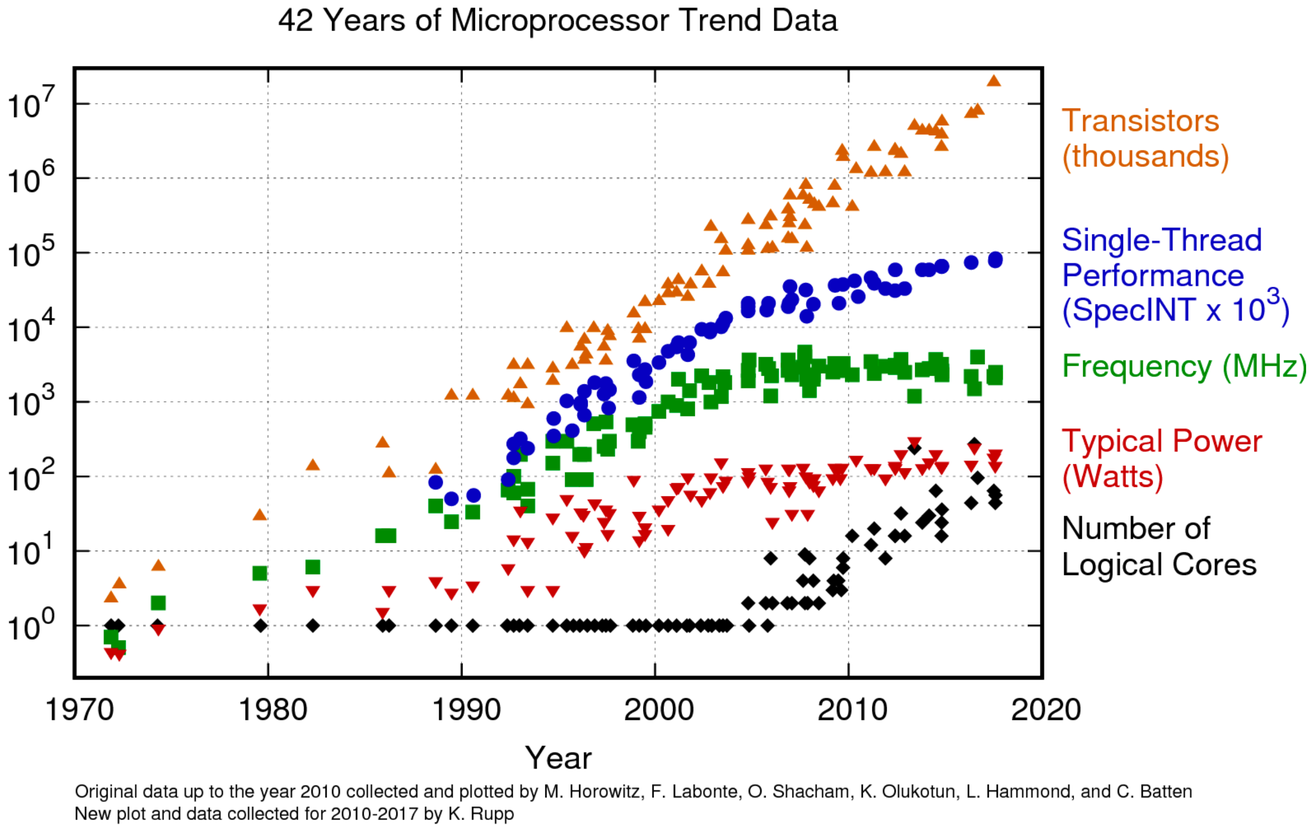
# Breaking Band

35 56

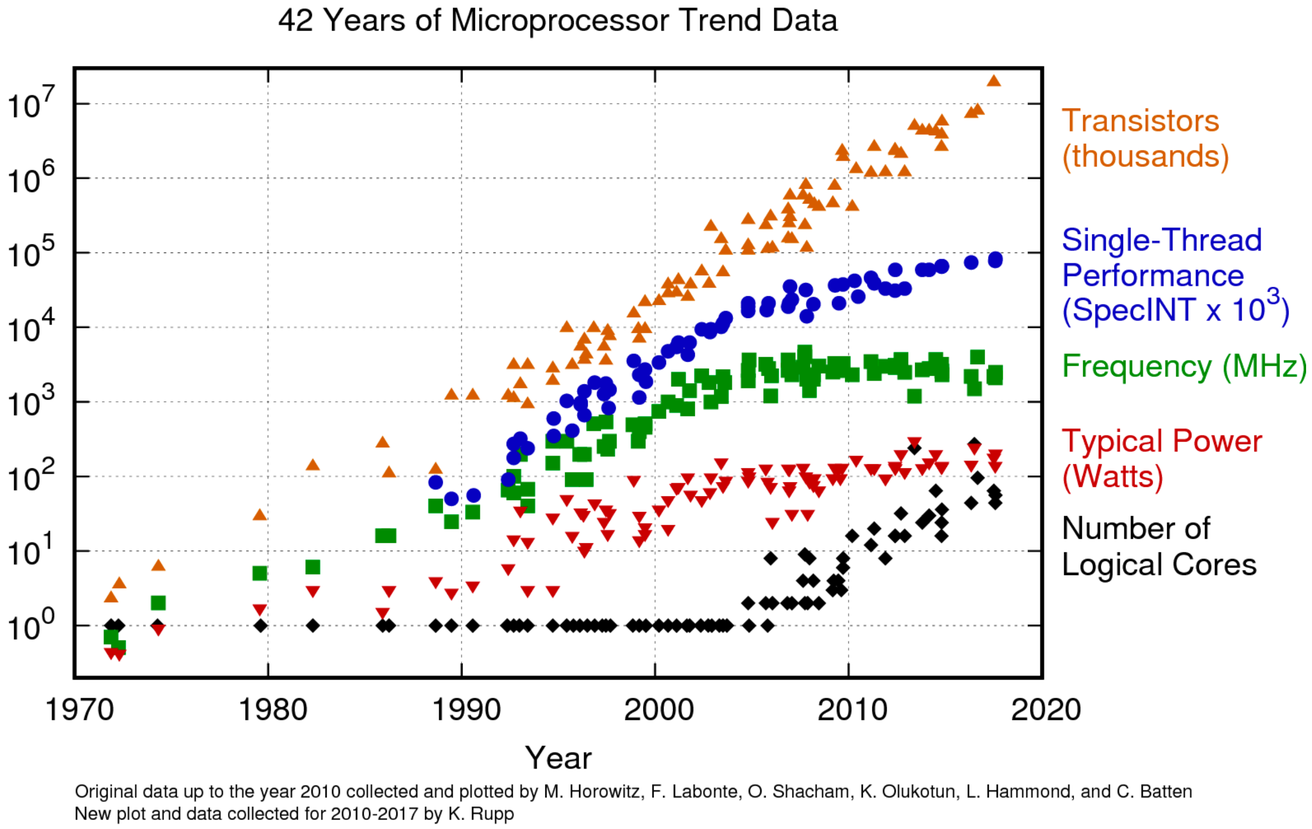
## A Breakdown of High-performance Communication

Rohit Zambre,* Megan Grodowitz,^ Aparna Chandramowlishwaran,* Pavel Shamis^
*University of California, Irvine
^Arm Research

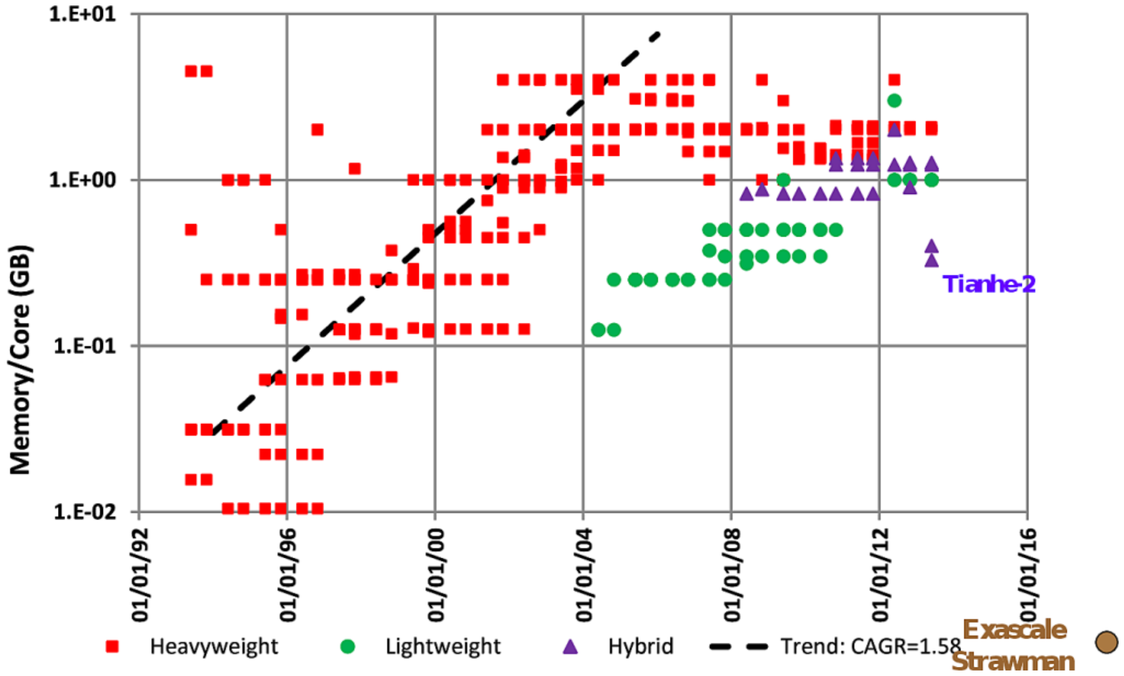

Original data up to the year 2010 collected and plotted by M. Horowitz, F. Labonte, O. Shacham, K. Olukotun, L. Hammond, and C. Batten
New plot and data collected for 2010-2017 by K. Rupp

https://www.karlrupp.net/2018/02/42-years-of-microprocessor-trend-data/

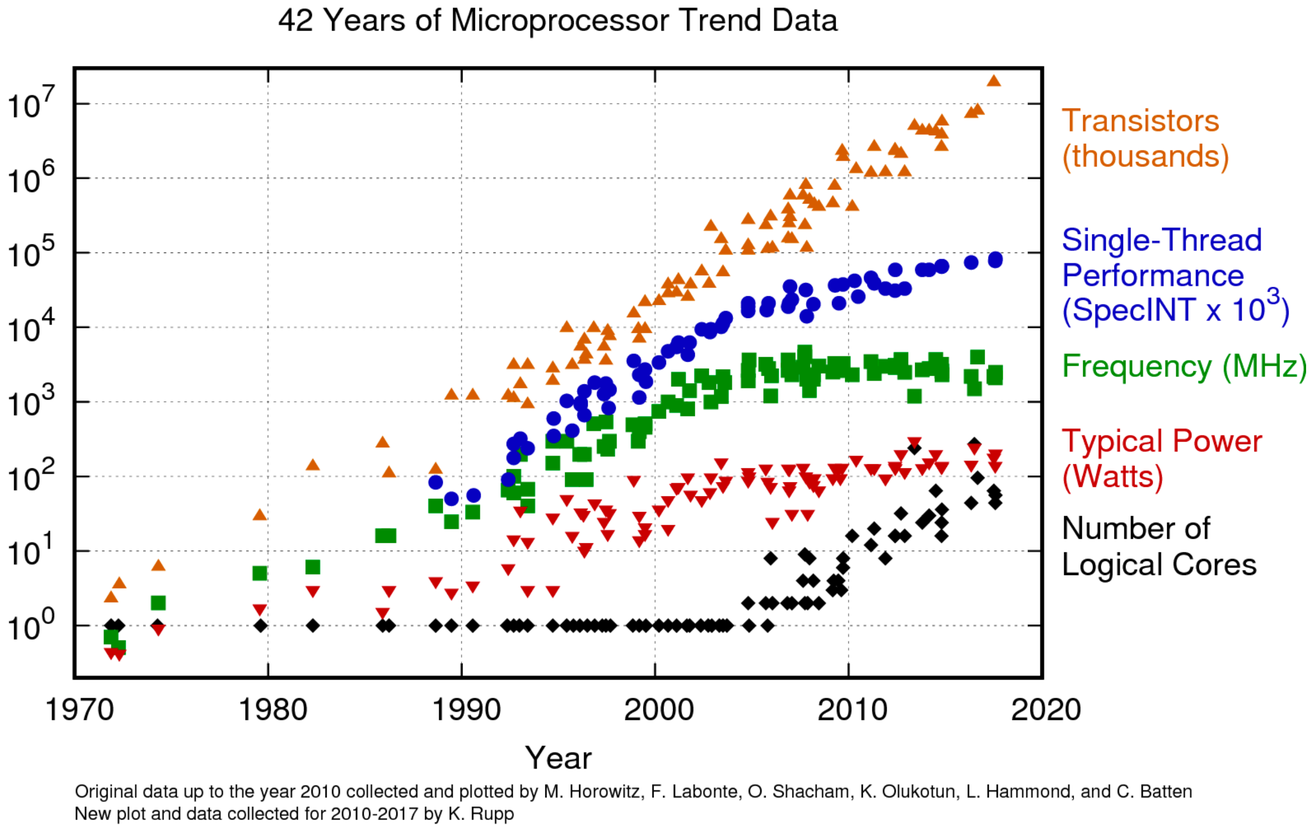



Original data up to the year 2010 collected and plotted by M. Horowitz, F. Labonte, O. Shacham, K. Olukotun, L. Hammond, and C. Batten
New plot and data collected for 2010-2017 by K. Rupp

https://www.karlrupp.net/2018/02/42-years-of-microprocessor-trend-data/

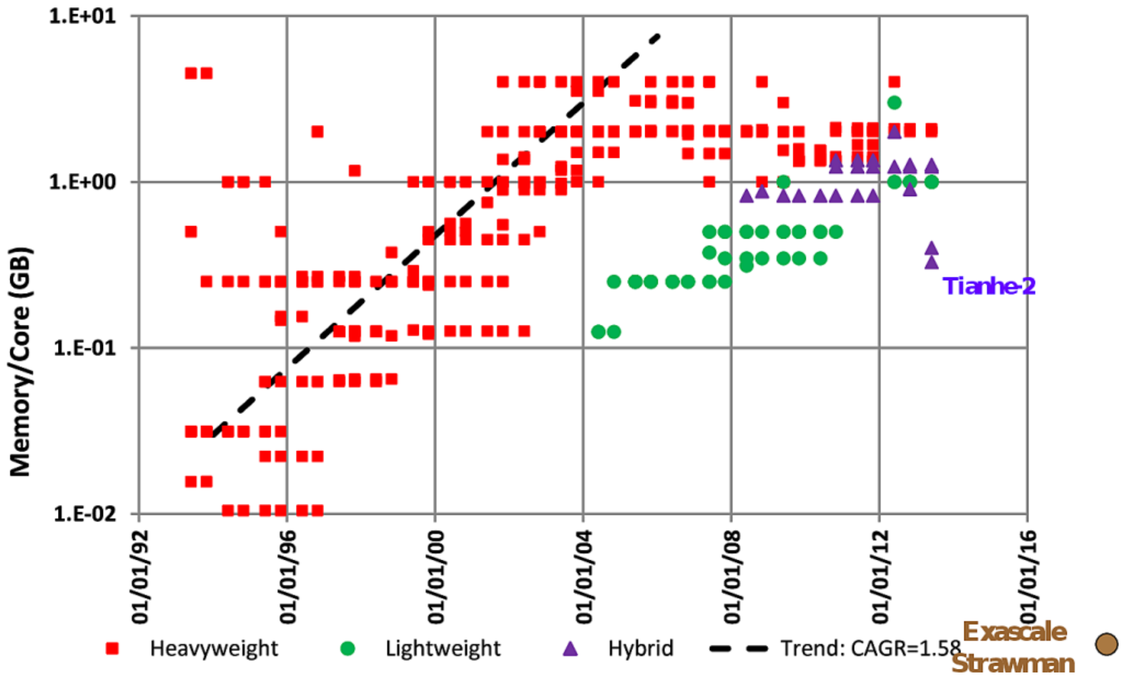



**Evolution of the memory capacity per core in the Top500 list**
(Peter Kogge. Pim & memory: The need for a revolution in architecture.)

