

RDMA

ECE/CS598HPN

Radhika Mittal

Last week's reading

Empowering Azure Storage with RDMA (NSDI'23)

- What was the primary usecase?
- What are the other challenges that arise?
- What do you all think of using RDMA for this usecase?

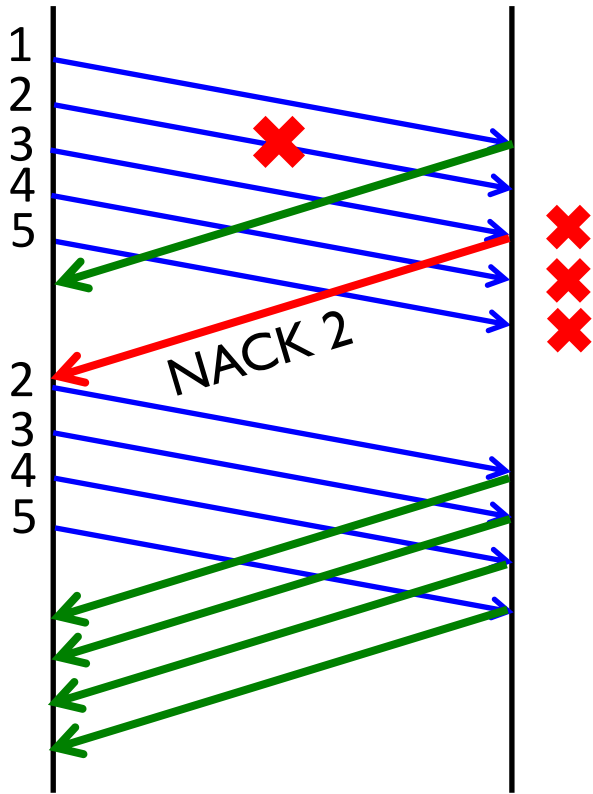
Focus of today's lecture

- Overview of RDMA
- Challenges with large-scale RDMA deployments

Conventional RDMA

- RDMA traditionally used in Infiniband clusters.
 - A different network protocol supporting high bandwidth.
- Infiniband links use credit-based flow control.
 - Losses are rare.
- Transport layer in RDMA NICs not designed to deal with losses efficiently.
 - Receiver discards out-of-order packets.
 - Sender does *go-back-N* on detecting packet loss.

Go-back-N Loss Recovery



Receiver discards all out-of-order packets.

Sender retransmits all packets sent after the last acked packet.

Conventional RDMA

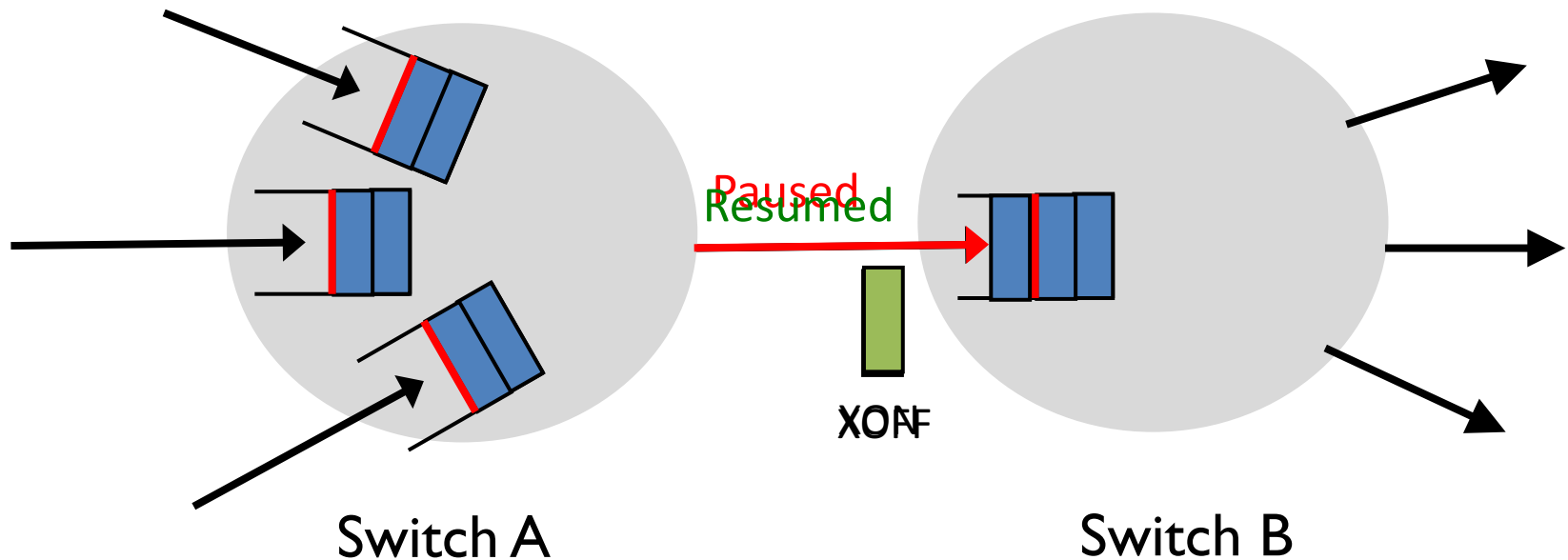
- RDMA traditionally used in Infiniband clusters.
 - A different network protocol supporting high bandwidth.
- Infiniband links use credit-based flow control.
 - Losses are rare.
- Transport layer in RDMA NICs not designed to deal with losses efficiently.
 - Receiver discards out-of-order packets.
 - Sender does *go-back-N* on detecting packet loss.

RDMA in datacenters

- Desire to run RDMA over commodity Ethernet.
- RoCE: RDMA over Ethernet fabric.
 - RoCEv2: RDMA over IP-routed networks.
- Infiniband transport was adopted as it is.
 - Go-back-N loss recovery.
 - Needs a lossless network for good performance.

