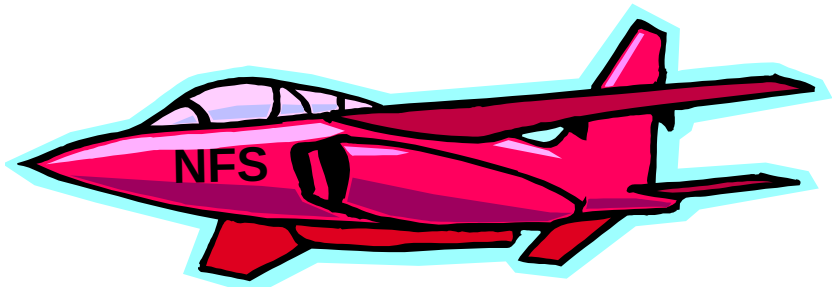


NFS over RDMA

Brent Callaghan, Theresa Lingutla-Raj,
Alex Chiu, Peter Staubach, Omer Asad

Sun Microsystems, Inc.

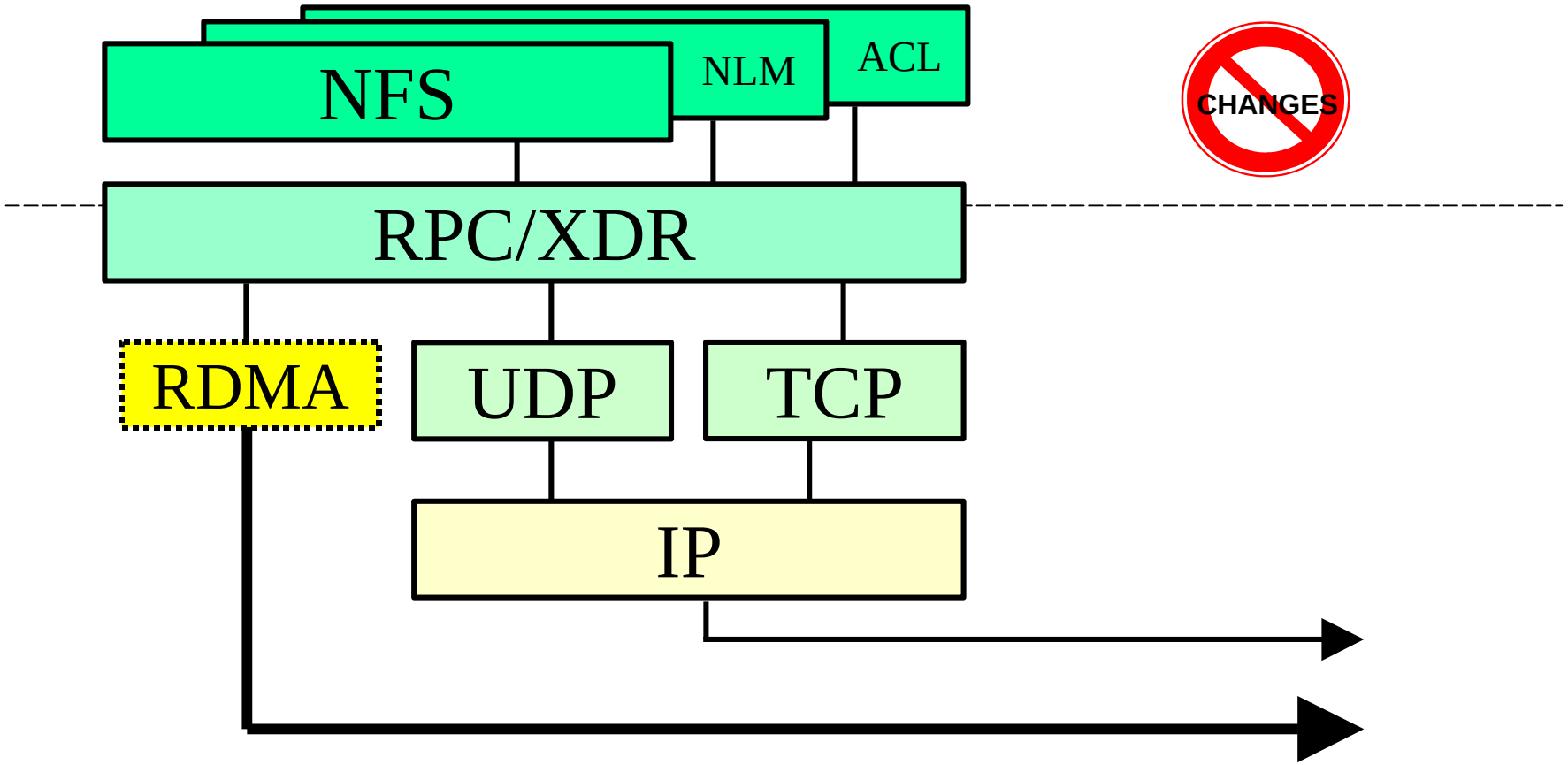
Why RDMA as a Transport?

