A Performance Analysis of the iSCSI Protocol

Stephen Aiken, Andrew Pleszkun, Dirk Grunwald, Jesse Willeke

Colorado Center for Information Storage
University of Colorado, Boulder

Goals

- Examine the Overall Performance of the iSCSI Protocol in a Number of Different Configurations

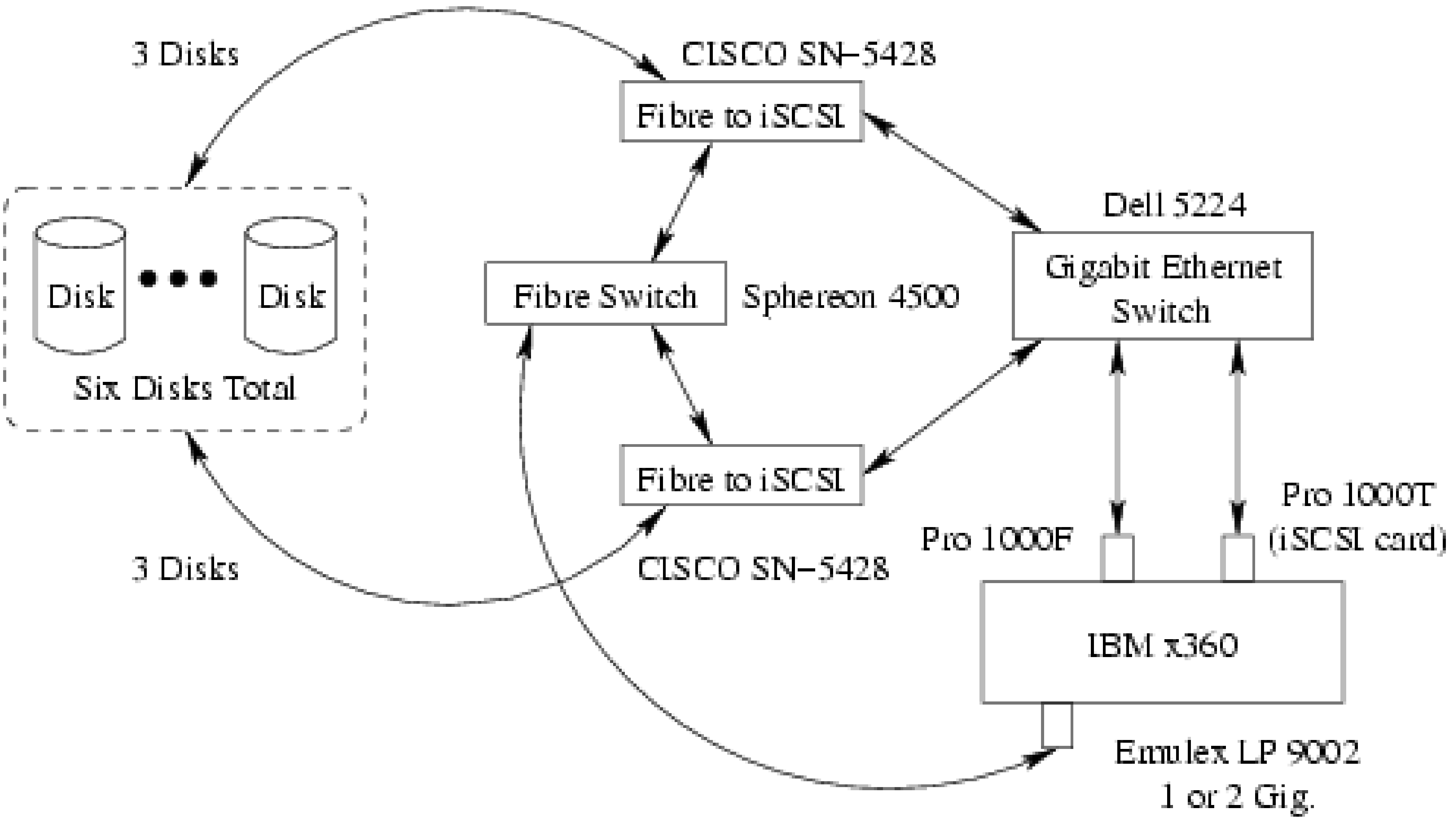
Experimental Configurations

- Commercial Deployment
  - Hardware Target, Software Initiator
  - Hardware Target, Hardware Initiator
- Inexpensive Software Deployment
  - SAN, Gigabit Ethernet
  - WAN, Fast Ethernet