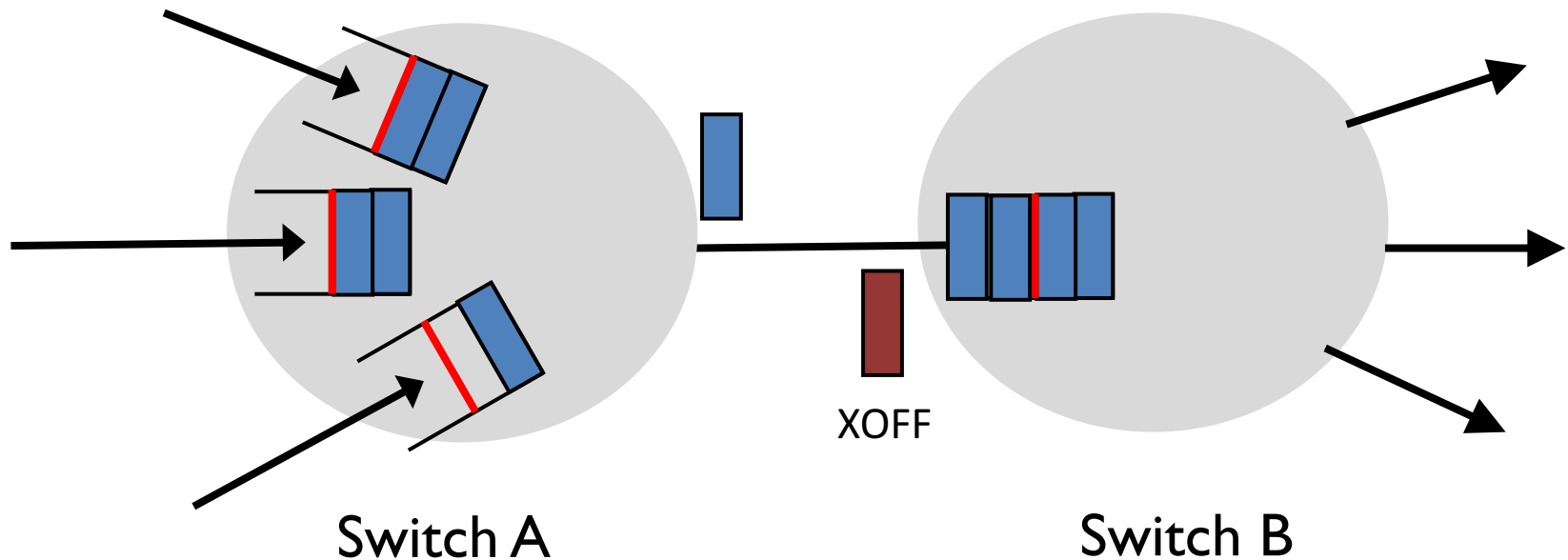
Network made lossless by enabling PFC

- Priority Flow Control (PFC)
 - Pause transmission when queuing exceeds a certain threshold.



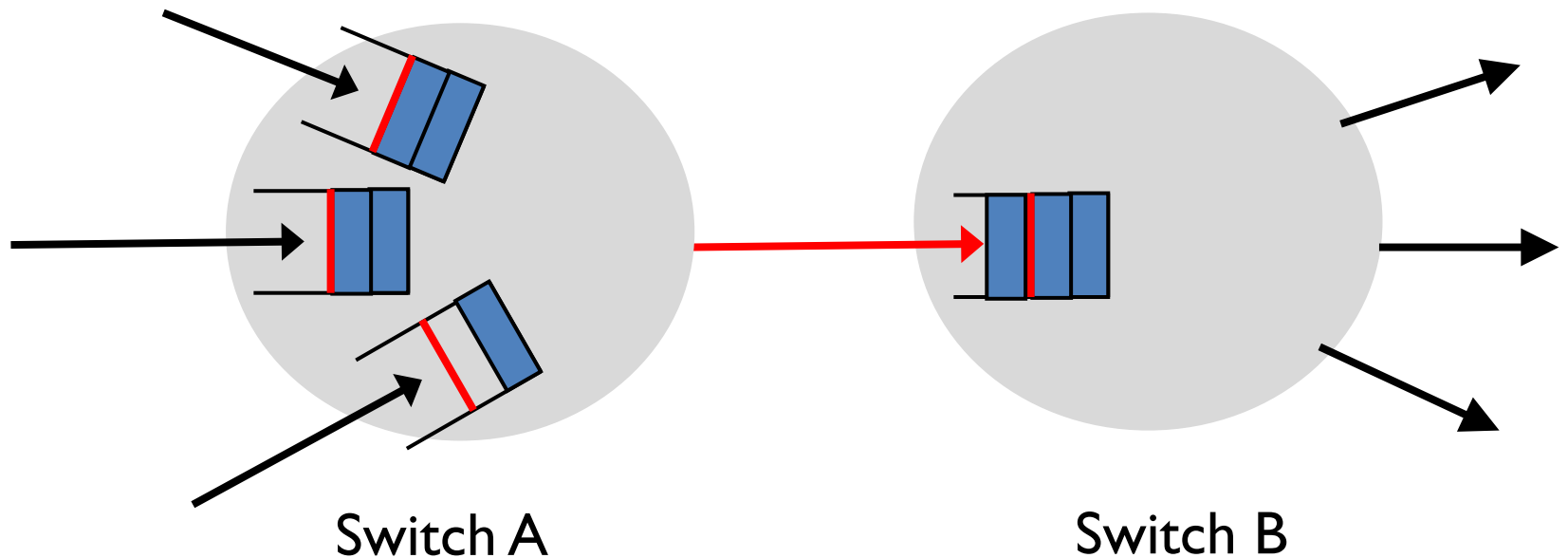
Drawbacks of PFC

Complicates network management.
PFC threshold requires careful configuration.