- Nice to have at 1 Gb/sec but *must* have for 10 Gb/sec
- Offload protocol processing from general purpose CPU to dedicated protocol hardware
- Offload host memory/IO bus with direct data placement (DDP)

NFS is an RDMA Sweet Spot

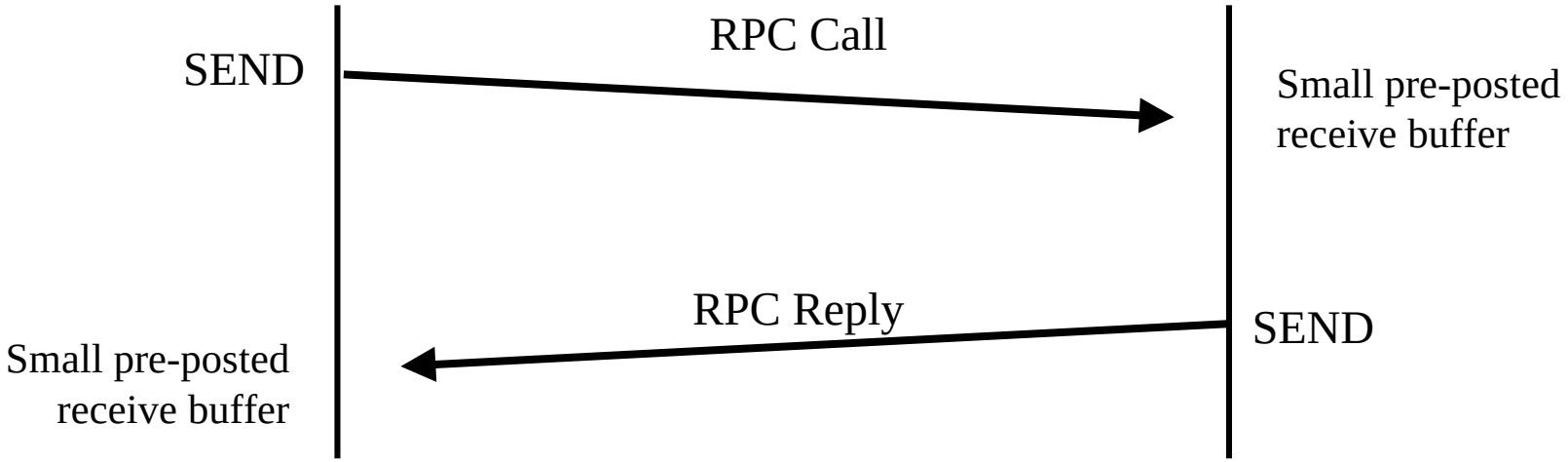
- Clients and servers are close
  - Most commonly on a LAN
  - Often in the same server room or rack
  - Bandwidth high - latency low
- NFS moves big chunks of data
  - 8 KB for NFS version 2
  - No limit for NFS version 3
    - Most clients read & write 32 KB chunks
    - Solaris servers accept up to 1 MB reads/writes

RDMA as a new RPC Transport



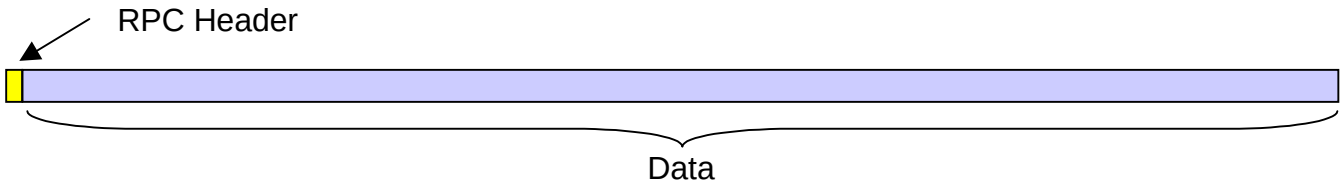
Small RPC Messages

- Most NFS messages are quite small
  - Less than 1 KB
- No RDMA needed - just use SENDs

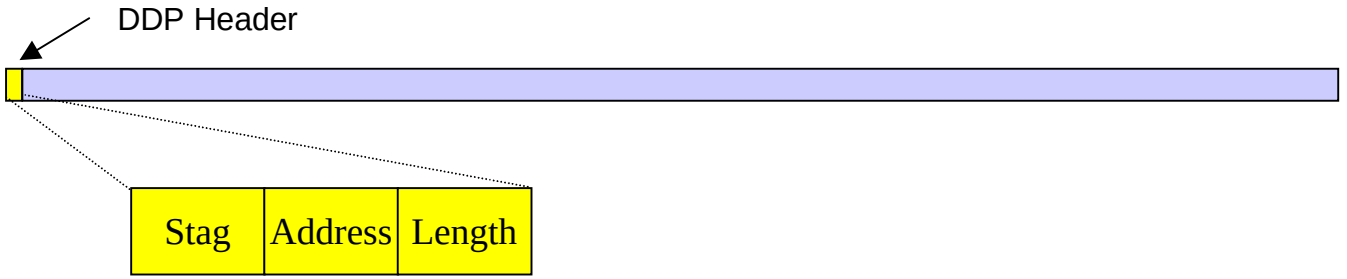


Moving NFS data with RDMA

An NFS read reply or write request is a large chunk of data with a variable length RPC & NFS header.