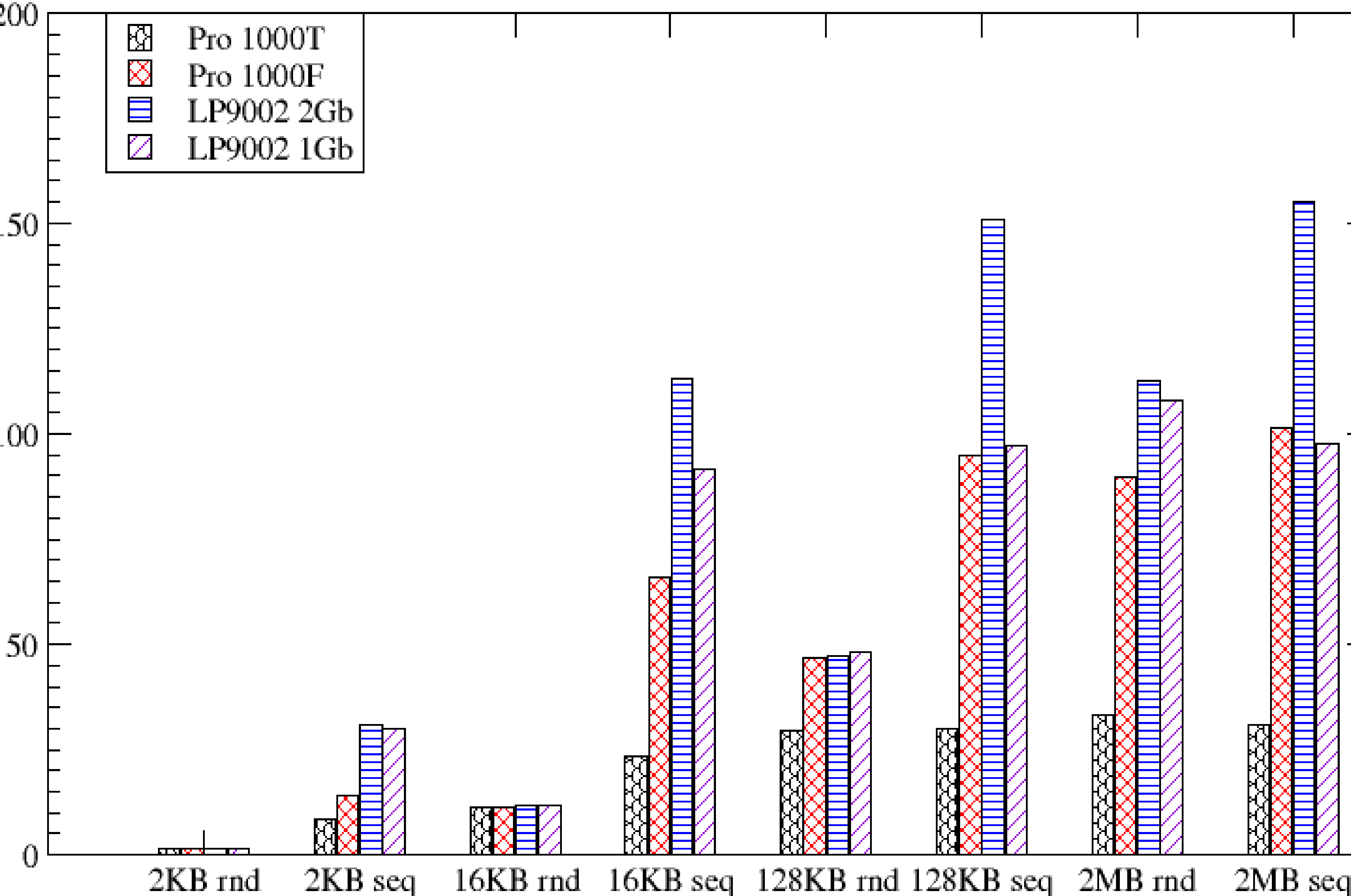
Commercial Deployment – Experimental Design

- Wanted to compare iSCSI to fibre channel
  - Over a 1Gbps connection
  - Software-based iSCSI on 1Gb ethernet
  - Specialized hardware iSCSI HBA
- Tested overall disk throughput
  - 1GB files used
  - Sequential reads/writes
  - Randomized access
  - Examine throughput for varying block sizes

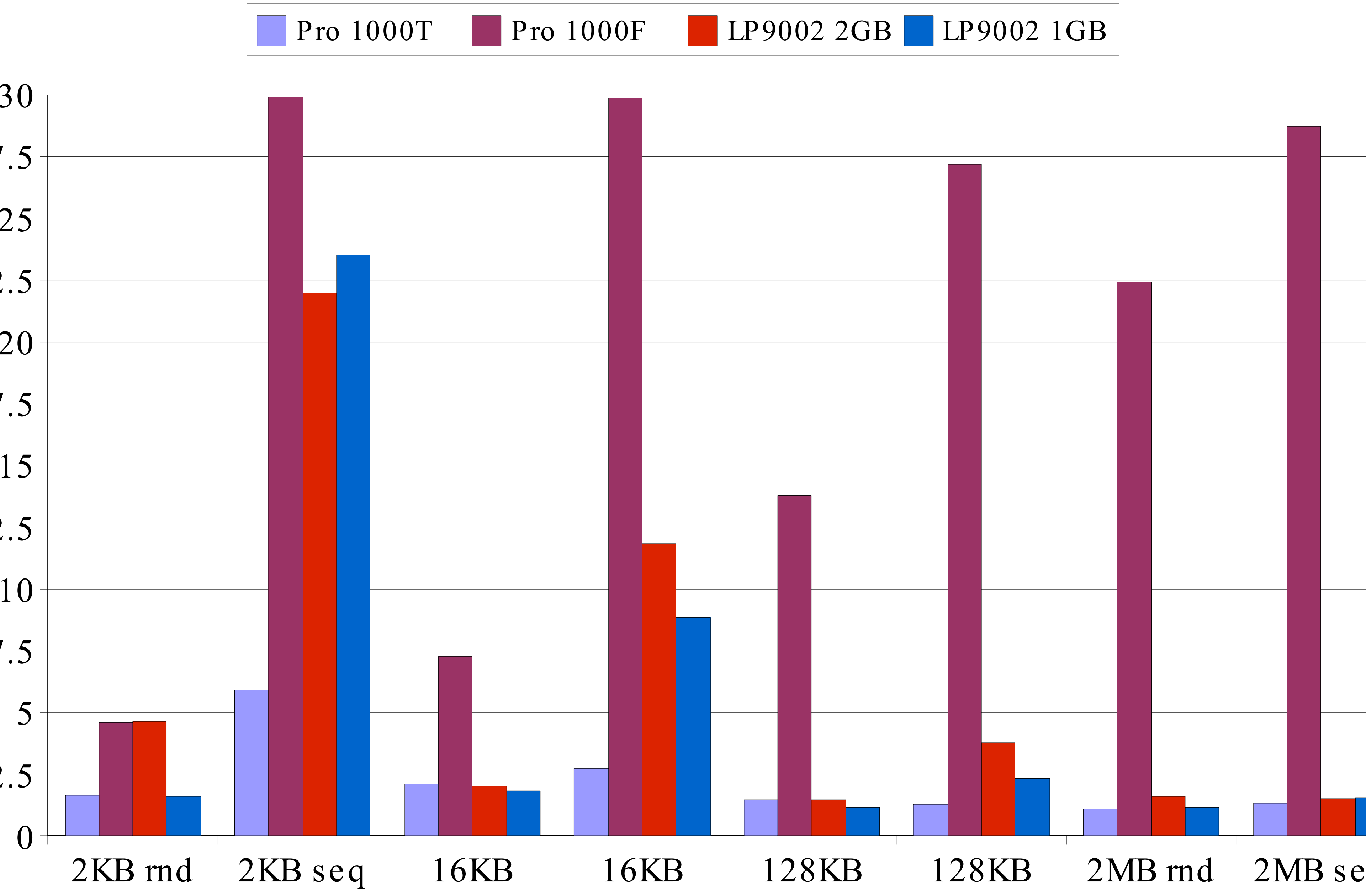
Commercial Deployment - Experimental Setup