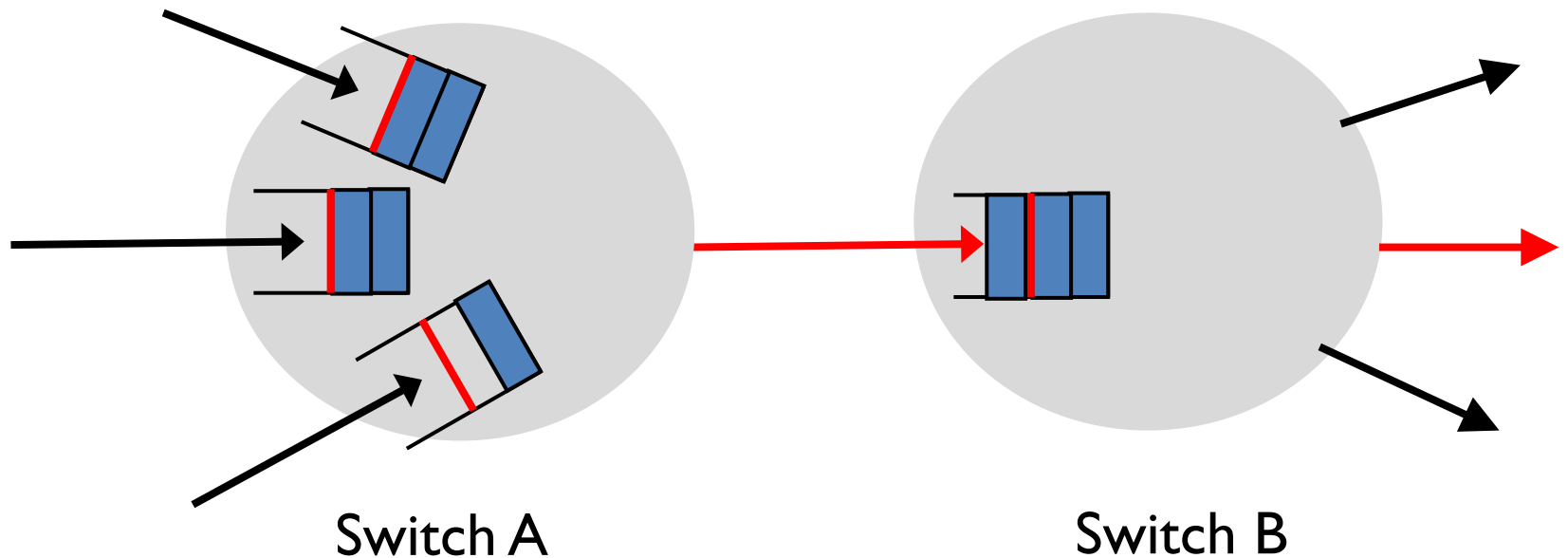
Drawbacks of PFC

Unfairness and Head-of-Line blocking



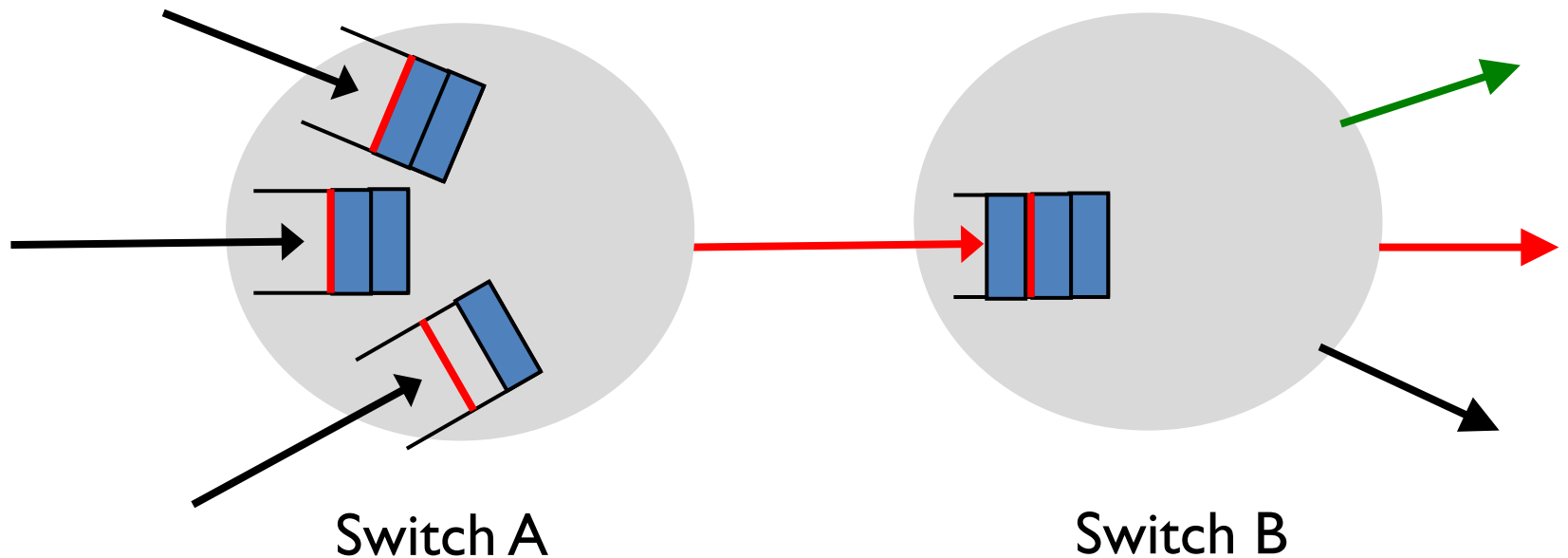
Drawbacks of PFC

Unfairness and Head-of-Line blocking



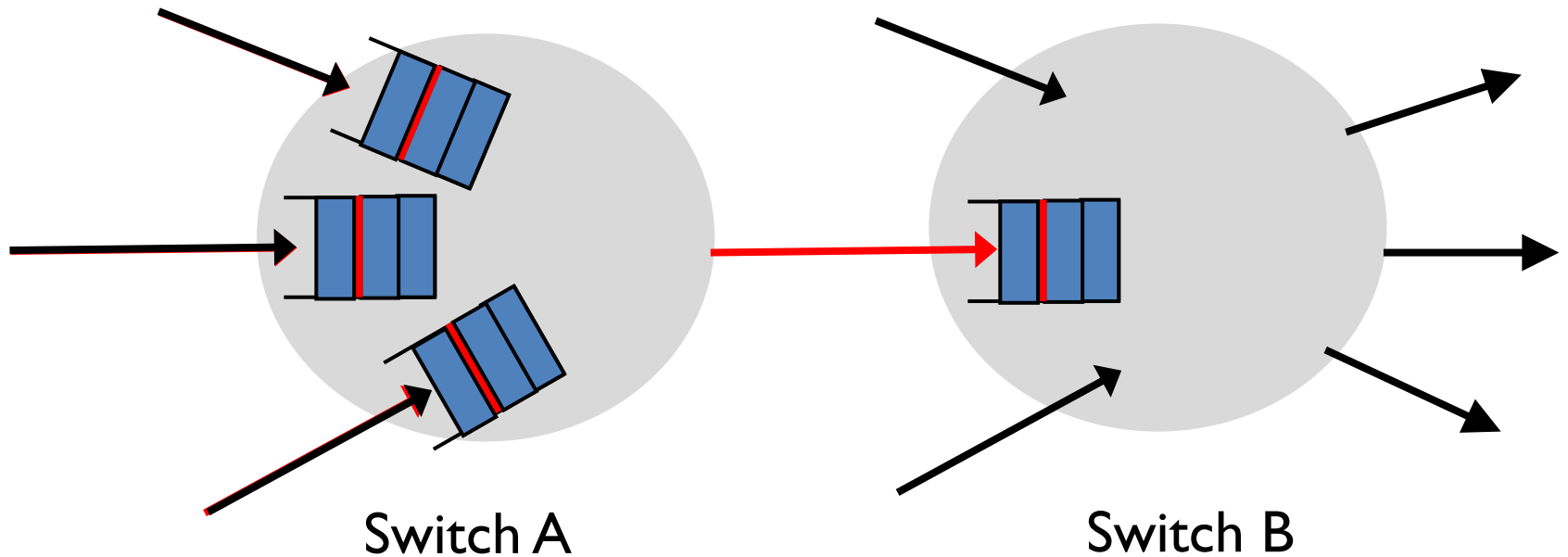
Drawbacks of PFC

Unfairness and Head-of-Line blocking



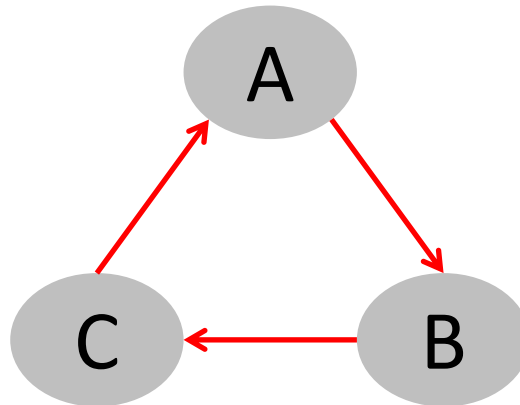
Drawbacks of PFC

Congestion Spreading



Drawbacks of PFC

Deadlocks caused by cyclic buffer dependency



Lots of work highlighting PFC issues

- **Congestion control to mitigate PFC issues**
 - Delay-based TIMELY, *Mittal et al, SIGCOMM'15*
 - ECN-based DCQCN, *Zhu et al, SIGCOMM'15*
 - Recent update: Zero-touch RoCE based on delay-based congestion control (NVIDIA/Microsoft)
- **Deployment experience**
 - RDMA over commodity Ethernet at scale, *Guo et al, SIGCOMM'16*
- **Deadlock avoidance**
 - Deadlocks in datacenter: why do they form and how to avoid them, *Hu et al, HotNets 2016*
 - Unlocking credit loop deadlock, *Shpiner et al, HotNets 2016*
 - Tagger: Practical PFC deadlock prevention in datacenter networks, *Hu et al, CoNext 2017*

Can we alter the RoCE NIC design