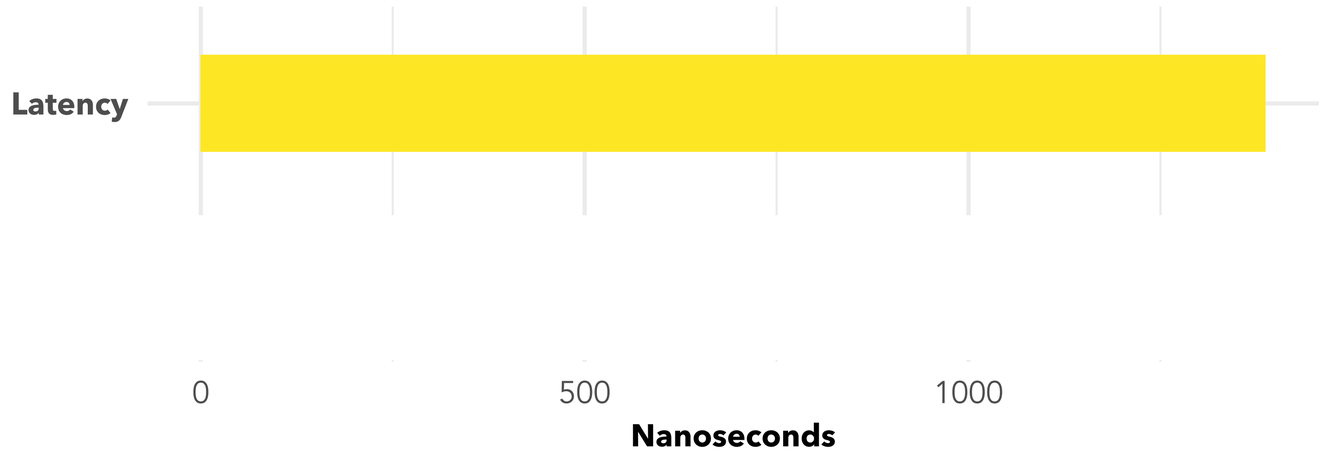


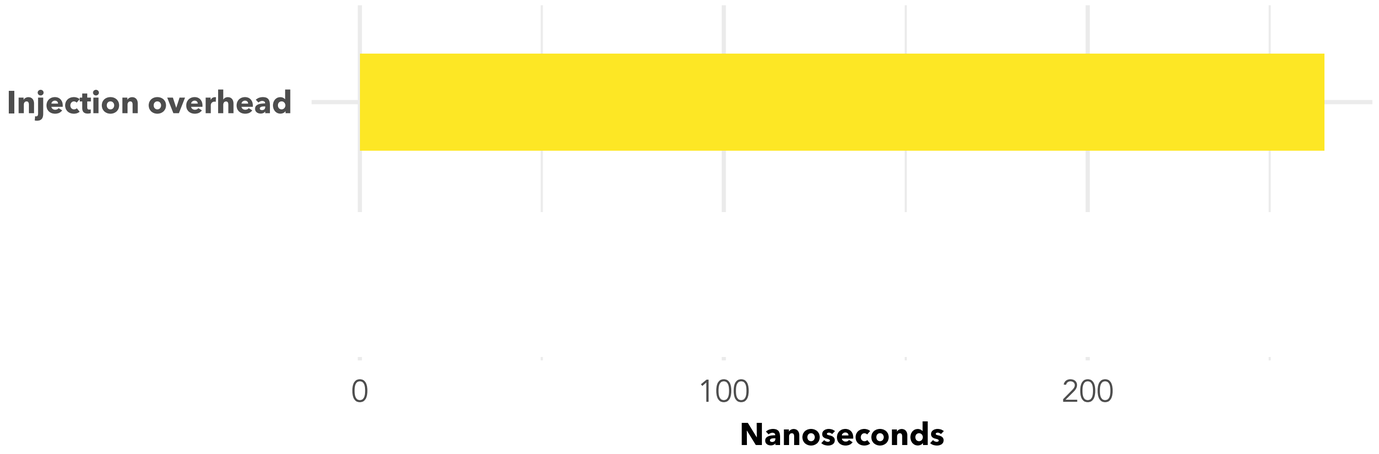

https://www.karlrupp.net/2018/02/42-years-of-microprocessor-trend-data/

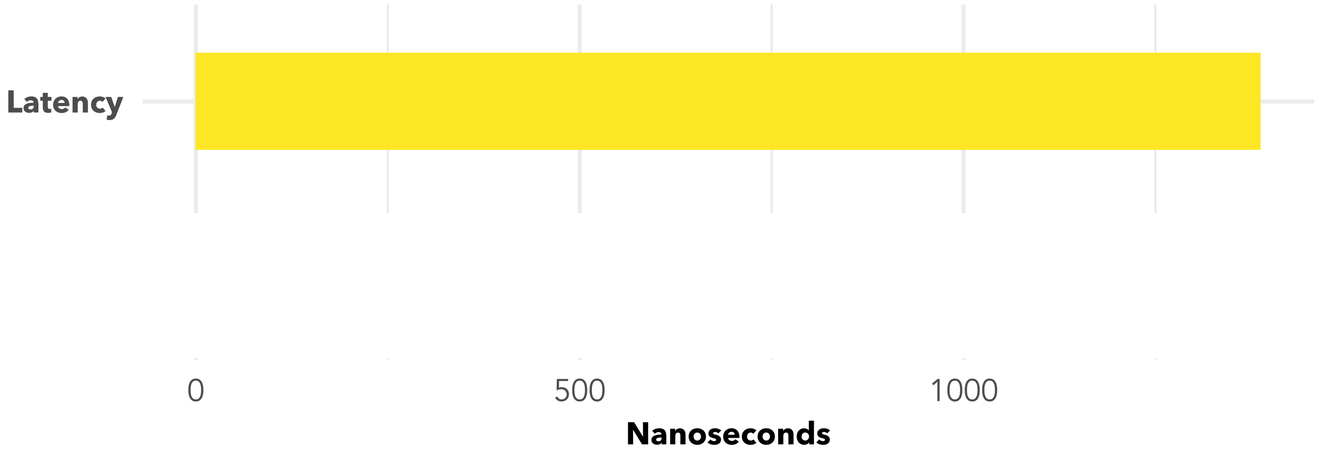
- Strong scaling is the way forward.
- Small messages at the limits of strong scaling.



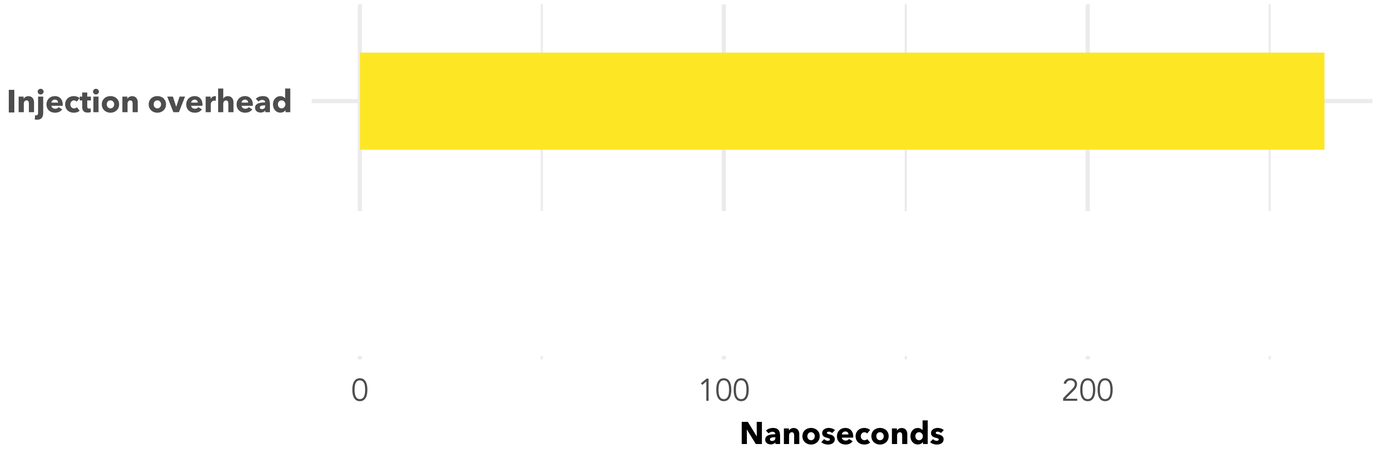

**Evolution of the memory capacity per core in the Top500 list**
(Peter Kogge. Pim & memory: The need for a revolution in architecture.)

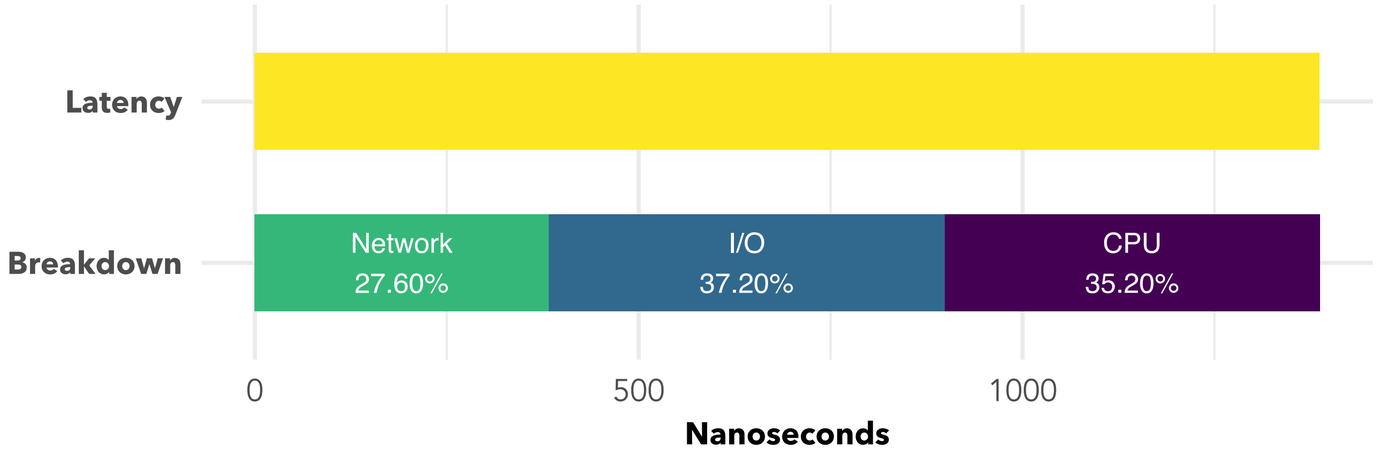

Latency
0
500
1000
Nanoseconds


Injection overhead
0
100
200
Nanoseconds

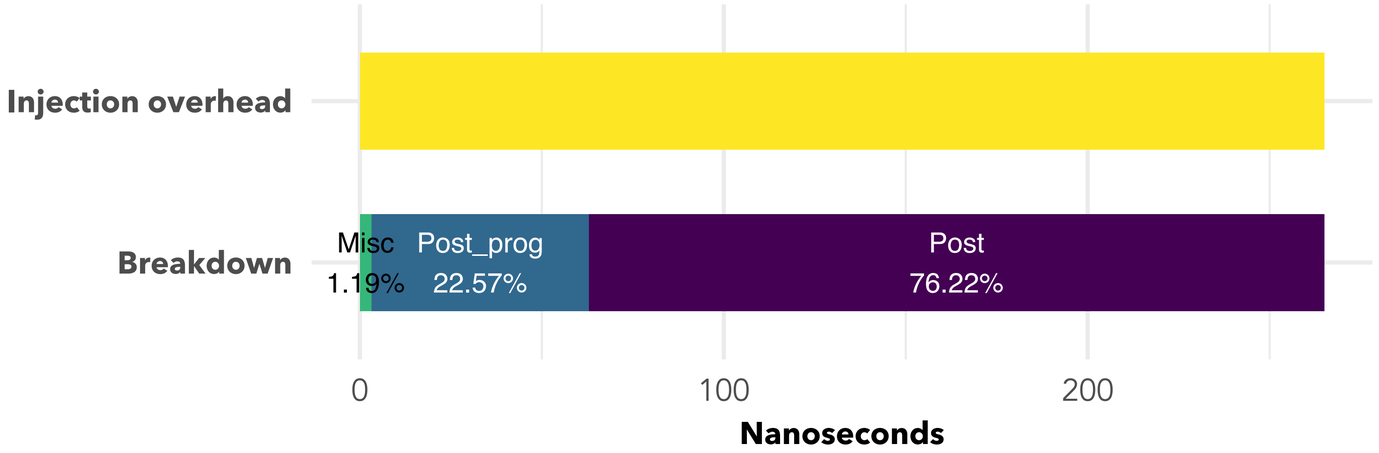


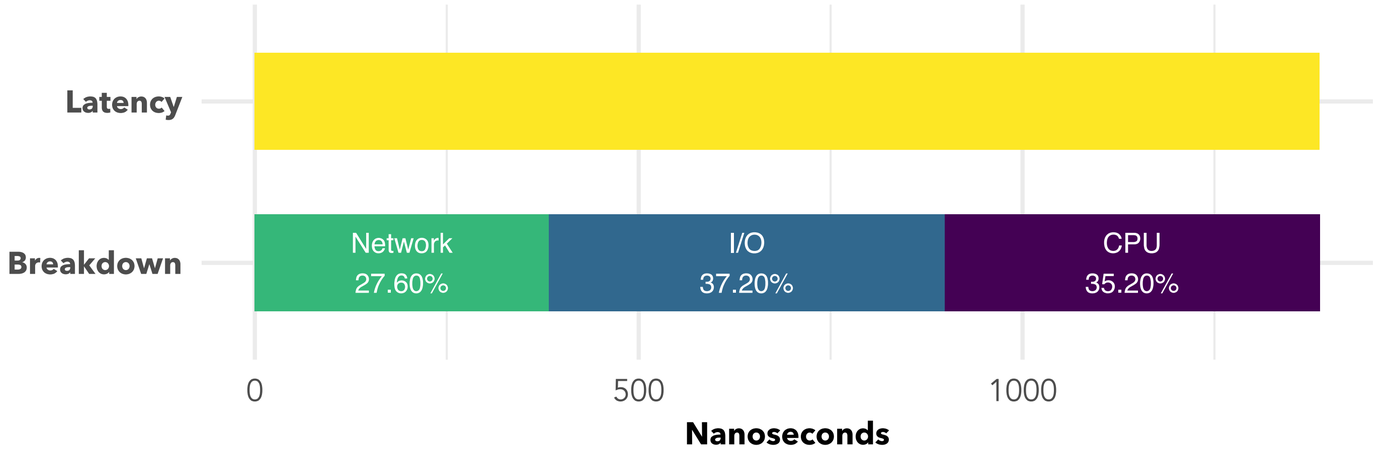

- How much does a component contribute?

