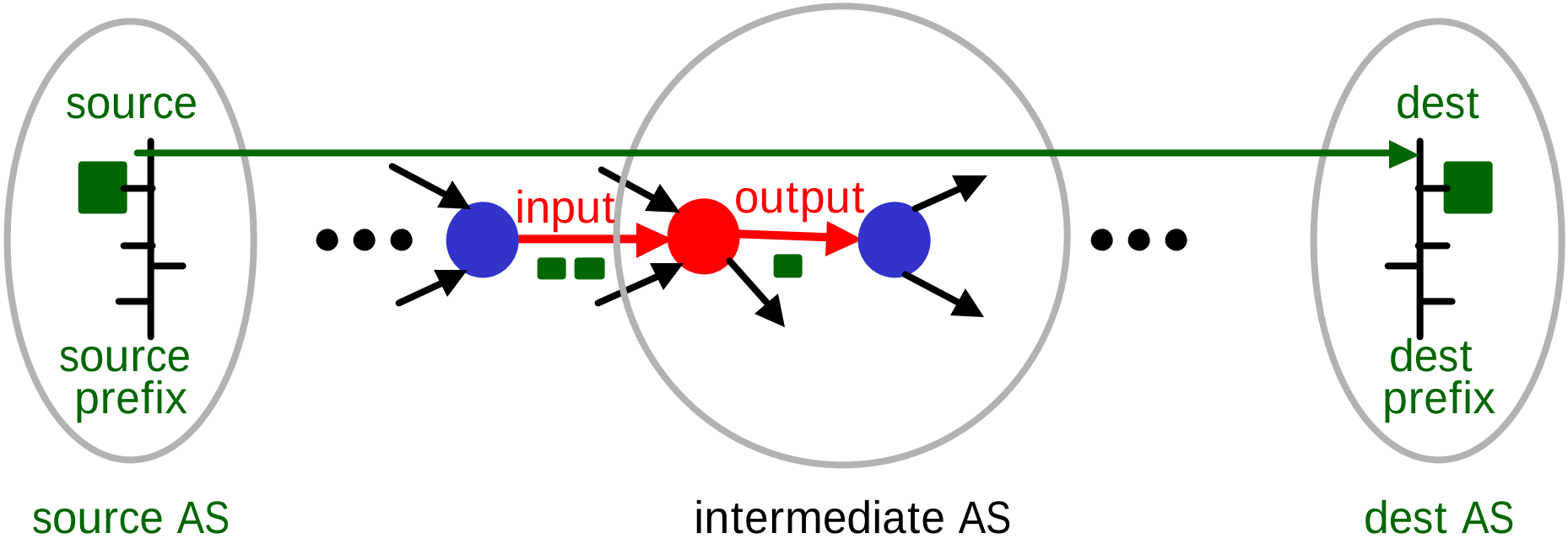
4 packets
1436 bytes
SYN, ACK, & FIN

Recording Routing Info (e.g., Netflow)

- Input and output interfaces
  - Input interface is where the packets entered the router
  - Output interface is the “next hop” in the forwarding table
- Source and destination IP prefix (mask length)
  - Longest prefix match on the src and dest IP addresses
- Source and destination autonomous system numbers
  - Origin AS for src/dest prefix in the BGP routing table



Measuring Traffic as it Flows By



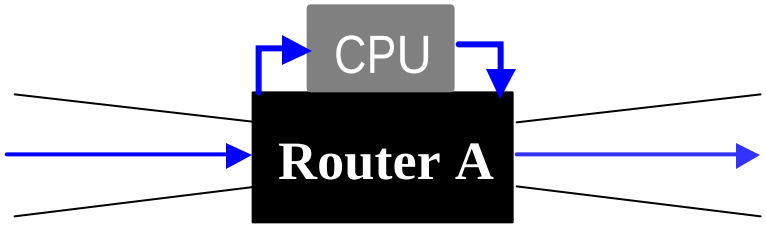
Source and destination: IP header
Source and dest prefix: forwarding table or BGP table
Source and destination AS: BGP table

