Open Fabrics Adoption and Use

OPENFABRICS ALLIANCE

John F. Russo Moderator

Introduction

- In order to facilitate wider adoption of OFED there are a number of roadblocks that must be removed or minimized.
- Barriers to HPC include concern about the availability of software that will run ISV applications on HPC servers and lack of people skilled in using HPC hardware and software systems

* The Council on Competitiveness & IDC (2008)

OFED Software

- Has become a standard with its adoption into the kernel
- Long term relevance may require a new mindset
  - Fewer changes being made in the base stack and the Upper Layer Protocols.
  - Refocus into areas that aid wider adoption
  - Group cooperation for common goals

Areas of Improvement

- User documentation
  - Make it simpler for new users to understand the capabilities and power of OFED
- Developer documentation
  - Creating a “how-to” guide to aid developers who wish to write native applications
- Changes to certain areas of the stac
  - Make such development simpler (e.g. CM)

Connection Manager (CM)

- Provide complete sample programs using CM
  - Designed for easy cut/paste into real applications.
  - Removed unnecessary complexity in OFED APIs leading to potential simplification in the APIs.

Error Messages

- Improve error messages.
  - Too many of the errors from kernel and user space are not obvious to users and often refer to source code, modules, etc
  - Errors must assume the user does not have source code or does not have the time to study source to diagnose the error
  - Errors should provide as much useful information as possible, such as remote node names (not just LIDs)

Barriers to OFED Adoption

OPENFABRICS
ALLIANCE

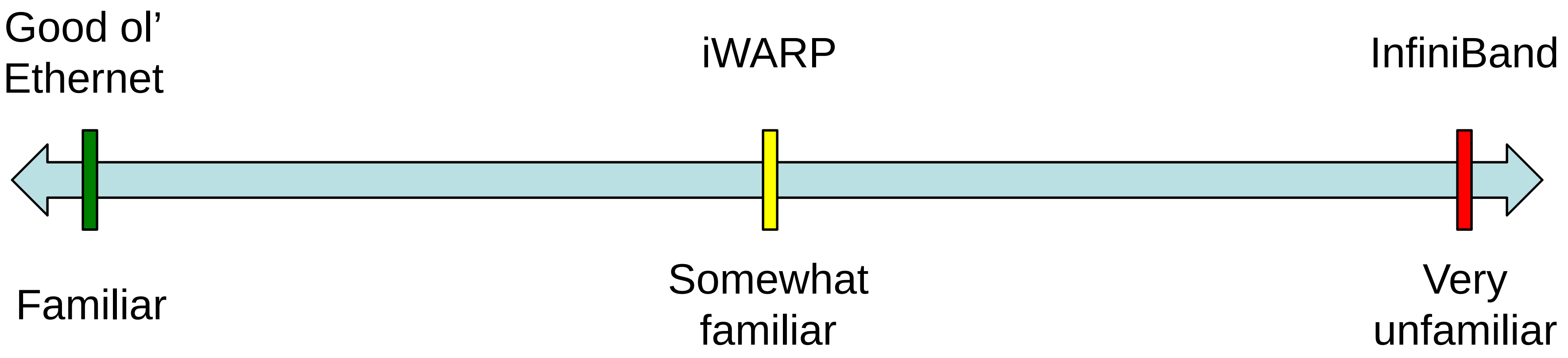
Jeff Squyres, Cisco Systems

Two main categories of problems

- Applications
- Administrators

Common theme

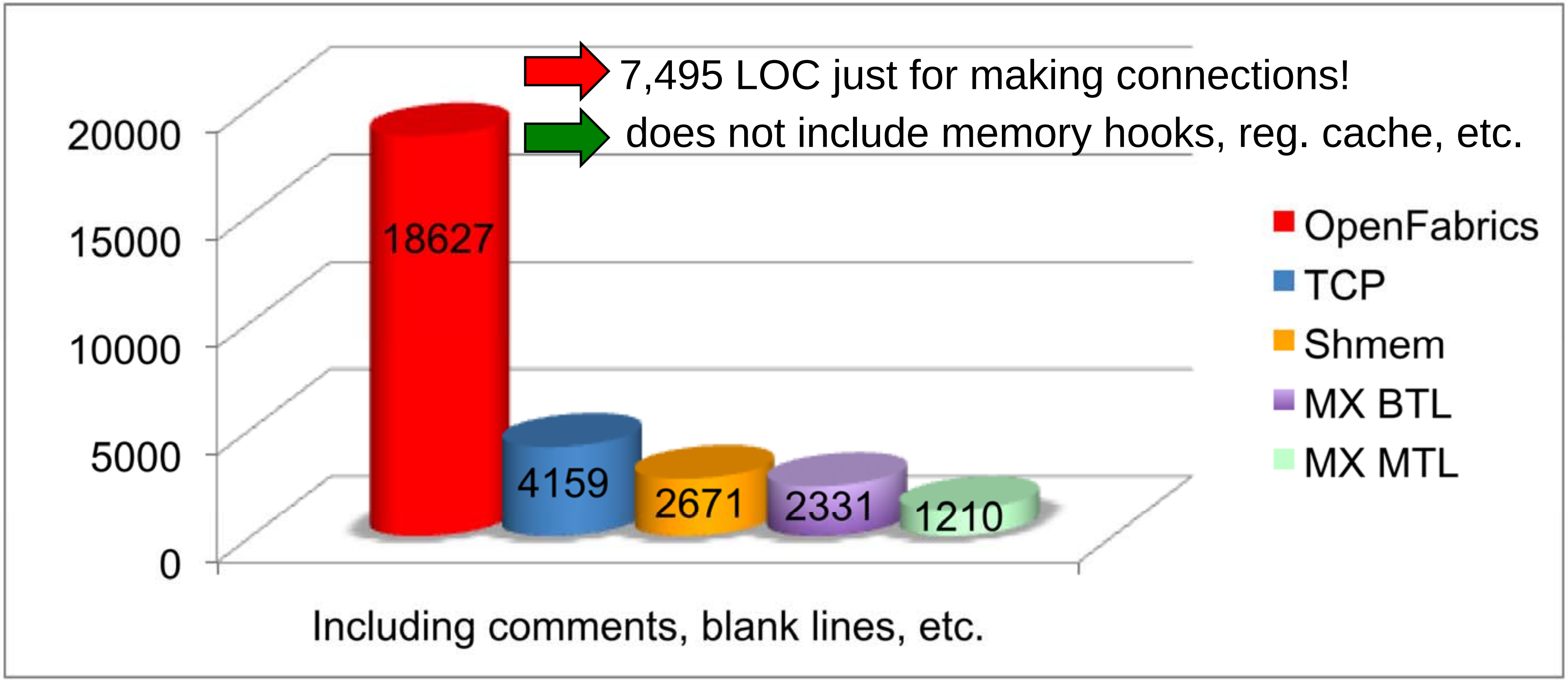
- OpenFabrics is not just different
  - It’s completely, totally, utterly, wholly different
    (I’m not saying this is fair)