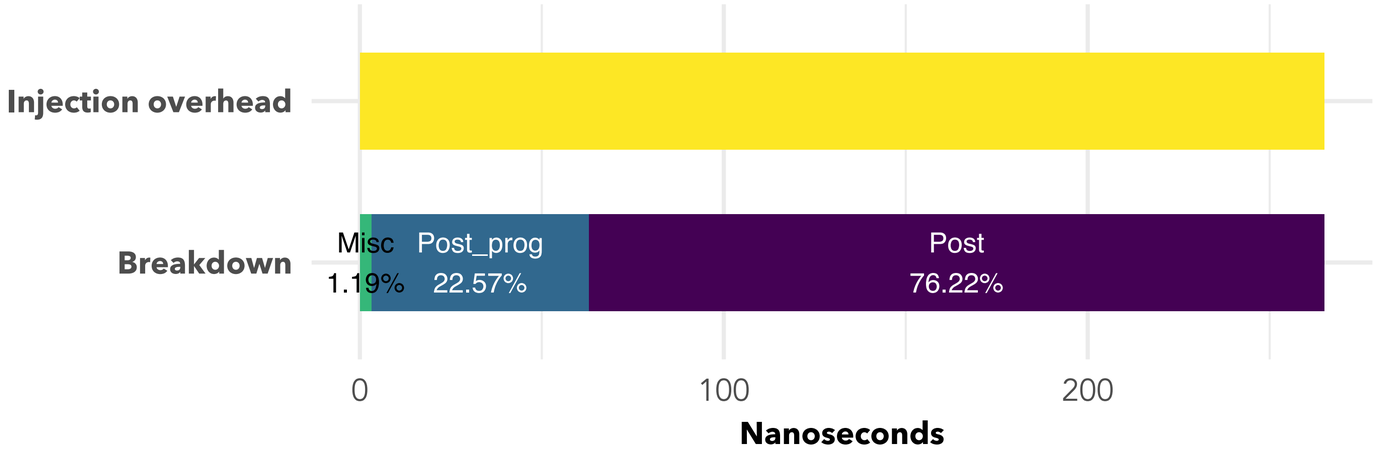





- How much does a component contribute?






- How much does a component contribute?
- If we optimize component X by Y%, by how much will communication performance improve?

# CONTRIBUTIONS OF THE PAPER

- *A detailed breakdown of communication performance of small messages.*
- Analytical models to explain the injection and latency.
  - Effective within 5% of the observed performance.

# CONTRIBUTIONS OF THE PAPER

- *A detailed breakdown of communication performance of small messages.*
- Analytical models to explain the injection and latency.
  - Effective within 5% of the observed performance.
- Detailed measurement methodology to produce breakdown on any other system configuration.

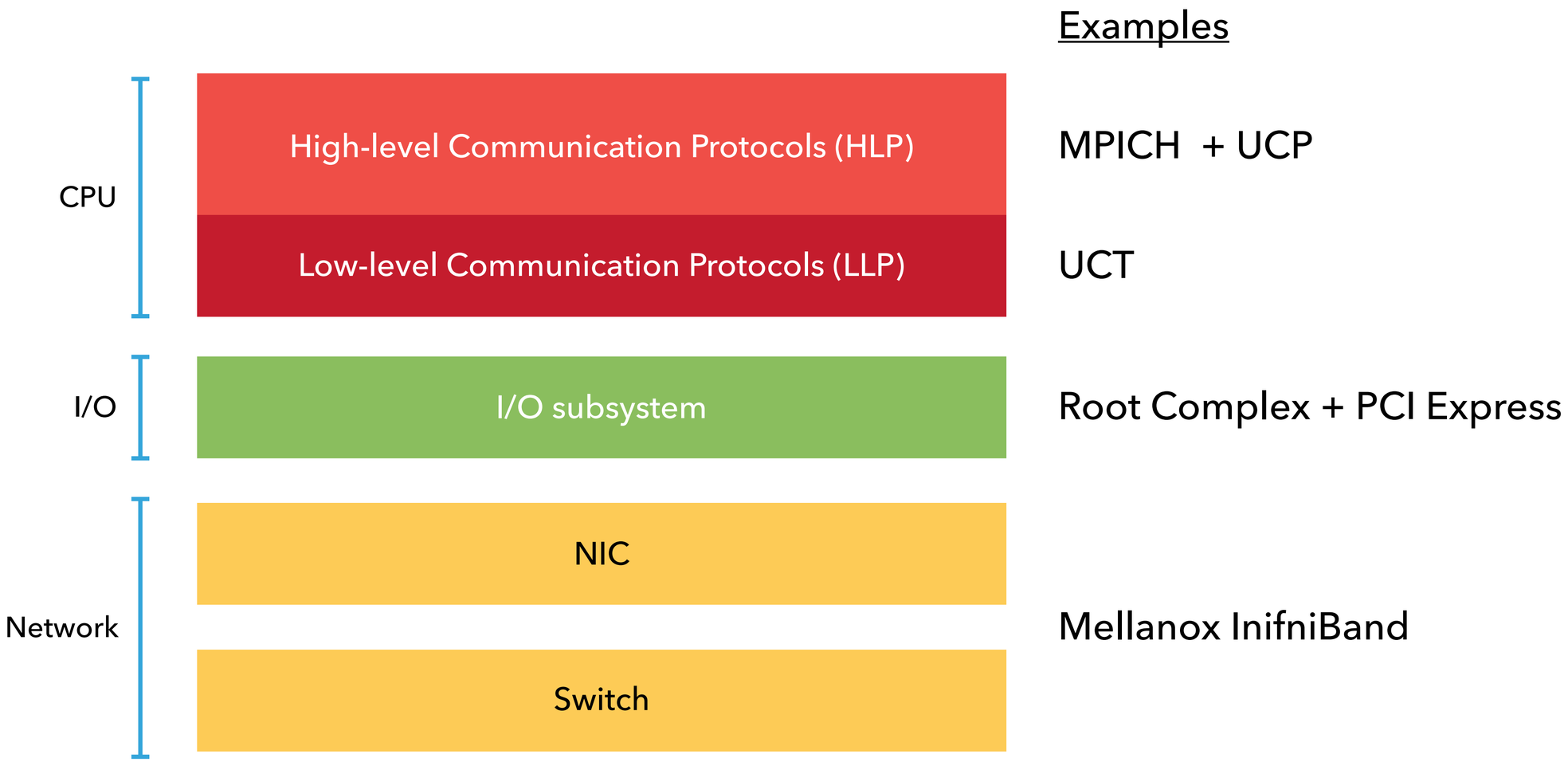
# CONTRIBUTIONS OF THE PAPER

- *A detailed breakdown of communication performance of small messages.*
- Analytical models to explain the injection and latency.
  - Effective within 5% of the observed performance.
- Detailed measurement methodology to produce breakdown on any other system configuration.
- What-if analysis for a set of optimizations.
- First work of its kind on an Arm-based server.

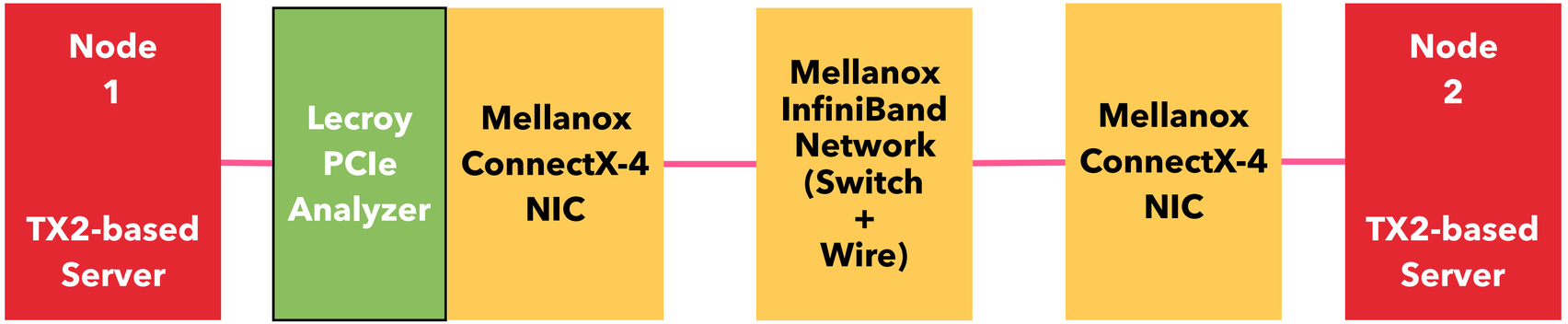
# OUTLINE

- Introduction
- Experimental setup & Measurement methodology
- Injection overhead: Modeling and breakdown
- Latency: Modeling and breakdown
- Simulated optimizations