Packet vs. Flow Measurement

- Basic statistics (available from both techniques)
  - Traffic mix by IP addresses, port numbers, and protocol
  - Average packet size
- Traffic over time
  - Both: traffic volumes on a medium-to-large time scale
  - Packet: burstiness of the traffic on a small time scale
- Statistics per TCP connection
  - Both: number of packets & bytes transferred over the link
  - Packet: frequency of lost or out-of-order packets, and the number of application-level bytes delivered
- Per-packet info (available only from packet traces)
  - TCP seq/ack #s, receiver window, per-packet flags, ...
  - Probability distribution of packet sizes
  - Application-level header and body (full packet contents)

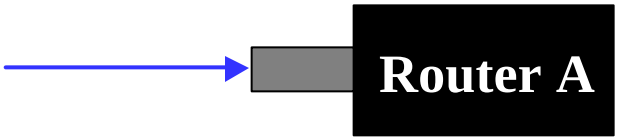
Collecting Flow Measurements

Route CPU that generates flow records



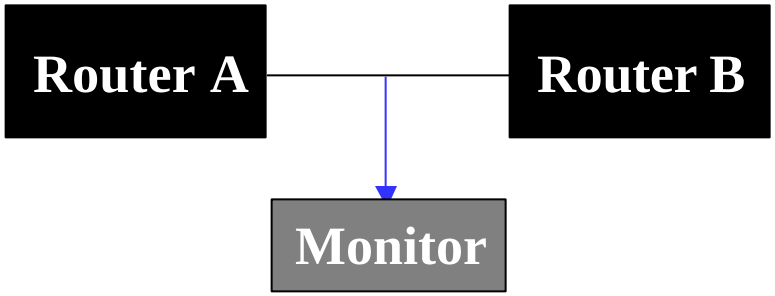
…may degrade forwarding performance

Line card that generates flow records



…more efficient to support measurement in each line card

Packet monitor that generates flow records



…third party

Router Collecting Flow Measurement

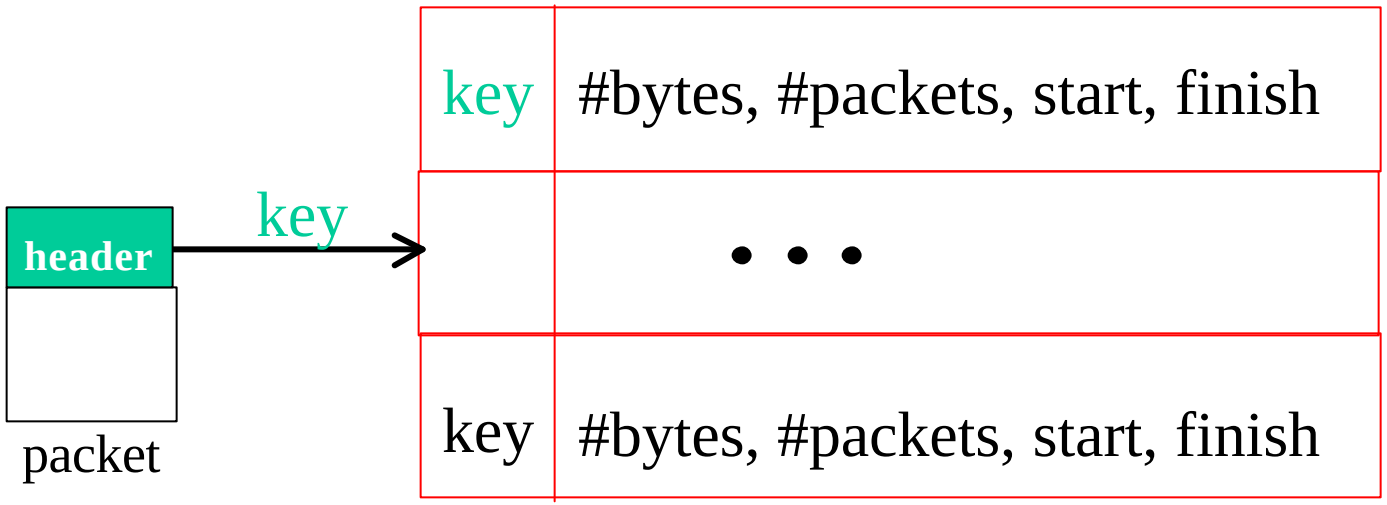
- Advantage
  - No need for separate measurement device(s)
  - Monitor traffic over all links in/out of router (parallelism)
  - Ease of providing routing information for each flow
- Disadvantage
  - Requirement for support in the router product(s)
  - Danger of competing with other 1st-order router features
  - Possible degradation of the throughput of the router
  - Difficulty of online analysis/aggregation of data on router
- Practical application
  - View from multiple vantage points (e.g., all edge links)