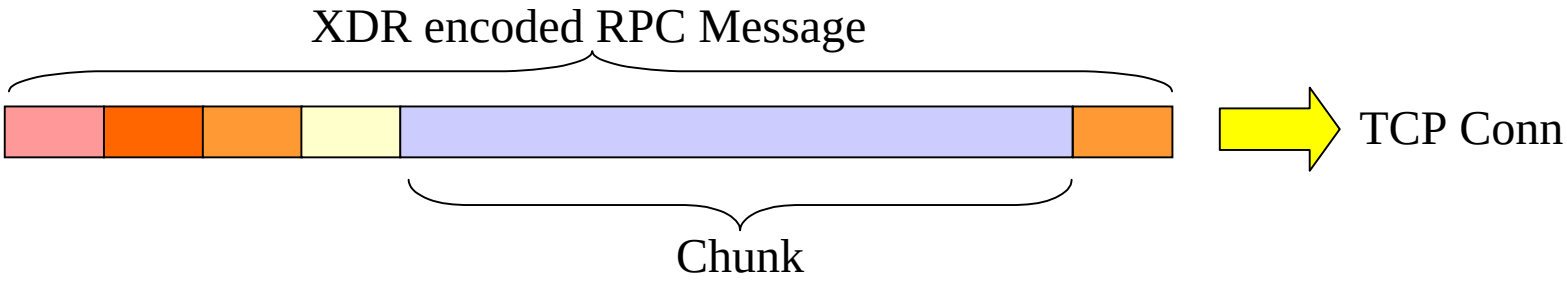


That large chunk of data could be moved more efficiently if we could move it instead with DDP.