Commercial – Throughput



Commercial – CPU Utilization



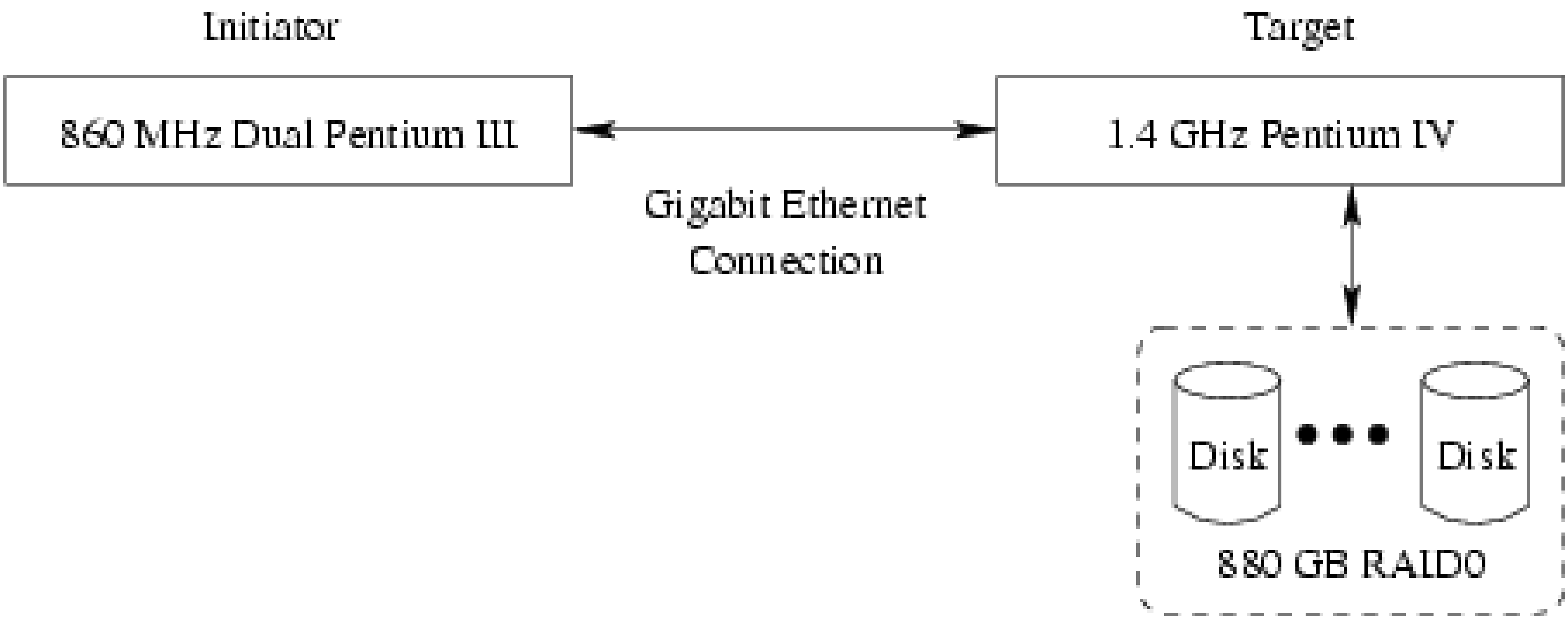
Commercial Deployment – Summary

- Software iSCSI initiator over gigabit network is comprable to 1Gbps fibre channel for large block sizes
- iSCSI hardware HBA adaptor performed poorly in all cases
  - Processor on card not able to keep up with network traffic
  - Server CPU utilization was minimized