Packet Monitor Collecting Flow Records

- Advantages
  - No performance impact on packet forwarding
  - No dependence on support by router vendor
  - Possibility of customizing the thinning of the data
- Disadvantages
  - Overhead/cost of tapping a link & reconstructing packets
  - Cost of buying, deploying, and managing extra equipment
  - No access to routing info (input/output link, IP prefix, etc.)
- Practical application
  - Selective monitoring of a small number of links
  - Deployment in front of particular services or sites
- Packet monitor vendors support flow-level output

Mechanics: Flow Cache

- Maintain a cache of active flows
  - Storage of byte/packet counts, timestamps, etc.
- Compute a key per incoming packet
  - Concatenation of source, destination, port #s, etc.
- Index into the flow cache based on the key
  - Creation or updating of an entry in the flow cache



Mechanics: Evicting Cache Entries

- Flow timeout
  - Remove flows that have not received a packet recently
  - Periodic sequencing through the cache to time out flows
  - New packet triggers the creation of a new flow
- Cache replacement
  - Remove flow(s) when the flow cache is full
  - Evict existing flow(s) upon creating a new cache entry
  - Apply eviction policy (LRU, random flow, etc.)
- Long-lived flows
  - Remove flow(s) that persist for a long time (e.g., 30 min)
  - ... otherwise flow statistics don't become available
  - ... and the byte and packet counters might overflow

Measurement Overhead: Per Packet/Flow

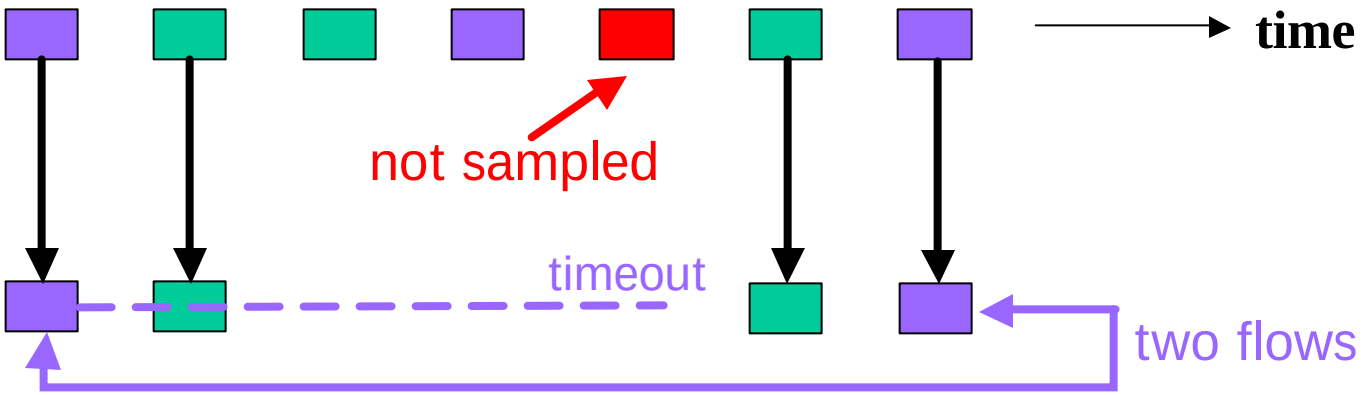
- Per-packet overhead
  - Computing the key for the packet
  - Indexing into the flow cache based on the key
  - More work when the average packet size is small
  - May not be able to keep up with the link speed
- Per-flow overhead
  - Creation and eviction of entry in the flow cache
  - Volume of measurement data (# of flow records)
  - Larger number of flows when #packets per flow is small
  - Extreme case of one-packet, 40-byte flows (SYN attack!)
  - May overwhelm the system collecting/analyzing the data