Verbs API

- Very hard to write verbs-based applications
  - Significantly more complex than sockets
  - “Common” verbs practices are not well known
  - Different API stacks for different OS’s
  - Man pages are not sufficient documentation
  - Tutorials, books, programming workshops, etc.
- Does not address many needs of its biggest current customer (MPI)
  - See MPI Panel for more details (yesterday)

Example: Open MPI lines of code



7,495 LOC just for making connections!

does not include memory hooks, reg. cache, etc.

Including comments, blank lines, etc.

Lack of (performance) tools

- No equivalents to tcpdump, wireshark, …
  - Cannot tell what is happening on RNIC / HCA
- Many OF tools do not work on iWARP – why?
- Much OF validation done with MPI – why?
- Network administrators are greatly hampered
  - Wholly reliant on the vendor for support
  - Use other tools (e.g., MPI) to validate the network
- Don’t say: “it’s open source; go look yourself”

Lack of (performance) tools

- Great difficulty in answering the following common questions:
  - Why am I not getting full bandwidth?
  - Why is my 0-byte HRT latency so high?
  - Is the QP cache being thrashed?
  - Is there congestion in the network?
  - What is the queue depth utilization?
- There should be common OF tools that can answer most / all of these questions

Policy enforcement

- Network cannot force traffic to be distinct
  - No way to *force* all MPI apps to specific network parameters (e.g., MPI can pick any SL it wants)
  - TCP (iWARP) has source / destination port traffic classification
- Want to *force* MPI traffic to X, I/O traffic to Y
- This is but one example (!)

Security response

- No security team / policy in place
- Root exploit recently found in a network vendor kernel driver
  - What about the released OFEDs with this bug?
- Why doesn't security@openfabrics.org exist?
  - What is OpenFabrics' defined response?
  - When can you guarantee a fix to customers?
  - When can you guarantee a fix to OS distros?
  - How would the cross-org. coordination work?

“OFED? What is that?”

- Many customers want an OS distribution
  - Enterprise networks and filesystems (NFS (!!) …and SCSI for FCoE?) should be part of the OS
  - …so why so much effort on OFED?
- Centralized integration and testing is good
  - But OS distros re-package everything
  - They don’t want or benefit from OFED integration
- Rather than have them take our table scraps, give them what they want

Conclusions

- Push all the code upstream
  - Make it easy to use
  - Make it familiar to use
- Make OS’s be the main distribution effort
  - Actively work to give them what they want
  - Align with their schedules
  - Align with their requirements

Softiwarp

A Software iWARP Driver for OpenFabrics

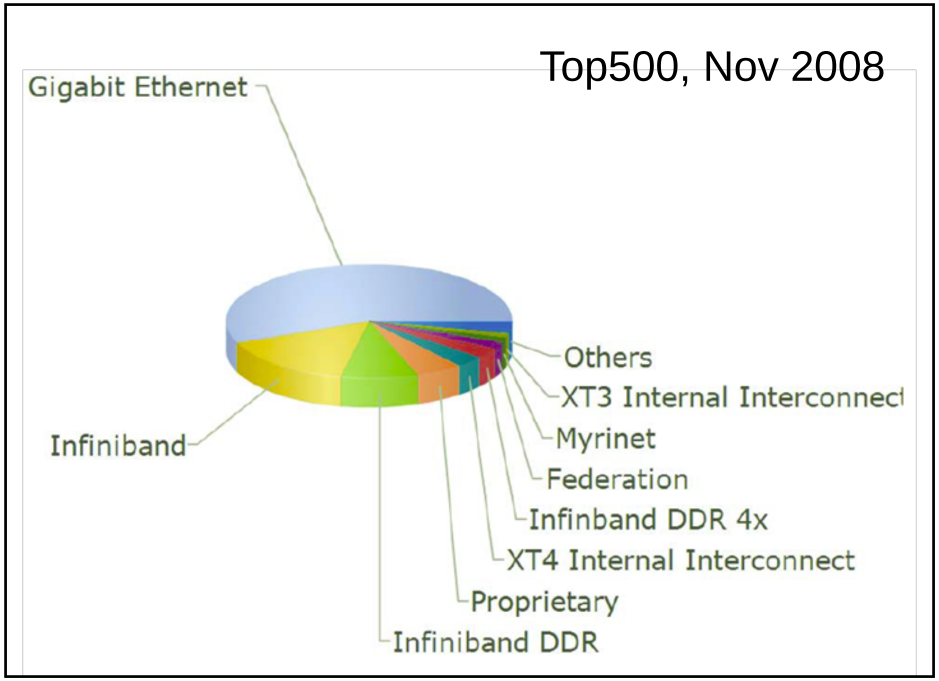
Bernard Metzler, Fredy Neeser, Philip Frey
IBM Zurich Research Lab

Contents

- Background
- What is it?
- Do we need Software RDMA?
- How is it made?
- Some first Test Results
- Feedback: OFED Issues
- Project Status & Roadmap
- Summary

