

IPC-Direct: Fast and Compatible Inter-Process Communication in User Space

SOSP'17 SRC #39

Microsoft Research

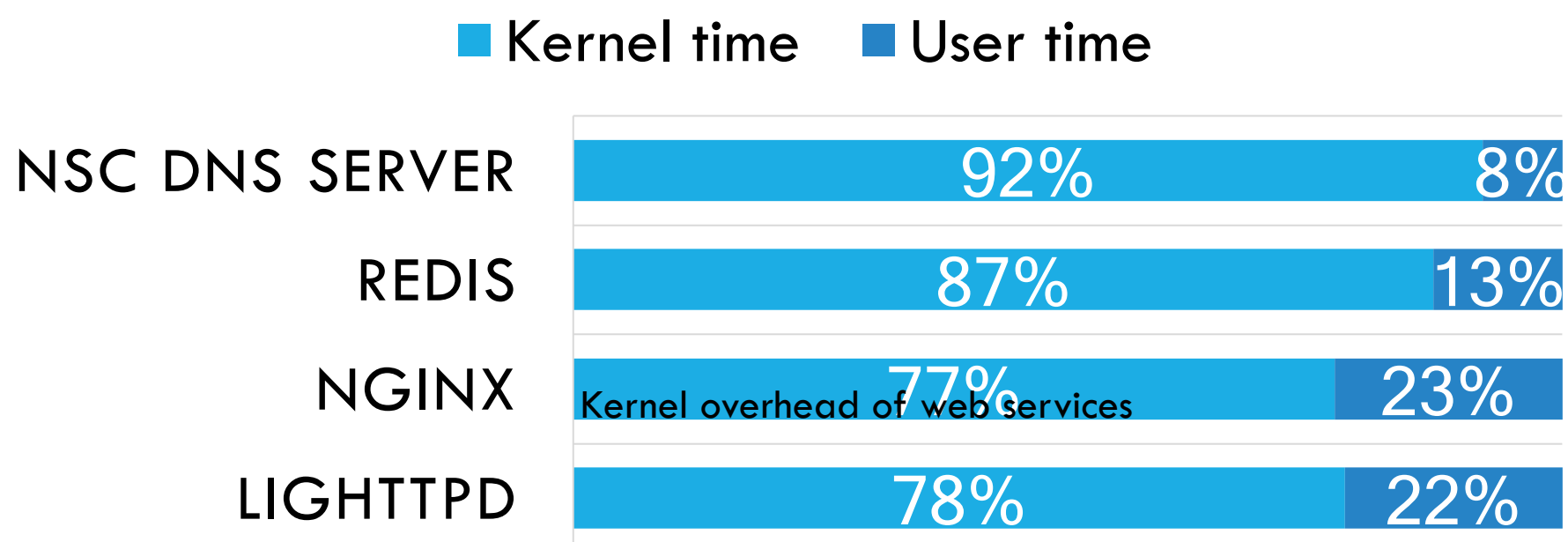


Bojie Li, Tianyi Cui, Zibo Wang, Lintao Zhang

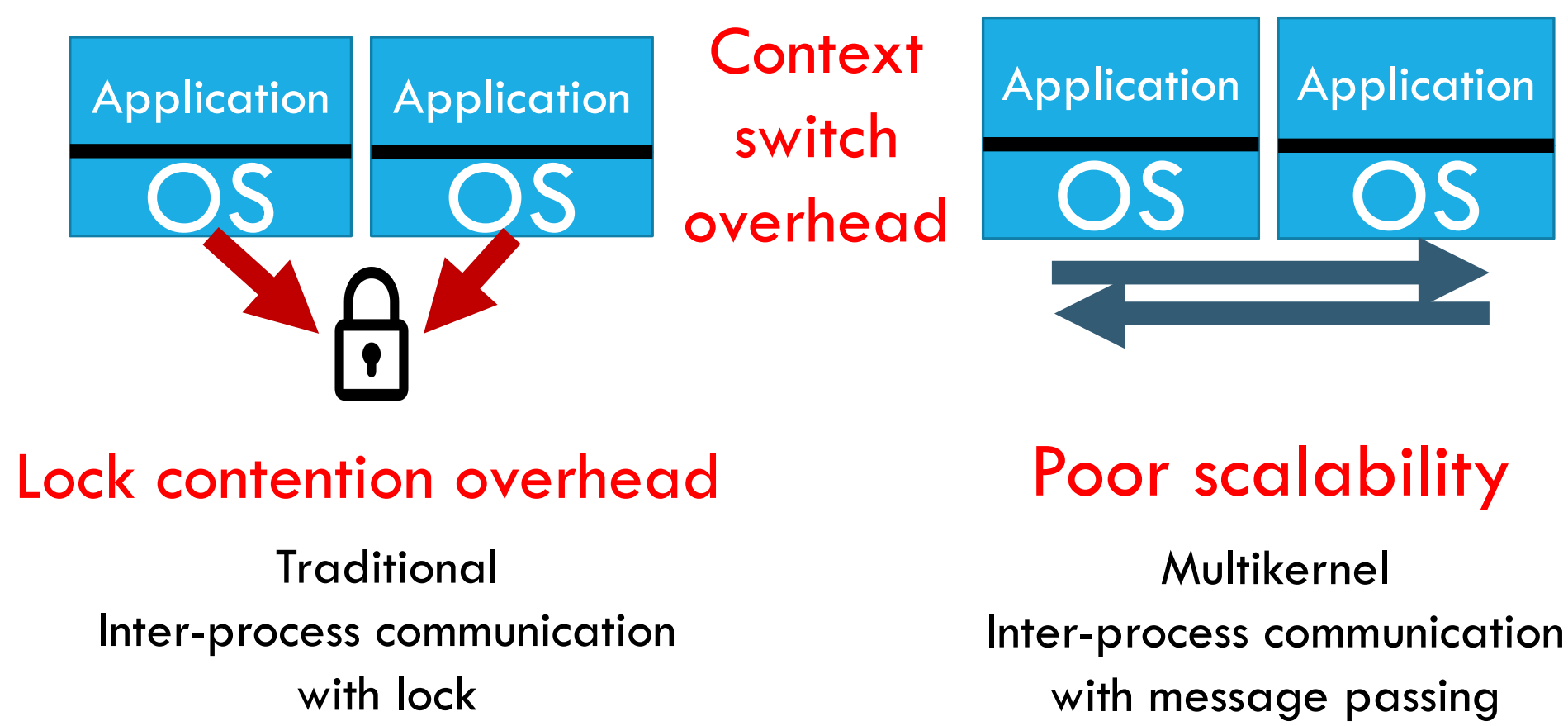
