Patchwork

Testbed-wide Profiling for FABRIC

ILLINOIS TECH

Nishanth Shyamkumar

KNIT9 – Sep 24, 2024

CNS-2346499

Motivation

- Knowing what's on the dataplane and profile it…
  … *feasibly* and *flexibly*.

- "Feasible?"
  - Port mirroring is a basic and ubiquitous low-level primitive on switches.
  - Tcpdump is deployed widely, easily available and well tested.
  - Telemetry helps, but Patchwork provides finer granularity.

- "Flexible?"
  - What to capture?
  - How much to sample?
  - How much to truncate?

Patchwork's Goal

Providing a network profiler for FABRIC users.

Abstract away the FABRIC APIs.

Two usage modes:

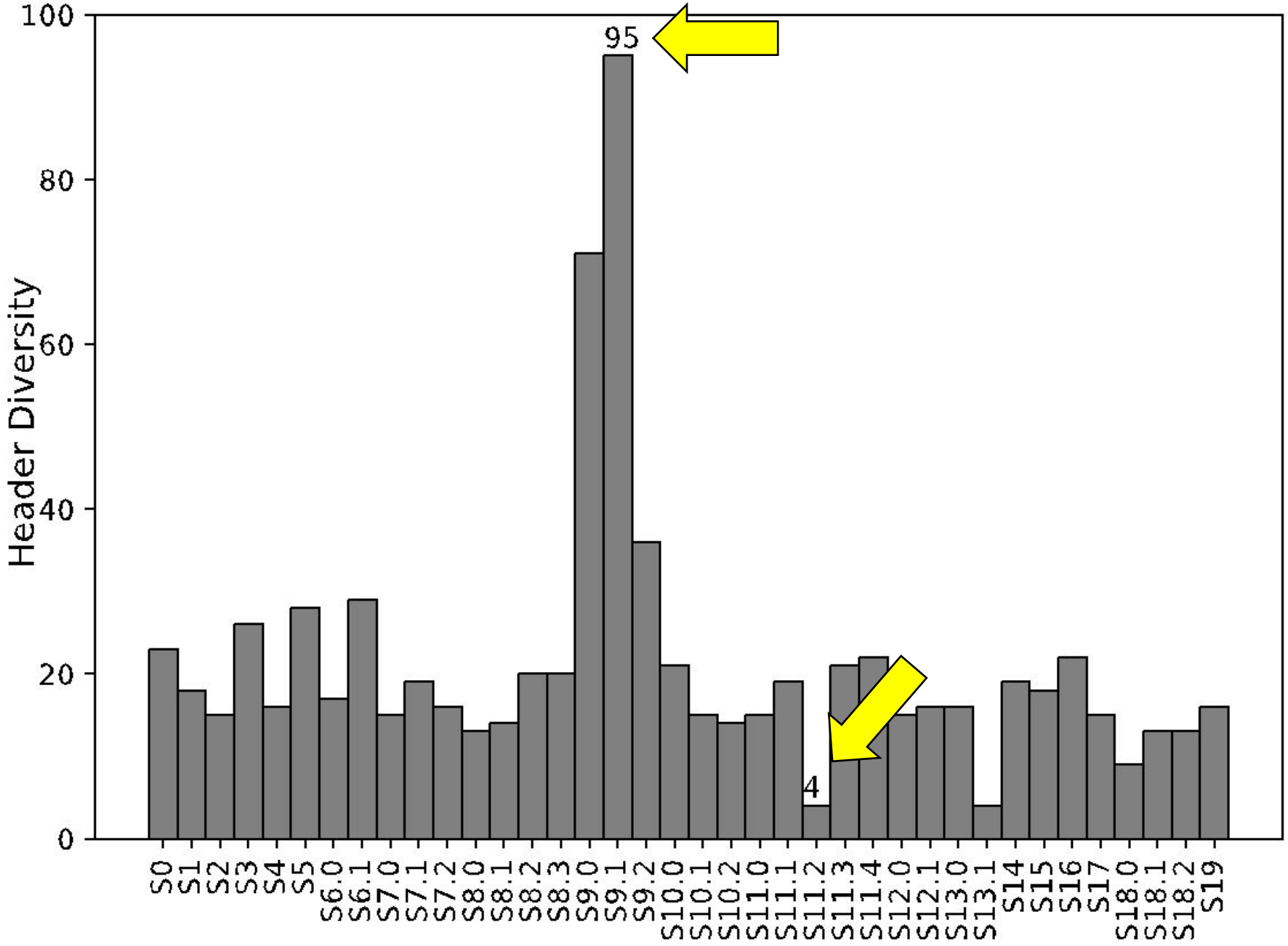
1. Users: Building a profile of individual experiments.
2. Operators: Building a profile of FABRIC's network traffic.

How can Researchers benefit?