# INTERNODE COMMUNICATION COMPONENTS IN HPC

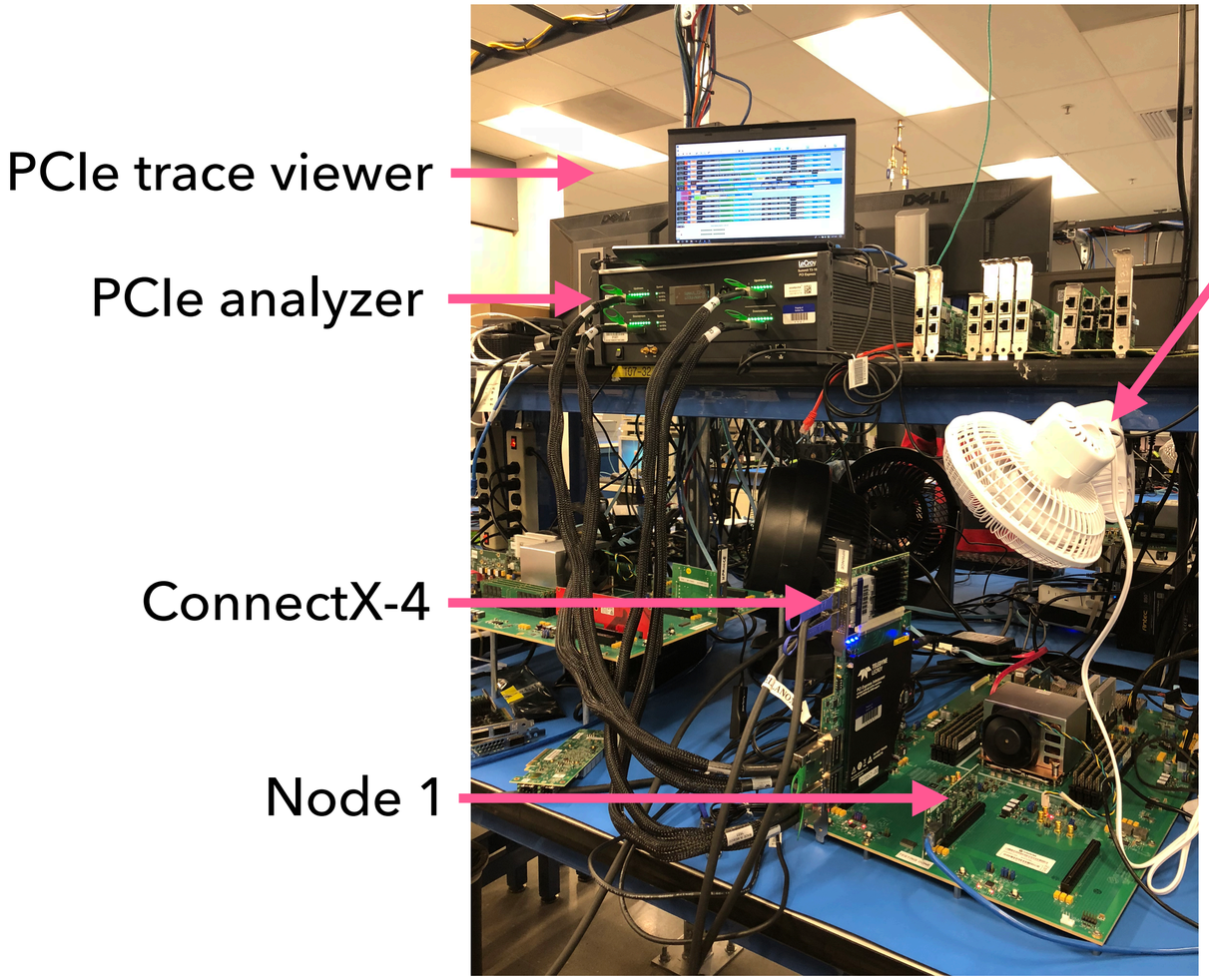


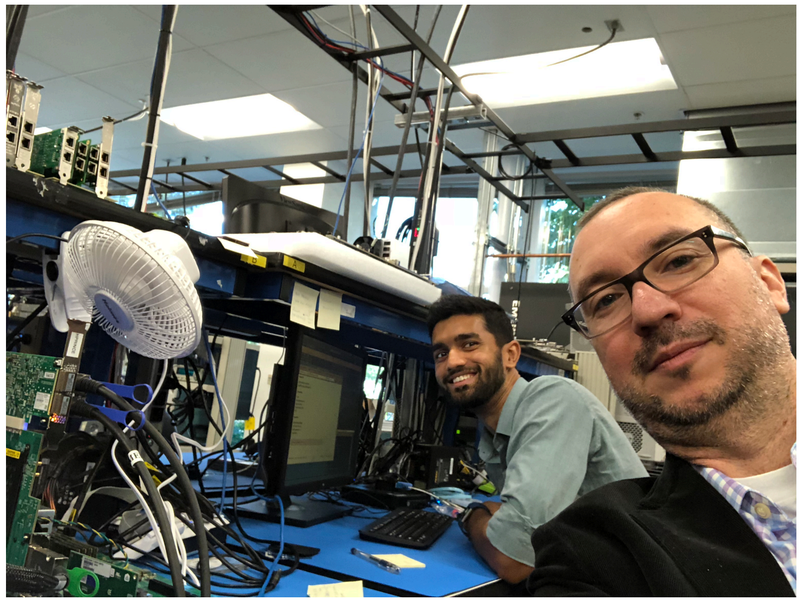
# EXPERIMENTAL SETUP



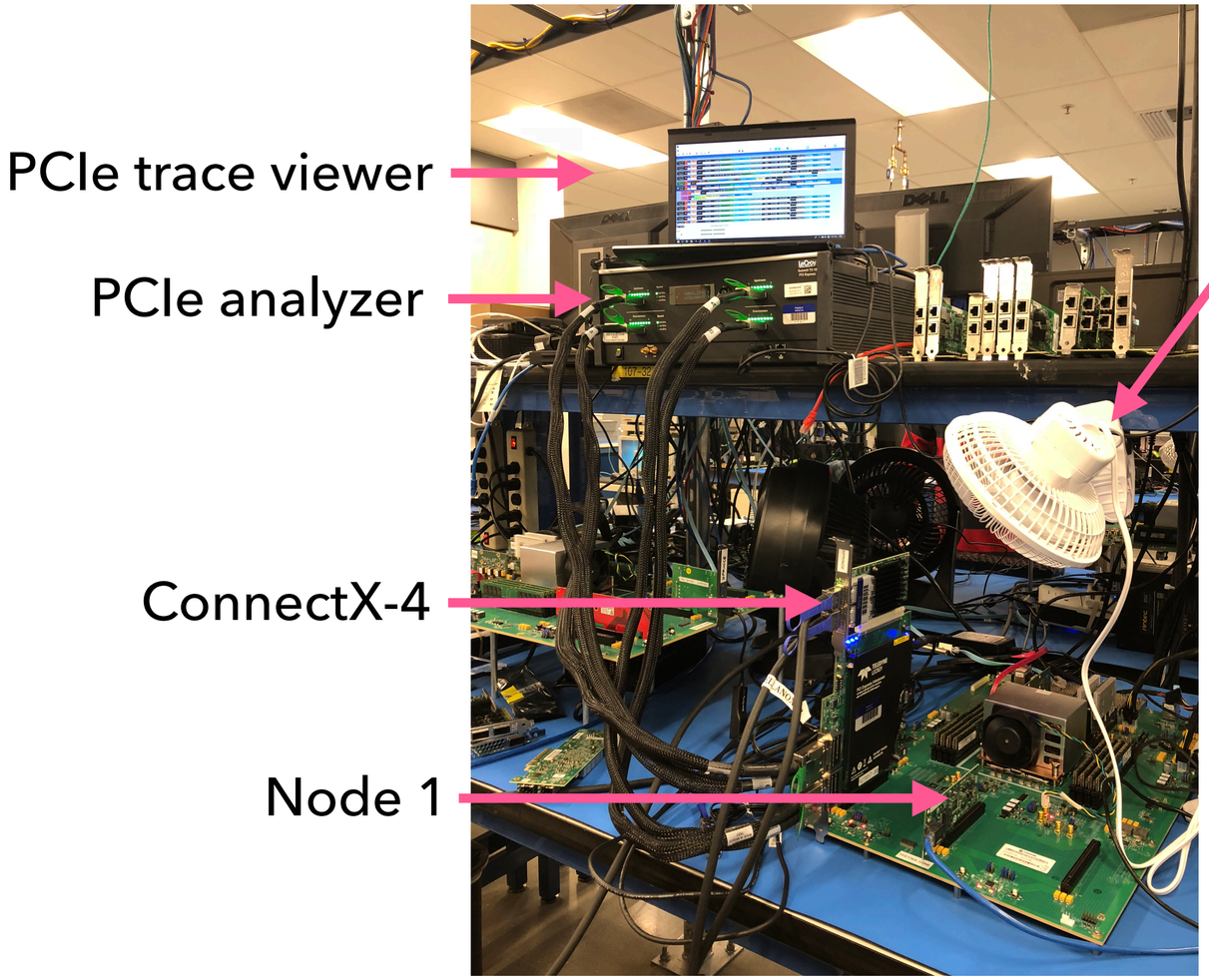

- Software: MPICH CH4 + UCX; Hardware: Arm TX2 + PCIe + Mellanox IB
- CPU timer registers to measure CPU time.
- PCIe analyzer to measure time in other components through traces.

# EXPERIMENTAL SETUP (WHAT IT ACTUALLY LOOKED LIKE)



# EXPERIMENTAL SETUP (WHAT IT ACTUALLY LOOKED LIKE)




State-of-the-art cooling

# USING CPU TIMERS

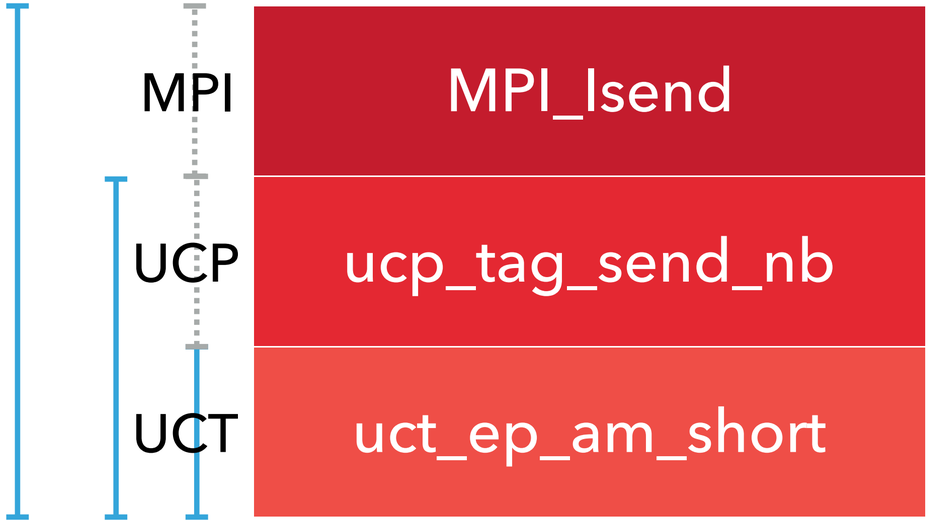
*Timer start*

```
<code>
<of>
<interest>
```

*Timer end*

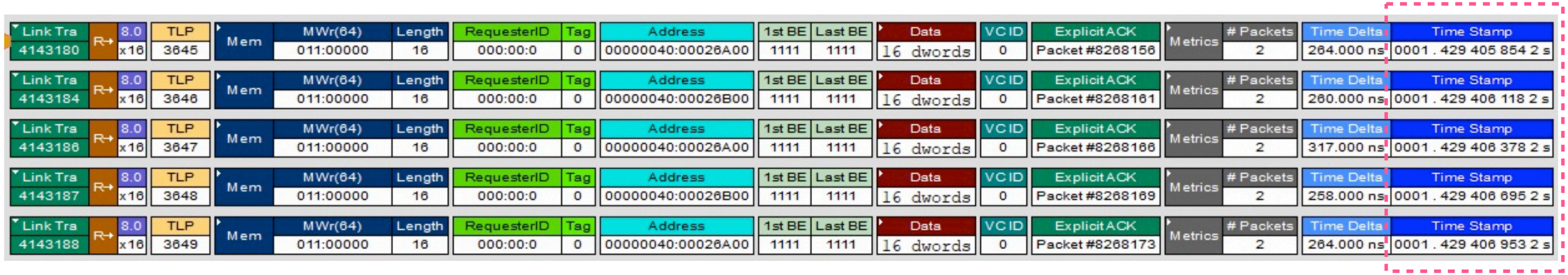
*Time for code of interest = Timer end - Timer start - Timer overhead*

# USING CPU TIMERS




- Measured time in different components using deltas.
- Carefully isolated callbacks/functions between layers (details in paper).

# USING PCIE ANALYZER



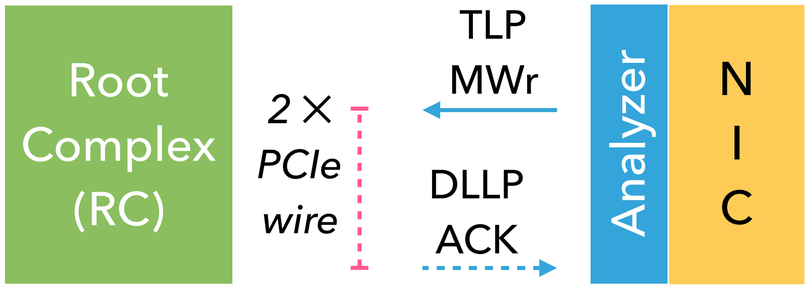
| Link Tra | | 8.0 | TLP | Mem | MWr(64) | Length | RequesterID | Tag | Address | 1st BE | Last BE | Data | VC ID | ExplicitACK | Metrics | # Packets | Time Delta | Time Stamp |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 4143180 | R→ | x16 | 3645 | | 011:00000 | 16 | 000:00:0 | 0 | 00000040:00026A00 | 1111 | 1111 | 16 dwords | 0 | Packet #8268156 | | 2 | 264.000 ns | 0001 . 429 405 854 2 s |
| 4143184 | R→ | x16 | 3646 | | 011:00000 | 16 | 000:00:0 | 0 | 00000040:00026B00 | 1111 | 1111 | 16 dwords | 0 | Packet #8268161 | | 2 | 260.000 ns | 0001 . 429 406 118 2 s |
| 4143186 | R→ | x16 | 3647 | | 011:00000 | 16 | 000:00:0 | 0 | 00000040:00026A00 | 1111 | 1111 | 16 dwords | 0 | Packet #8268166 | | 2 | 317.000 ns | 0001 . 429 406 378 2 s |
| 4143187 | R→ | x16 | 3648 | | 011:00000 | 16 | 000:00:0 | 0 | 00000040:00026B00 | 1111 | 1111 | 16 dwords | 0 | Packet #8268169 | | 2 | 258.000 ns | 0001 . 429 406 695 2 s |
| 4143188 | R→ | x16 | 3649 | | 011:00000 | 16 | 000:00:0 | 0 | 00000040:00026A00 | 1111 | 1111 | 16 dwords | 0 | Packet #8268173 | | 2 | 264.000 ns | 0001 . 429 406 953 2 s |

*Time of event = Timestamp of packet after event -*
*Timestamp of packet before event*

# USING PCIE ANALYZER

| Link Tra | | 8.0 | TLP | Mem | MWr(64) | Length | RequesterID | Tag | Address | 1st BE | Last BE | Data | VC ID | Explicit ACK | Metrics | # Packets | Time Delta | Time Stamp |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 4143180 | R→ | x16 | 3645 | | 011:00000 | 16 | 000:00:0 | 0 | 00000040:00026A00 | 1111 | 1111 | 16 dwords | 0 | Packet #8268156 | | 2 | 264.000 ns | 0001 . 429 405 854 2 s |
| Link Tra | | 8.0 | TLP | Mem | MWr(64) | Length | RequesterID | Tag | Address | 1st BE | Last BE | Data | VC ID | Explicit ACK | Metrics | # Packets | Time Delta | Time Stamp |
| 4143184 | R→ | x16 | 3646 | | 011:00000 | 16 | 000:00:0 | 0 | 00000040:00026B00 | 1111 | 1111 | 16 dwords | 0 | Packet #8268161 | | 2 | 260.000 ns | 0001 . 429 406 118 2 s |
| Link Tra | | 8.0 | TLP | Mem | MWr(64) | Length | RequesterID | Tag | Address | 1st BE | Last BE | Data | VC ID | Explicit ACK | Metrics | # Packets | Time Delta | Time Stamp |
| 4143186 | R→ | x16 | 3647 | | 011:00000 | 16 | 000:00:0 | 0 | 00000040:00026A00 | 1111 | 1111 | 16 dwords | 0 | Packet #8268166 | | 2 | 317.000 ns | 0001 . 429 406 378 2 s |
| Link Tra | | 8.0 | TLP | Mem | MWr(64) | Length | RequesterID | Tag | Address | 1st BE | Last BE | Data | VC ID | Explicit ACK | Metrics | # Packets | Time Delta | Time Stamp |
| 4143187 | R→ | x16 | 3648 | | 011:00000 | 16 | 000:00:0 | 0 | 00000040:00026B00 | 1111 | 1111 | 16 dwords | 0 | Packet #8268169 | | 2 | 258.000 ns | 0001 . 429 406 695 2 s |
| Link Tra | | 8.0 | TLP | Mem | MWr(64) | Length | RequesterID | Tag | Address | 1st BE | Last BE | Data | VC ID | Explicit ACK | Metrics | # Packets | Time Delta | Time Stamp |
| 4143188 | R→ | x16 | 3649 | | 011:00000 | 16 | 000:00:0 | 0 | 00000040:00026A00 | 1111 | 1111 | 16 dwords | 0 | Packet #8268173 | | 2 | 264.000 ns | 0001 . 429 406 953 2 s |

## NIC WRITING COMPLETION

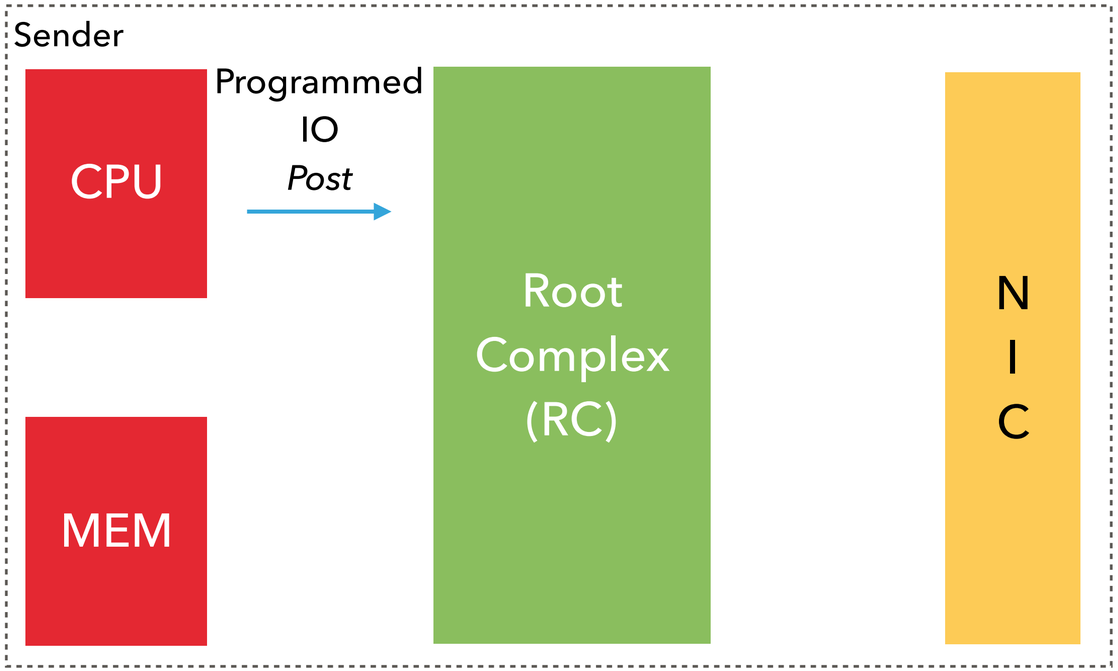


# OUTLINE

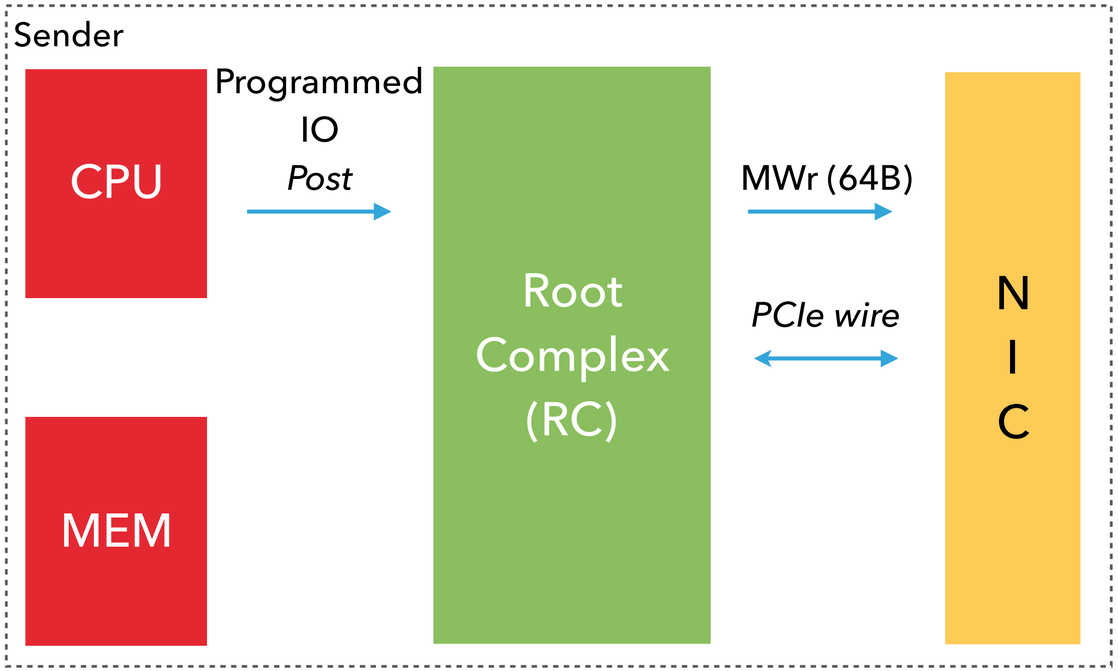
- Introduction
- Experimental setup & Measurement methodology
- Injection overhead: Modeling and breakdown
- Latency: Modeling and breakdown
- Simulated optimizations