such that a lossless network
is not required?

Why not iWARP?

- Designed to support RDMA over a fully general network.
 - Implements entire TCP stack in hardware.
 - Needs translation between RDMA and TCP semantics.
- General consensus:
 - iWARP is more complex, more expensive, and has worse performance.

iWARP vs RoCE

NIC	Cost in Dec 2016	Throughput	Latency
iWARP: Chelsio T-580-CR	\$760	3.24Mpps	2.89us
RoCE: Mellanox MCX 416A-BCAT	\$420	14.7Mpps	0.94us

**Could be due to a number of reasons besides transport design: different profit margin, engineering effort, supported features etc.*

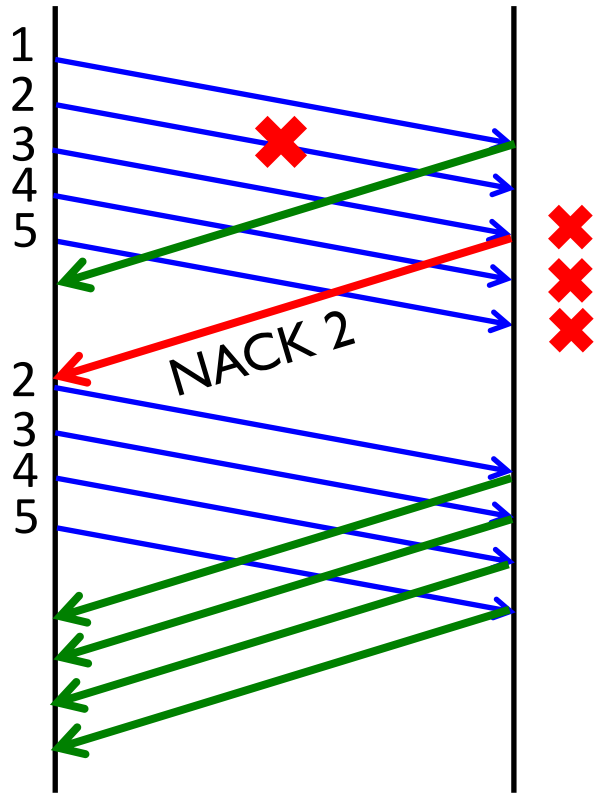
Our work shows that

- iWARP had the right philosophy.
 - NICs should efficiently deal with packet losses.
 - Performs better than having a lossless network.
- But we can have a design much closer RoCE.
 - No need to support the entire TCP stack.
 - Identify incremental changes for better loss recovery.
 - Less complex and more performant than iWARP.

Improved RoCE NIC (IRN)

1. Better loss recovery.

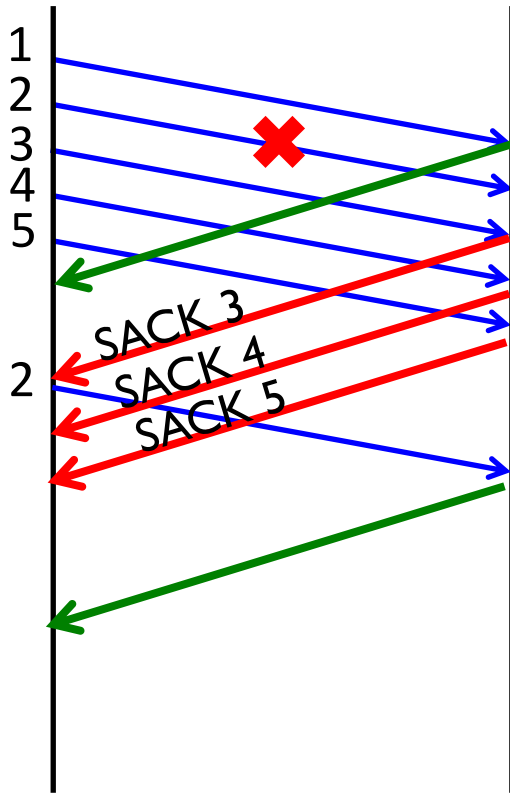
Instead of go-back-N loss recovery...



