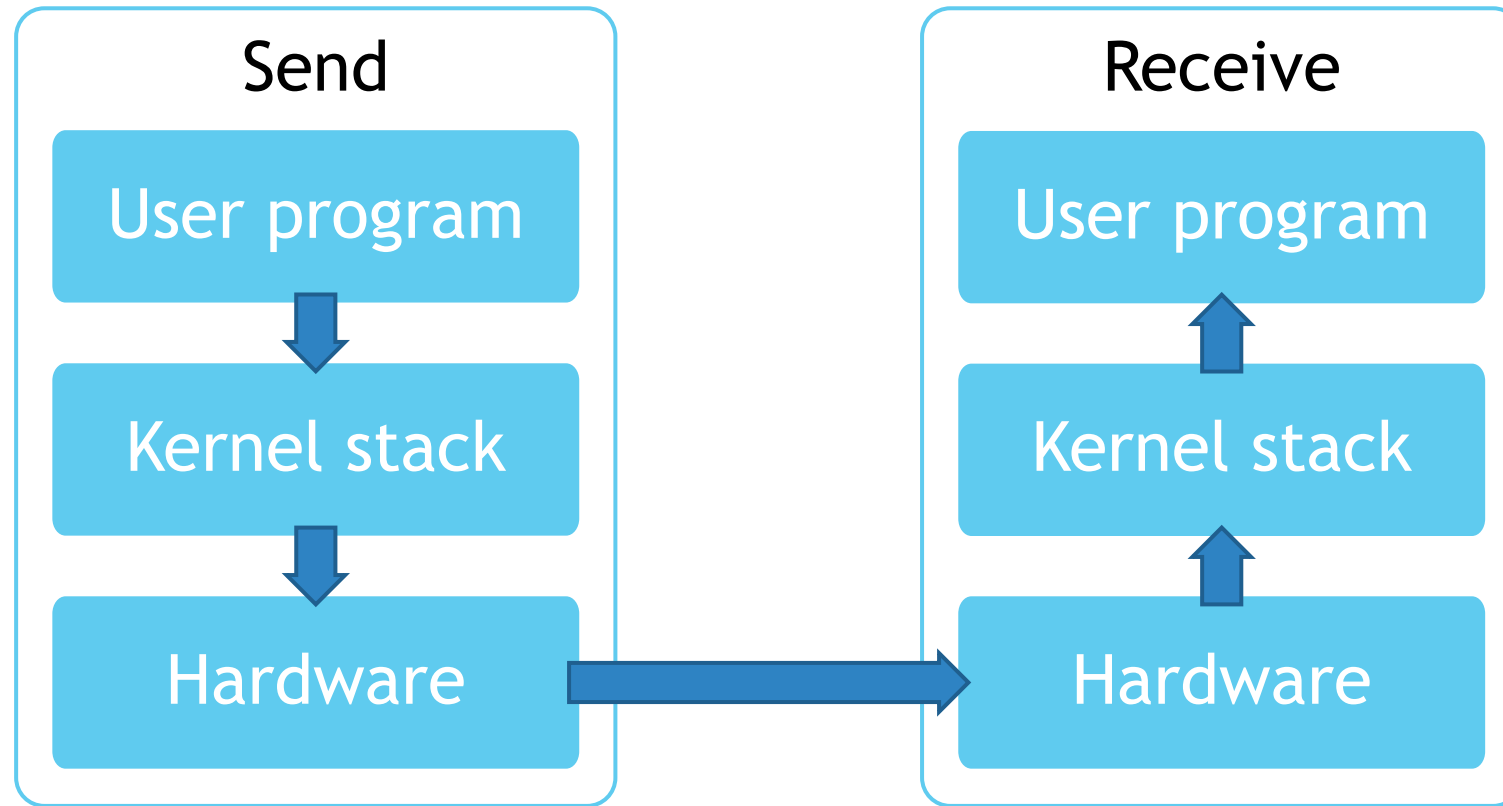


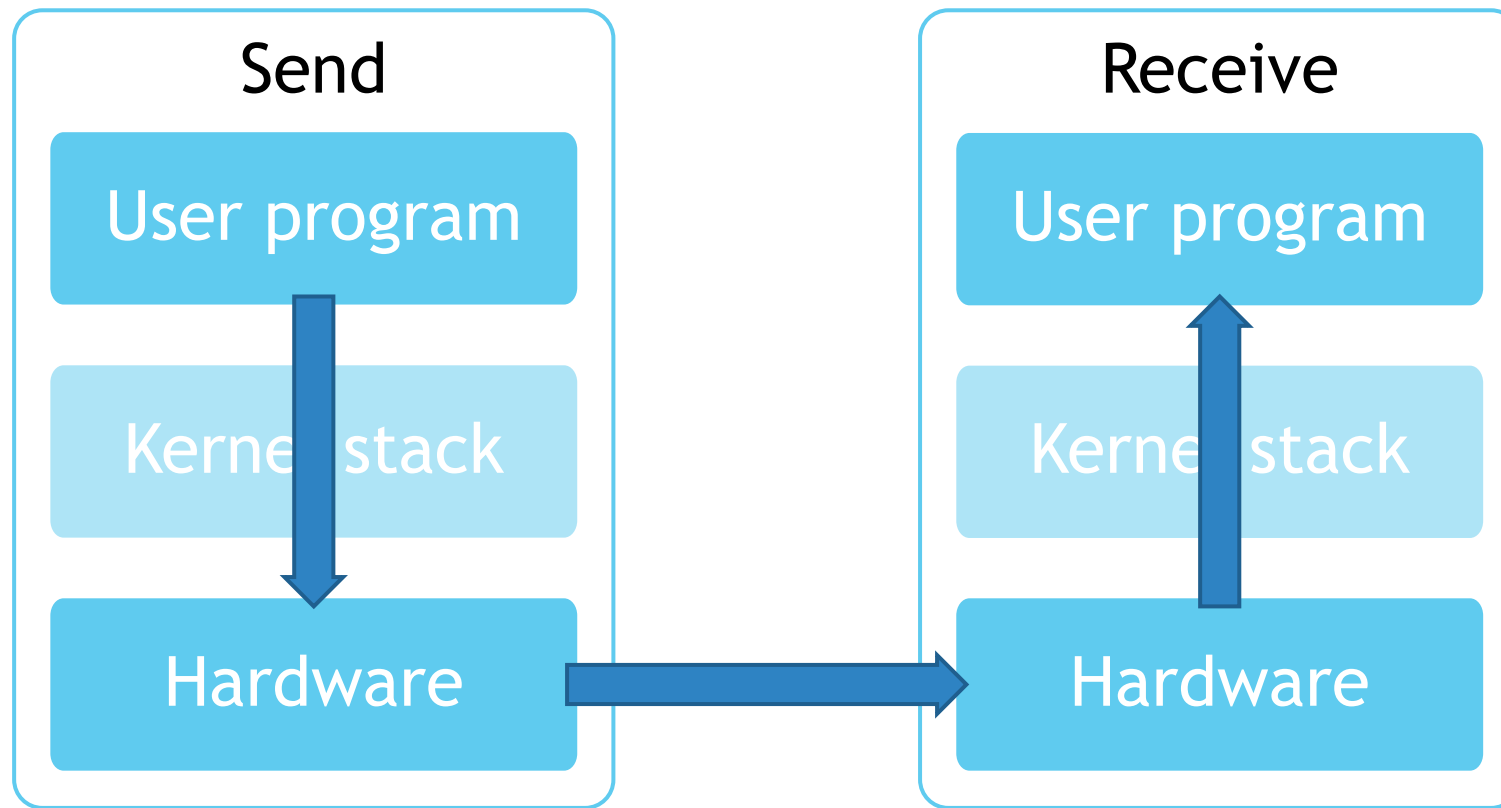
RDMA Networking

By Nathan

Networking *without* RDMA



Networking *with* RDMA



Benefits of RDMA

- ▶ Bypass kernel networking stack
 - ▶ Use networking hardware directly
 - ▶ Minimize data copying
 - ▶ Improve networking performance
- ▶ *Note: RDMA is a technique, not one specific implementation...*