Measurement Overhead: Active Flows

- Number of active flows on the link
  - Depends on the link speed and per-flow bandwidth
  - Memory requirements of the flow cache
  - Overhead for indexing into the flow cache
  - Overhead for sequencing through the flow cache
- "Working set" may exceed the cache size
  - #of active flows may be larger than # cache entries
  - Repeated eviction and creation of new cache entries
  - Division of single flow into multiple flow records
  - Increase in volume of flow measurement records

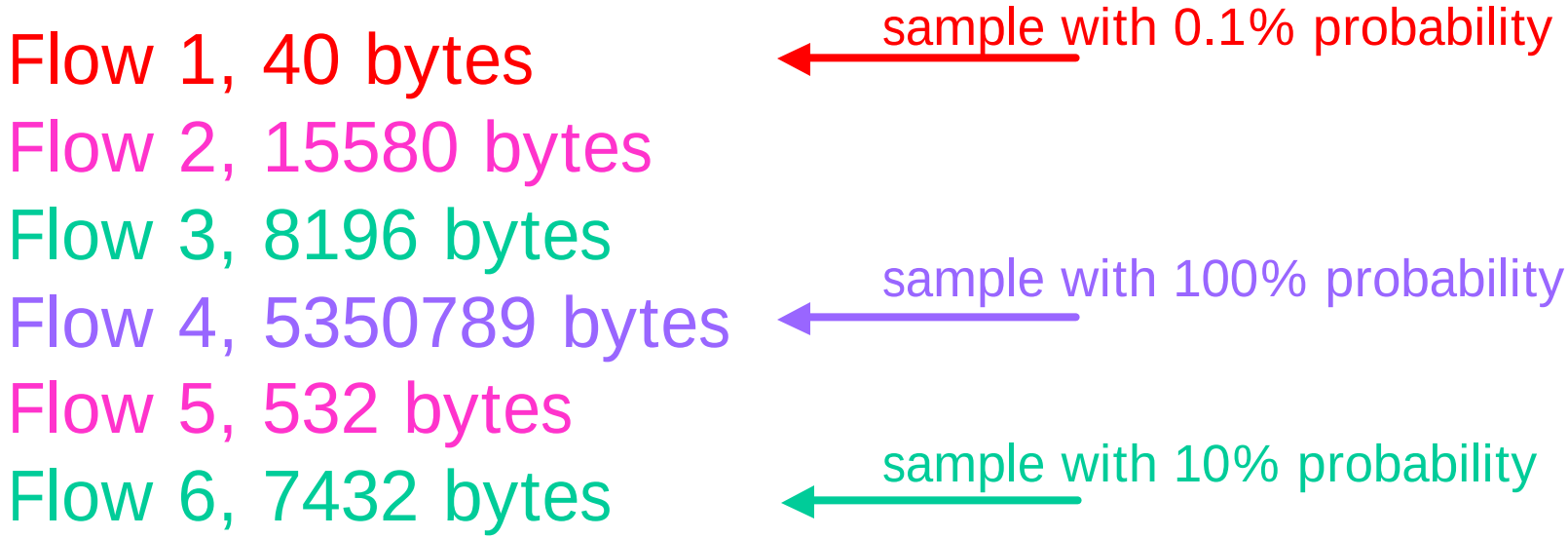
Sampling: Packet Sampling

- Packet sampling before flow creation (Sampled Netflow)
  - 1-out-of-m sampling of individual packets (e.g., m=100)
  - Create of flow records over the sampled packets
- Reducing overhead
  - Avoid per-packet overhead on (m-1)/m packets
  - Avoid creating records for a large number of small flows
- Increasing overhead (in some cases)
  - May split some long transfers into multiple flow records
  - ... due to larger time gaps between successive packets



Sampling: Flow-Level Sampling

- Sampling of flow records evicted from flow cache
  - When evicting flows from table or when analyzing flows
- Stratified sampling to put weight on “heavy” flows
  - Select all long flows and sample the short flows
- Reduces the number of flow records
  - Still measures the vast majority of the traffic