Receiver discards all out-of-order packets.

Sender retransmits all packets sent after the last acked packet.

...use selective retransmission



Receiver does not discard out-of-order packets and *selectively acknowledges* them.

Sender retransmits only the lost packets.

Use bitmaps to track lost packets.

0	1	1	1	0	0
---	---	---	---	---	---



Seq. No. = 2

Handling timeouts

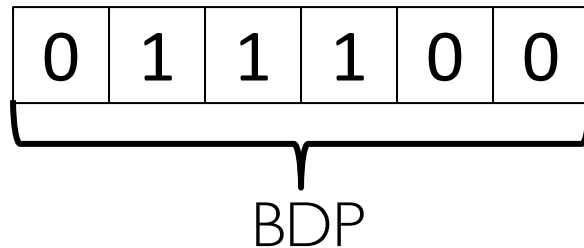
- Very small timeout value
 - Spurious retransmissions.
- Very large timeout value
 - High tail latency for short messages.
- IRN uses two timeout values
 - RTO_{low} : Less than N packets in flight.
 - RTO_{high} : Otherwise.

Improved RoCE NIC (IRN)

1. Better loss recovery.
 - Selective retransmission instead of go-back-N.
 - Inspired from traditional TCP, but simpler.
 - Two timeout values instead of one.
2. BDP-FC: BDP based flow control.

BDP-FC

- Bound the number of in-flight packets by the bandwidth-delay product (BDP) of the network.
- Reduces unnecessary queuing.
- Strictly upper-bounds the amount of required state.



Improved RoCE NIC (IRN)

1. Better loss recovery.

- Selective retransmission instead of go-back-N.
 - Inspired from traditional TCP, but simpler.
- Two timeout values instead of one.

2. BDP-FC: BDP based flow control.

- Bound the number of in-flight packets by the bandwidth-delay product (BDP) of the network.

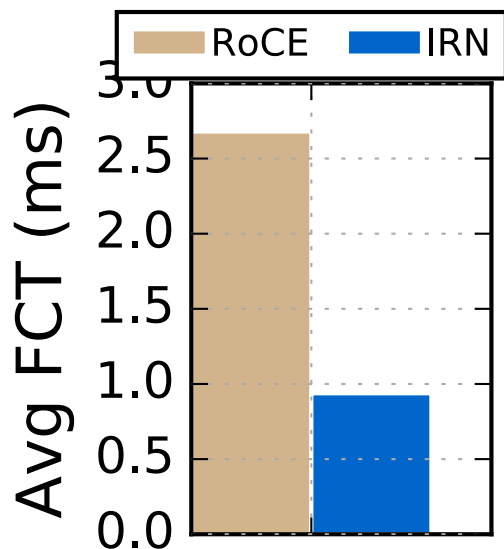
Can IRN eliminate the need for a lossless network?

Default evaluation setup

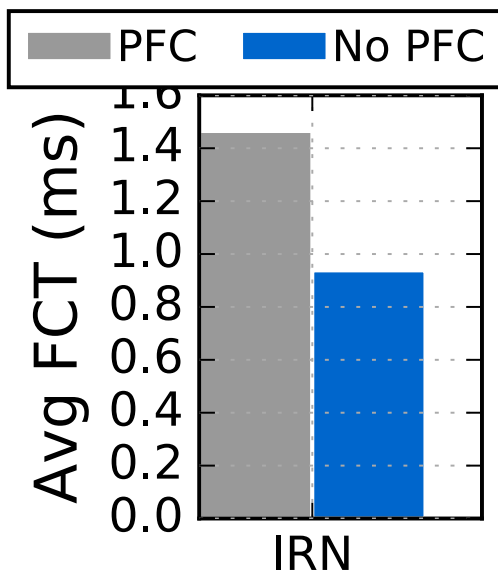
- Mellanox simulator modeling ConnectX4 NICs.
 - Extended from Omnet/Inet.
- Three layered fat-tree topology.
- Links with capacity 40Gbps and delay 2us.
- Heavy-tailed distribution at 70% utilization.
- Per-port buffer of $2 \times$ (bandwidth-delay product).

Key results

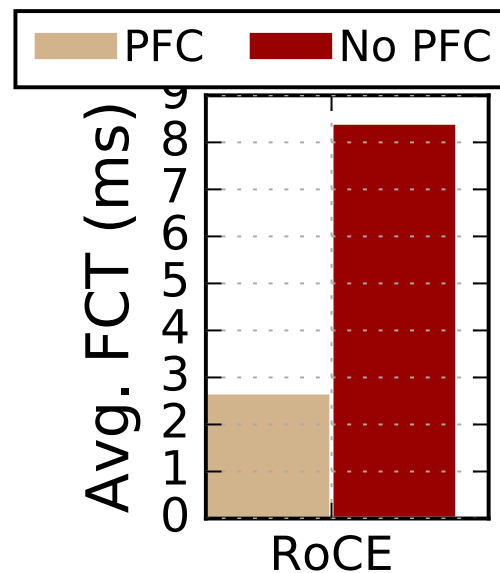
IRN *without PFC*
performs
better than
RoCE *with PFC*.



IRN does not
require PFC.

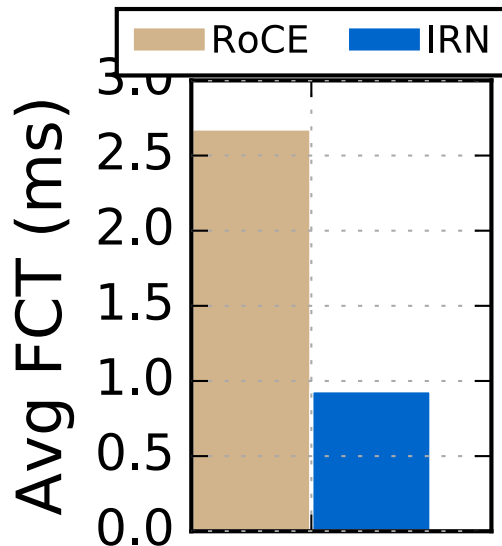


RoCE requires
PFC.