Background

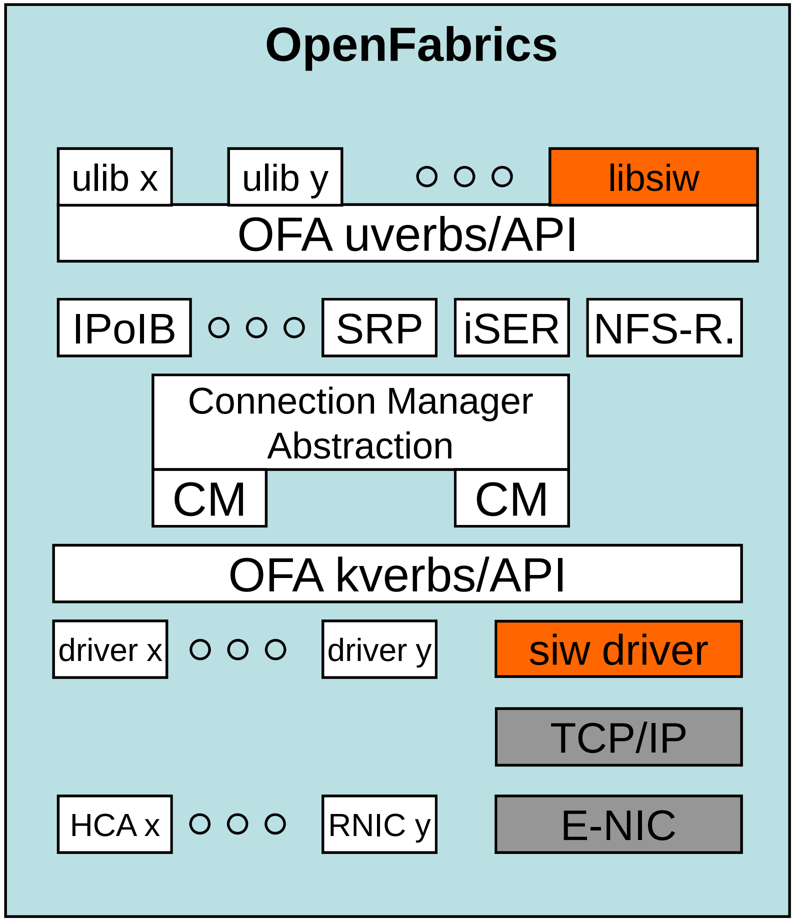
- RDMA: via, InfiniBand, iWARP
- Ethernet 1,10,100,1000,10000,40000,…MBit
- Unified Wire: Single link, single switch, single or no adapter


Top500, Nov 2008

- OpenIB
  - Focussed on InfiniBand
- OpenFabrics
  - InfiniBand + iWARP HW
  - + iWARP SW?
- IBM Zurich Research
  - RDMA API standardization
  - IETF work on iWARP
  - Software iWARP stack

Softiwarp: What is it?

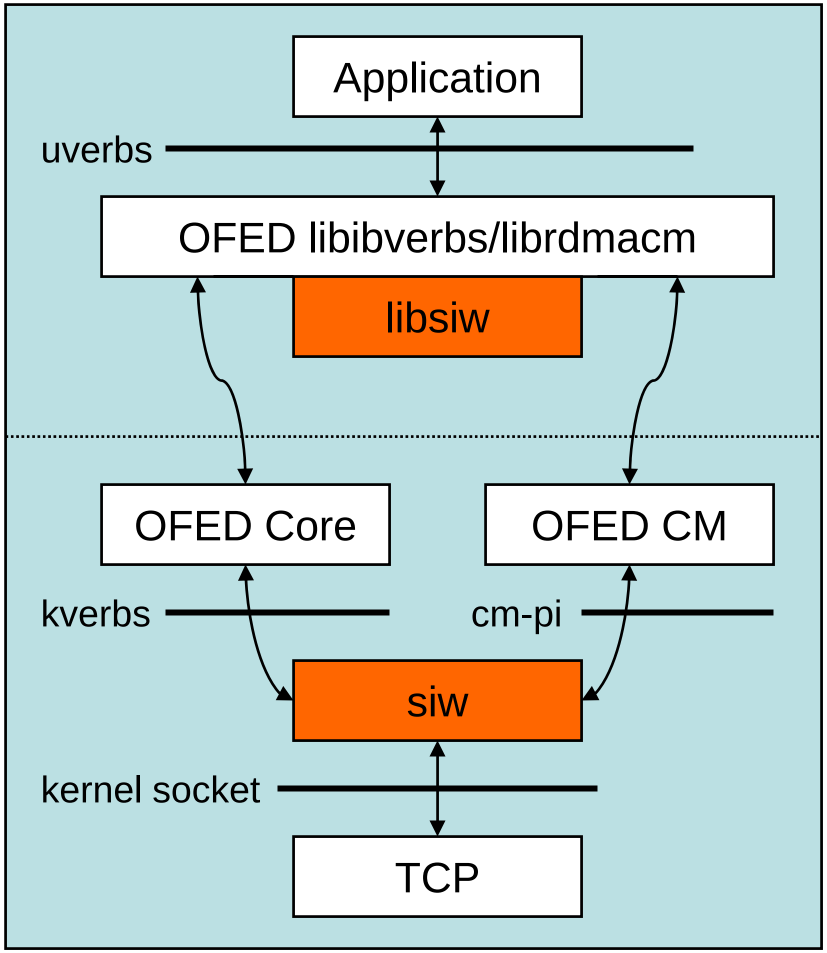
- Just another OFED iWARP driver
  - ../hw/cxgb3/, ../hw/siw,
- Purely software based iWARP protocol stack implementation
  - Kernel module
  - Runs on top of TCP kernel sockets
  - Exports OFED Interfaces (verbs, IWCM, management, …)
- Client support
  - Currently only user level clients
  - libsiw: user space library to integrate with libibverbs, librdmacm
- Current build
  - OFED 1.3
  - Linux 2.6.24