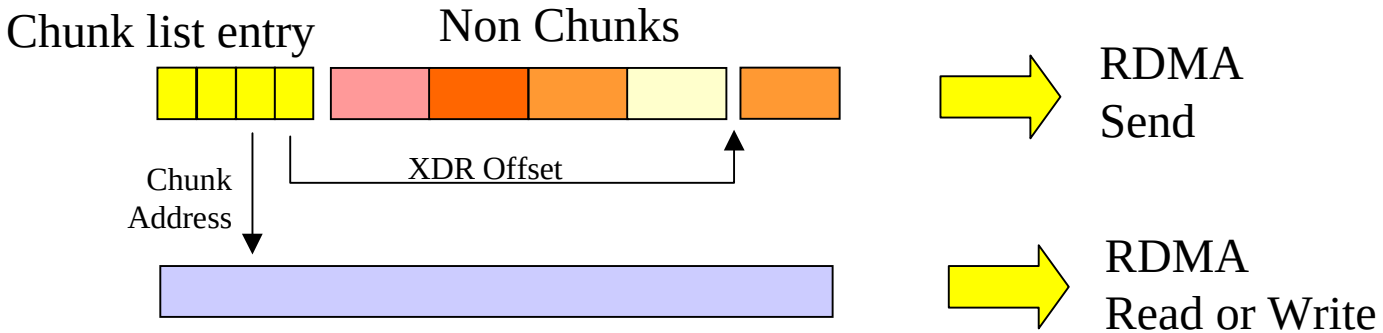


XDR De-Chunking the Message

- Encoded message for TCP transport

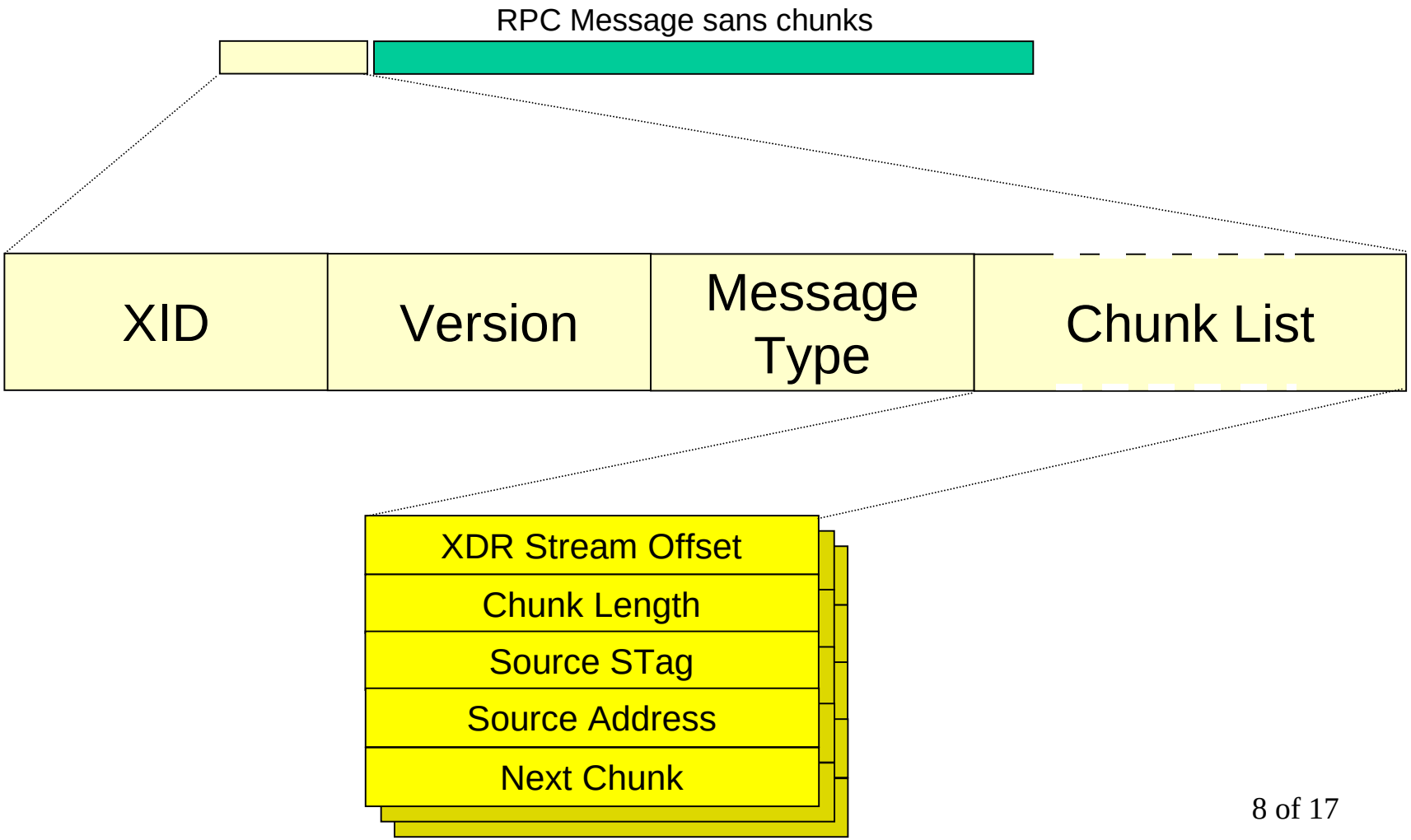