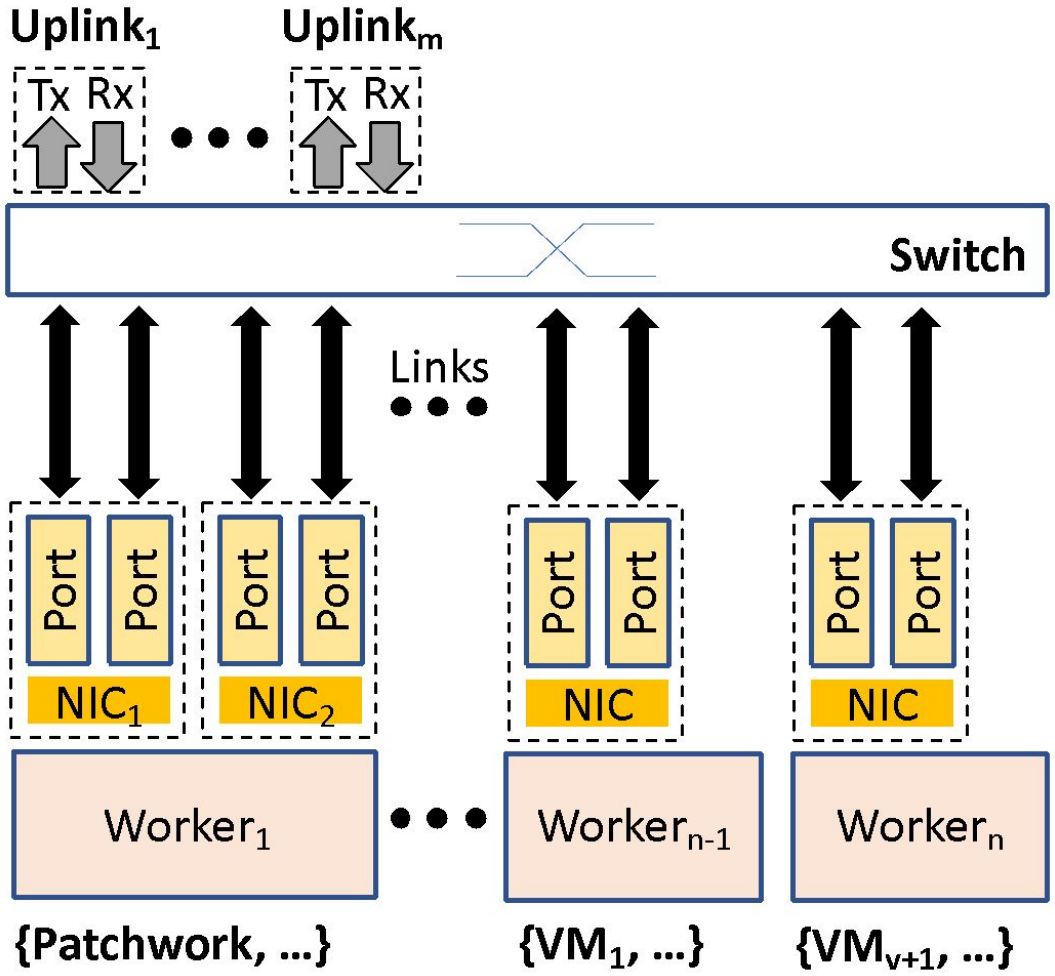
- Profile your experiment:
  - Characterize the workload of your service.
  - Is your network behaving the way you expect it to? e.g., frequent RSTs?

- Traffic profile composition
  - Header types
  - Encapsulation patterns
  - Flows
    - Number of packets
    - Frame sizes
    - Inter-packet delays

Early study


Header Diversity
100
80
60
40
20
0
95
4
S0
S1
S2
S3
S4
S5
S6.0
S6.1
S7.0
S7.1
S7.2
S8.0
S8.1
S8.2
S8.3
S9.0
S9.1
S9.2
S10.0
S10.1
S10.2
S11.0
S11.1
S11.2
S11.3
S11.4
S12.0
S12.1
S13.0
S13.1
S14
S15
S16
S17
S18.0
S18.1
S18.2
S19

Patchwork runs as a FABRIC experiment



Performance

- Overhead
- Tcpdump throughput limited to 8.5Gbps for 1500 Byte packets
- Kernel NW stack + CPU processing

- Performance improvement
  - Offload filtering and truncation to Alveo U280 FPGA using P4.
  - DPDK pcap filter to create pcap files of traffic
  - However, CPU becomes bottleneck at ~2Mpps/core, due to packet copy