OFED and Kernel Integration

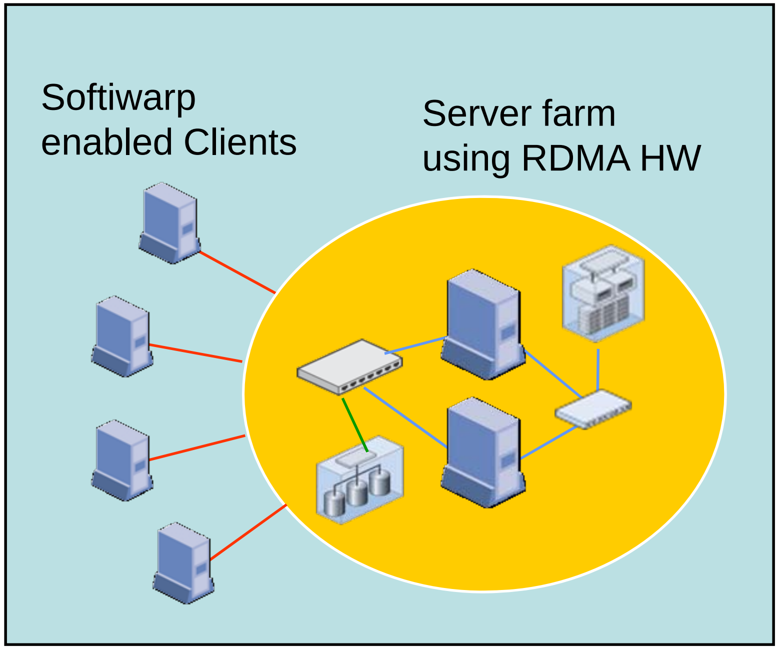
Approach: **Keep things simple and standard**

- TCP interface: Kernel Sockets
  - TCP stack completely untouched
  - Non-blocking write() with pause and resume
  - softirq-based read()
- Linux Kernel Services
  - List-based QP/WQE management
  - Workqueue-based asynchronous sending/CM
  - …
- OFED interface
  - verbs,
  - Event callbacks,
  - Device registration
- Fast Path
  - No private interface between user lib and kernel module
  - Syscall for each post(SQ/RQ) or reap(CQ) operation



Why RDMA in Software?

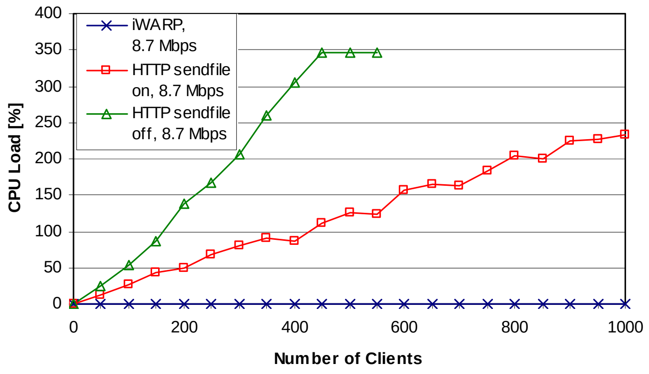
- Enable systems without RNIC to speak RDMA
  - Conventional ENIC sufficient
  - Peer with real RNICs
    - Help busy server to offload
    - Speak RDMA out of the Cluster
    - Enable real RNICs(!)
  - Benefit from RDMA API semantics
    - Application benefits
      - Async. comm., parallelism
      - One-sided operations
    - CPU benefits
      - Copy avoidance in tx
      - Named buffers in rx
- Early system migration to RDMA
  - Migrate applications before RNIC avail.
  - Mix RNIC equipped systems with ENICs
- Test/Debug real HW
- RDMA transport redundancy/failover
- Help to grow OFED Ecosystem for Adoption and Usage beyond HPC

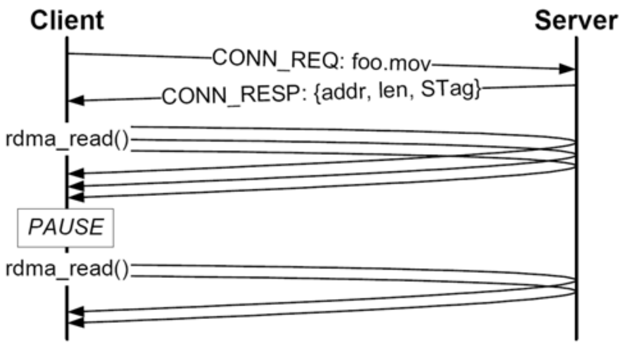


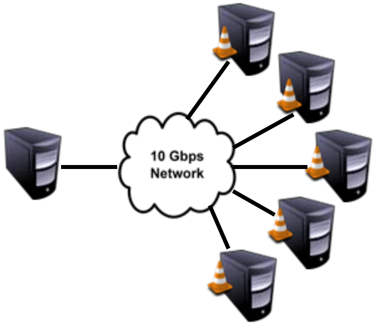
RDMA Use Case != HPC

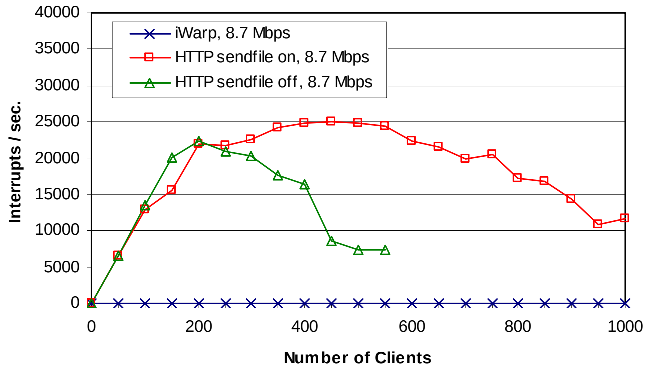
Multimedia Data Dissemination via RDMA

- RNIC-equipped video server, Chelsio t3 10Gb
  - Complete content in Server RAM
  - IBM HS21 BladeServers (4core Xeon 2.33 GHz, 8GB Mem.)
- Up to 1000 VLC clients to pull FullHD (8.7Mbps)
  - VLC client extended for OFED verbs
  - Client may seek in data stream
- HTTP get (Apache w/sendfile()) or RDMA READ
  - Service degradation w/o sendfile
  - Increasing load with sendfile
  - Zero server CPU load for RDMA
- Very simple pull protocol for RDMA
  - Minimum iWARP server state per Client: RDMA READ!



