



# RSOCKETS

## RDMA for Dummies

Sean Hefty  
Intel Corporation

# Last Time on RSOCKETS

- Programming to verbs ***at all*** is hard

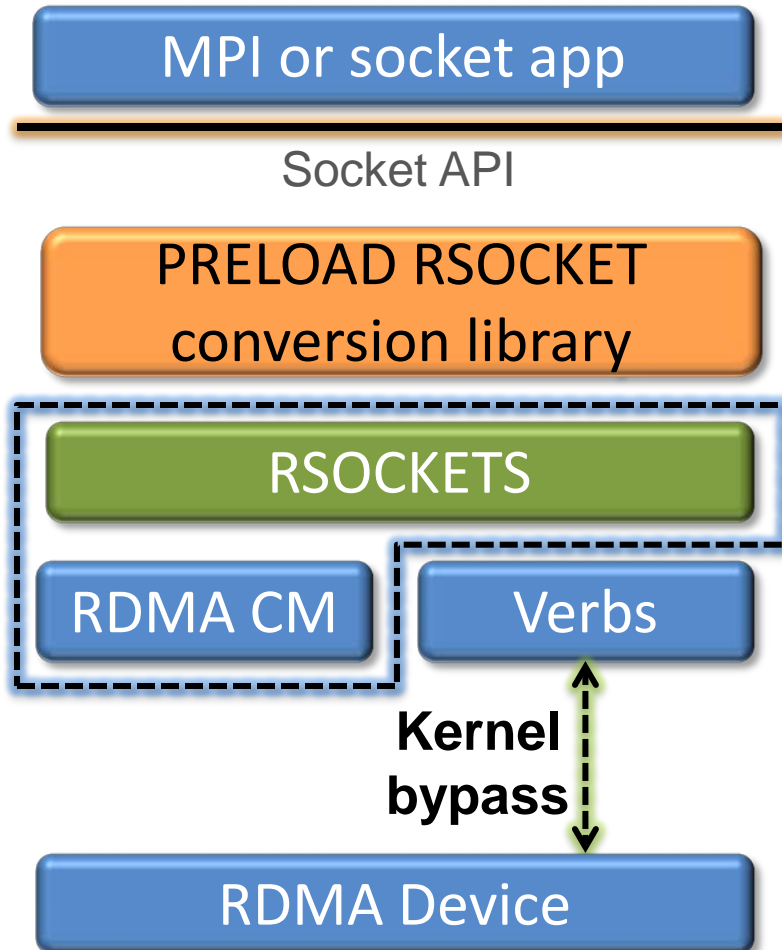


***Just give me fast  
sockets. Please?***

Programming to verbs  
***well*** is ***really*** hard

# RSOCKETS

## SOCKETS Over RDMA Solution



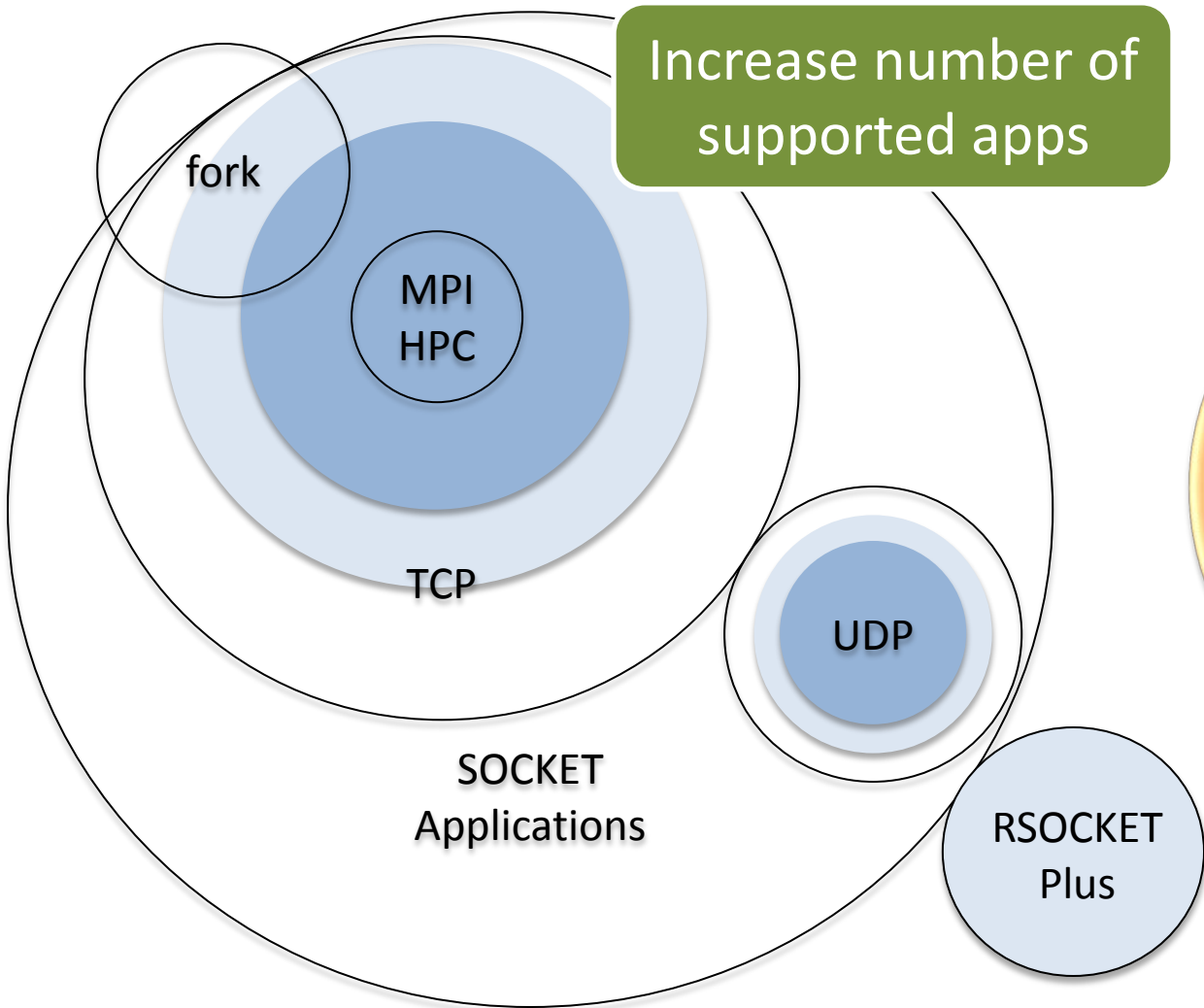
Role within the industry

Approach to the 'performance gap'

Future features and development

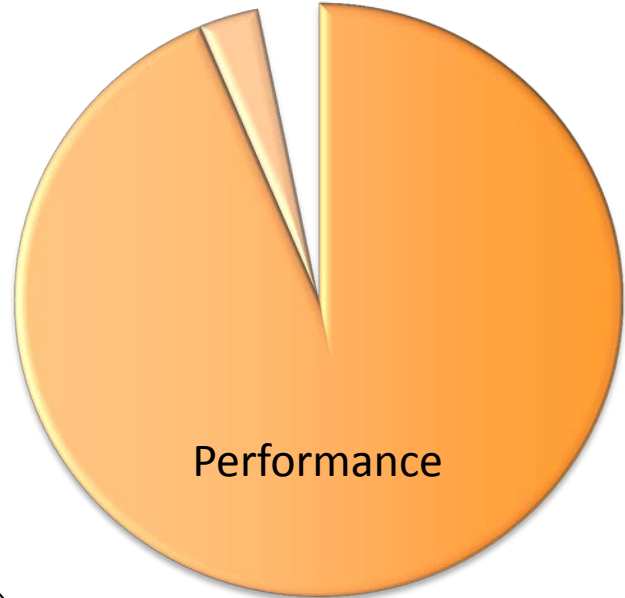
# RSOCKETS Role

**STRONG**  
industry interest



Increase number of supported apps

Decrease performance gap



**Performance is priority**

# RSOCKETS Role

Extend reach and  
impact

RSOCKETS

Support other  
HW/SW interfaces

Optimized HW Interfaces

Verbs

PSM

MXM

Support other  
RDMA devices

RDMA Devices

Infiniband

RoCE

iWarp

Extend to other platforms (Windows)

# Closing the Performance Gap

## RDMA Specific Socket Options

- Programmable (setsockopt)
- Configuration files (/etc/rdma/\*)

## 'Natural' API Extensions

- Follow (non)blocking socket model
- Selective optimizations
- (riomap/riounmap, riowrite, rioread)

## Asynchronous IO Extensions

- For usable zero-copy
- Selective re-design of transfers
- Under consideration

# Future

## Data Stream (TCP)

- Real keep-alive support
- iWarp support

## Datagram (UDP)

- Multicast

## Compatibility

- Expand socket options as needed
- Kernel assistance for full compatibility

## Performance

- Native IB addressing
- Direct read capability (rioread)

# Challenges

Socket is an fd, an rsocket is not  
Still want kernel bypass

- A socket fd can be passed into many calls
  - dup, fstat, epoll, etc.
- Support for existing apps requires intercepting all these calls
- Most troublesome call list:
  - epoll - possible, but significant work
  - chroot - requires kernel change to RDMA stack
  - fork – cannot share resources between processes