- Encoded message for RDMA transport

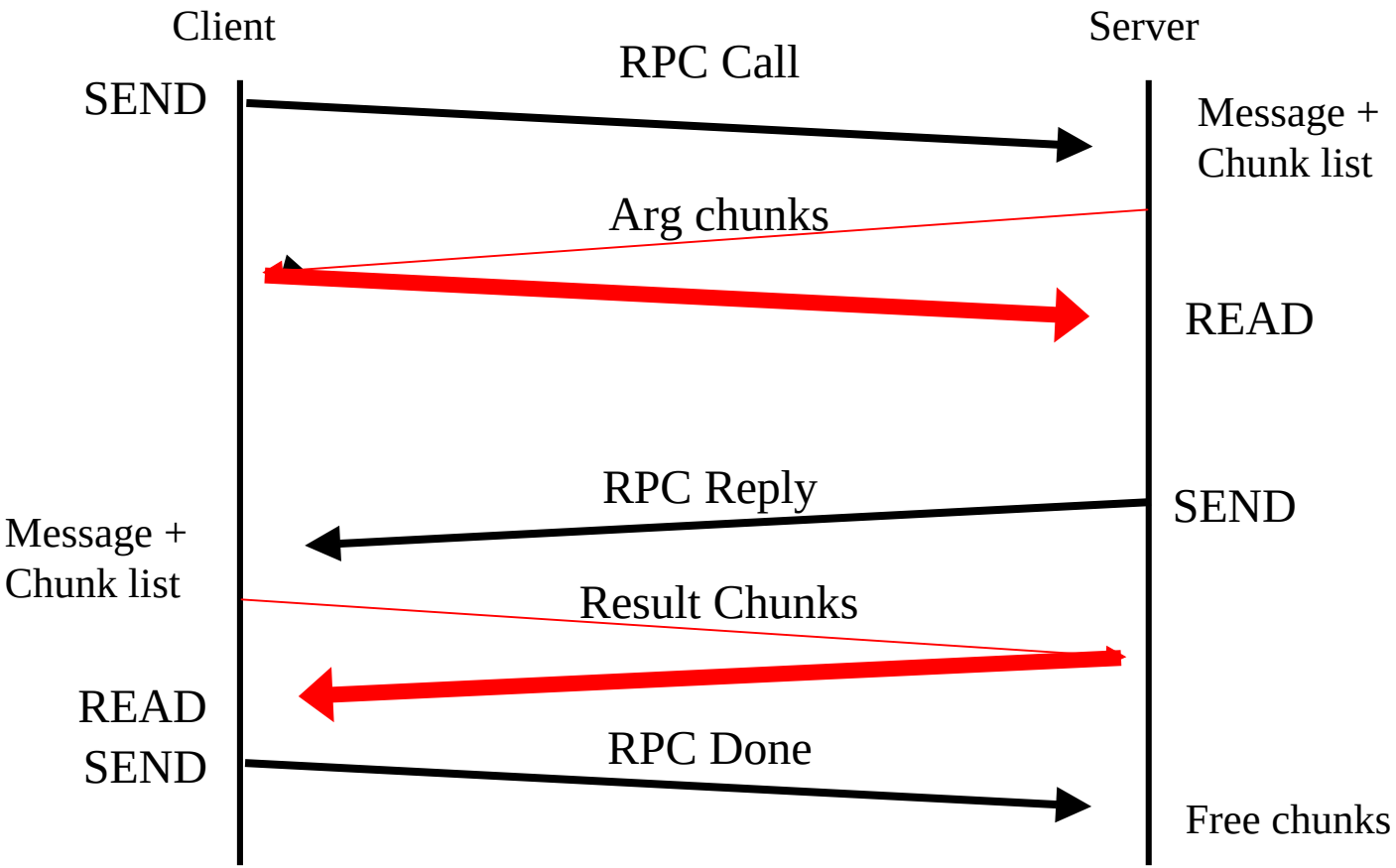


NFS over RDMA

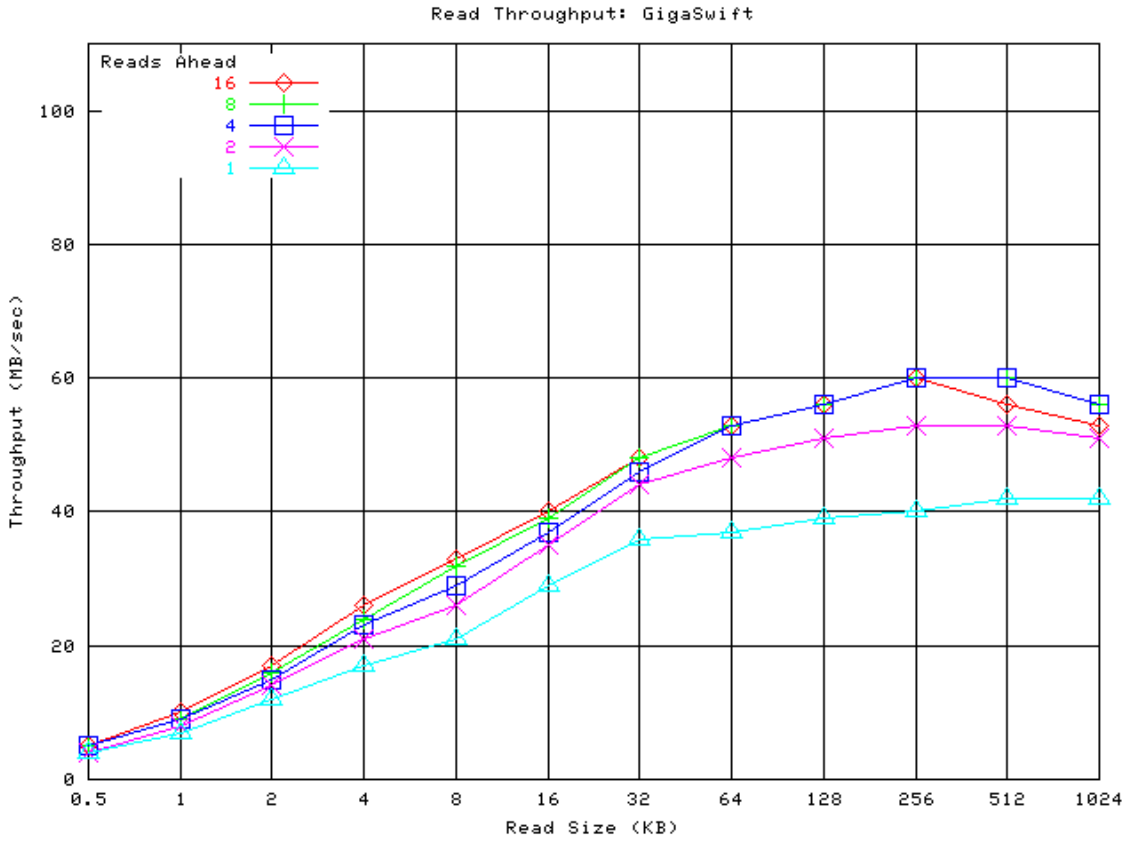
RDMA Transport Header



Read-Read Protocol