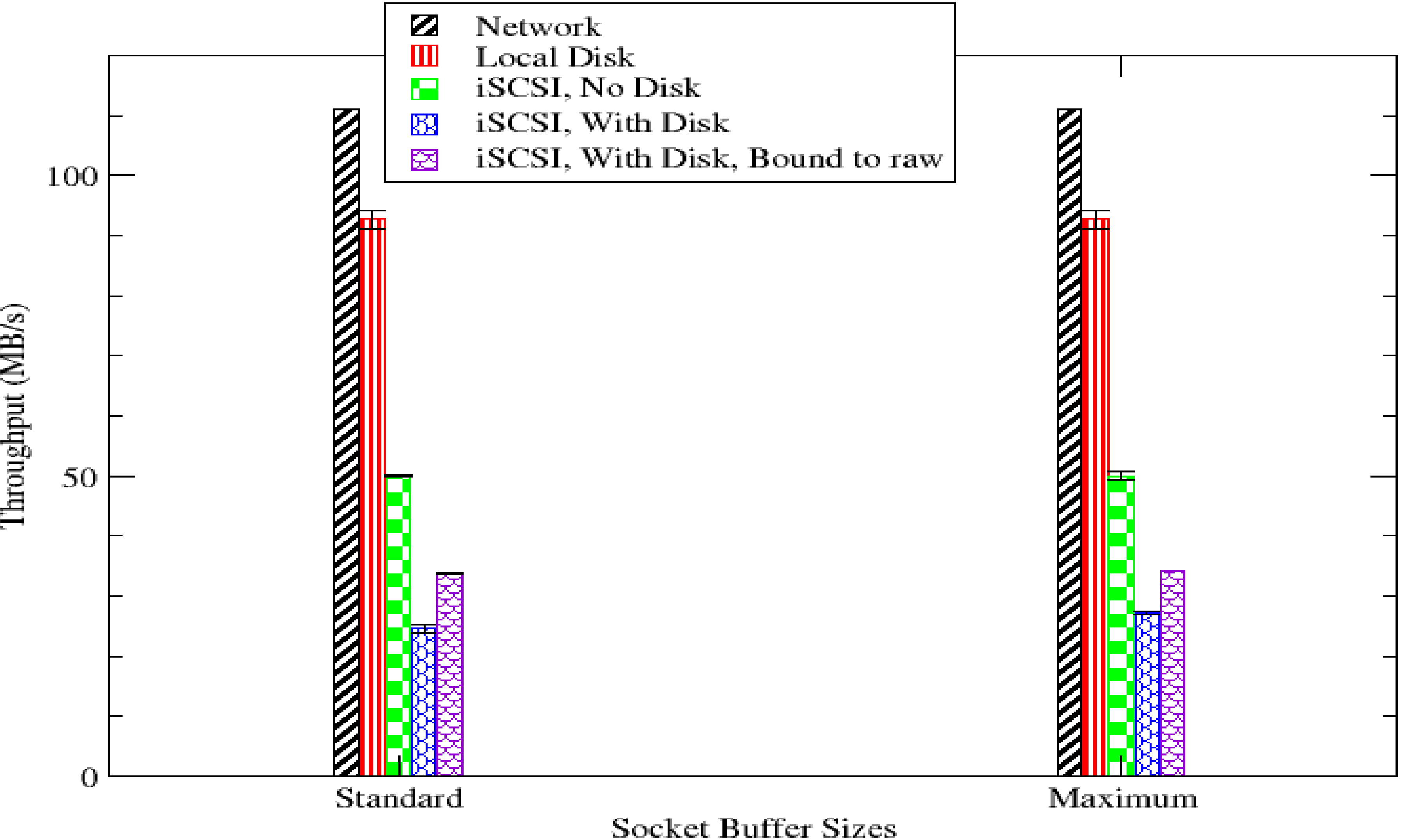
Software SAN – Experimental Design

- Examine performance at the block level using LMdd
  - Determine whether block size has an impact on performance
  - 1GB sizes used in order to reduce system buffering effects
  - Study protocol performance with no disk overhead
- Examine filesystem performance using Bonnie
  - Ext3 filesystem
  - Increasing file sizes from 100MB to 1.6GB
  - Determine buffer cache effects
- Investigate network settings effects on performance
  - TCP window sizes
    - Determined by the Bandwidth Delay Product