# INJECTION OVERHEAD

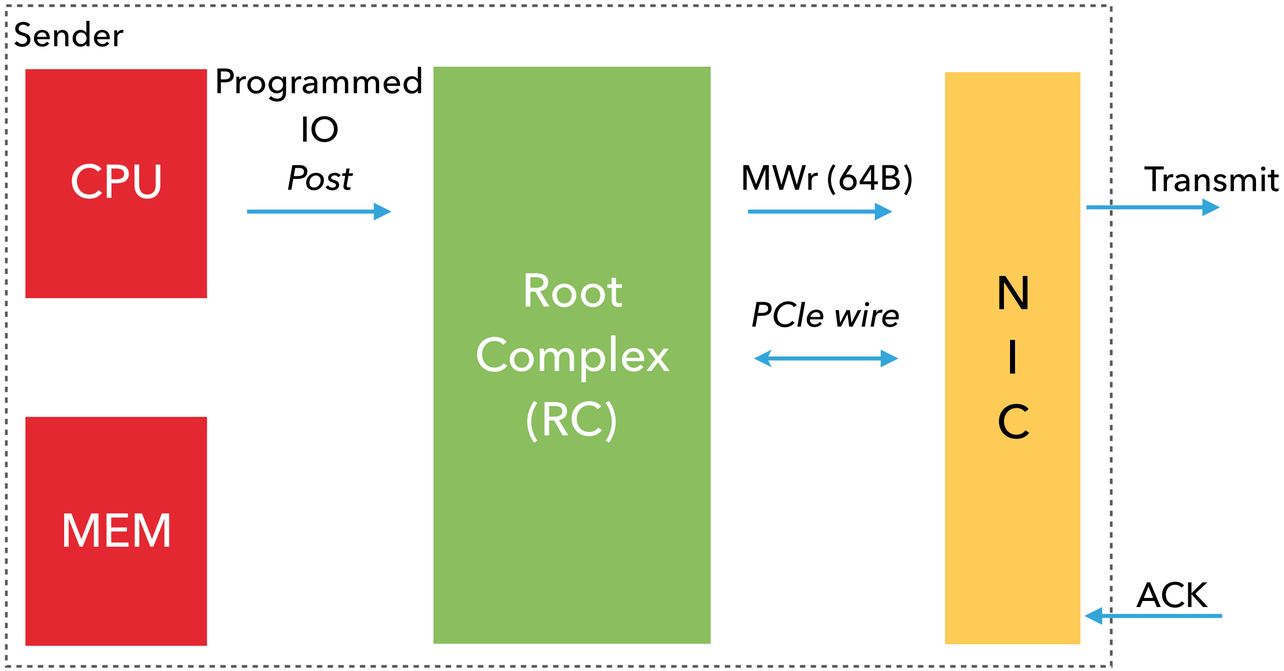
# INJECTION OVERHEAD: BACKGROUND



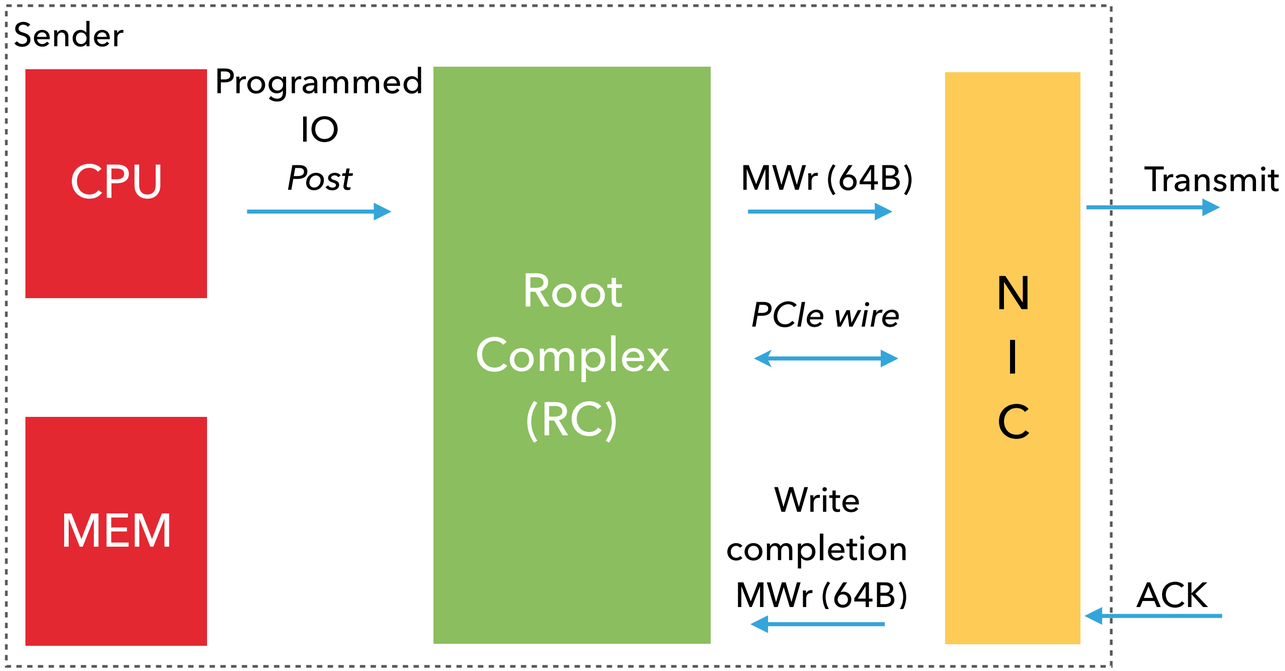
# INJECTION OVERHEAD: BACKGROUND



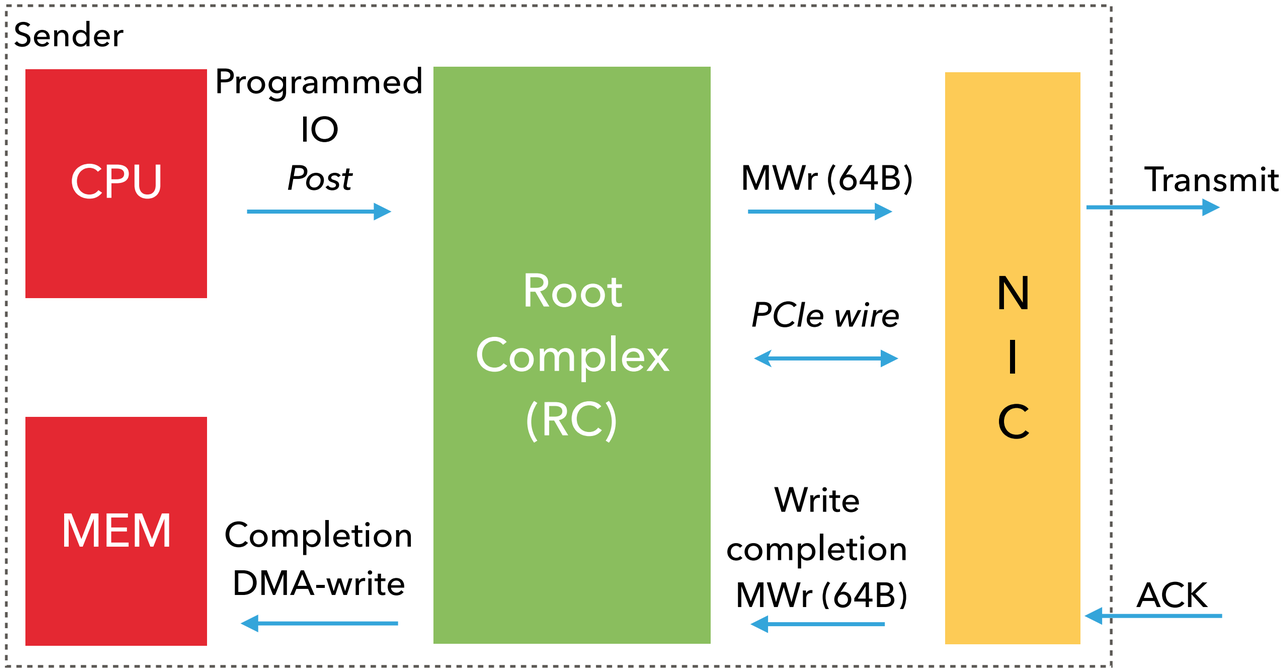
# INJECTION OVERHEAD: BACKGROUND



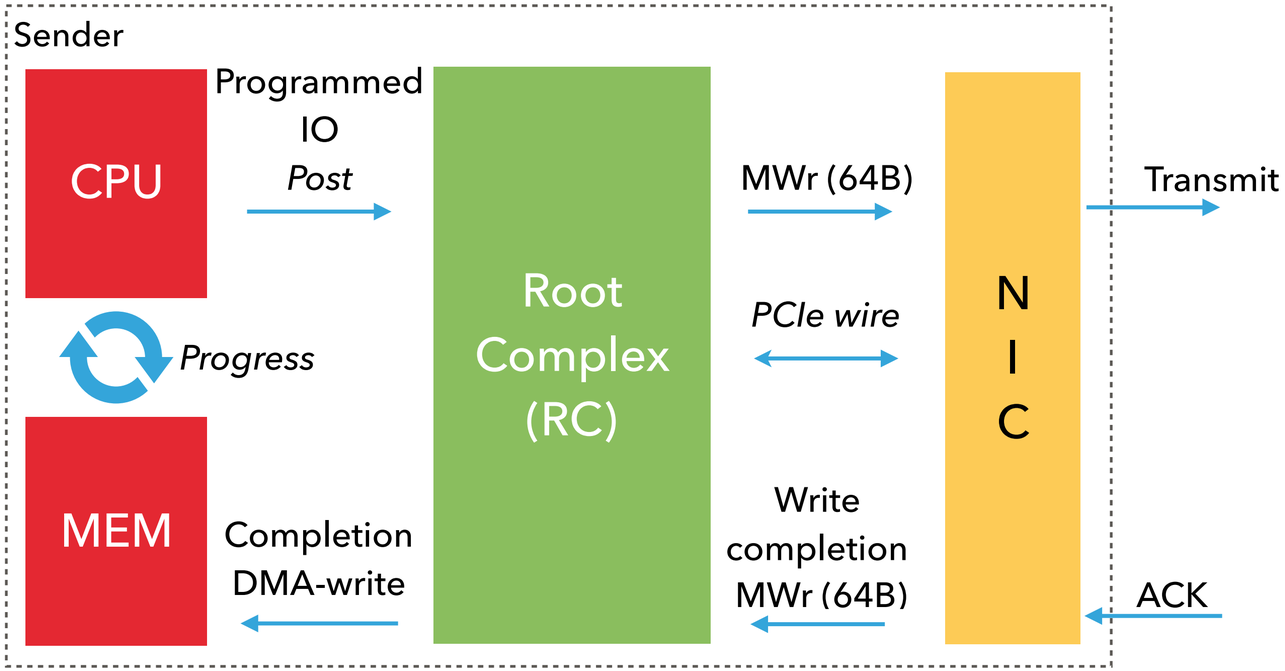
# INJECTION OVERHEAD: BACKGROUND



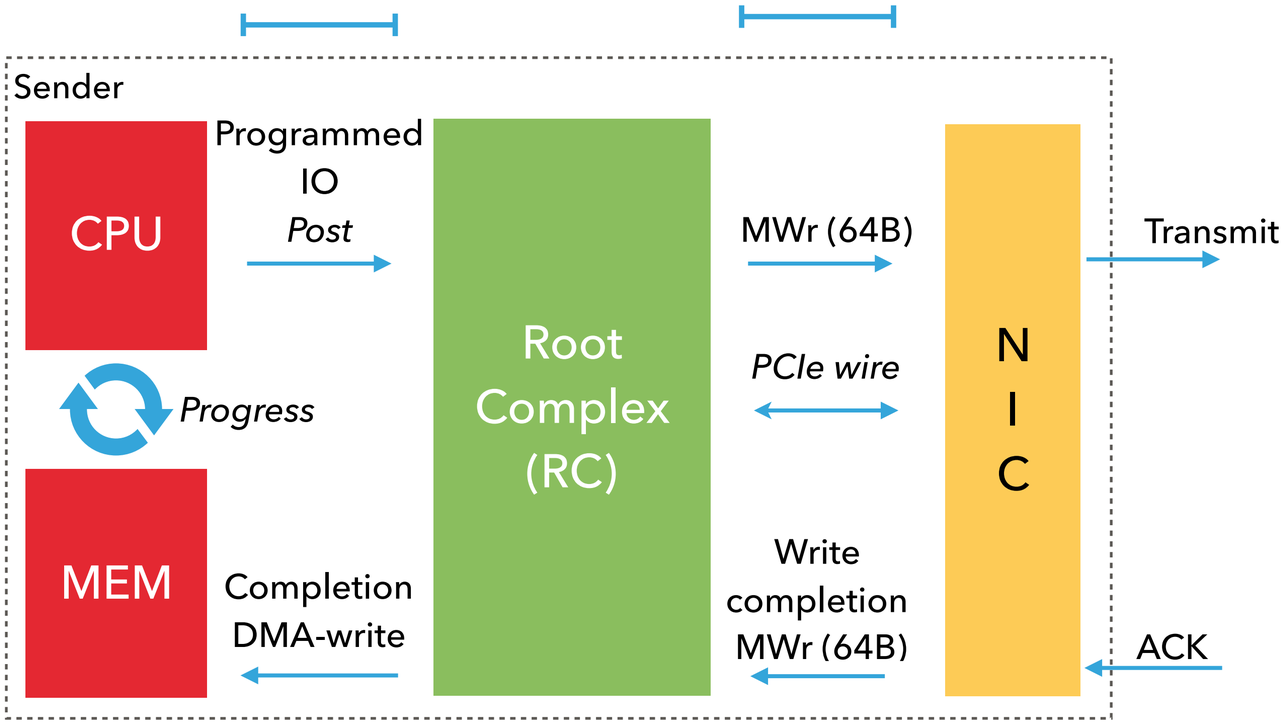
# INJECTION OVERHEAD: BACKGROUND



# INJECTION OVERHEAD: BACKGROUND

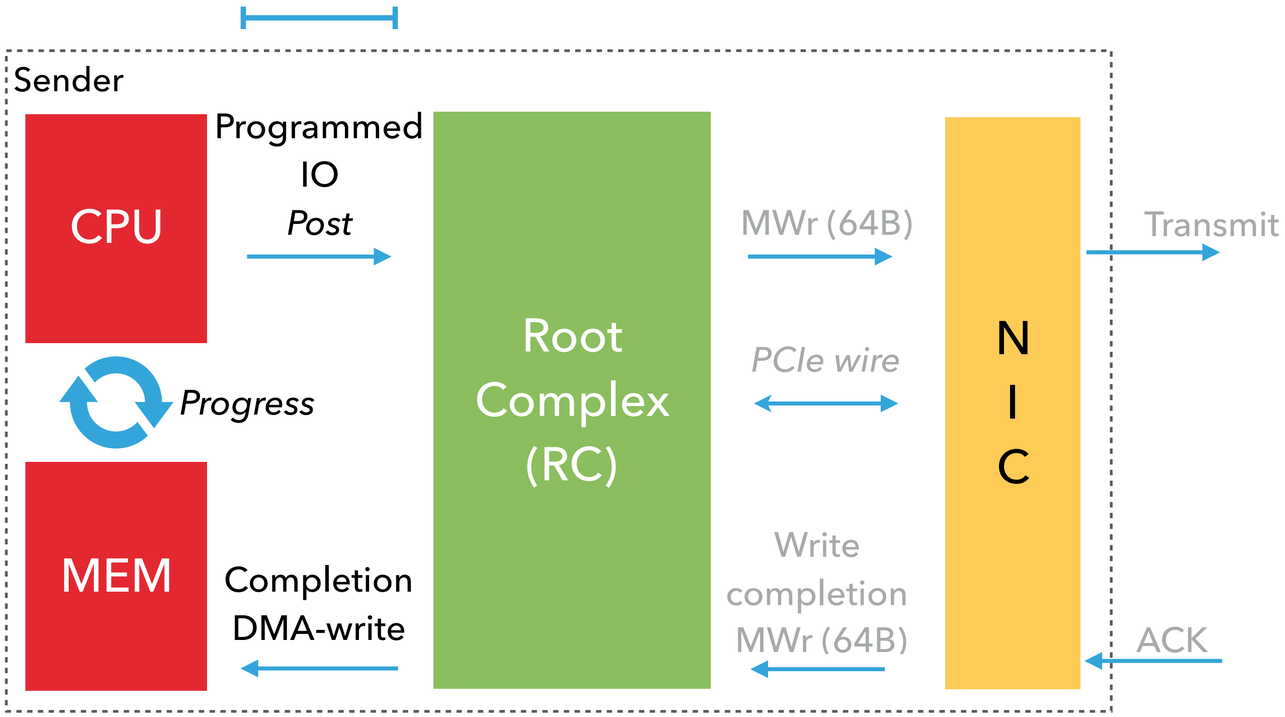


# INJECTION OVERHEAD




- Overhead observed by RC
- Overhead observed by NIC

# INJECTION OVERHEAD



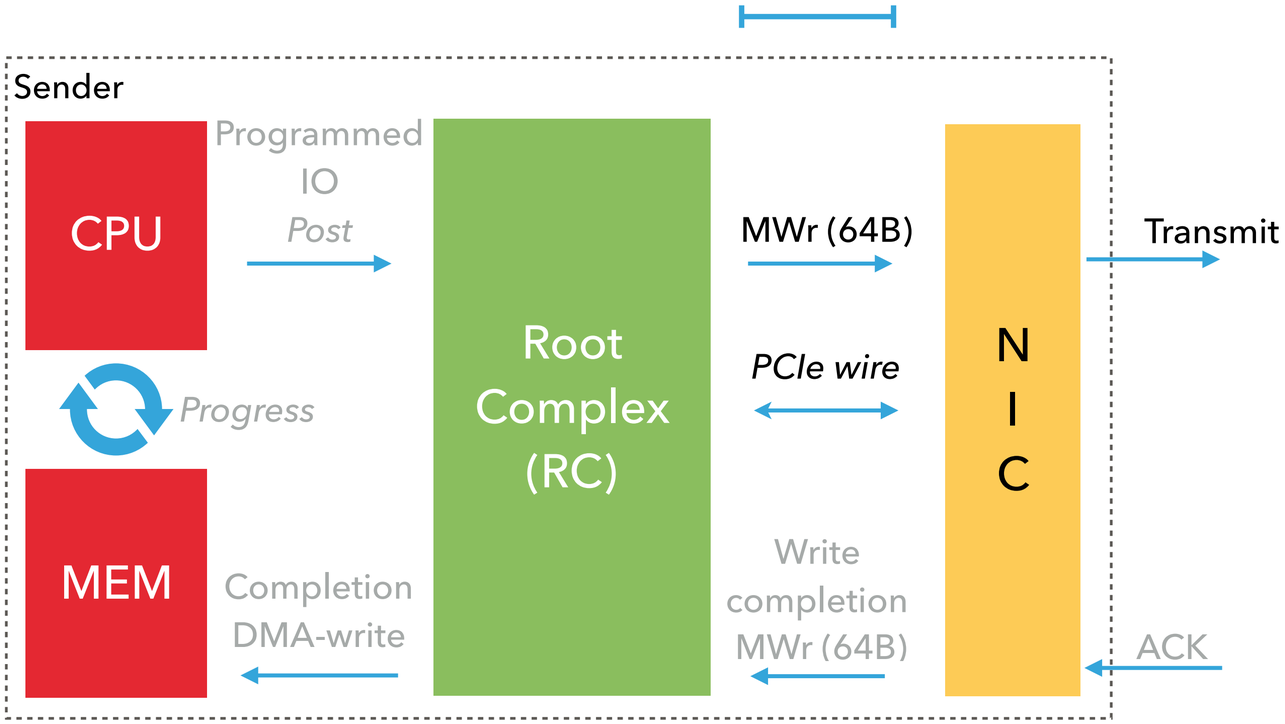

- Overhead observed by RC

$$\frac{b \times Post + b \times Progress + tot\_Misc}{b}$$