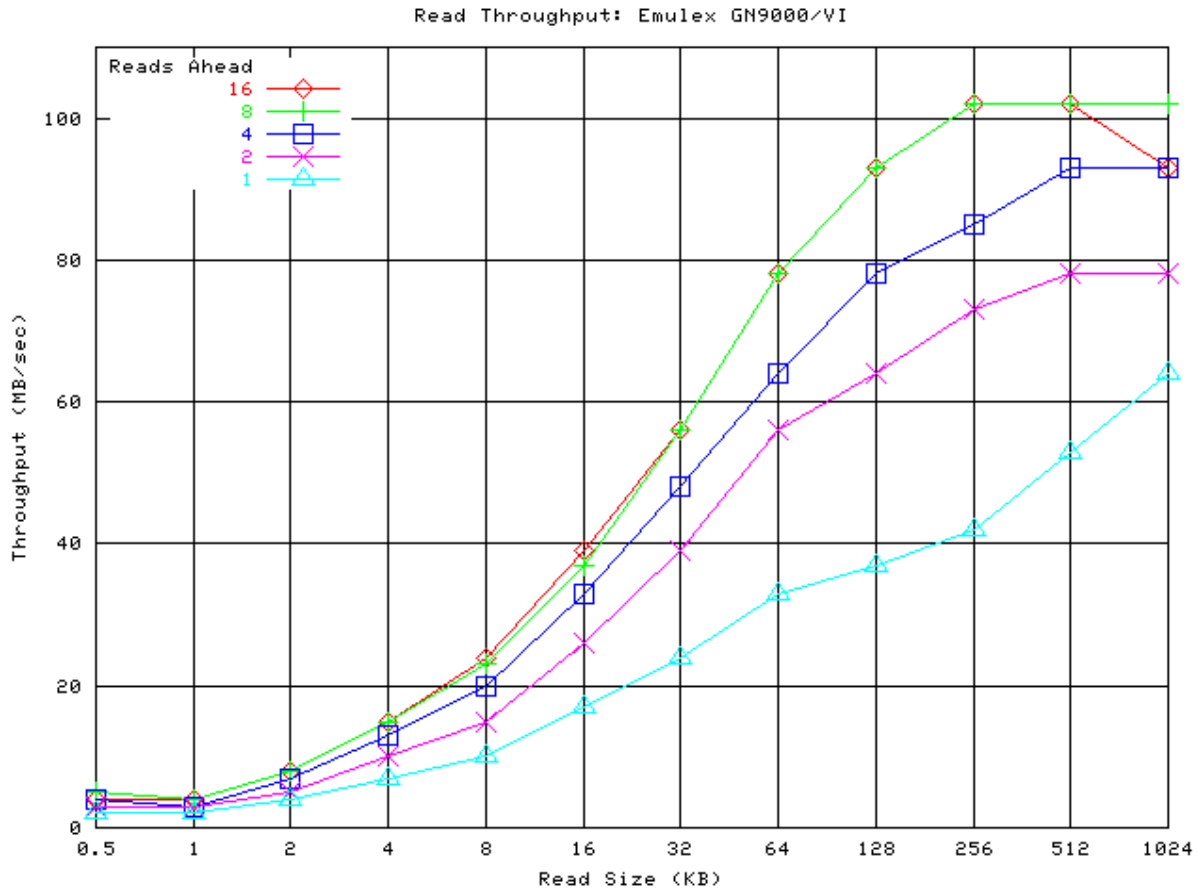


NFS/TCP Throughput



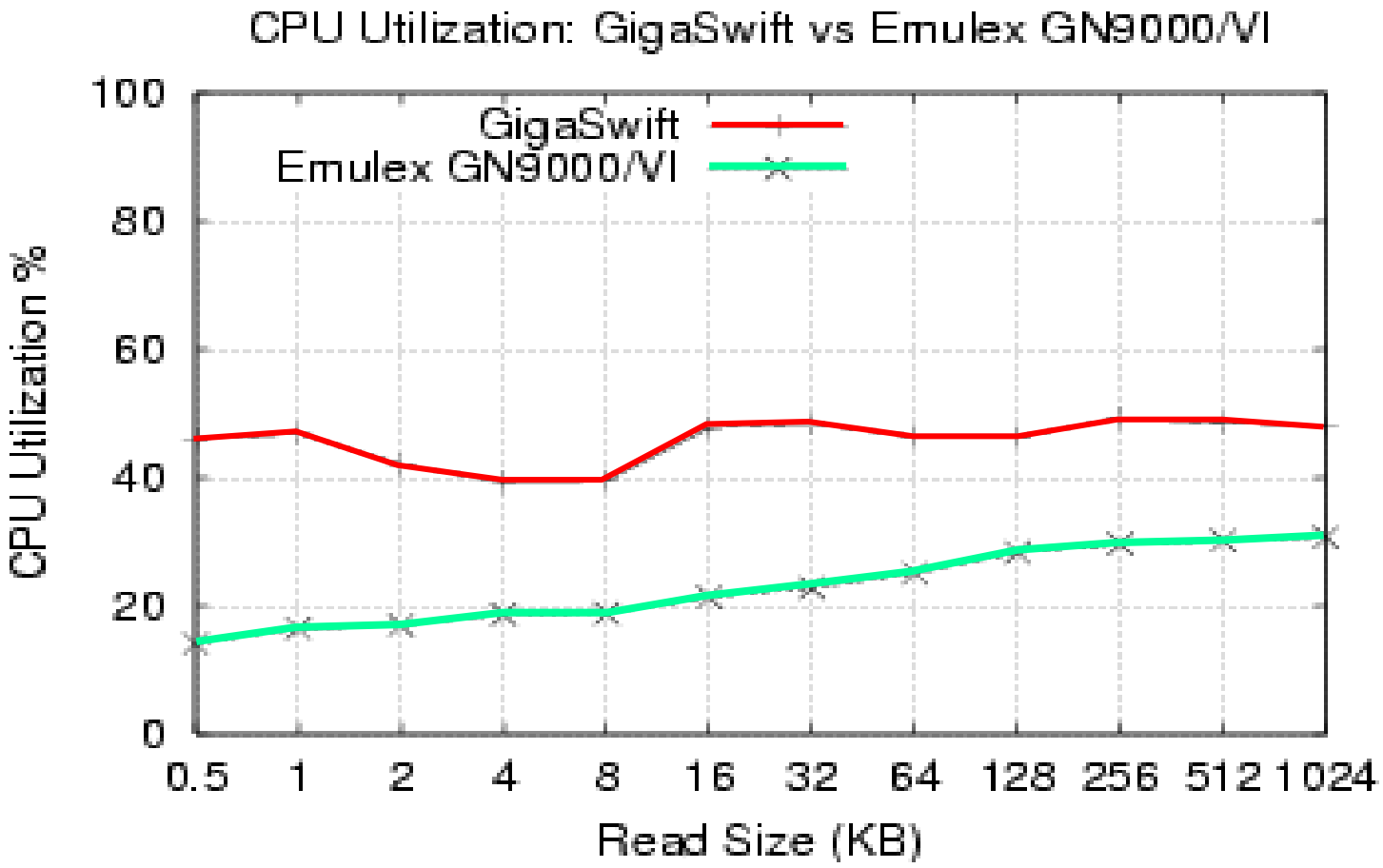
Peak throughput 60 MB/sec @ 256 KB reads & 4 reads-ahead