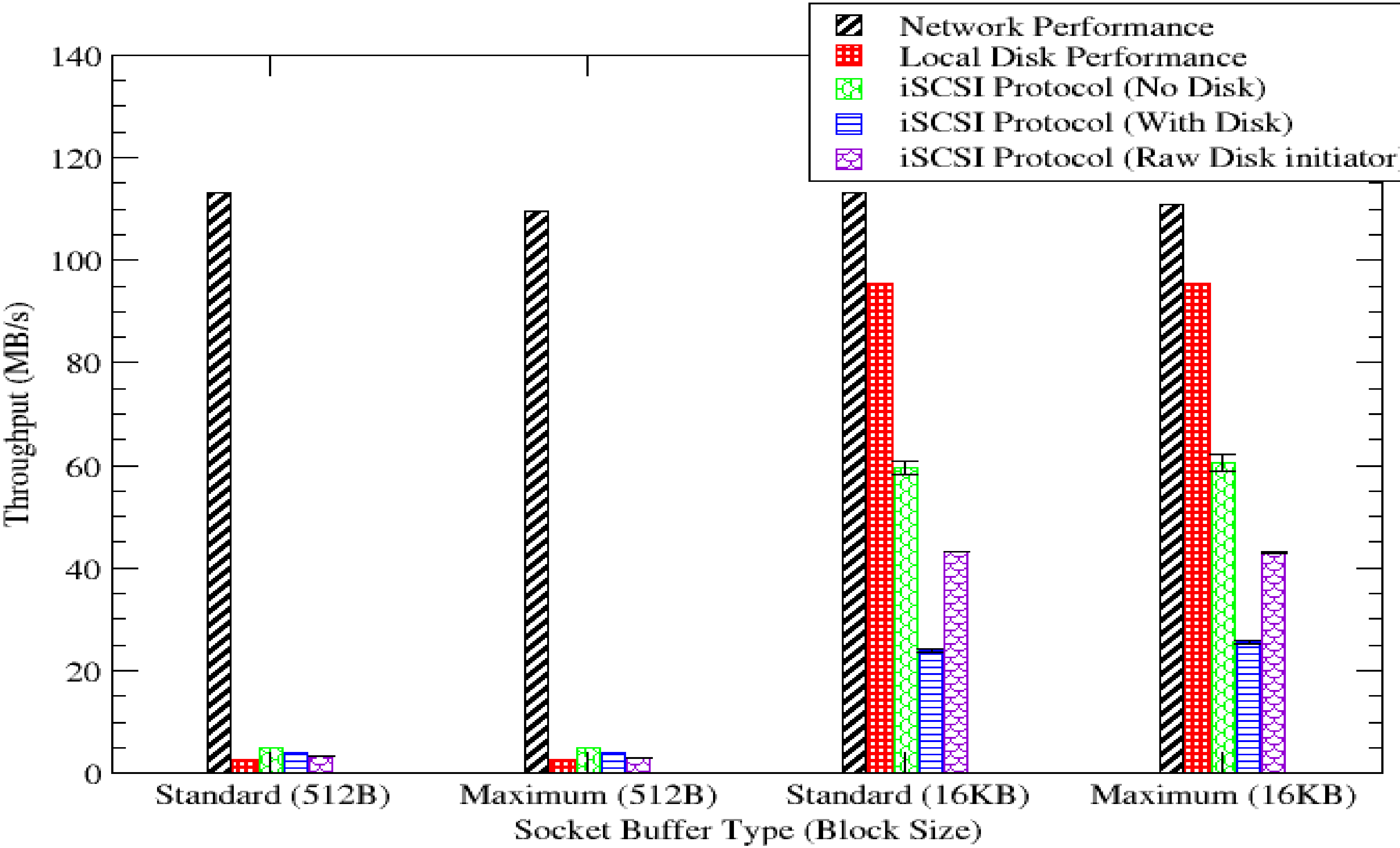
Software SAN – Experimental Setup



SAN – Block Reads, Block Size and Socket Buffer Size Don't Matter

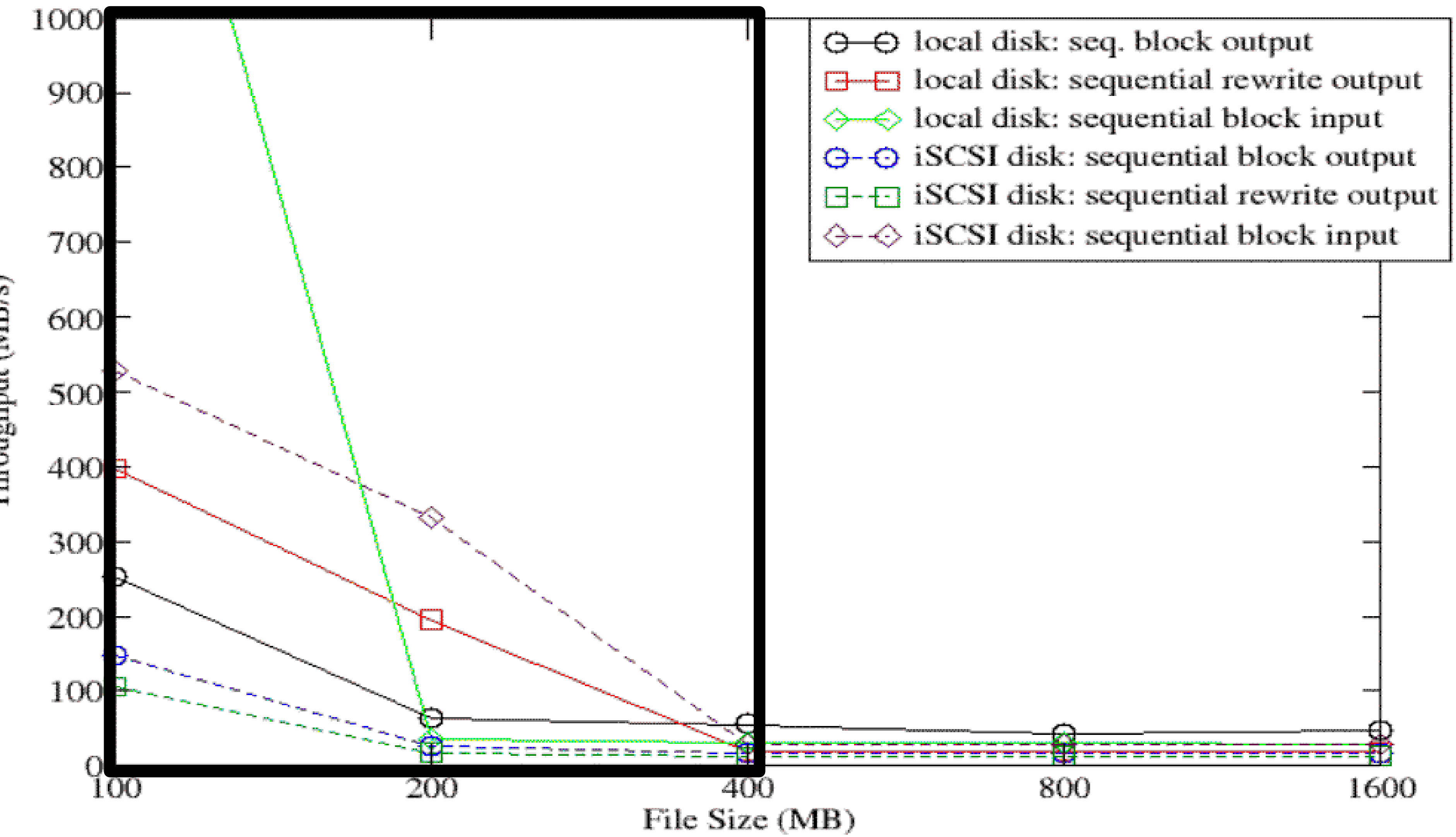


Software SAN – Block Size Matters, Socket Buffer Doesn't