$$= CPU\_time = Post + Progress + Misc$$

- Overhead observed by NIC

# INJECTION OVERHEAD




- Overhead observed by RC

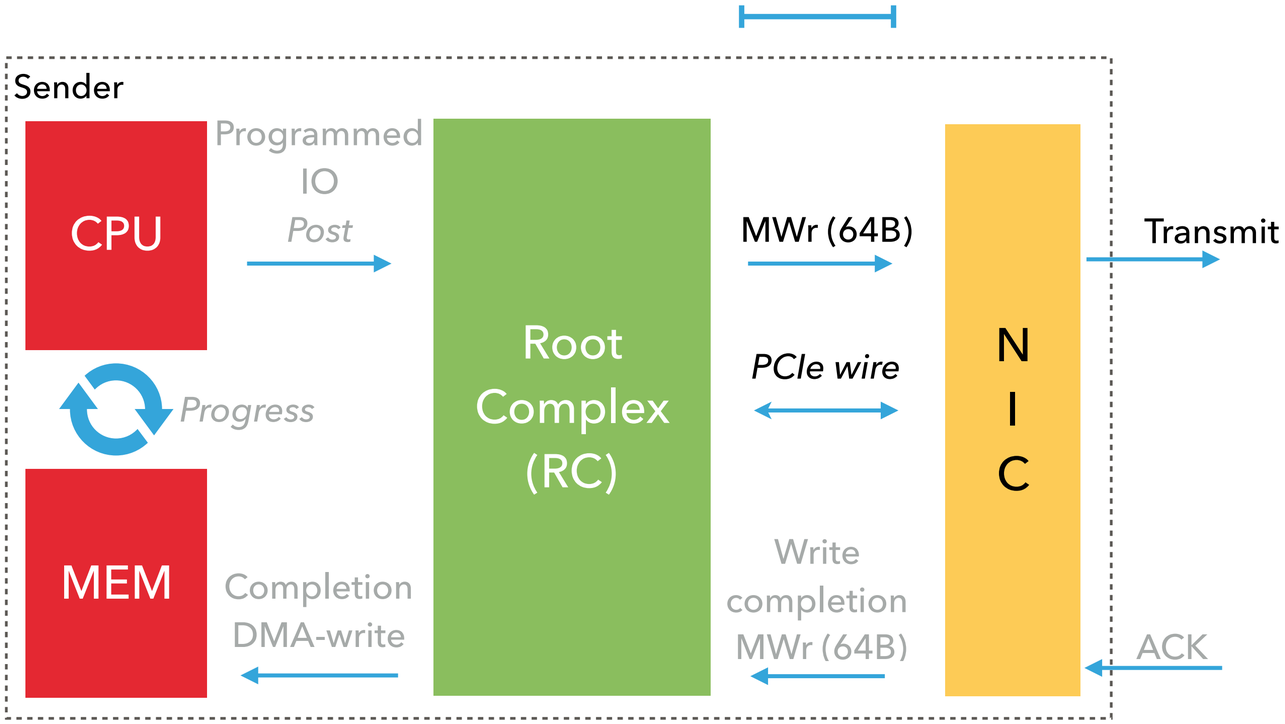
$$\frac{b \times Post + b \times Progress + tot\_Misc}{b}$$

$$= CPU\_time = Post + Progress + Misc$$

- Overhead observed by NIC

(1) Credit-based flow control
(2) Multiple outstanding PCIe transactions

# INJECTION OVERHEAD




- Overhead observed by RC

$$\frac{b \times Post + b \times Progress + tot\_Misc}{b}$$

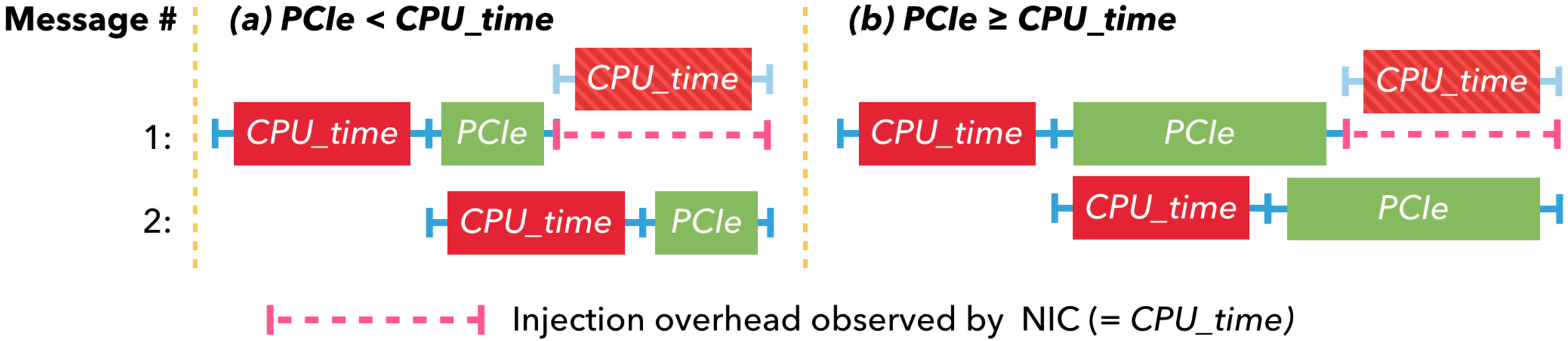
$$= CPU\_time = \boxed{Post + Progress + Misc}$$

- Overhead observed by NIC

= Overhead observed by RC

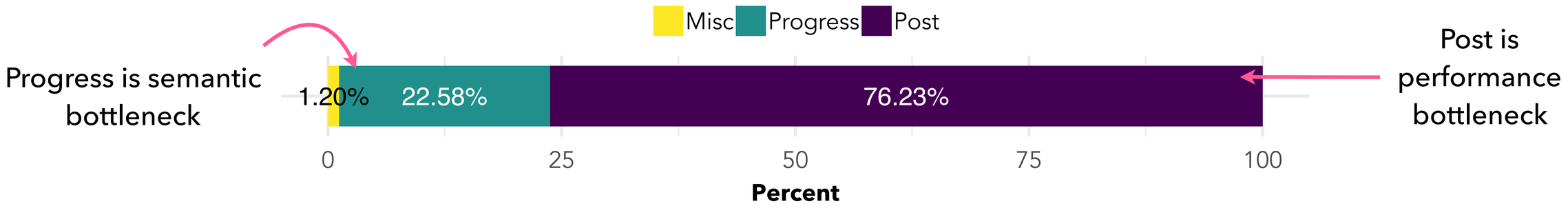
(1) Credit-based flow control
(2) Multiple outstanding PCIe transactions