Kernel overhead dominates web service performance

sources of kernel overhead:

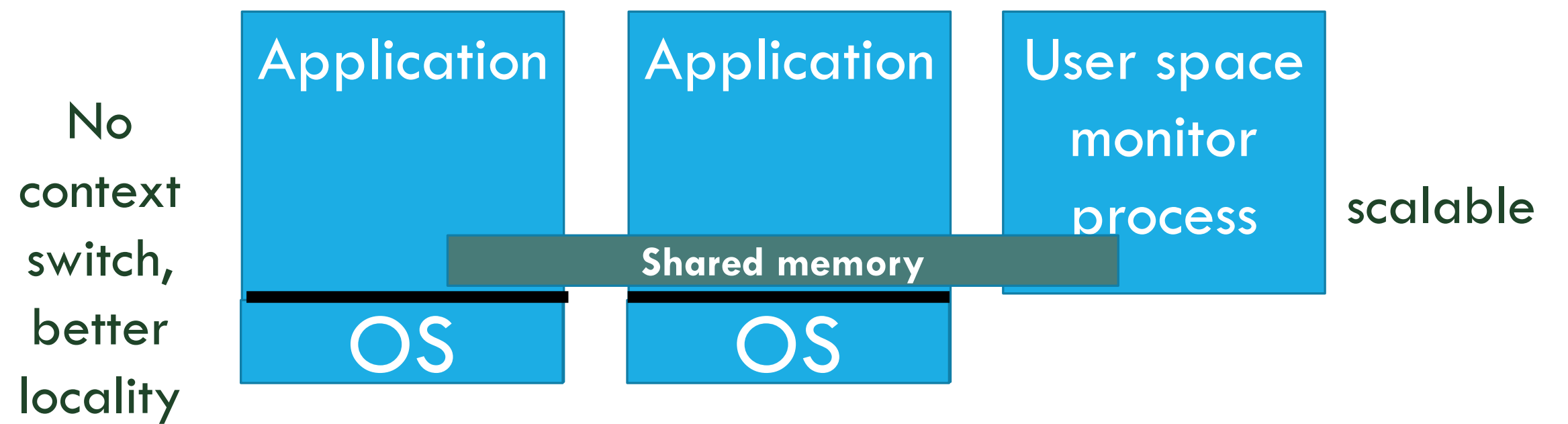
- unnecessary processing in kernels
- context switch
- distributed coordination has high contention