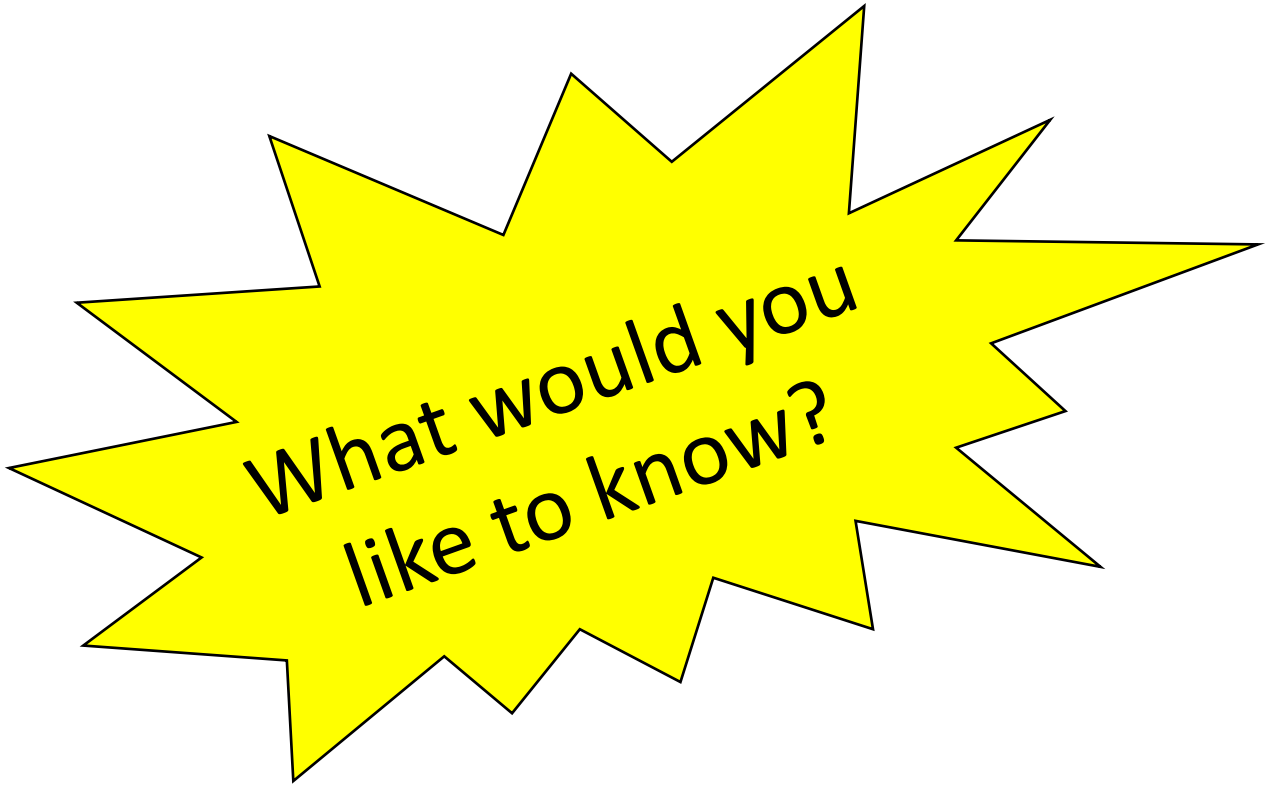
Performance: Alveo FPGA offloads

Send rate(Line rate%)	Packet Loss (%)	Packet Size	Cores used
20	0	1024	5 cores
40	0.078770608	1024	5 cores
60	0.034233449	1024	5 cores
80	0.203838958	1024	6 cores
100	0.135046993	1024	7 cores
100	8.672533493	1024	7 cores

Line rate of **100Gbps** for packet sizes of 1024 Bytes and above.