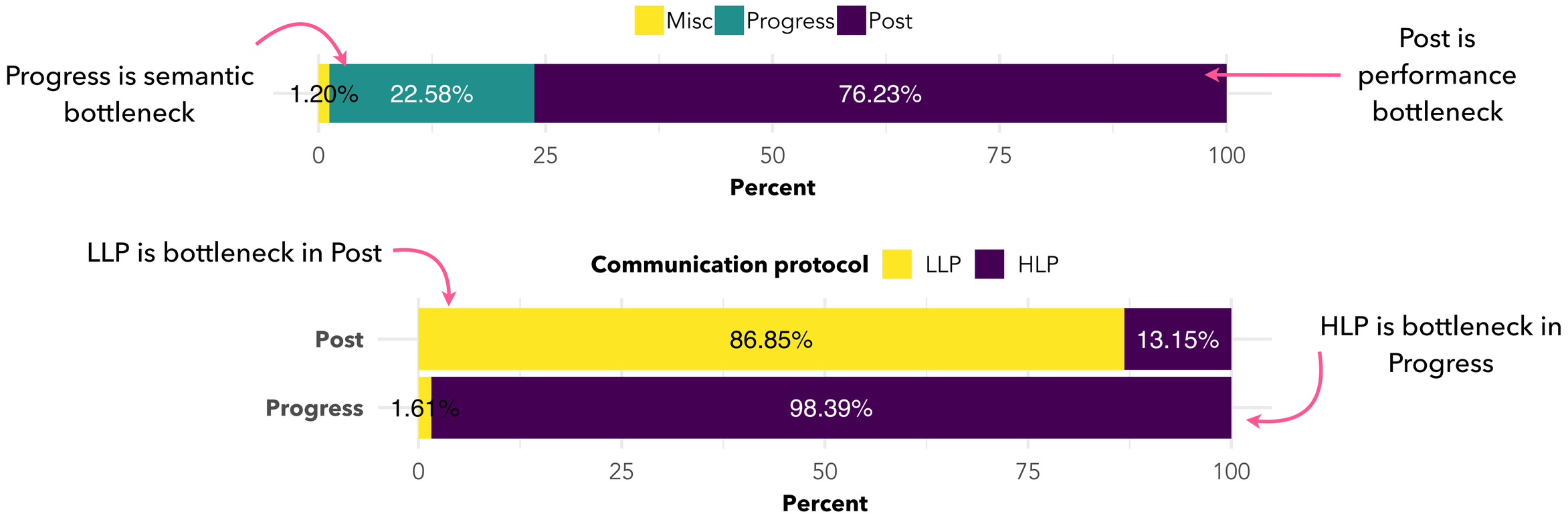
# INJECTION OVERHEAD

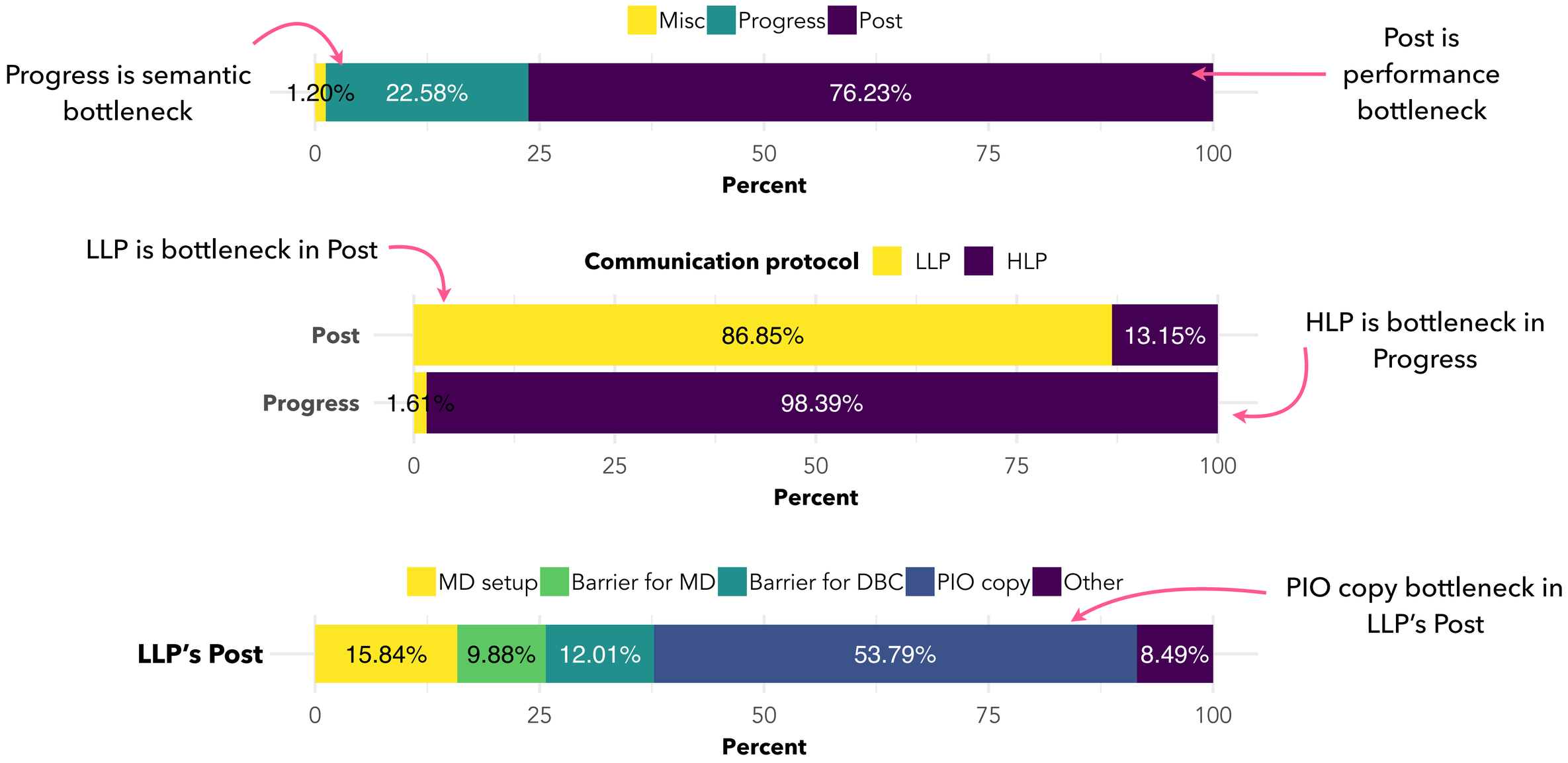



$$Injection\ overhead = CPU\_time = Post + Progress + Misc$$

CPU timer registers


Misc
Progress
Post
Progress is semantic bottleneck
1.20%
22.58%
76.23%
Post is performance bottleneck
0
25
50
75
100
Percent

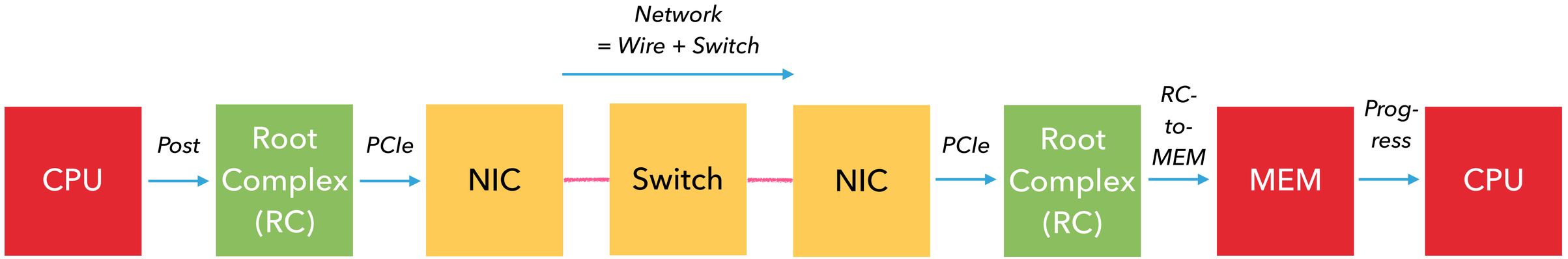

Misc
Progress
Post
Progress is semantic bottleneck
1.20%
22.58%
76.23%
Post is performance bottleneck
0
25
50
75
100
Percent
LLP is bottleneck in Post
Communication protocol
LLP
HLP
Post
86.85%
13.15%
Progress
1.61%
98.39%
HLP is bottleneck in Progress
0
25
50
75
100
Percent


Misc
Progress
Post
Progress is semantic bottleneck
1.20%
22.58%
76.23%
Post is performance bottleneck
0
25
50
75
100
Percent
LLP is bottleneck in Post
Communication protocol
LLP
HLP
Post
86.85%
13.15%
Progress
1.61%
98.39%
HLP is bottleneck in Progress
0
25
50
75
100
Percent
MD setup
Barrier for MD
Barrier for DBC
PIO copy
Other
PIO copy bottleneck in LLP's Post
LLP's Post
15.84%
9.88%
12.01%
53.79%
8.49%
0
25
50
75
100
Percent

# OUTLINE

- Introduction
- Experimental setup & Measurement methodology
- Injection overhead: Modeling and breakdown
- Latency: Modeling and breakdown
- Breakdown
- Simulated optimizations

# LATENCY OVERHEAD: MODELING



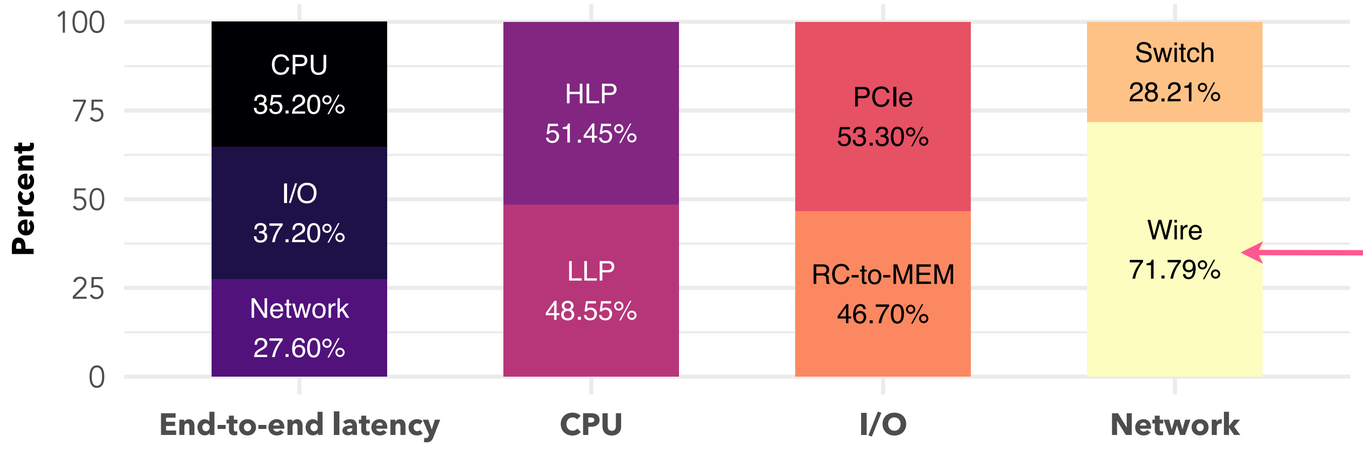

*Latency = Post + 2 (PCIe) + Network + RC-to-MEM + Progress*

CPU timer-registers

System measurements

CPU timer-registers

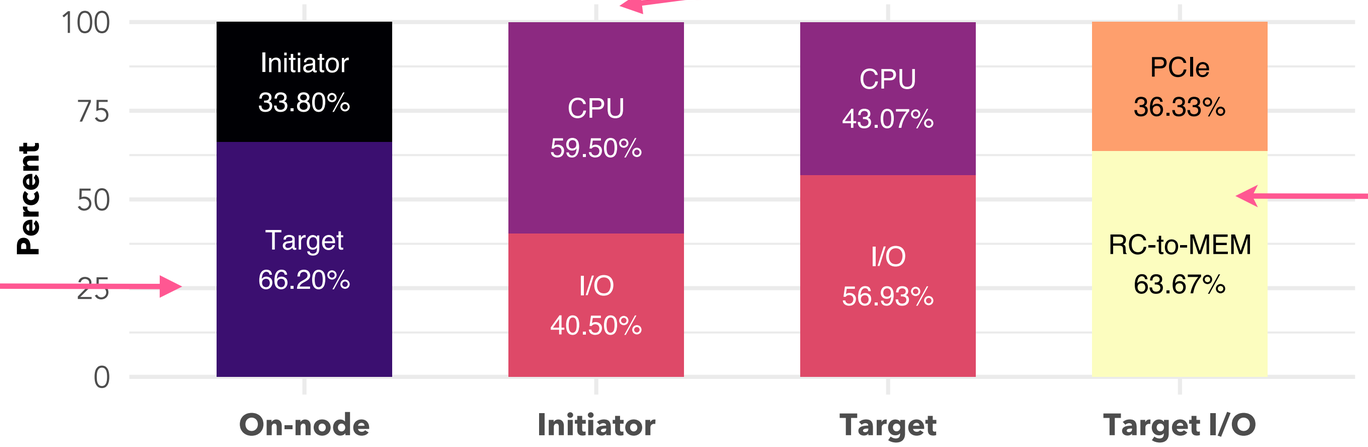
On-node time dominates latency




Wire dominates off-node time

CPU is the majority on initiator due to PIO

Target dominates on-node time



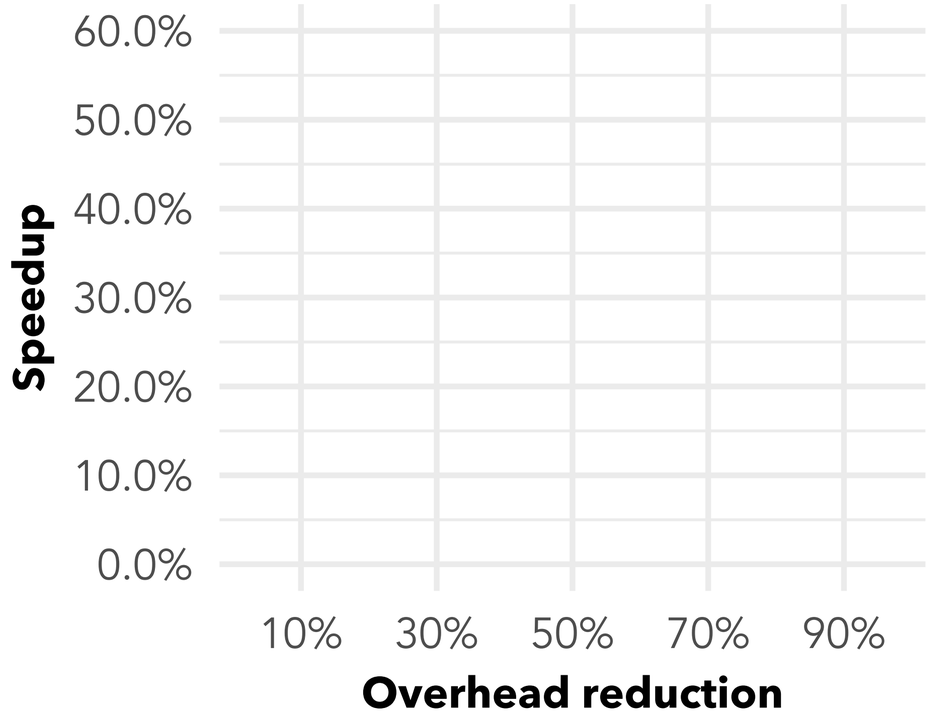

I/O is the majority on target due to RC-to-MEM

# OUTLINE

- Introduction
- Experimental setup & Measurement methodology
- Injection overhead: Modeling and breakdown
- Latency: Modeling and breakdown
- Simulated optimizations

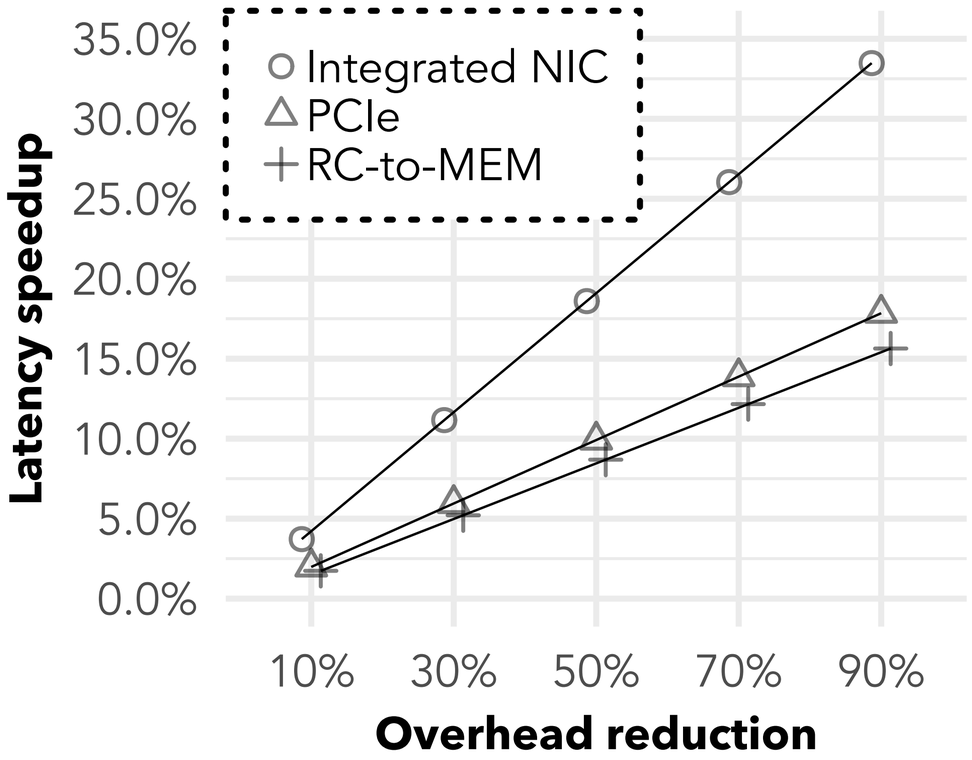
# SIMULATED OPTIMIZATIONS

- *If we optimize component X by Y%, what is the corresponding speedup in latency and injection overhead?*