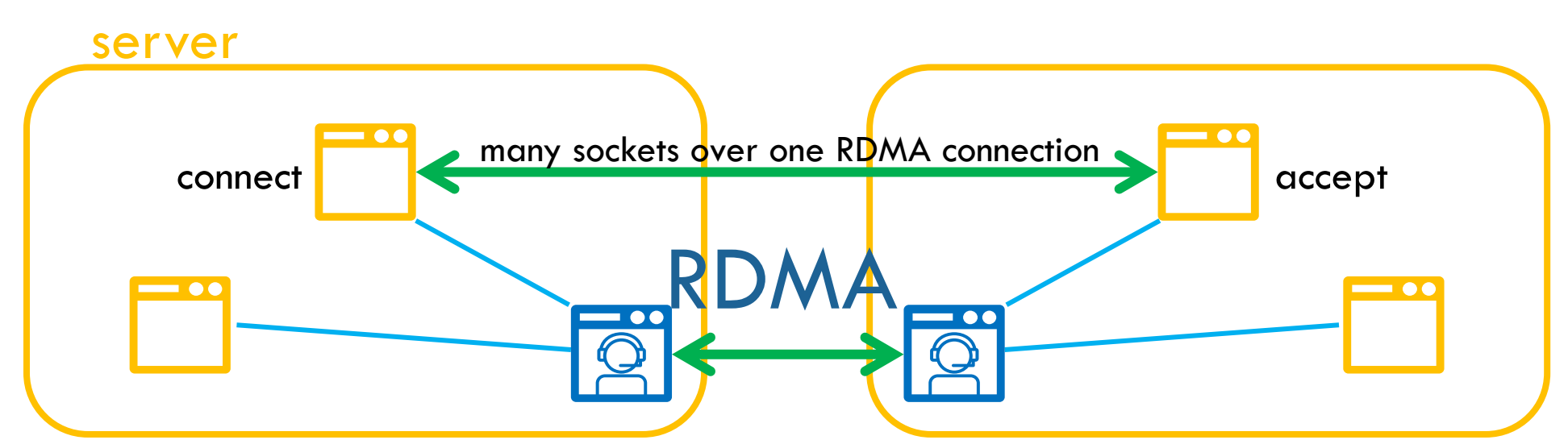
Bottleneck of the current design



Our design: use dedicated coordination core for monitor process



Inter-process communication with centralized coordinator



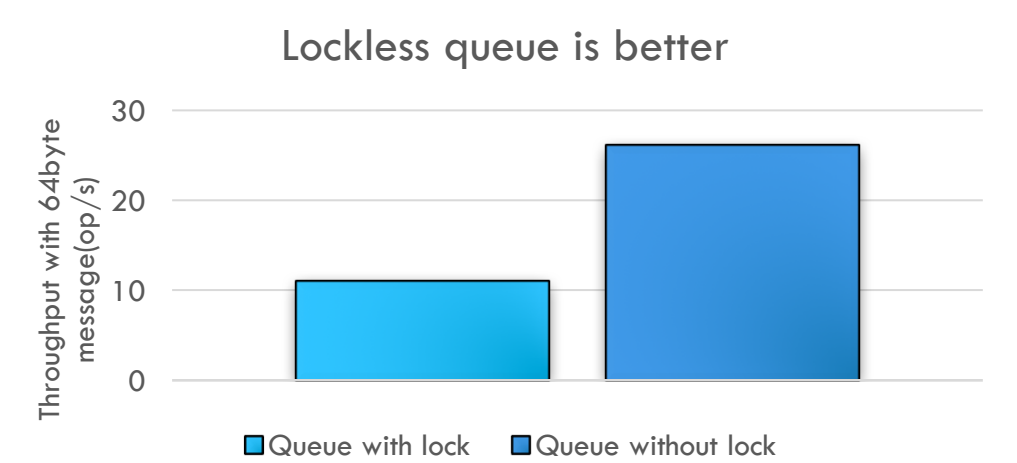
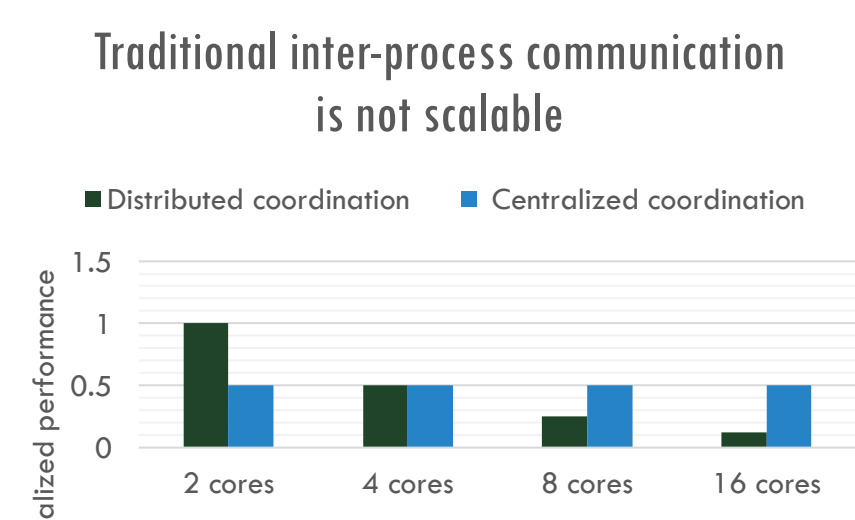
Our advantage

- High throughput
- Low latency
- Compatible with POSIX API (using LD_PRELOAD)