My Project: Benchmark Program

Server

- ▶ Establish connection
- ▶ Start timer
- ▶ Send messages
- ▶ Stop timer
- ▶ Print results

Client

- ▶ Establish connection
- ▶ Start timer
- ▶ Receive messages
- ▶ Stop timer
- ▶ Print results

...implemented with 4 different technologies!

Tech 1: Linux Socket API

- ▶ Standard networking API
 - ▶ General purpose tool
 - ▶ Optimized and easy to use
- ▶ Does *not* use RDMA
 - ▶ Point of comparison for the other 3 technologies, which do use RDMA

Sample benchmark data

- 100,000 messages
- 64 bytes/message
- 0.18 seconds, 35.3 MB/s (server)
- 0.10 seconds, 67.2 MB/s (client)

Tech 2: InfiniBand IB-Verbs API

- ▶ Designed to use RoCE hardware
 - ▶ Can be run using software as well (Soft-RoCE)
- ▶ Not particularly robust
 - ▶ Couldn't handle very many or very large messages
 - ▶ Occasionally crashed due to race conditions under the hood
- ▶ Ran 10-20x slower than socket
 - ▶ Moved on to VPP instead of getting to the bottom of this

Sample benchmark data

- 1,000 messages
- 64 bytes/message
- 0.02 seconds, 3.20 MB/s (server)
- 0.02 seconds, 3.20 MB/s (client)

Tech 3: Vector Packet Processing (VPP)

- ▶ Weird code (too many macros)
- ▶ Poor documentation
- ▶ Refused to compile



Tech 4: eXpress Data Path (XDP)

- ▶ Uses two programs
 - ▶ Kernel program: intercepts incoming packets, sends them to user program's AF_XDP socket
 - ▶ User program: reads and writes packets using AF_XDP socket
- ▶ Ran 5-10x slower than socket
 - ▶ Couldn't enable zero-copy mode due to system limitations
 - ▶ Nearly 90% of server runtime was the kernel doing packet processing

Sample benchmark data

- 100,000 messages
- 64 bytes/message
- 1.03 seconds, 6.19 MB/s (server)
- 1.01 seconds, 6.32 MB/s (client)

Results

- ▶ 1st place: Socket
 - ▶ It's not easy to beat the Linux developers at their own game!
- ▶ 2nd place: XDP
 - ▶ Most straightforward RDMA tech
 - ▶ Some clear ways to try to improve
- ▶ 3rd place: IB-Verbs
 - ▶ More opaque API than XDP
 - ▶ Harder to find and fix errors
- ▶ (Dis)honorable mention: VPP



Challenges

- ▶ Switching between technologies over the course of the summer
- ▶ Using APIs with limited documentation
- ▶ Debugging asynchronous and time-sensitive code
- ▶ Trying to outperform the Linux kernel developers

Challenges

- ▶ Switching between technologies over the course of the summer
- ▶ Using APIs with limited documentation
- ▶ Debugging asynchronous and time-sensitive code
- ▶ Trying to outperform the Linux kernel developers
- ▶ My mentor having a baby



What I learned

- ▶ Better C and Make skills
- ▶ Profiling my code (gprof)
- ▶ Reading source code
- ▶ Low-level networking

Thank you!

The background features abstract, overlapping geometric shapes in various shades of blue, ranging from light sky blue to deep navy blue. These shapes are primarily located on the right side of the frame, creating a modern, layered effect against the white background.