Aggregation

- Define flows at a coarser level
  - Ignore TCP/UDP port numbers, ToS bits, etc.
  - Source & destination IP prefix (not full addresses)
  - Source & destination Autonomous System number
- Advantages
  - Substantially reduce the size of the flow cache
  - Substantially reduce the number of flow records
  - Accurate information for a variety of applications
- Disadvantage
  - Lost information for basic traffic reporting
  - Impacted by the view of routing (prefixes) at this router
- Could aggregate *after* creating fine-grain flows
  - When evicting flows from the flow cache
  - When analyzing the flow records (offline)

IETF Standards Activity

- Real-Time Traffic Flow Meter (RTFM)
  - Past working group on describing and measuring flows
  - Meter with flow table and packet matching/handling
  - Meter readers that transport usage data from the meters
  - Manager for downloading rule sets to the meter
  - SNMP for downloading rules and reading usage data
- Internet Protocol Flow eXport (IPFX)
  - Proposed working group (BoF at August'01 IETF)
  - Distinguishing flows (interfaces, IP header fields, transport header fields, MPLS label, diff-serv code point)
  - Metering (reliability, sampling, timestamps, flow timeout)
  - Data export (information model, reliability, confidentiality, integrity, anonymization, reporting times)

Conclusions

- Flow measurement
  - Medium-grain view of traffic on one or more links
- Advantages
  - Lower measurement volume than full packet traces
  - Available on high-end line cards (Cisco Netflow)
  - Control over overhead via aggregation and sampling
- Disadvantages
  - Computation and memory requirements for the flow cache
  - Loss of fine-grain timing and per-packet information
  - Not uniformly supported by router vendors

Data Interpretation

Data Interpretation: Outline

- Problem
  - Measurement data collected *inside* the network
  - Ambiguity about applications, hosts, & institutions
  - An issue for both packet and flow measurement
- Outline
  - Identifying hosts and institutions
  - Dynamic IP addresses
  - Port number ambiguity
  - Multiple identifiers across data sets

Identifying Hosts and Institutions

- nslookup
  - Translate domain name to an IP address
  - Translate IP address into domain name(s)
  - Reverse queries often unsuccessful
- whois
  - Identify institution responsible for an IP address
  - Identify institution responsible for an AS number
  - Information often incomplete or out-of-date
- traceroute
  - Identify names of interfaces near the destination address
- BGP tables
  - Associate an IP address with an IP prefix
  - Associate an address or prefix with an origin AS
  - Different vantage points, multiple origin ASes, etc.

nslookup Example

```
jrex% nslookup 199.222.69.151

Name:    lovelace.acm.org
Address:  199.222.69.151
```

```
jrex% nslookup 135.207.38.125

*** alliance.research.att.com can't find 135.207.38.125:
Non-existent host/domain
```

whois Examples

```
jrex% whois -h whois.arin.net 199.222.69.151
ANS Communications (NETBLK-ANS-C-BLOCK2) ANS-C-BLOCK2
                                  199.219.0.0 - 199.222.255.255
Association for Computing Machiner (NETBLK-ACM-NET) ACM-NET
                                  199.222.69.0 - 199.222.69.255
```

```
jrex% whois -h whois.arin.net 135.207.38.125
AT&T ITS (NETBLK-ATT-135-197-219-B) ATT-135-197-219-B
                                  135.197.0.0 - 135.219.255.255
AT&T Corporate ITS (NET-ATT-135-207-B) ATT-135-207-B
                                  135.207.0.0 - 135.207.255.255
```

traceroute example

```
jrex% traceroute 199.222.69.151
traceroute to 199.222.69.151 (199.222.69.151), 30 hops max, 40 byte packets
 1  oden (135.207.31.1)  1 ms  1 ms  1 ms
 2  opsec (135.207.1.2)  2 ms  2 ms  2 ms
 3  argus (192.20.225.225)  33 ms  219 ms  4 ms
 4  Serial1-4.GW4.EWR1.ALTER.NET (157.130.0.177)  4 ms  3 ms  4 ms
 5  117.ATM4-0.XR1.EWR1.ALTER.NET (152.63.25.186)  4 ms  4 ms  4 ms
 6  293.ATM6-0.XR1.NYC4.ALTER.NET (146.188.178.57)  4 ms  5 ms  11 ms
 7  189.ATM8-0-0.GW2.NYC4.ALTER.NET (146.188.178.133)  5 ms  5 ms  6 ms
 8  acm-gw.customer.ALTER.NET (157.130.17.238)  11 ms 10 ms  10 ms
 9  lovelace.acm.org (199.222.69.151)  9 ms  8 ms  8 ms
```