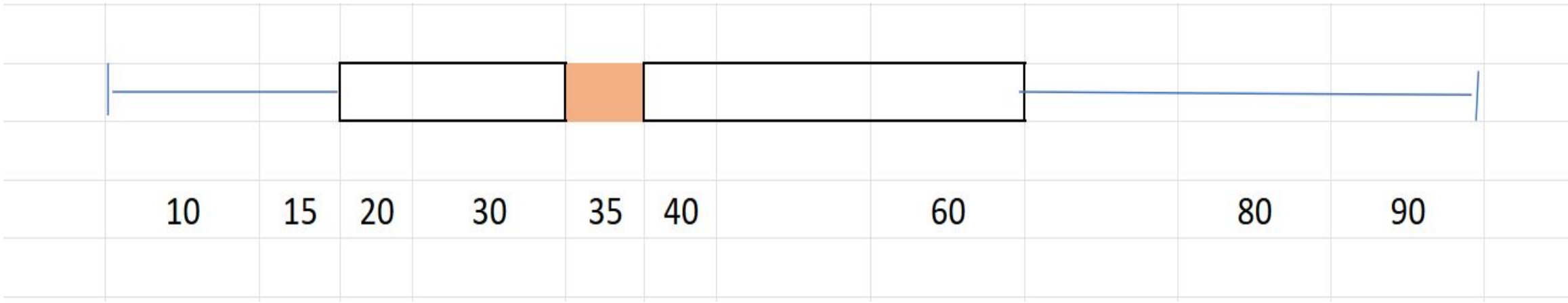
Dataplane Analyses

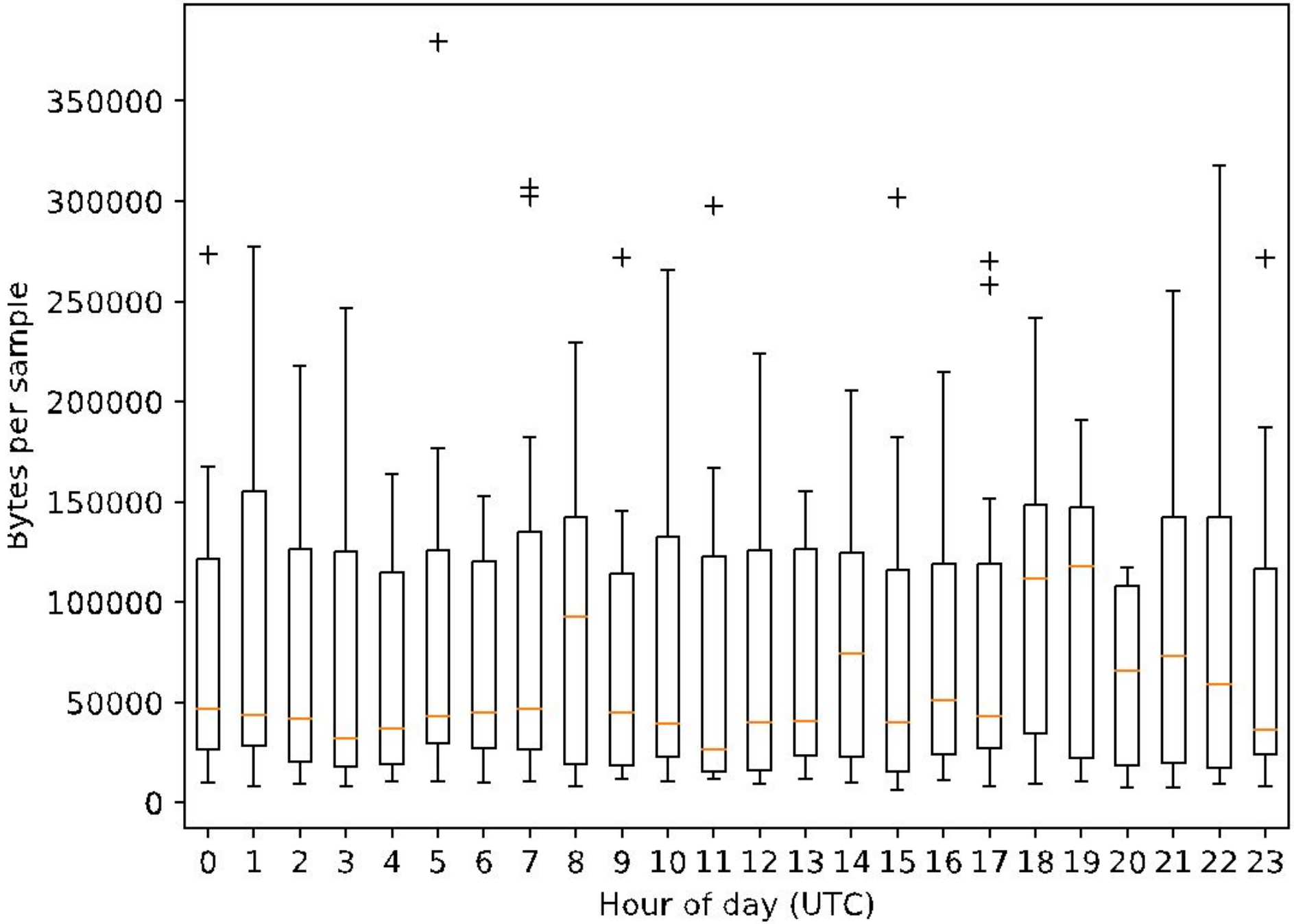
- Early study
  - Frame sizes across FABRIC sites
  - Headers across FABRIC
  - Header diversity across FABRIC sites
- Later analyses
  - Encapsulation depth + patterns
  - #Flows across FABRIC sites
  - Flow composition (headers, frame sizes, burstiness)
  - Flow persistence (across #Samples)
  - Hour, Day, Month