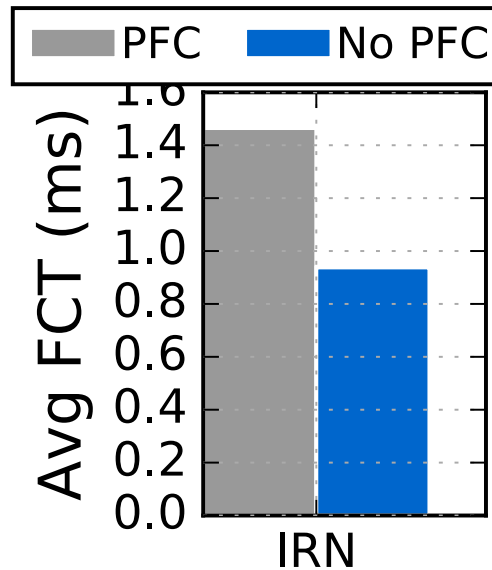


Average flow completion times

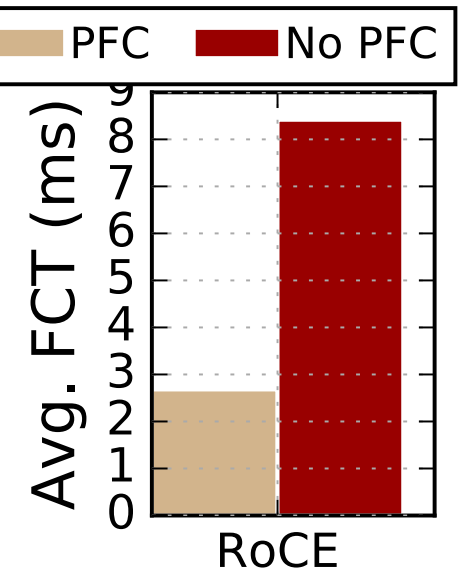
IRN *without PFC*
performs
better than
RoCE *with PFC*.



IRN does not
require PFC.

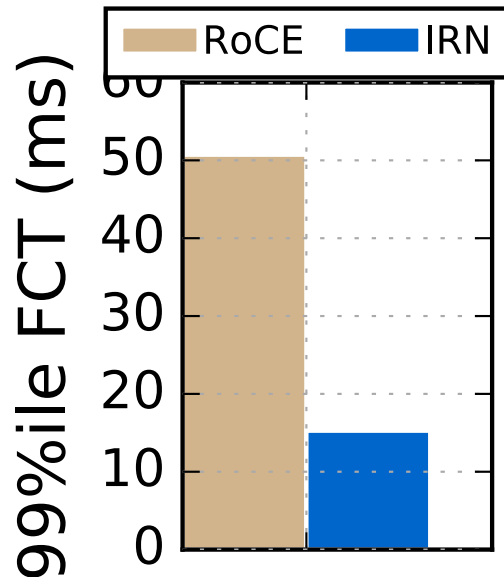


RoCE requires
PFC.

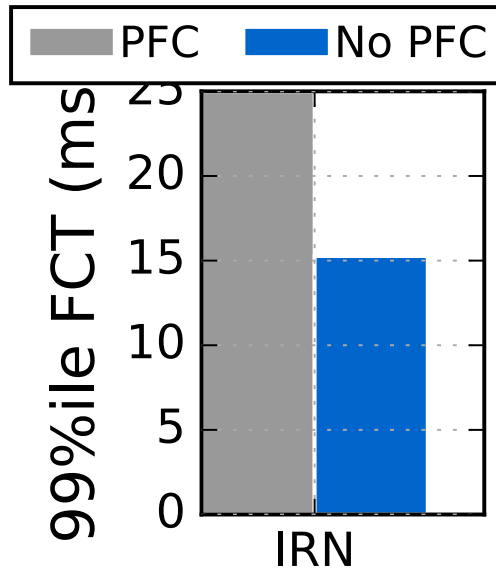


Tail flow completion times

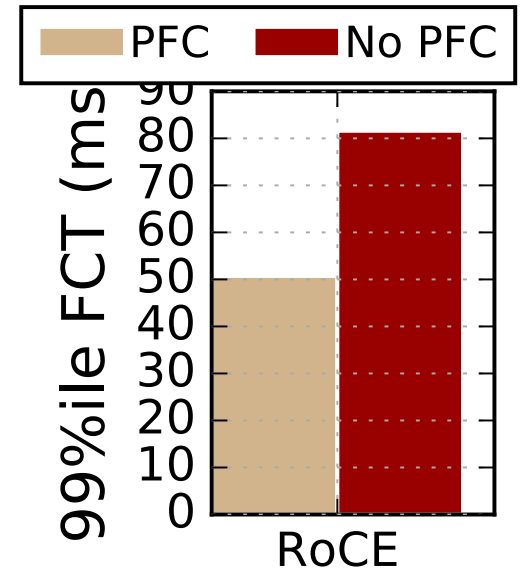
IRN *without PFC*
performs
better than
RoCE *with PFC*.



IRN does not
require PFC.

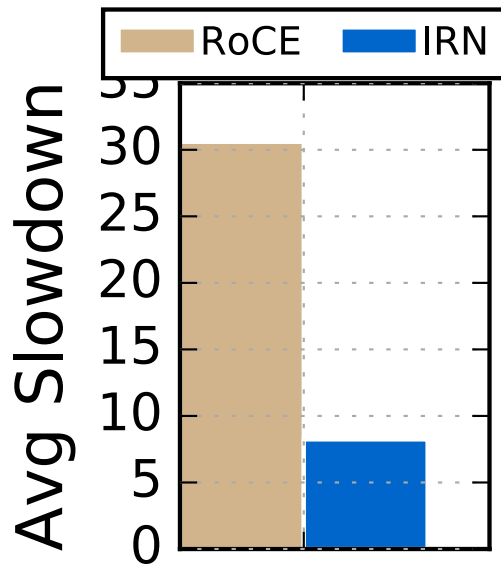


RoCE requires
PFC.