BGP Table Example (Using RouteViews)

*	199.222.69.0	167.142.3.6	0	5056 701 7046 i
*		4.0.0.2	0	1 701 7046 i
*		204.42.253.253	0	267 2914 701 7046 i
*		212.4.193.253	0	8918 701 7046 i
*		205.215.45.50	0	4006 701 7046 i
*		193.140.0.1	0	8517 9000 2548 701 7046 i
*		165.87.32.5	0	2685 701 7046 e
*		206.220.240.223	0	10764 1 701 7046 i
*		203.62.248.4	0	1221 16779 1 701 7046 i
*		203.62.252.21	0	1221 16779 1 701 7046 i
*		157.22.9.7	0	715 1 701 7046 i
*		193.0.0.56	0	3333 9057 3356 701 7046 i
*		195.219.96.239	0	8297 6453 701 7046 i

Prefix 199.222.69.0/24 has origin AS 7046
(whois says that 7046 is ASN-UUNET-CUSTOMER)

Associating IP Address With Host

- Problem: dynamic assignment of IP addresses
  - Difficult to associate traffic with client or user over time
- Track assignment of IP address
  - Modem records that indicate IP address assignment
  - Logs or traces of address allocations by DHCP servers
- Timeout heuristic
  - Assume a single client is responsible for the traffic
  - ... until a period of inactivity (IP address may be reassigned)
  - Useful for studying “session-level” user behavior
- Application-level headers
  - Extract client/user-level information from layer-4 header
  - Unique user with From: jrex@att.com or Cookie: user17
  - Change from User-Agent: Mozilla/4.03 to Mozilla/2.0

Port Numbers: Ambiguity

- Associate well-known ports with applications
  - Port 53 for DNS, port 80 for HTTP, port 119 for NNTP, ...
- Expect unusual use of port numbers
  - Keep track of de facto ports (e.g., 8000 and 8080 for HTTP)
  - Check for anomalous transfers on well-known ports
  - Explore sites using unexplained port #s (download games!)
- Expect dynamic assignment of port numbers
  - Check for parallel transfers between the same end points (e.g., to find FTP data transfer in parallel to FTP control)
  - Parse the control messages that select the dynamic port #s (e.g., parse FTP commands and RTSP messages)

Incoming Traffic Volume by Application

Application	July 1, 2001	Sep 13, 2000
HTTP	46.7%	53.2%
TCP unknown	17.2%	12.9%
KaZaA	9.9%	0.0%
FTP	5.7%	4.3%
UDP unknown	3.3%	3.2%
Napster	2.0%	10.7%
SMTP	1.5%	2.9%
NNTP	1.3%	3.2%
HTTPS	1.0%	1.0%

Multiple Identifiers of Interfaces

- Flow-level measurements
  - Netflow record identifies an interface by SNMP index
  - Flow with input interface “7” and output interface “3”
- SNMP interface MIB
  - Index (7), IP address (12.123.36.73), and name (POS6/0)
- Configuration files
  - Interface IP address (12.123.36.73) and name (POS6/0)
  - BGP session to a neighbor AS associated with the interface
- Joining multiple data sets
  - Netflow: traffic volume and SNMP interface index
  - SNMP MIB: interface index, IP address, and name
  - Configuration: neighbor AS associated with interface

Conclusions

- Identifying institutions
  - Tools like nslookup, whois, traceroute, BGP tables
- IP and port number ambiguity
  - Heuristics, appl-level info, and de facto standards
- Multiple identifiers for same network element
  - Join of information from multiple data sets