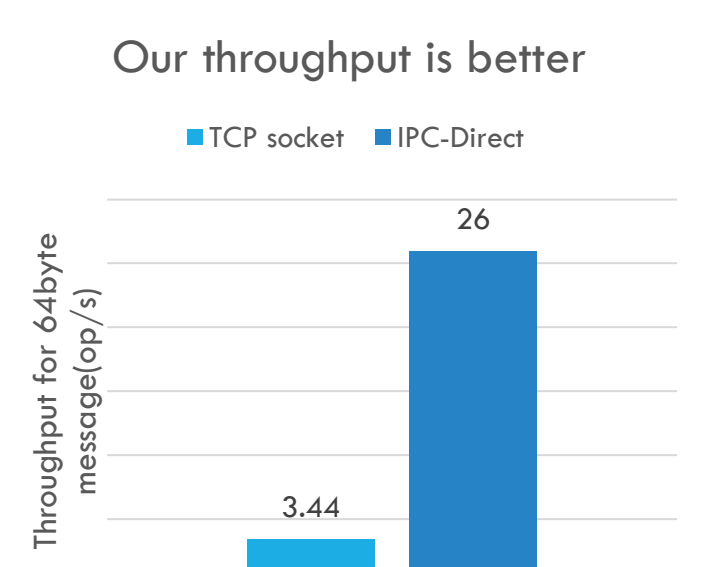
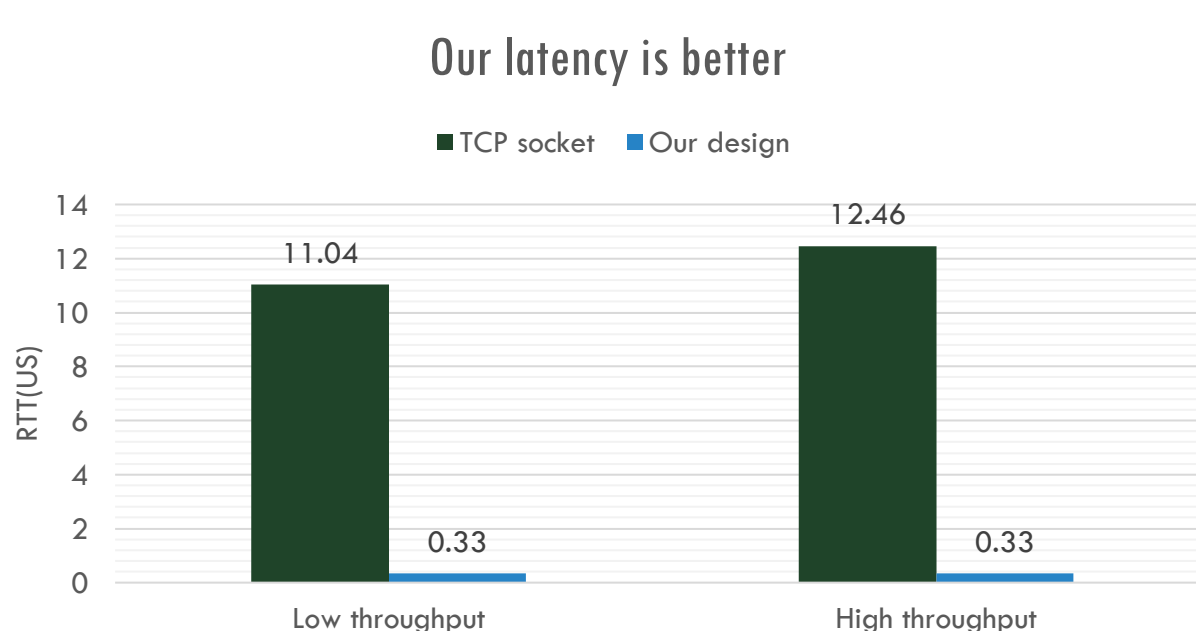