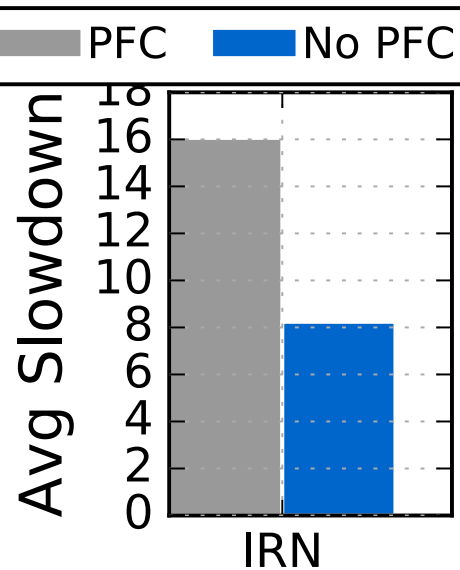


Average slowdown

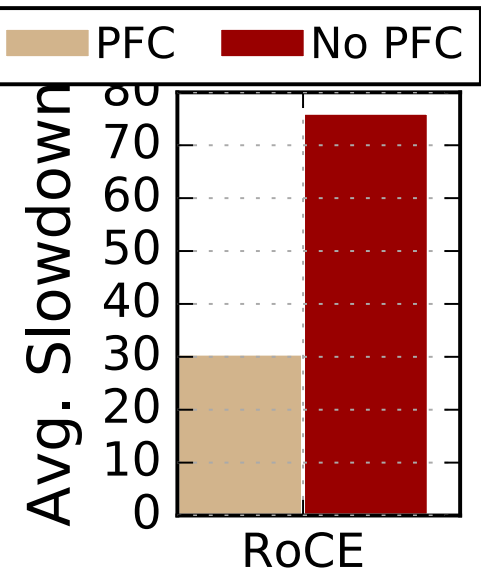
IRN *without PFC*
performs
better than
RoCE *with PFC*.



IRN does not
require PFC.

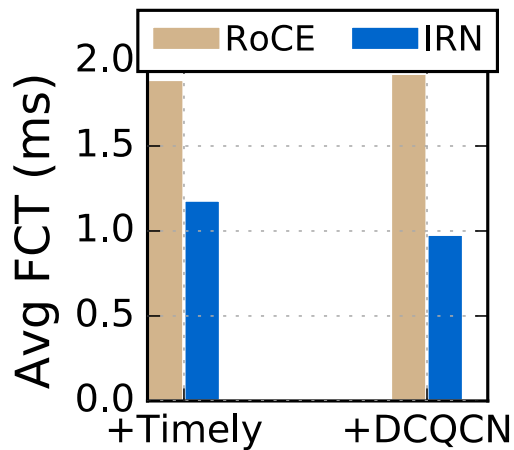


RoCE requires
PFC.

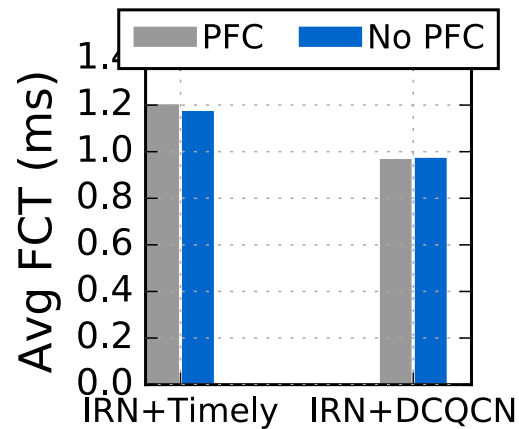


With explicit congestion control

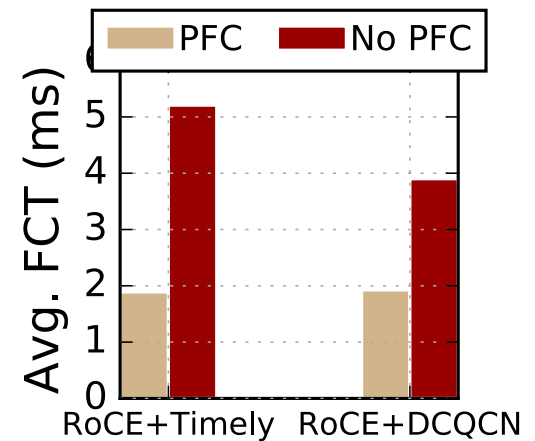
IRN *without* PFC
performs
better than
RoCE *with* PFC.



IRN does not
require PFC.

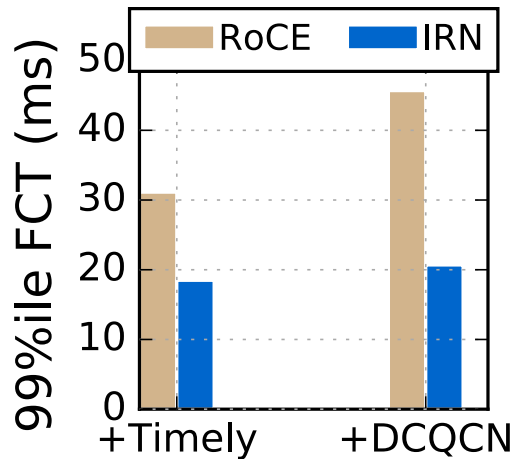


RoCE requires
PFC.

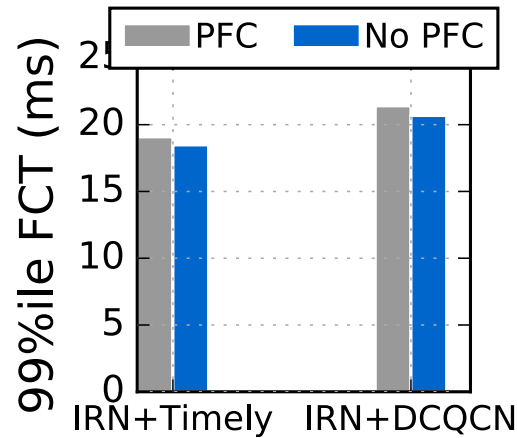


With explicit congestion control

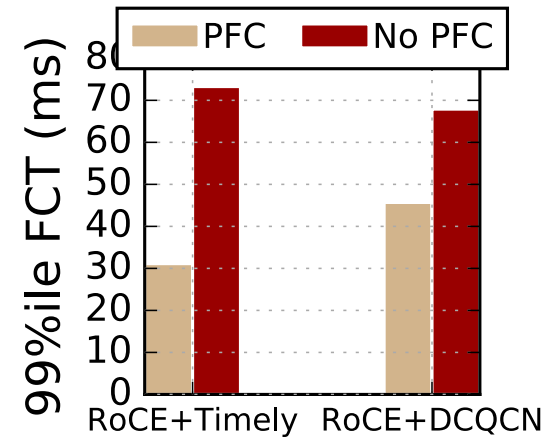
IRN *without* PFC
performs
better than
RoCE *with* PFC.