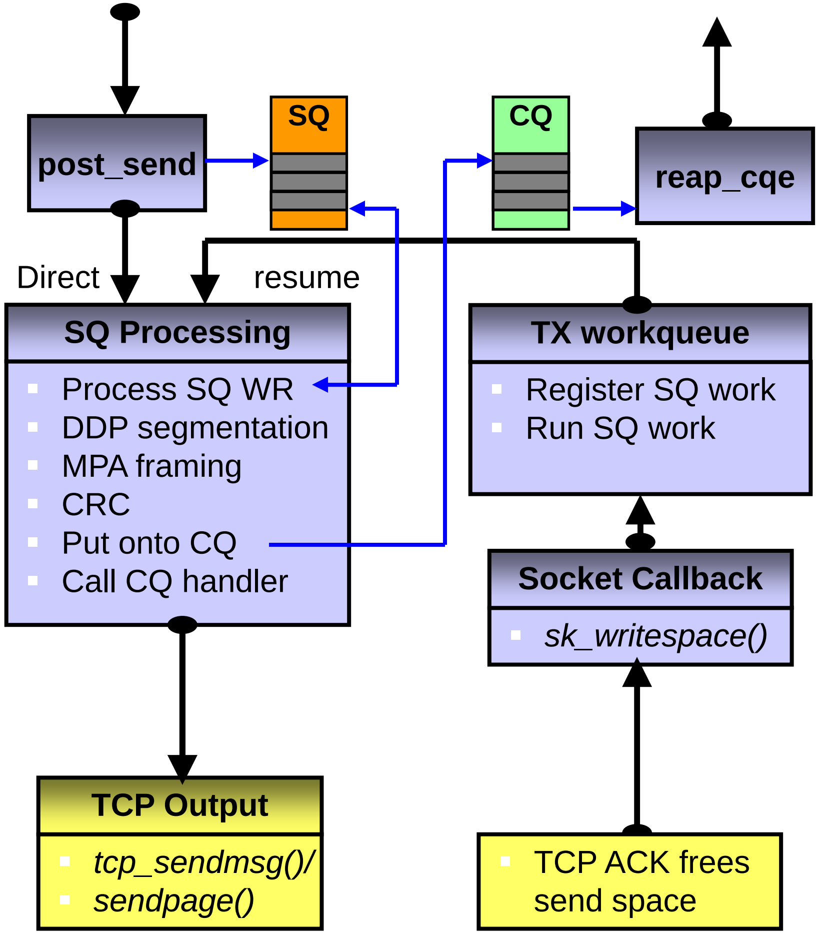






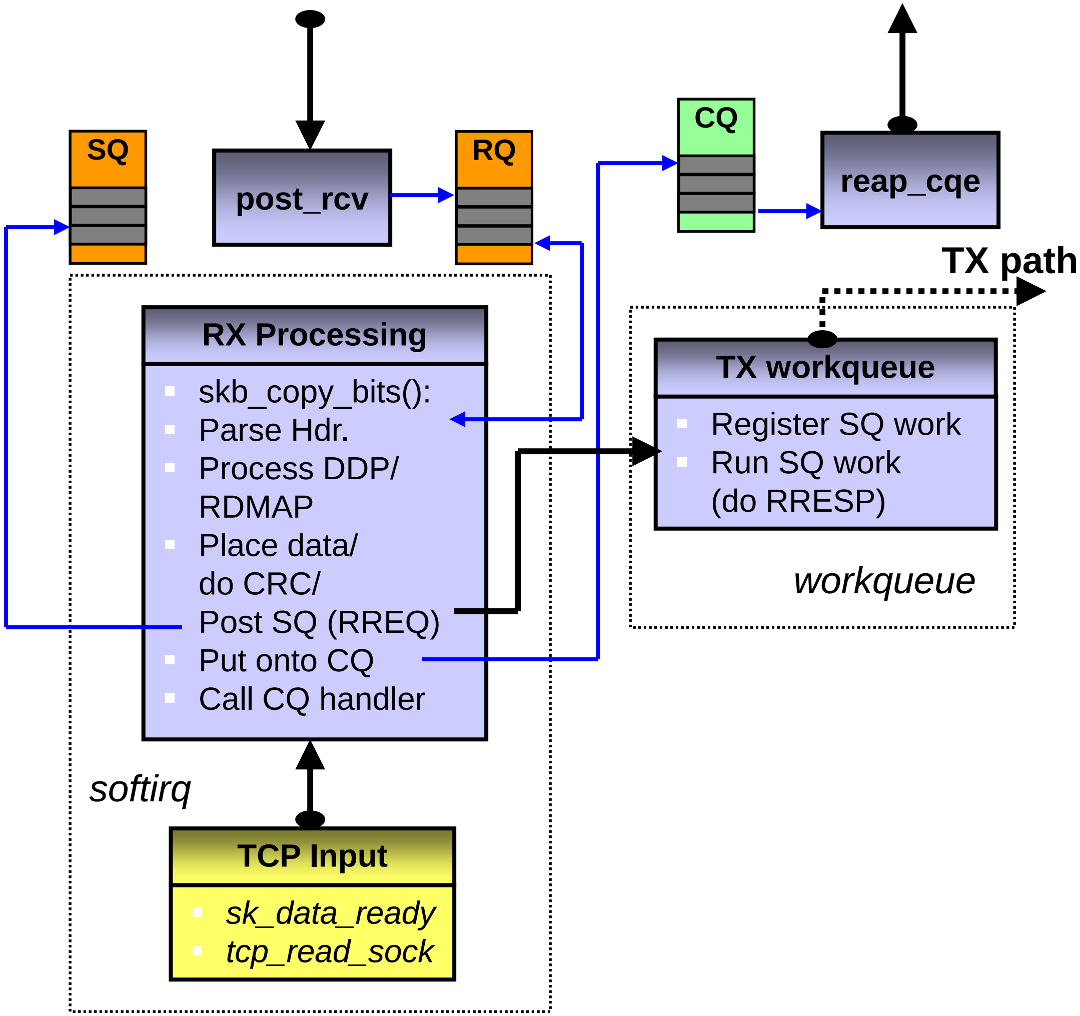
Softiwarp TX Path Design

- Syscall through OFED verbs API to post SQ work
- Synchronous send out of user context if socket send space available
- Nonblocking socket operation:
  - Pause sending if socket buffer full
  - Resume sending if TCP indicates sk_writespace()
    - Use Linux workqueue to resume sending
- Lock-free source memory validation on the fly
- sendfile()-semantic possible
- Post work completions onto CQ
- Reap CQE’s asynchronously