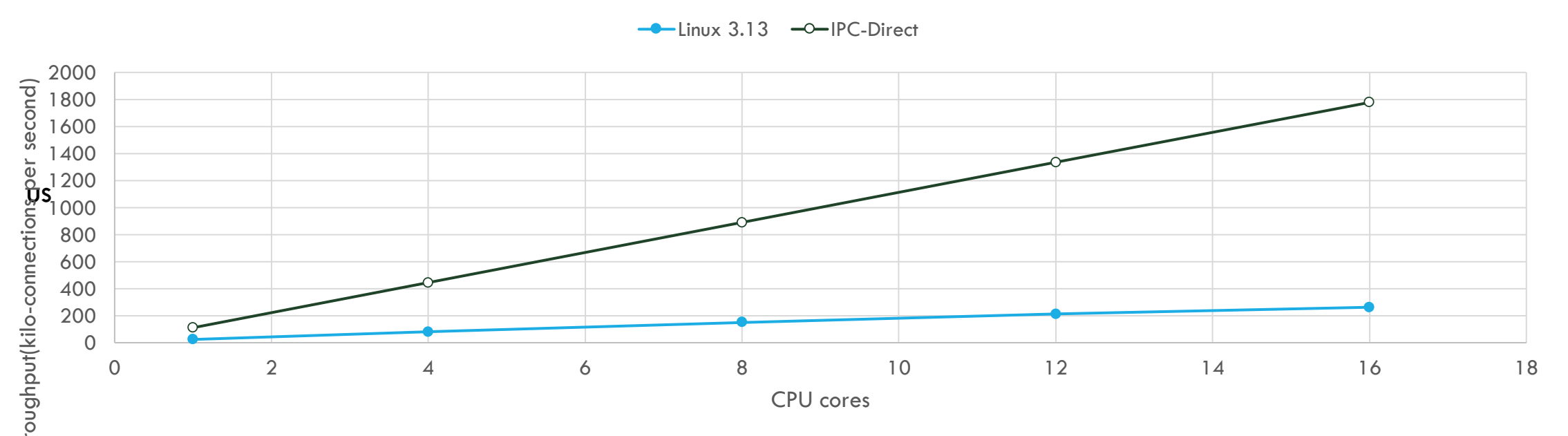
Two possible implementations of IPC-Direct