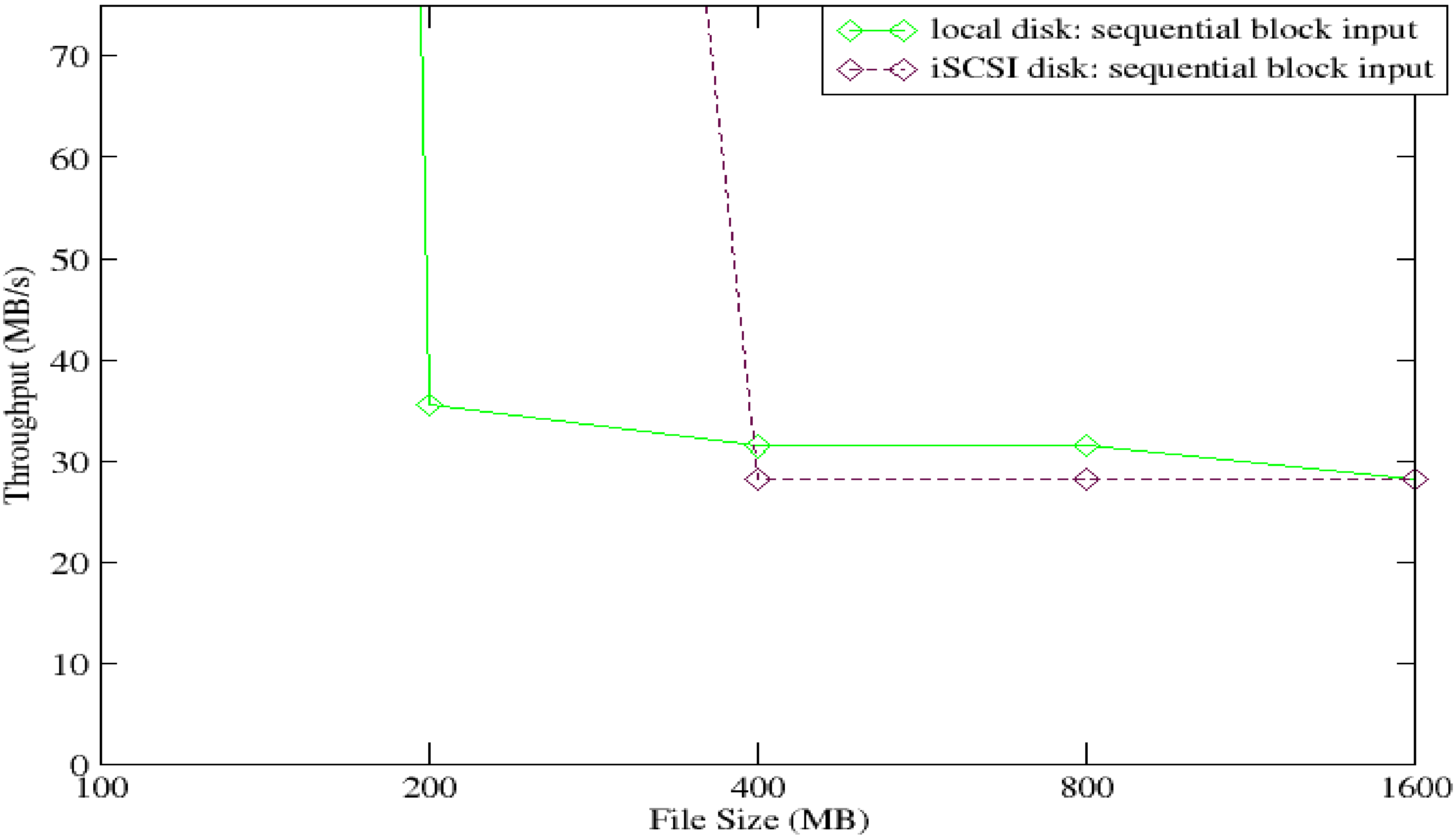


Software SAN – Filesystem Analysis

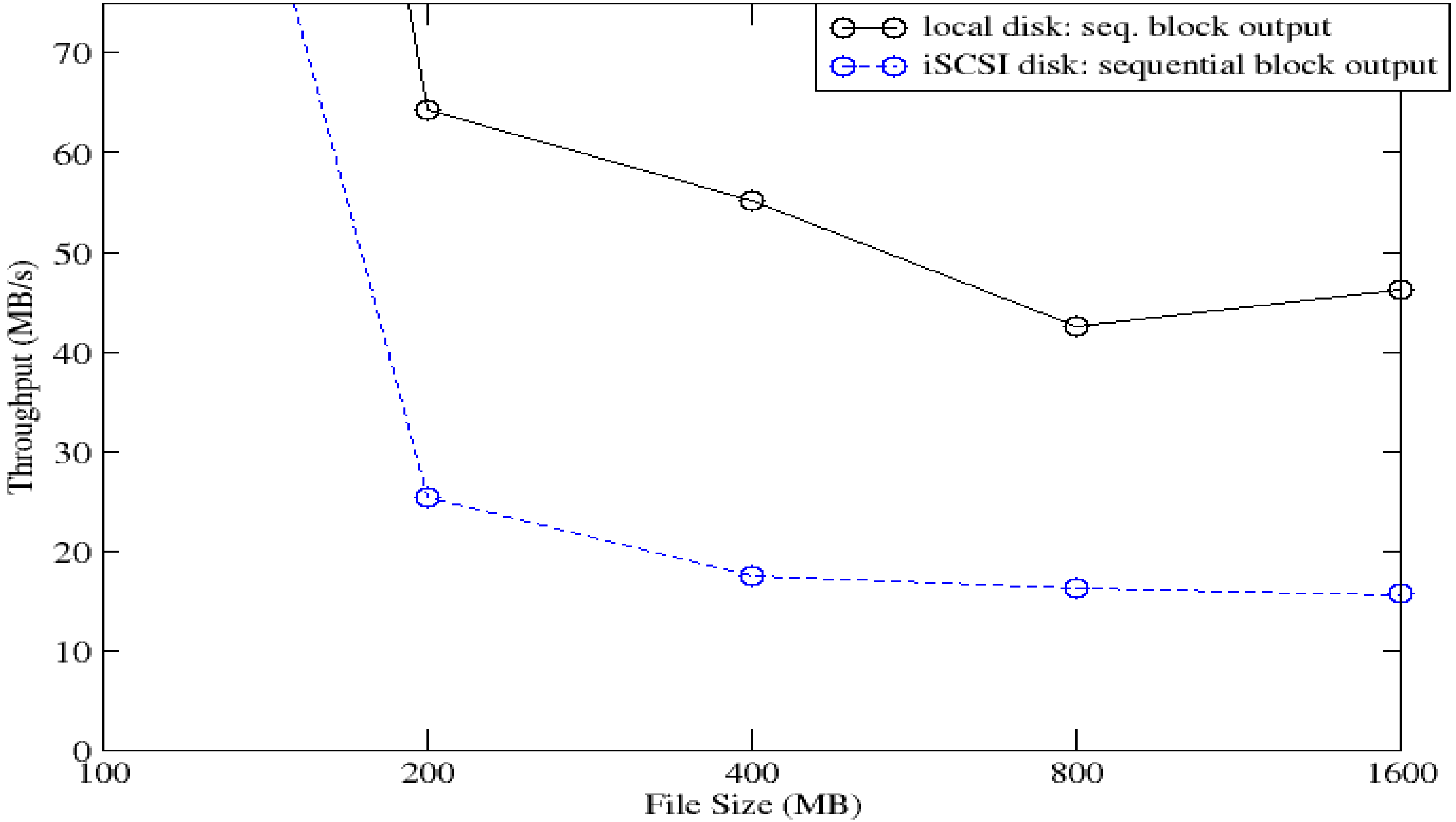
Buffer cache improves performance for small file i/o transactions