NFS/RDMA Throughput



Peak throughput 102 MB/sec @ 256 KB reads & 8 reads-ahead

CPU Utilization



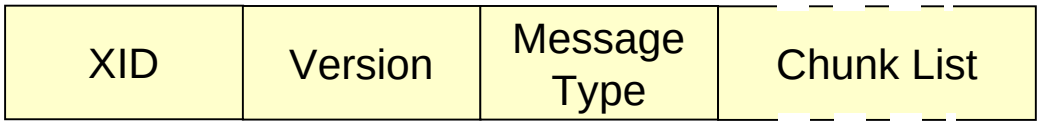
(with no async read-ahead)

Further Work

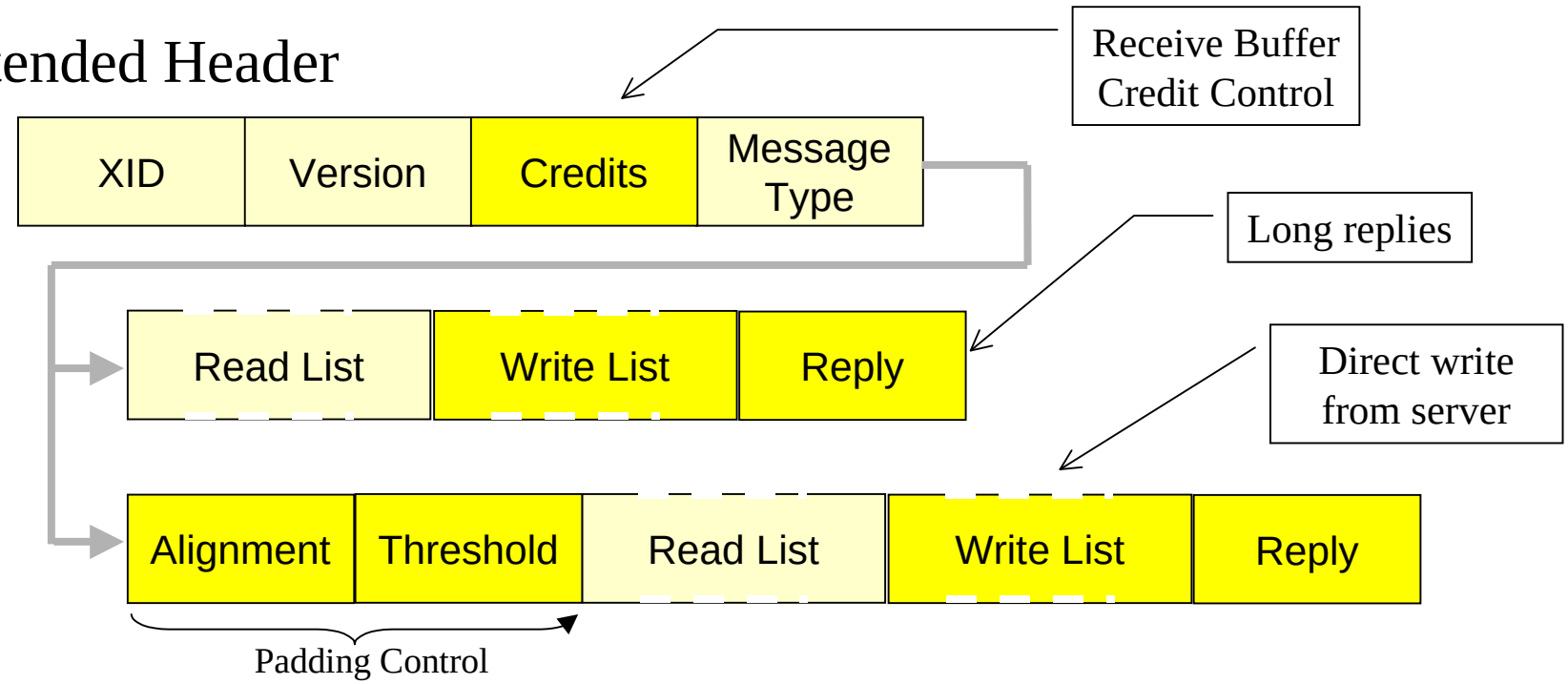
- NFS/RDMA protocol Internet Drafts submitted to IETF
- Extends basic “read-read” protocol to use RDMA write with ULP hooks: “read-write”
- Includes receive buffer request/grant credit control
- Support for alignment padding in RDMA SENDs
- Receive buffer size negotiation protocol
- Support in NFS version 4.1