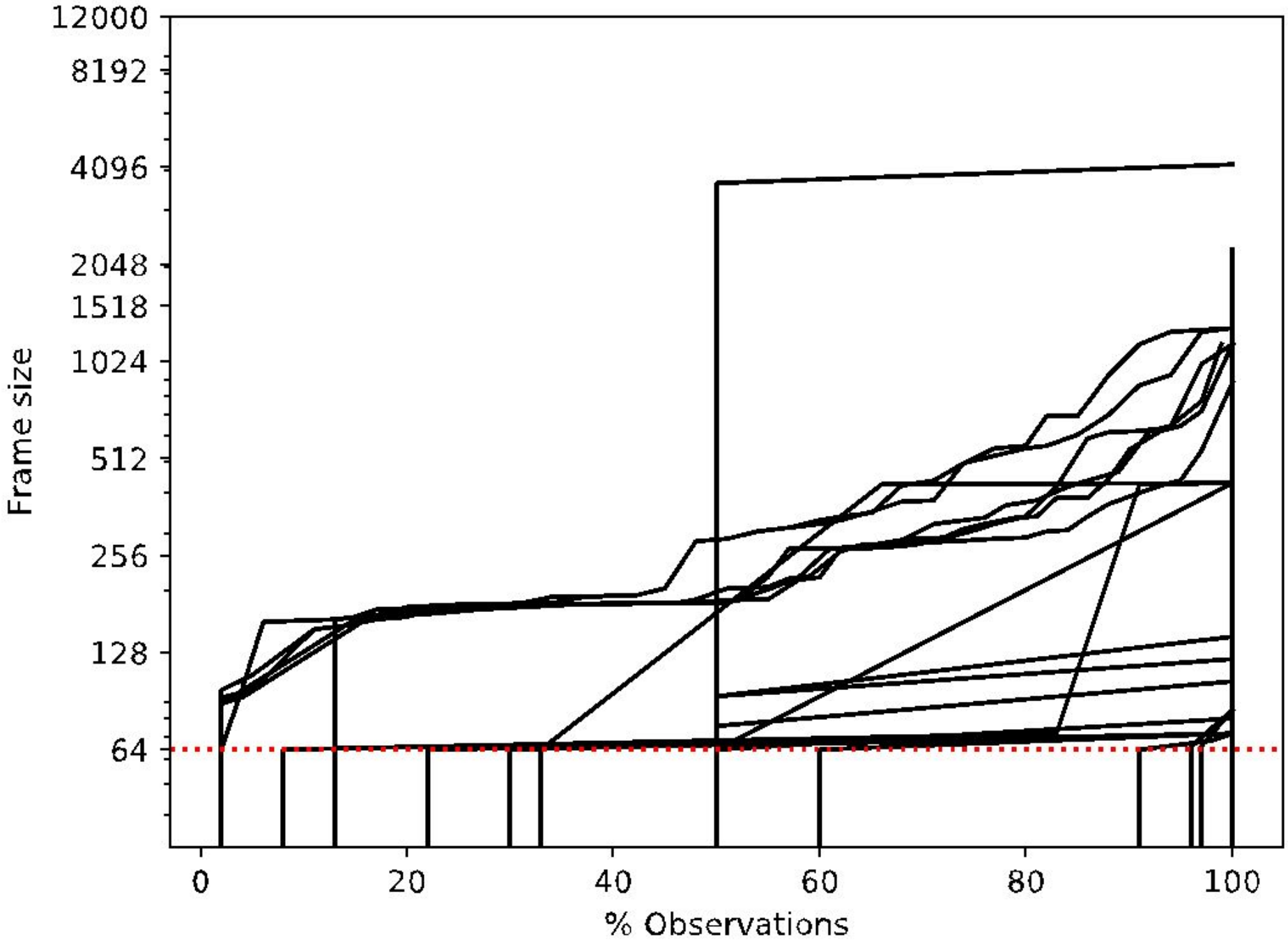
Traffic volume Characteristics

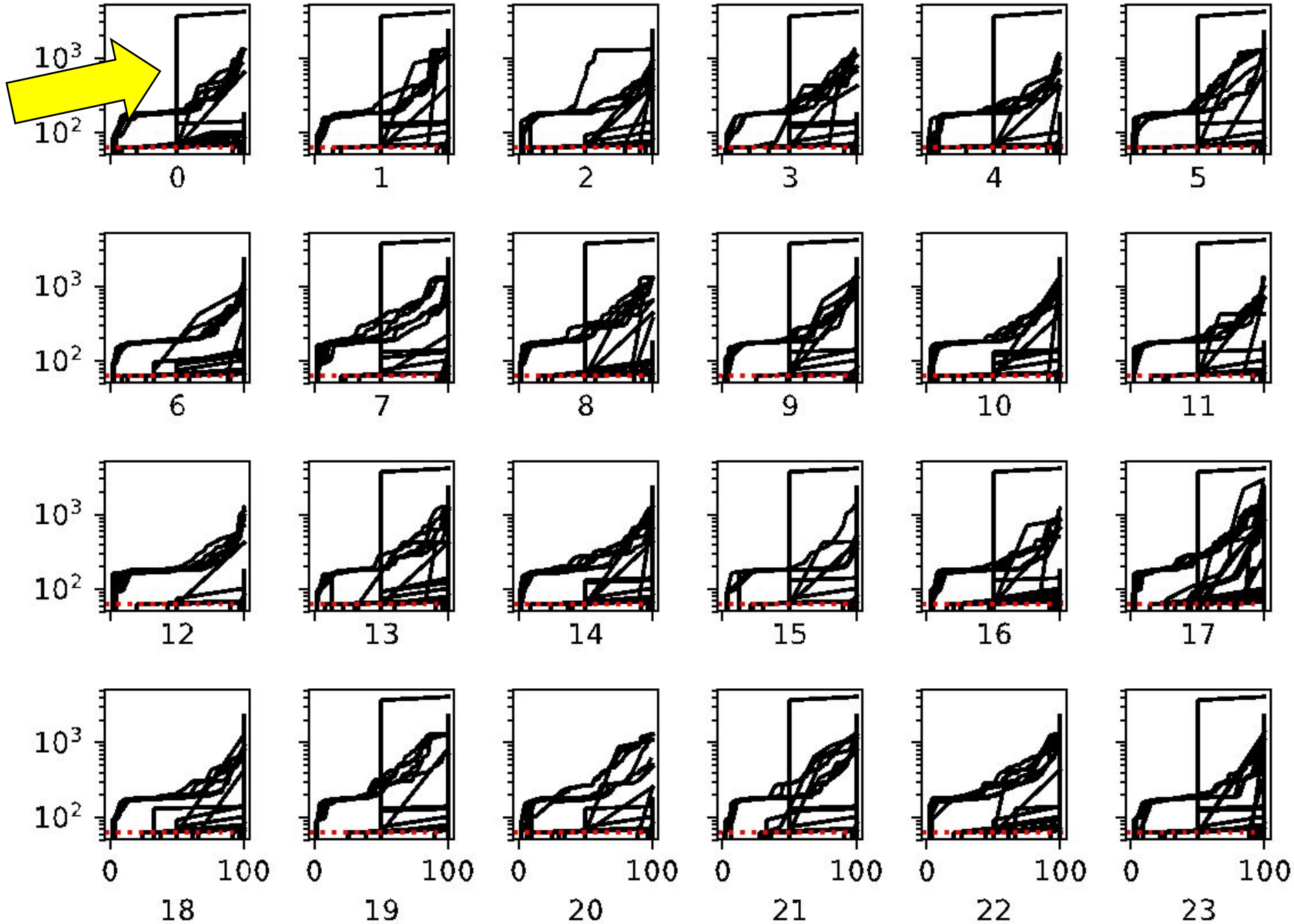
Box plot

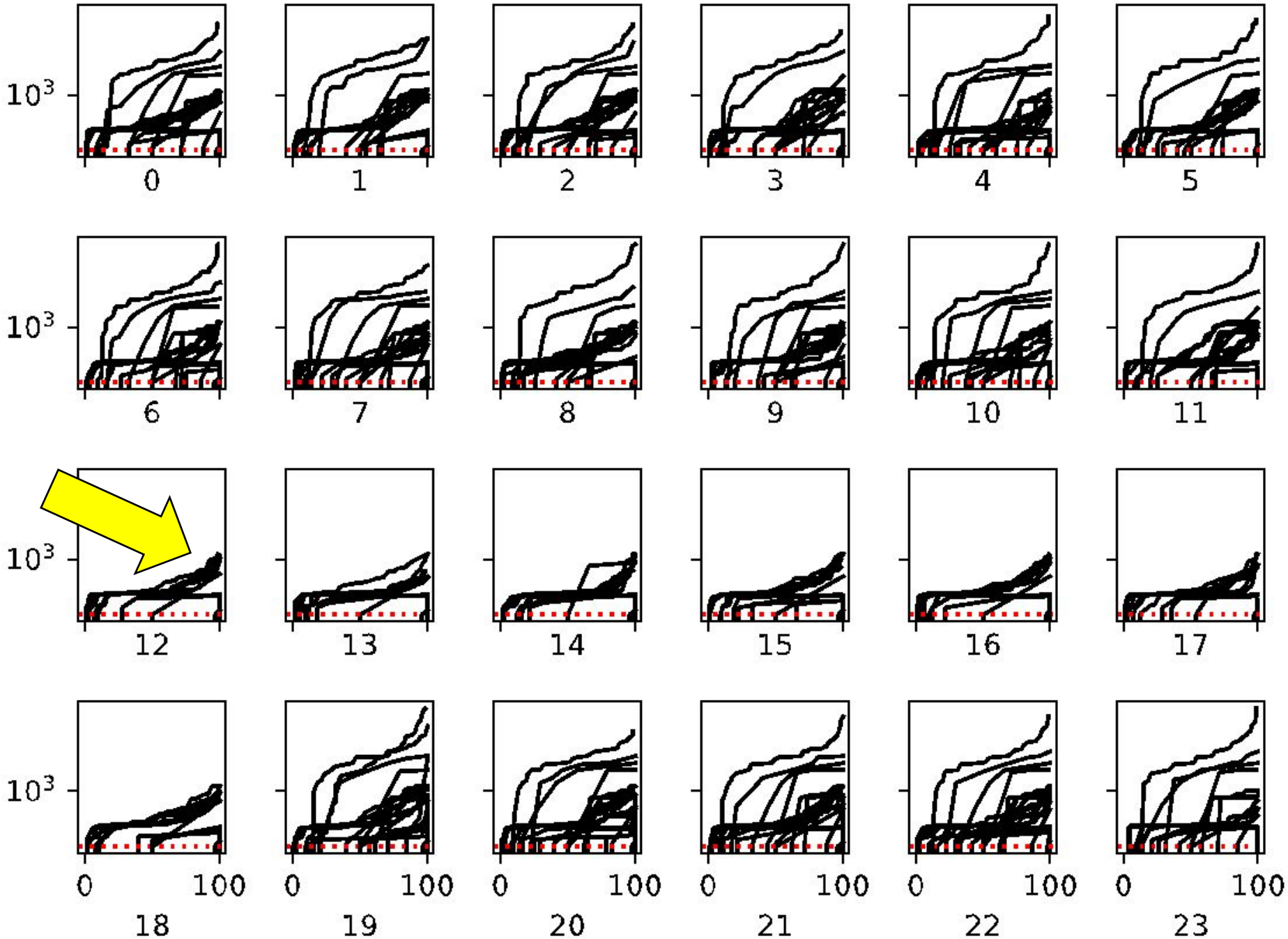



Bytes per sample
350000
300000
250000
200000
150000
100000
50000
0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
Hour of day (UTC)

FLOW CHARACTERISTICS


12000
8192
4096
2048
1518
1024
512
256
128
64
Frame size
0
20
40
60
80
100
% Observations


10^3
10^2
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
0
100


10^3
0
1
2
3
4
5
10^3
6
7
8
9
10
11
10^3
12
13
14
15
16
17
10^3
0
100
18
0
100
19
0
100
20
0
100
21
0
100
22
0
100
23

Summary

PATCHWORK is a ‘low barrier to entry’ network profiler

Can be used by Operators and End users

Analyze pcap files to glean information and obtain a model of traffic usage, Flow characteristics, header and frame size diversity etc.

Performance sensitive version implemented using FPGAs and DPDK.

Thank you

- Team at **Illinois Tech**: Hyunsuk Bang, Bjoern Sagstad, Prajwal Venkateshmurthy, Sean Cummings, Nik Sultana.
- Komal Thareja and Mert Cevik as **RENCI**,
  Charles Carpenter and Yongwook Song at **UKY**,
  and Stacey Sheldon, Peter Bengough, and Yatish Kumar at **ESnet**.

Additional Slides:

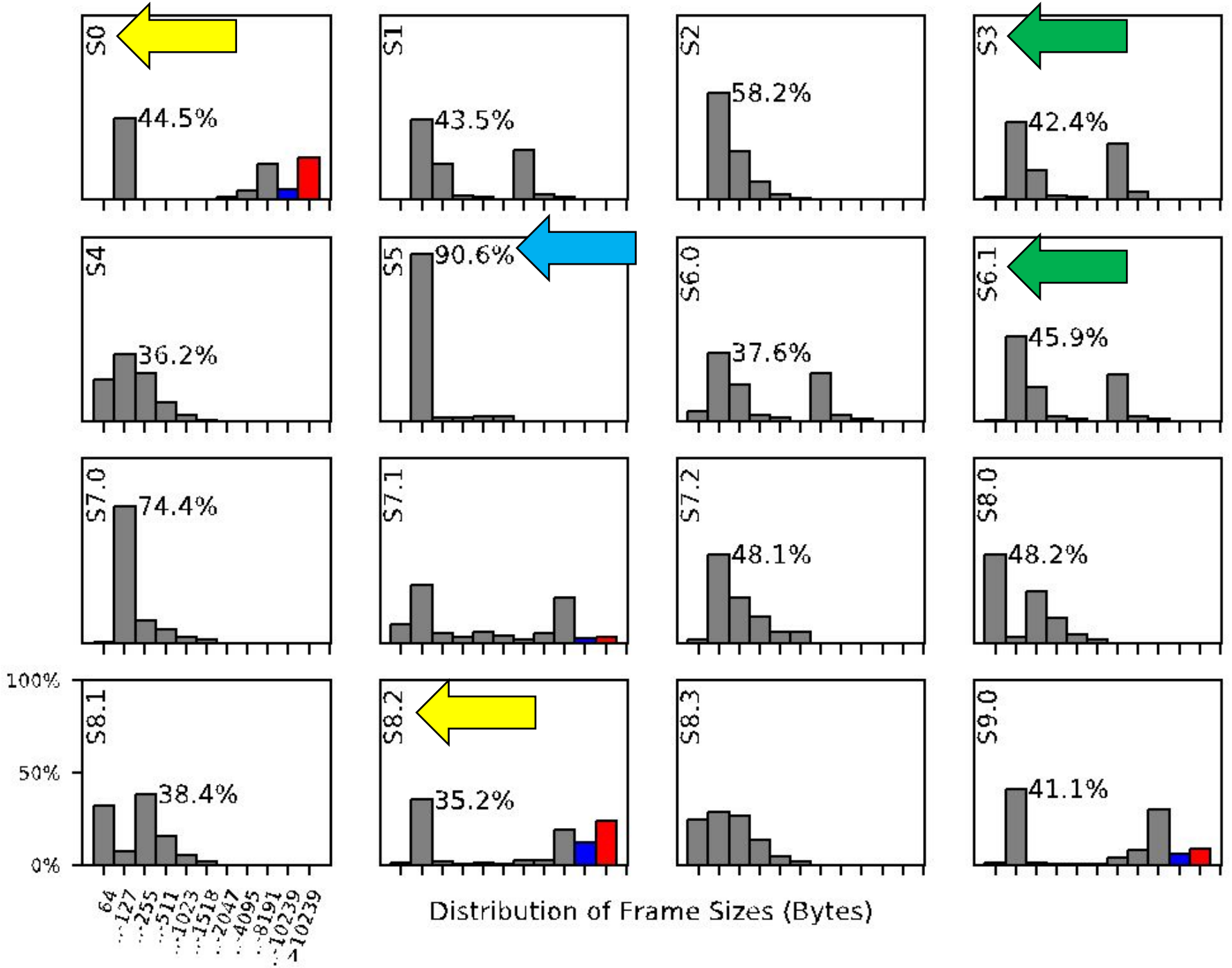
What have we learnt so far?

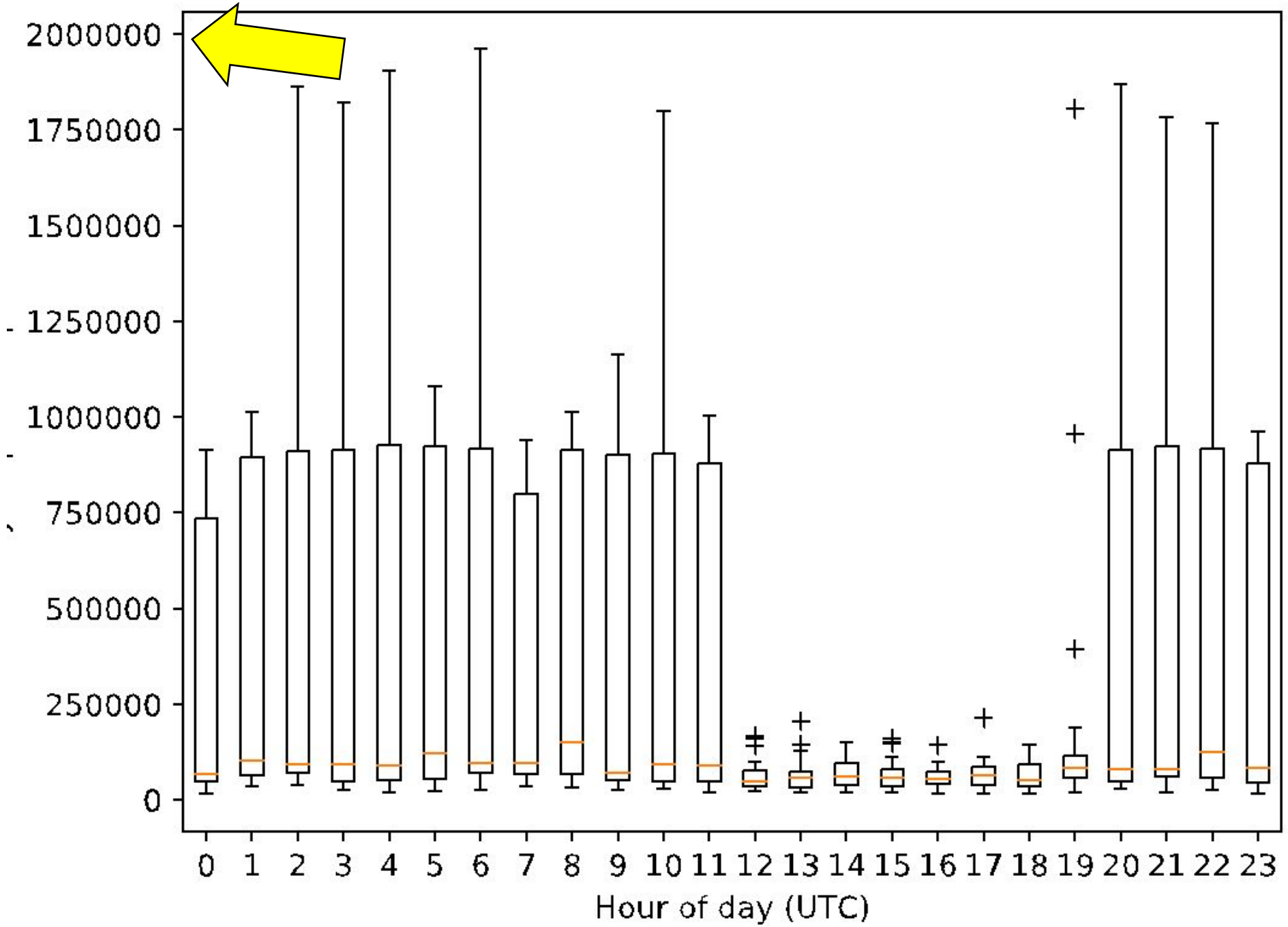
- “ssh” traffic seems to be the “radiological background”
  (A proxy value for FABRIC’s load – could watch this metric for “disturbances in the force”/anomalies?)