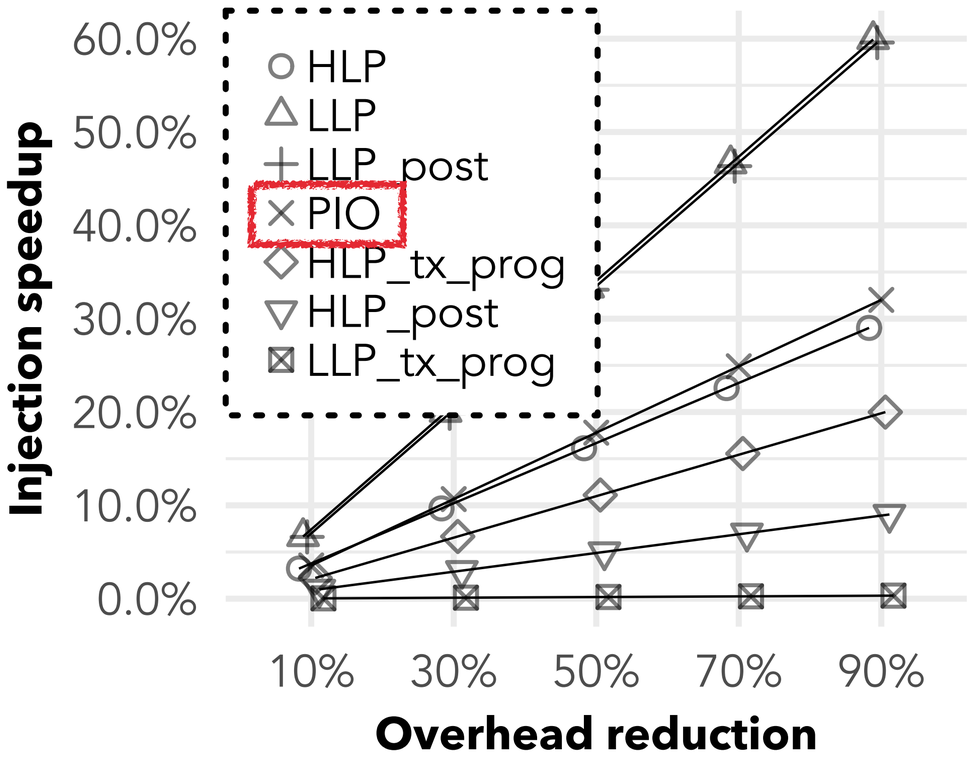
# NIC INTEGRATED ON CHIP

- Would eliminate most of I/O.
- Would make the CPU more available.
- Likelihood: Likely to become commonplace
  - Modest 50% reduction can speedup latency by 15%



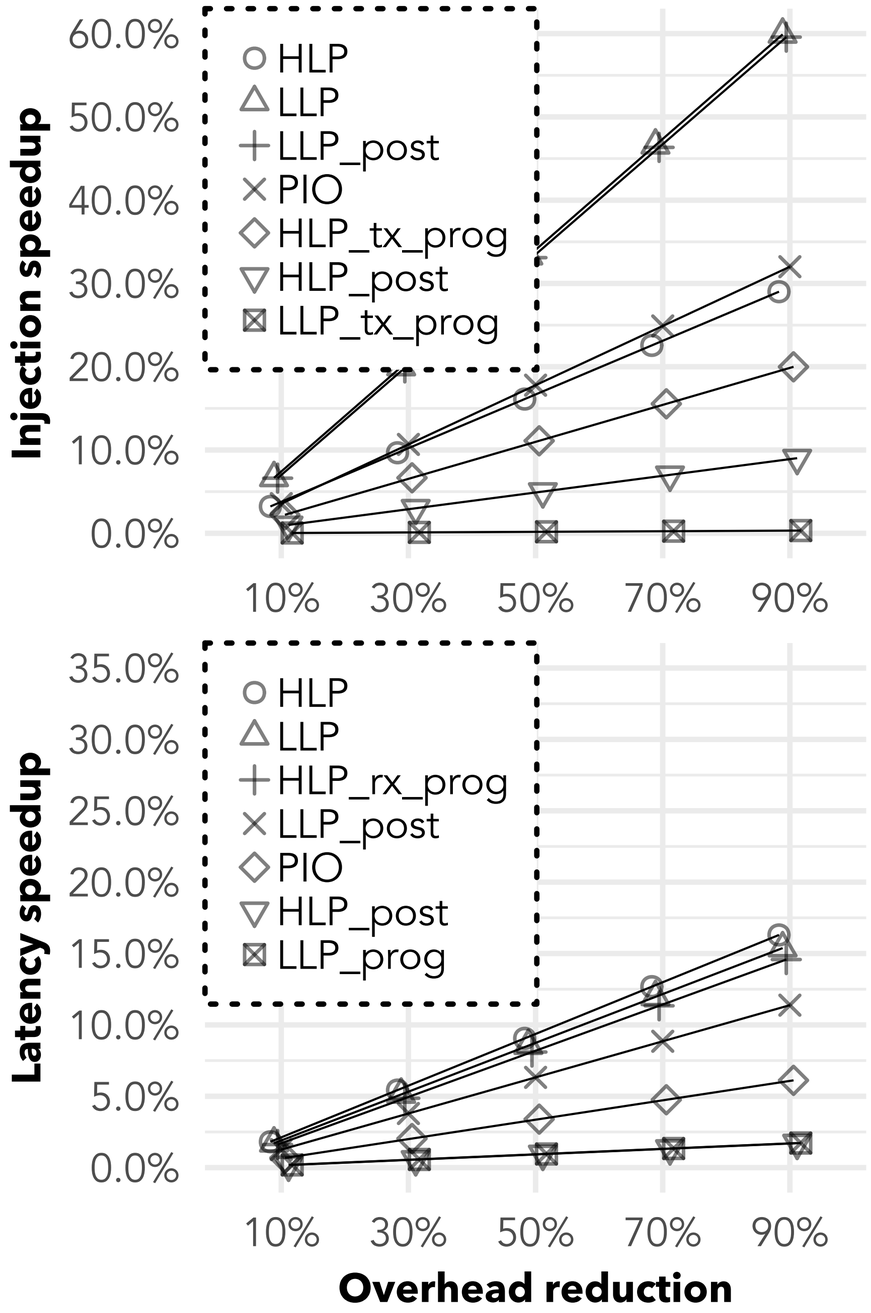
# FASTER LLP

- Microarchitectural improvements for writes to device memory most impactful.
- Likelihood: Likely since there seems to be room for improvement
  - PIO reduction to 15ns (84% reduction) can speedup injection by 25%



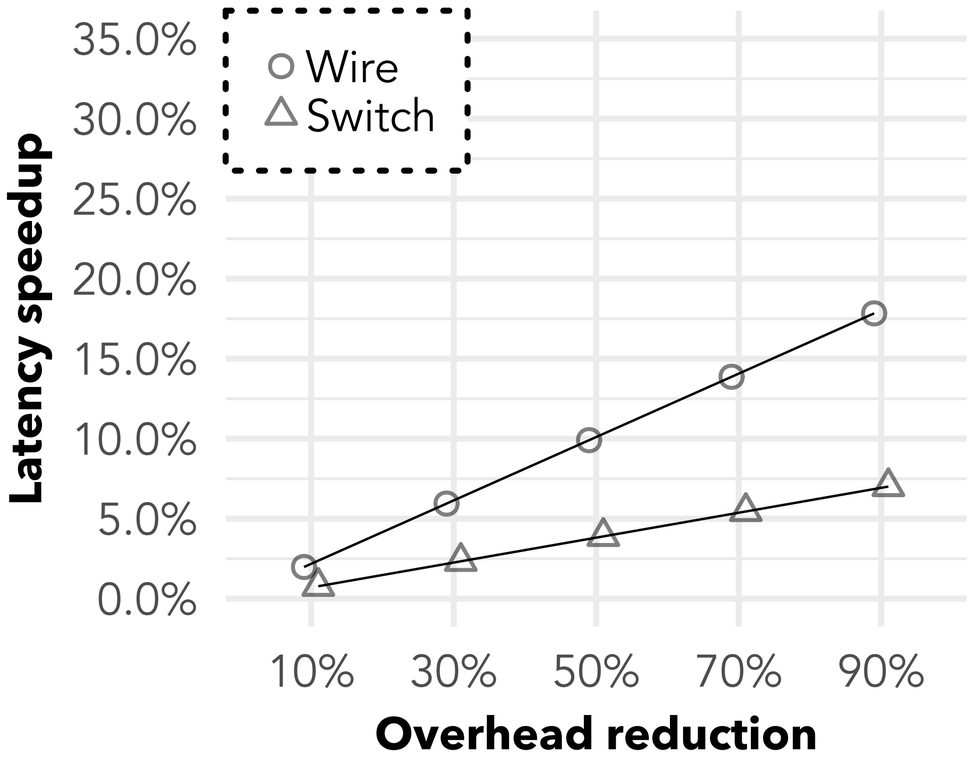
# HLP SOFTWARE IMPROVEMENTS

- HLP progress improvements would be closest to upper bounds.
- Likelihood: Overhead reductions not likely more than 20%.
  - Less than 5% latency speedup
  - 6.44% injection speedup



# NETWORK IMPROVEMENTS

- Likelihood: Further overhead reductions unlikely.
  - Wire latencies expected to increase.
  - Gen-Z switch overheads yet to be demonstrated.



# SUMMARY

- Our models explain observed performance within 5% margin of error.
- Breakdown explains where, why, and how much time is spent, providing key insights.
- Breakdown would help researchers guide their optimization efforts.

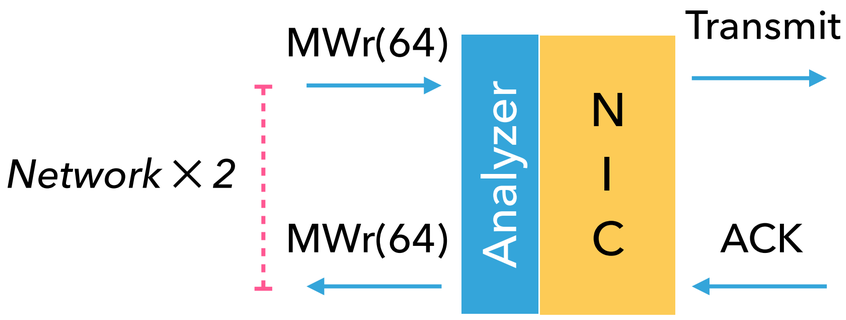
"To measure is to know." – Lord Kelvin

Special thanks to
Giri Chukkapalli, and Ham Prince from Marvell Technology Group,
Yossi Itigin from Mellanox Technologies, and
Pavan Balaji from Argonne National Laboratory

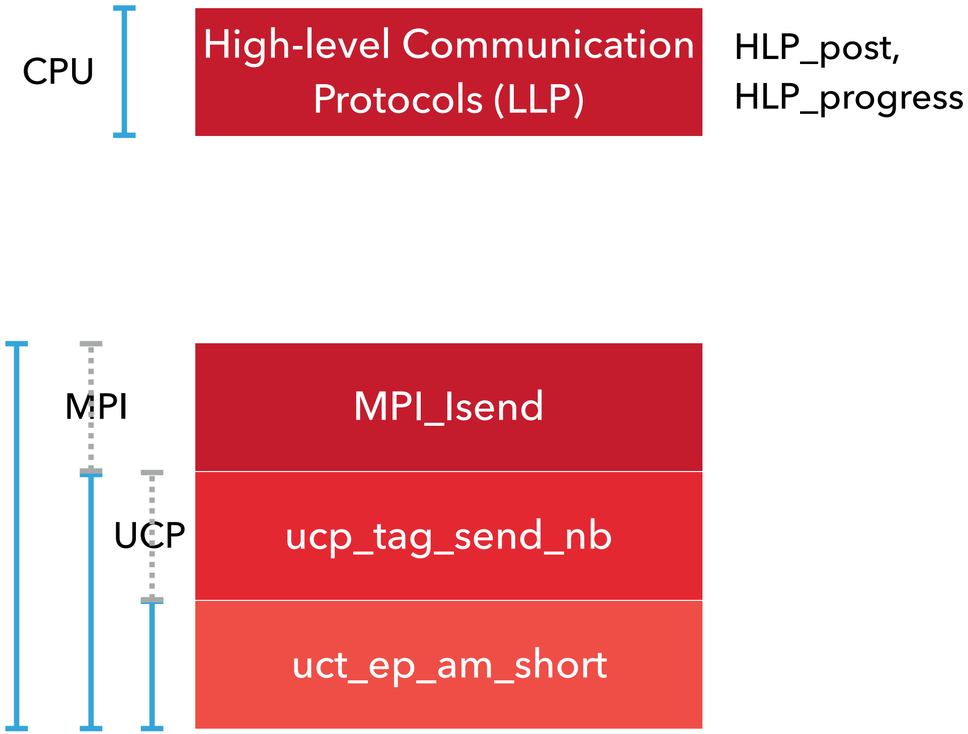
# USING PCIE ANALYZER

| Link Tra | | 8.0 | TLP | Mem | MWr(64) | Length | RequesterID | Tag | Address | 1st BE | Last BE | Data | VC ID | Explicit ACK | Metrics | # Packets | Time Delta | Time Stamp |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 4143180 | R↦ | x16 | 3645 | | 011:00000 | 16 | 000:00:0 | 0 | 00000040:00026A00 | 1111 | 1111 | 16 dwords | 0 | Packet #8268156 | | 2 | 264.000 ns | 0001 . 429 405 854 2 s |
| Link Tra | | 8.0 | TLP | Mem | MWr(64) | Length | RequesterID | Tag | Address | 1st BE | Last BE | Data | VC ID | Explicit ACK | Metrics | # Packets | Time Delta | Time Stamp |
| 4143184 | R↦ | x16 | 3646 | | 011:00000 | 16 | 000:00:0 | 0 | 00000040:00026B00 | 1111 | 1111 | 16 dwords | 0 | Packet #8268161 | | 2 | 260.000 ns | 0001 . 429 406 118 2 s |
| Link Tra | | 8.0 | TLP | Mem | MWr(64) | Length | RequesterID | Tag | Address | 1st BE | Last BE | Data | VC ID | Explicit ACK | Metrics | # Packets | Time Delta | Time Stamp |
| 4143186 | R↦ | x16 | 3647 | | 011:00000 | 16 | 000:00:0 | 0 | 00000040:00026A00 | 1111 | 1111 | 16 dwords | 0 | Packet #8268166 | | 2 | 317.000 ns | 0001 . 429 406 378 2 s |
| Link Tra | | 8.0 | TLP | Mem | MWr(64) | Length | RequesterID | Tag | Address | 1st BE | Last BE | Data | VC ID | Explicit ACK | Metrics | # Packets | Time Delta | Time Stamp |
| 4143187 | R↦ | x16 | 3648 | | 011:00000 | 16 | 000:00:0 | 0 | 00000040:00026B00 | 1111 | 1111 | 16 dwords | 0 | Packet #8268169 | | 2 | 258.000 ns | 0001 . 429 406 695 2 s |
| Link Tra | | 8.0 | TLP | Mem | MWr(64) | Length | RequesterID | Tag | Address | 1st BE | Last BE | Data | VC ID | Explicit ACK | Metrics | # Packets | Time Delta | Time Stamp |
| 4143188 | R↦ | x16 | 3649 | | 011:00000 | 16 | 000:00:0 | 0 | 00000040:00026A00 | 1111 | 1111 | 16 dwords | 0 | Packet #8268173 | | 2 | 264.000 ns | 0001 . 429 406 953 2 s |

## NIC RECEIVING ACK FROM TARGET NIC



# BREAKDOWN OF THE HIGHER LEVEL




- Measured initiation components using deltas.

# RELEVANT RESEARCH

- Most of the prior research tackle one component and show effect on overall performance.
  - Papadopoulou et al., Raffeneti et al. show instruction breakdown on UCX and MPICH respectively.
  - Ajima et al. show breakdown of RDMA-write latency on post-K using simulation waveforms.