Extended RDMA Transport Header

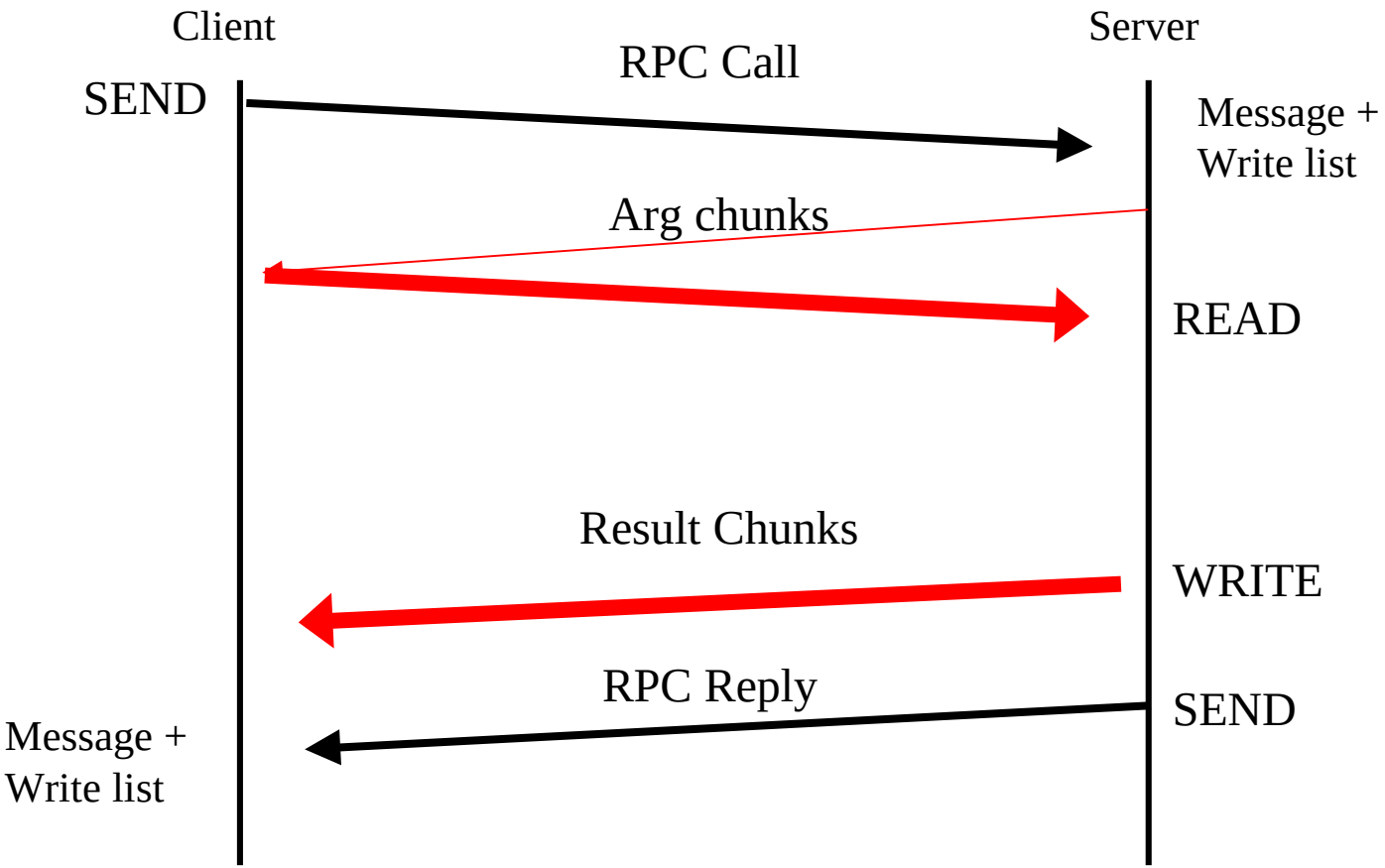
Old Header



Extended Header

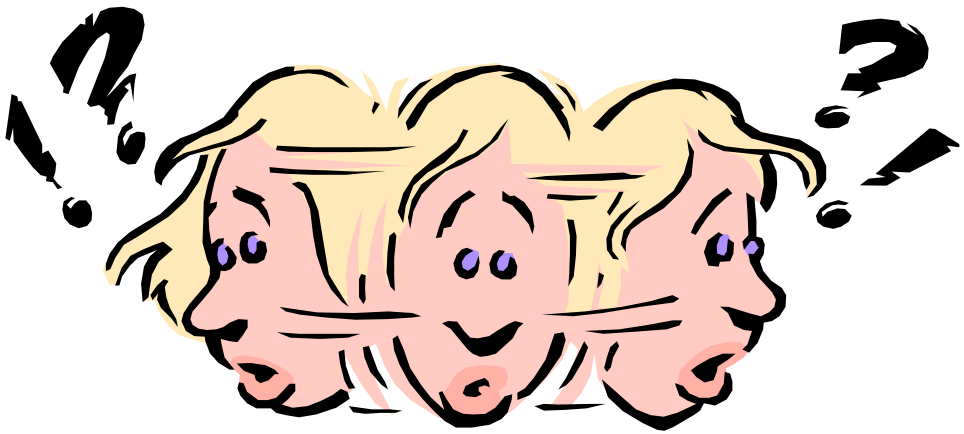


Read-Write Protocol



Project Status

- Solaris prototype
  - kVIPL with Emulex GN9000/VI, 1Gb link
  - Like a normal NFS mount
  - Demonstrated good performance
- Infiniband
  - Implementing extended “read-write” protocol
  - Mellanox Tavor, 10 Gb (4x) link
  - Evaluating performance



Questions & Answers