IRN does not
require PFC.

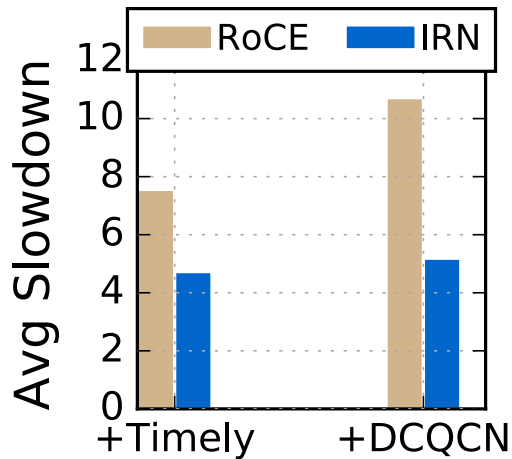


RoCE requires
PFC.

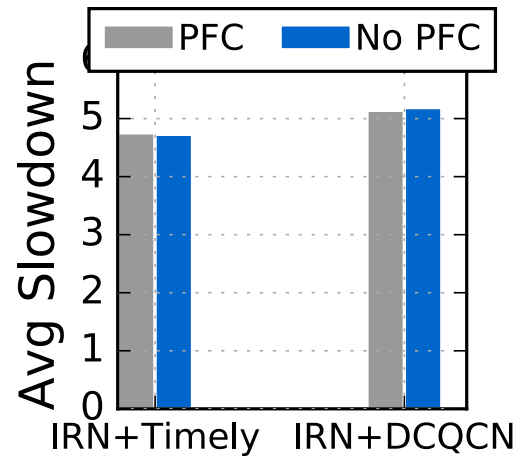


With explicit congestion control

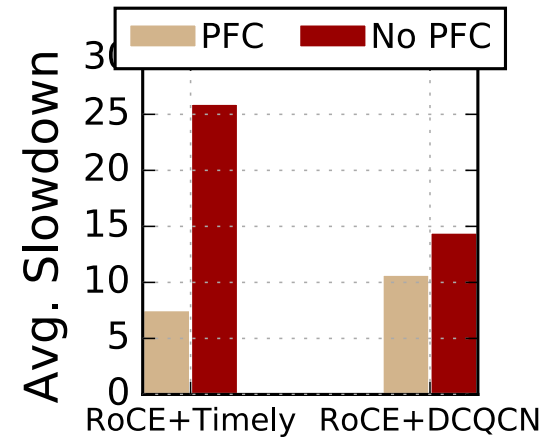
IRN *without* PFC
performs
better than
RoCE *with* PFC.



IRN does not
require PFC.



RoCE requires
PFC.



Robustness of results

- Tested a wide range of experimental scenarios:
 - Varying link bandwidth.
 - Varying workload.
 - Varying scale of the topology.
 - Varying link utilization.
 - Varying buffer size.
 - ...
- Our key takeaways hold across all of these scenarios.

Can IRN eliminate the need for a lossless network? **Yes.**

Can IRN be implemented easily?

Implementation challenges

- Need to deal with out-of-order packet arrivals.
 - Crucial information in first packet of the message.
 - Replicate in other packets.

Implementation challenges

- Need to deal with out-of-order packet arrivals.
 - Crucial information in first packet of the message.
 - Replicate in other packets.
 - Crucial information in last packet of the message.
 - Store it at the end-points.

Implementation challenges

- Need to deal with out-of-order packet arrivals.
 - Crucial information in first packet of the message.
 - Replicate in other packets.
 - Crucial information in last packet of the message.
 - Store it at the end-points.
 - Implicit matching between packet and work queue element (WQE).
 - Explicitly carry WQE sequence in packets.

Implementation challenges

- Need to deal with out-of-order packet arrivals.
 - Crucial information in first packet of the message.
 - Replicate in other packets.
 - Crucial information in last packet of the message.
 - Store it at the end-points.
 - Implicit matching between packet and work queue element (WQE).
 - Explicitly carry WQE sequence in packets.
- Need to explicitly send Read Acks.

Implementation overheads

- New packet types and header extensions.
 - Upto 16 bytes.
- Total memory overhead of 3-10%.
- FPGA synthesis targeting the device on an RDMA NIC.
 - Less than 4% resource usage.
 - 45.45Mpps throughput (without pipelining).

Can IRN eliminate the need for a lossless network? Yes.

Can IRN be implemented easily? Yes.

IRN Summary

- IRN makes incremental updates to the RoCE NIC design to handle packet losses better.
- IRN performs better than RoCE without requiring a lossless network.
- The changes required by IRN introduce minor overheads.

Your thoughts?

Updates since then....

- A variant got deployed on Mellanox (now NVIDIA) NICs.
 - has its limitations (as we will see in subsequent papers)
- That, along with CC, *enables* disabling PFC.
- Different datacenters have made different choices.....

Challenges of deploying RDMA in DCs

- Need for a lossless network
 - Congestion control to mitigate PFC issues (DCQCN, Timely, ZTR).
 - Better loss recovery in the NIC (IRN, SIGCOMM'18)
 - Large enough buffers + congestion control (eRPC, NSDI'19)
- Limited NIC cache:
 - Use bigger pages for memory translation (FaRM, NSDI'14).
 - Optimizing number of QPs (FaRM, NSDI'14; FASST, OSDI'16).
- Limited resource sharing and isolation
 - Kernel re-direction (LITE, SOSP'17)
- Supporting RDMA for VMs (para-virtual RDMA)
 - Commercial solution from VMWare requiring NIC support.
- Limited flexibility (tied to increased heterogeneity)
 - FPGA-based implementation / firmware patches.

Was RDMA the right choice?

Alternative designs?

Next few classes

- NVIDIA's RDMA NIC + switch package for AI workloads.
- Ultra Ethernet: clean-slate effort to replace Infiniband based RDMA with alternative hardware transport.
- ByteDance's CXL-CCL: CXL based communication interface to replace RDMA.
- Digression: software NICs, network virtualization.
 - Google's SNAP (software-based solution).
- Falcon: Google's hardware-based transport to replace RoCE.