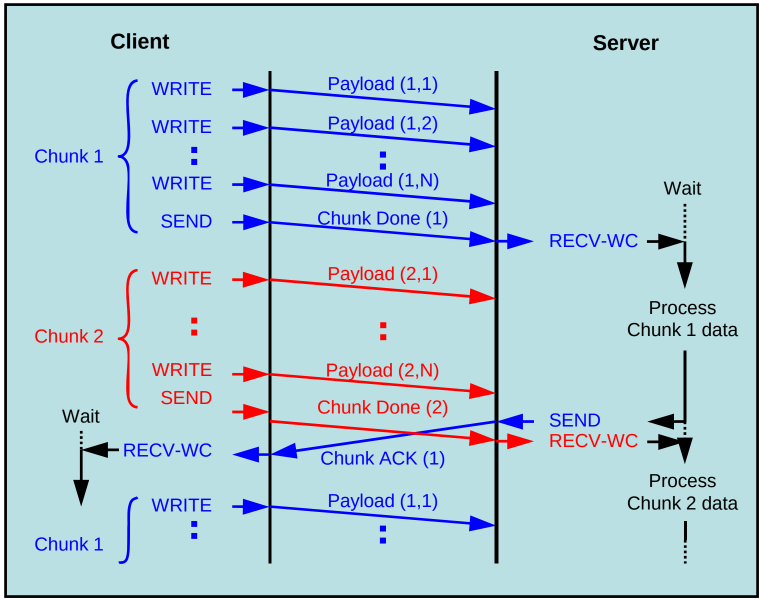
Softiwarp RX Path Design

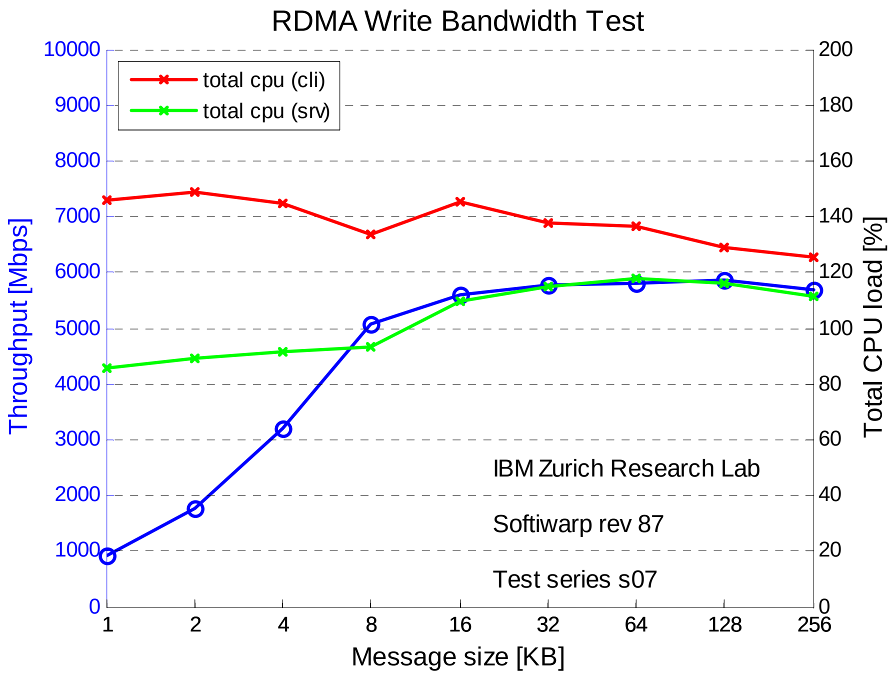
- All RX processing done in softirq context:
  - in sk_data_ready() upcall:
  - Header parsing
  - RQ access
  - Immediate data placement
  - CRC
  - No context switch
  - No extra thread
- Lock-free target memory validation on the fly
- Inbound RREQ just posted at SQ + SQ processing scheduled to resume later



First Tests: Softiwarp

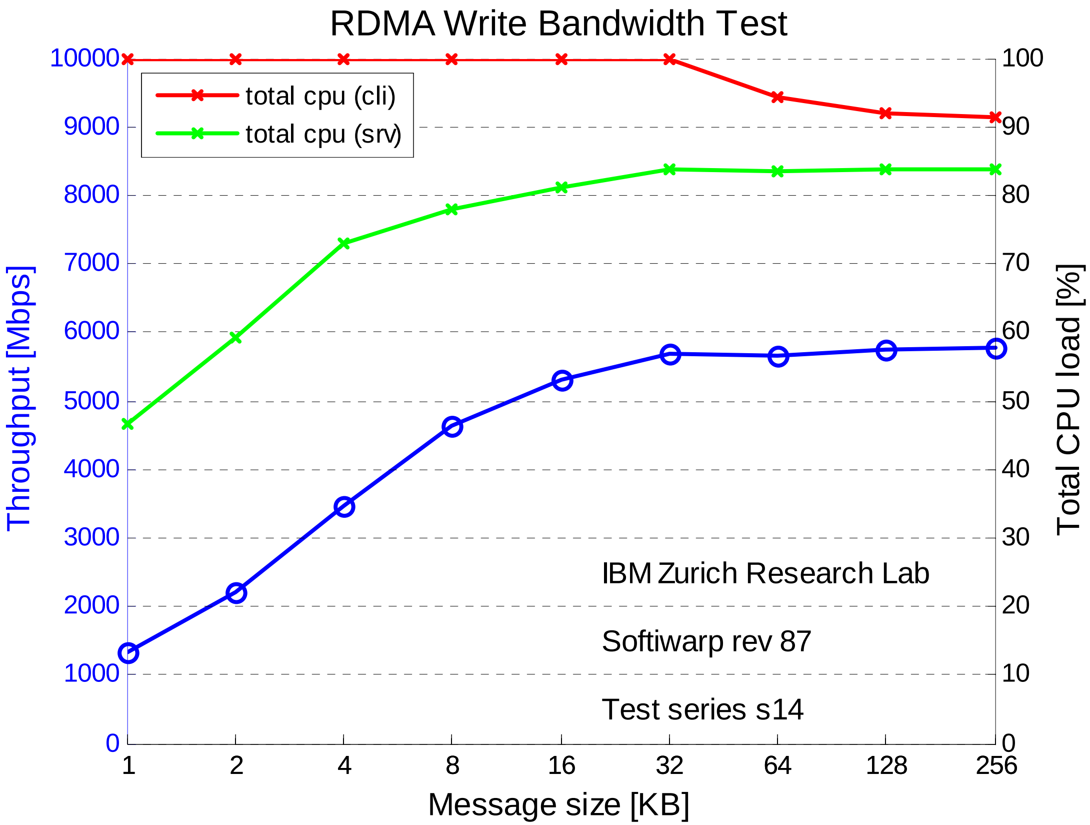
- Non-tuned software stack on both sides
- Application level flow control (ping-pong buffers)
- SEND‘s for synchronization
- 1 connection