- Traffic profiles do not follow a day/night cycle. Is more consistent

- Working in harmony with back-end dynamics – challenges and opportunities. (We try to mitigate these for Patchwork users.)

Early study

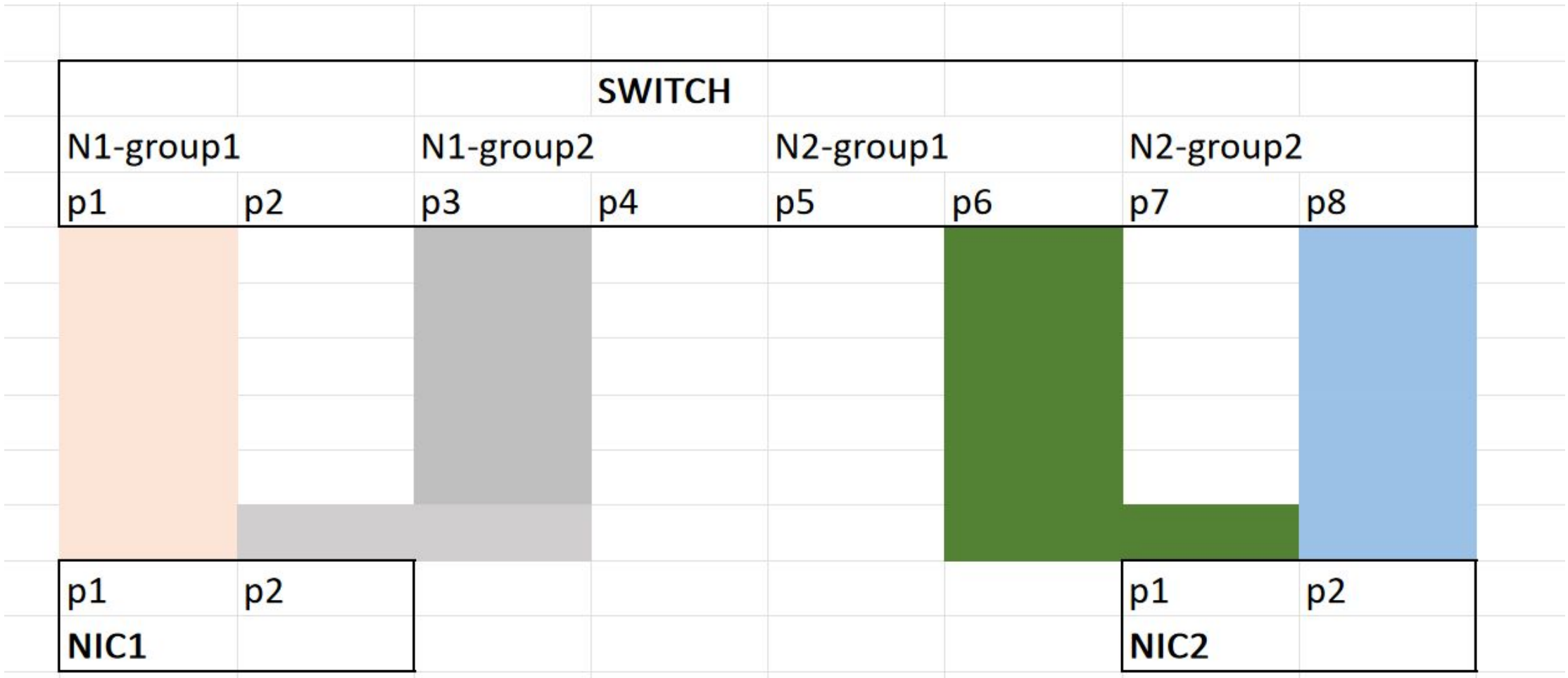

S0
44.5%
S1
43.5%
S2
58.2%
S3
42.4%
S4
36.2%
S5
90.6%
S6.0
37.6%
S6.1
45.9%
S7.0
74.4%
S7.1
S7.2
48.1%
S8.0
48.2%
S8.1
38.4%
S8.2
35.2%
S8.3
S9.0
41.1%
100%
50%
0%
64
...127
...255
...511
...1023
...1518
...2047
...4095
...8191
...10239
>10239
Distribution of Frame Sizes (Bytes)


2000000
1750000
1500000
1250000
1000000
750000
500000
250000
0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
Hour of day (UTC)

Patchwork's Design: Logical Structure

1. Setup phase: Get resources based on sites specified by user
2. Sampling phase: Listen for packets and write to pcap files; based on user parameters
3. Gathering phase: Download pcaps and consolidate into a site level directory
4. Analysis phase: Analyze packets using tshark primitive. Done offline

Patchwork's Design: Strobing phase



Patchwork's Design: Strobing phase

Today’s talk

- Patchwork’s relevance to FABRIC and its users.

- You can already use Patchwork on FABRIC!
  This talk will provide you with more information.