- Using divided queue for each process and polling from monitor
- Using shared queue for all processes and using atomic operation to tackle contention

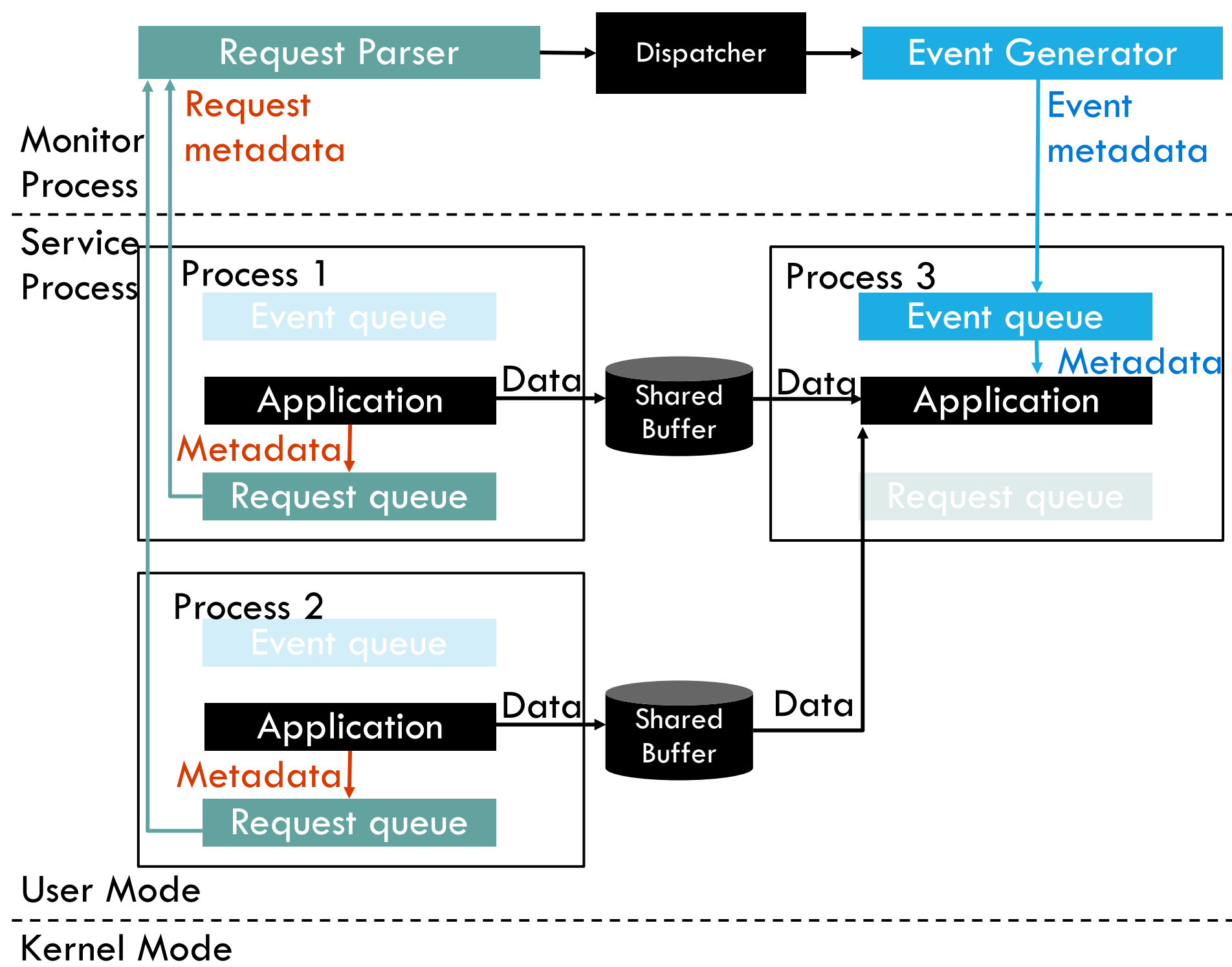
Microbenchmarks



Connection setup time for Nginx (Estimated)



Polling from different request queue



Using one shared request queue with atomic operation