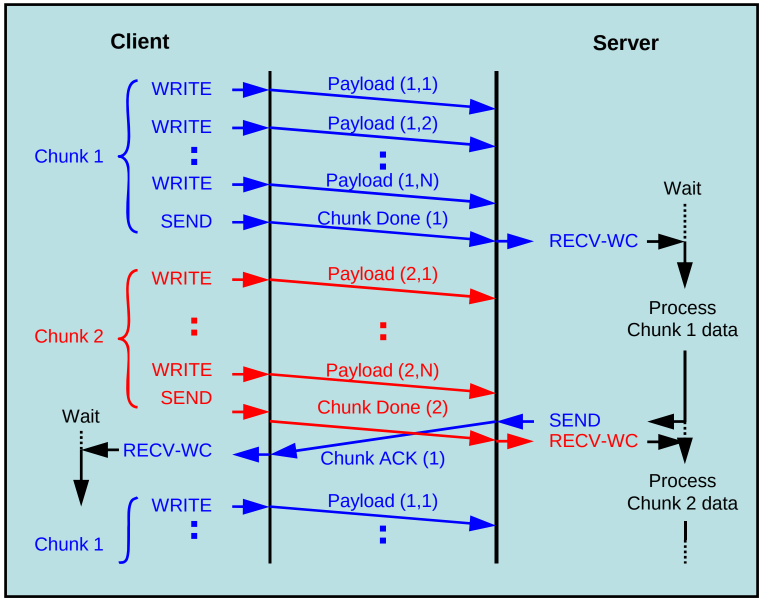




Write/read application data: off
MPA CRC32C: off
MTU = 9000

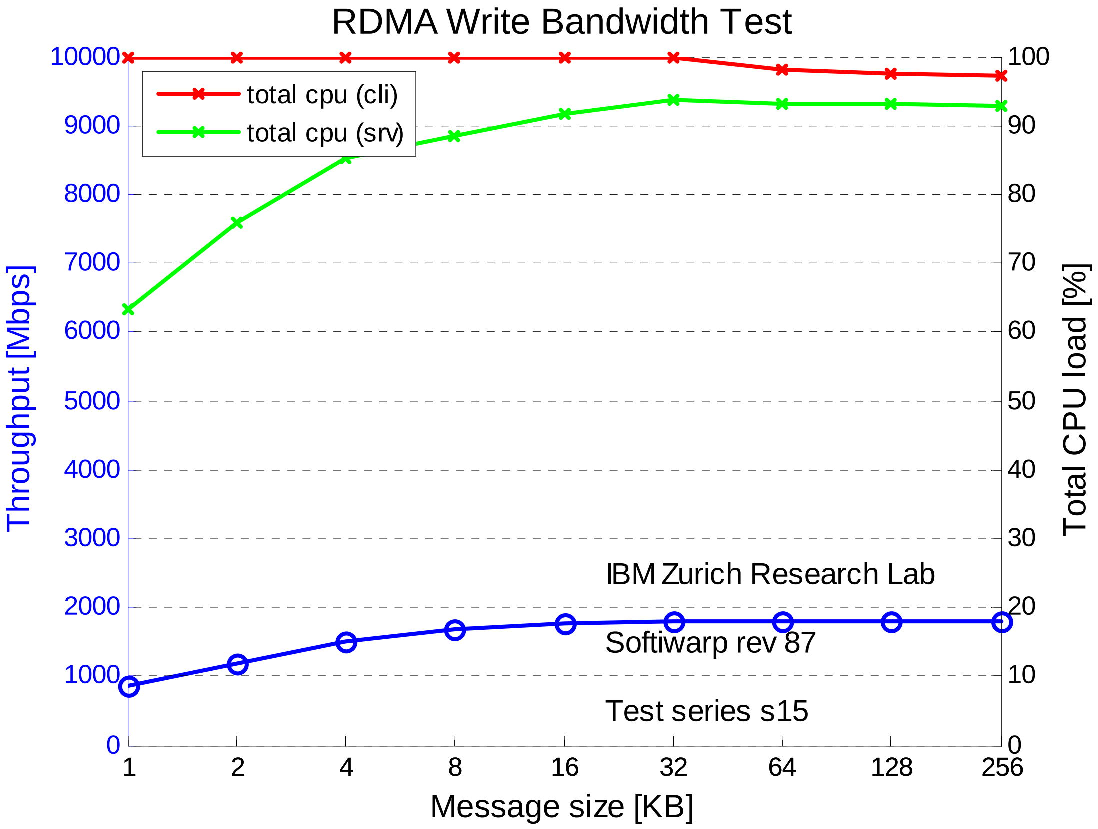
First Tests: Softiwarp

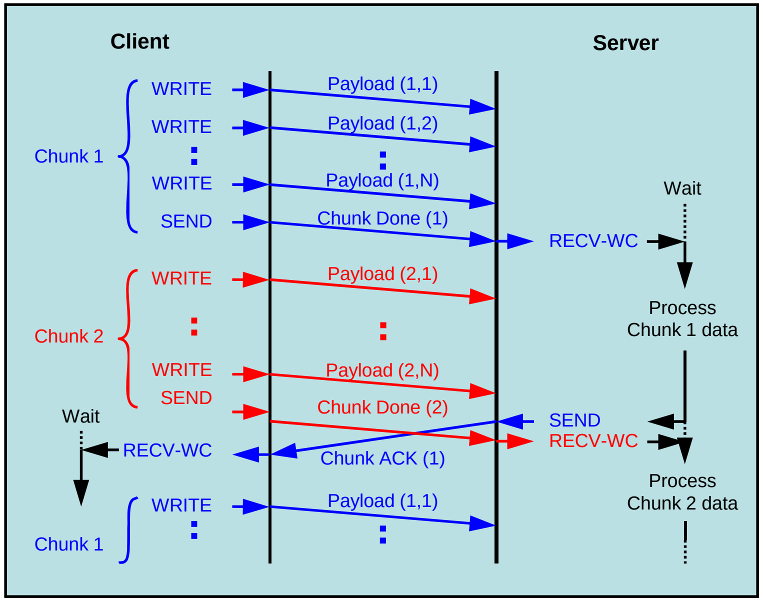




- Same application level flow control (ping-pong buffers) +
  - 1 Core only
  - MPA CRC off
  - MTU=9000
- Sending CPU on its limit