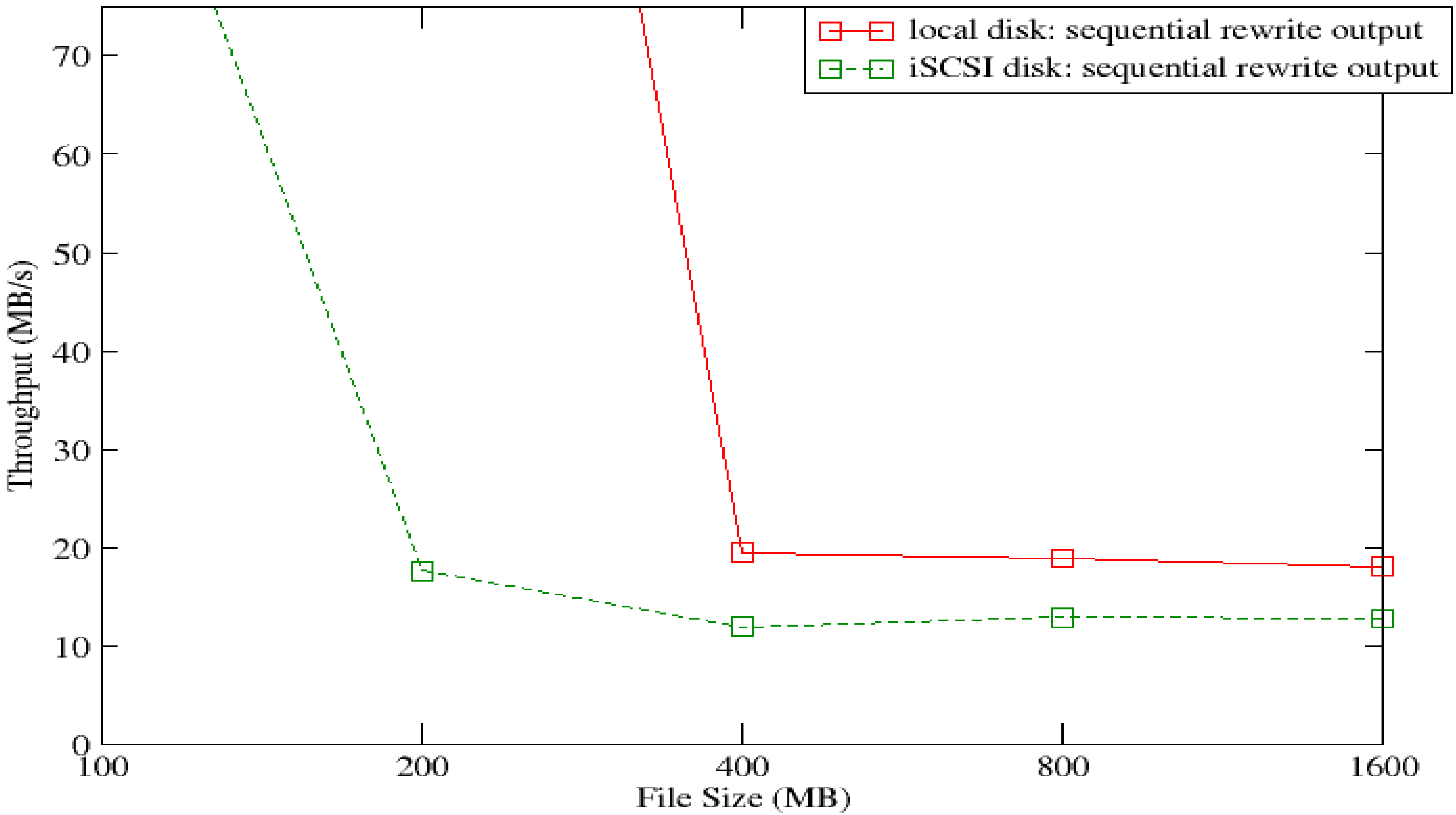
Software SAN – Filesystem Reads



Software SAN – Filesystem Writes



Software SAN – Filesystem Rewrites



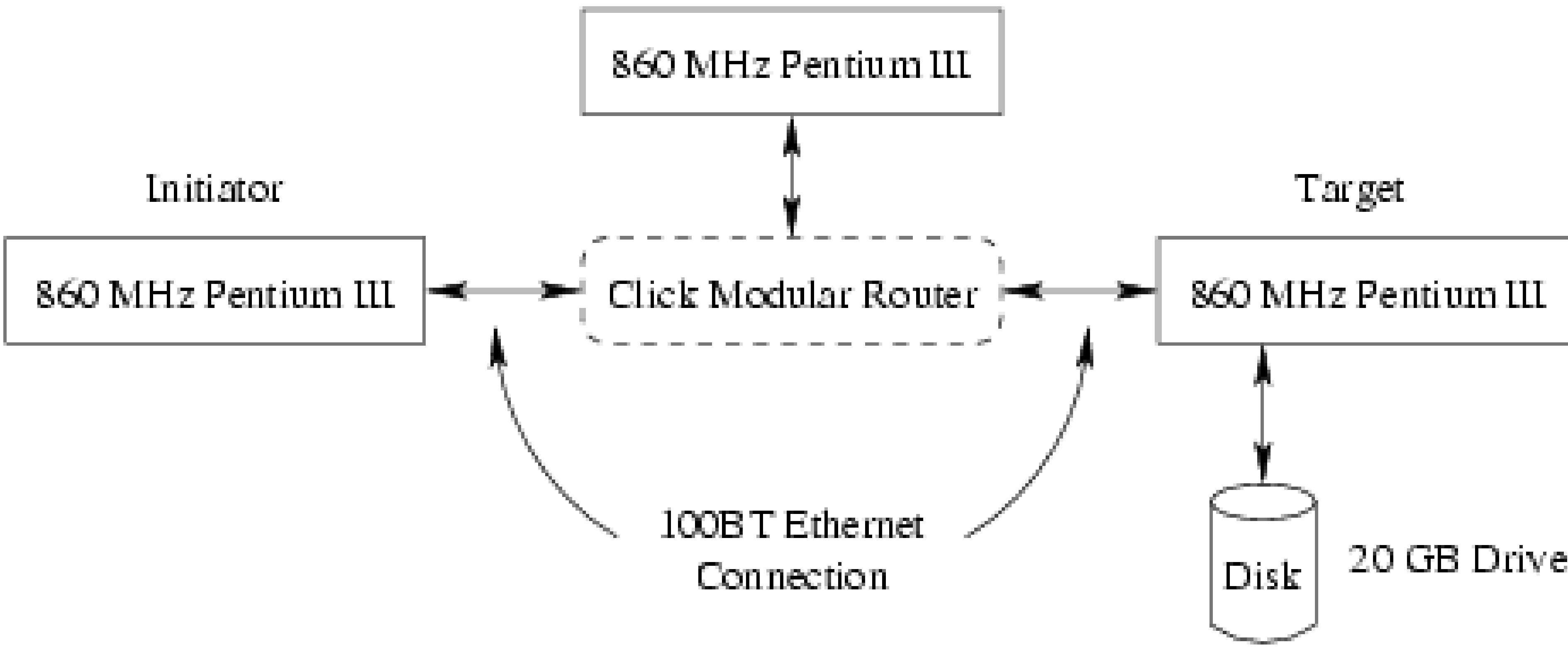
Software SAN – Summary

- Network layer parameters such as TCP window buffer sizes affect performance minimally
- System buffer cache plays a large role in performance
  - Hinders iSCSI at the block level