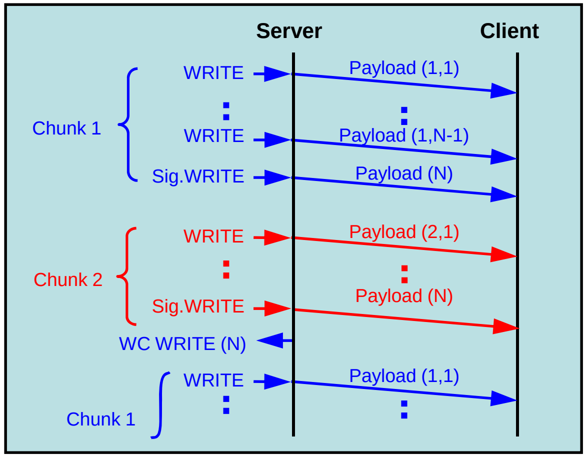
First Tests: Softiwarp + CRC

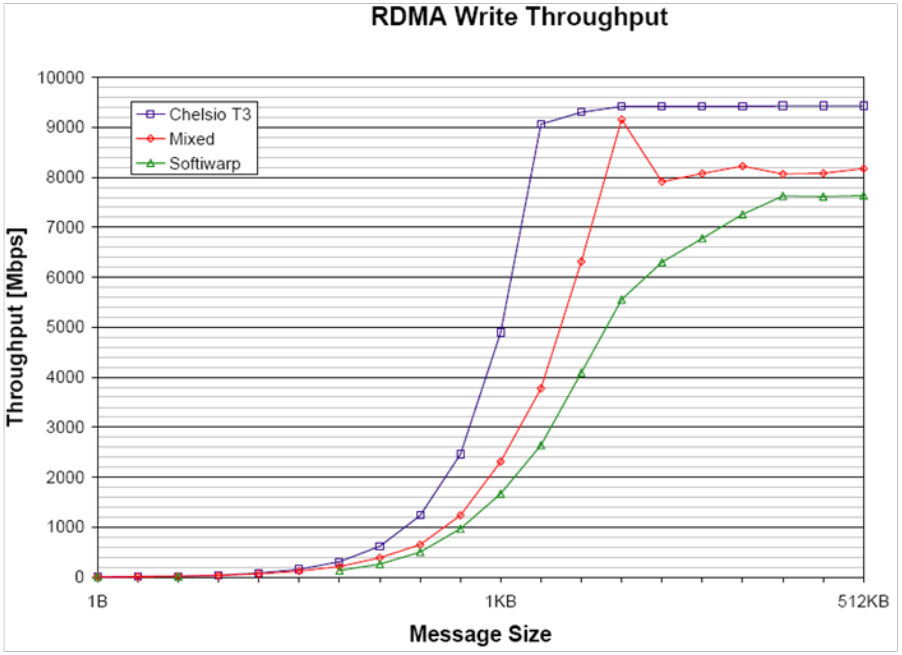




- Same application level flow control (ping-pong buffers) +
  - 1 Core only
  - **MPA CRC ON**
  - MTU=9000
- CRC is killing performance
- Still sending CPU on its limit

First Tests: Softiwarp-Chelsio





- Test 1: Softiwarp peering Chelsio T3
  - Setup:
    - RNIC sends WRITEs to Softiwarp target
    - 512KB kernel socket receive space
  - Result:
    - Close to line speed at 8KB
    - Uups - some issues at larger buffers

- Test 2: Softiwarp peering Softiwarp
  - Same setup
  - Result:
    - Maximum Bandwidth from 128KB on

- Conclusions:
  - Promising for first test on non-tuned stack
  - Software stack may server well on client side
  - Further improvement with sendfile() possible

Softiwarp: Work in progress

Core Functionality

RDMAP/DDP/MPA	x
QP/CQ/PD/MR Objects	x
Send	x
Receive	x
RDMA WRITE	x
RDMA READ	x
Connection Mgmt (IWCM, TCP)	x
Memory Management	x

Features (incomplete)…

MPA CRC	x
MPA Markers	-
Memory Windows	w
Inline Data	w
Shared Receive Queue	-
Fast Memory Registration	-
Termination Messages	w
Remote Invalidation	-
Stag 0	-
Resource Realloc. (MR/QP/CQ)	-
TCP header alignment	w
Relative adressing (ZBVA)	w

Softiwarp Roadmap

- Opensource very soon
- Discuss current code base in the community
  - Be open for changes/critics
  - Identify core must-haves which are missing
  - Stability!
  - Invite others to contribute
  - Feedback known issues of OFED core to team
  - Don’t touch TCP
- Start compliance testing (OFA IWG) soon
- Investigate private fast path user interface option
- Start working on kernel client support
- Investigate partially offloading of CPU intensive tasks
  - CRC, tx-markers
  - Data placement,..

Feedback: OFED Issues

- Late RDMA Transition
  - Something not part of RNIC integration is now possible
  - Very simple to do with Softiwarp, but:
  - OFED does not support TCP handover for good reasons
    - …think about iSER & Co
- OFED CM
  - How to coexist with RNIC if SW stack shares link, shall we?
  - Can we exist within OFED w/o full (complex) IWCM support?
- Device Management
  - Wildcard listen on multiple interfaces used by Softiwarp
- Zero based virtual adressing
- …

Summary

- Software RDMA is useful
- Software RDMA is efficient on client side (at least)
- RDMA semantics help to use transport efficiently
- Softiwarp helps to grow RDMA/OFED ecosystem
  - Establish RDMA communication model
  - Prepare applications to use RDMA
  - Prepare systems to introduce RDMA HW
  - Peer & thus enable RDMA HW
- Softiwarp is work in progress
  - Please join.