  - Greatly improve performance for small file transactions

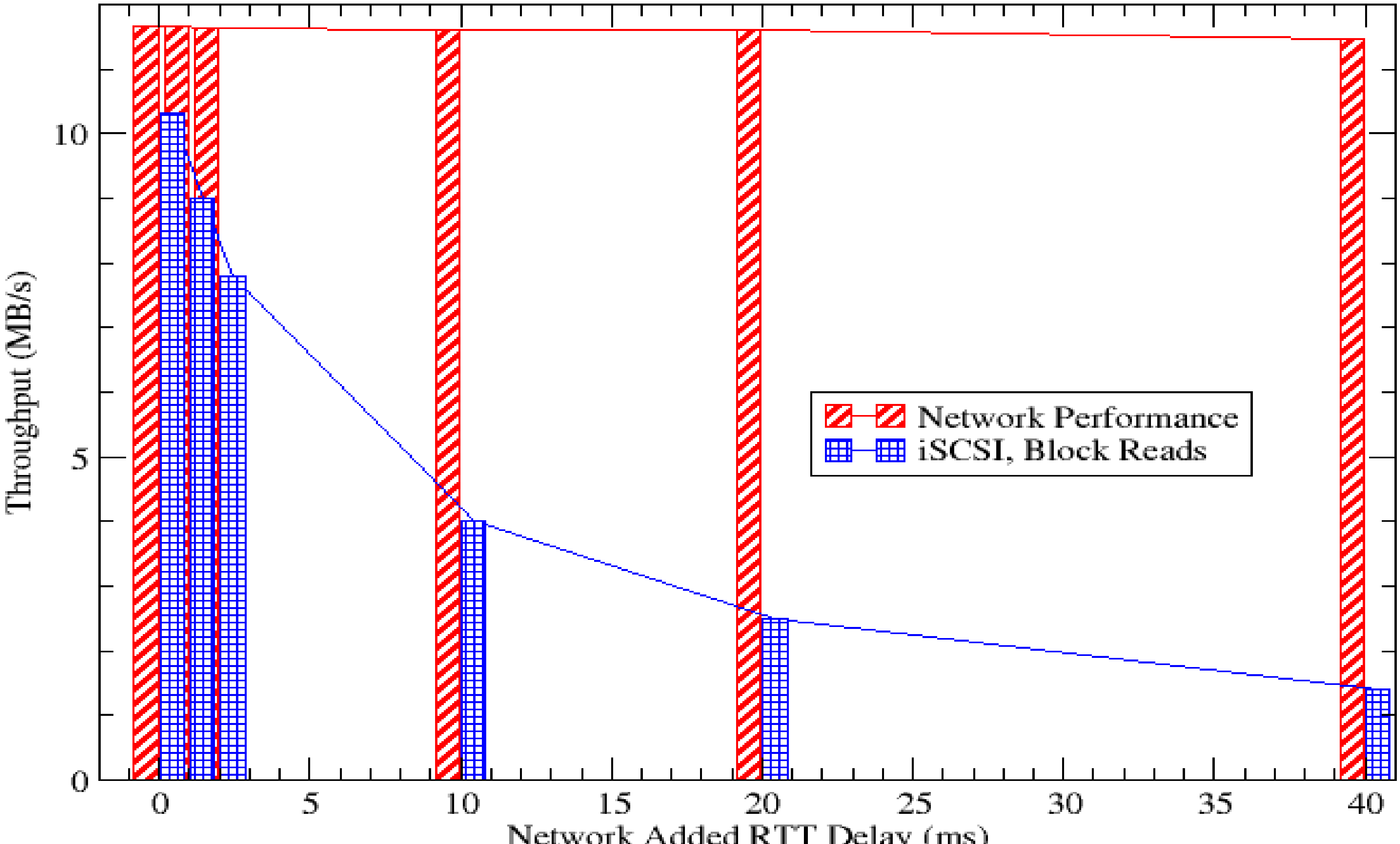
Software WAN – Experimental Design

- Examine block level throughput using Lmdd
- Investigate filesystem performance using Bonnie
- Examine influence of network parameters upon iSCSI
  - Network delay
  - Network pathologies
    - Packet loss, packet corruption, and reordering

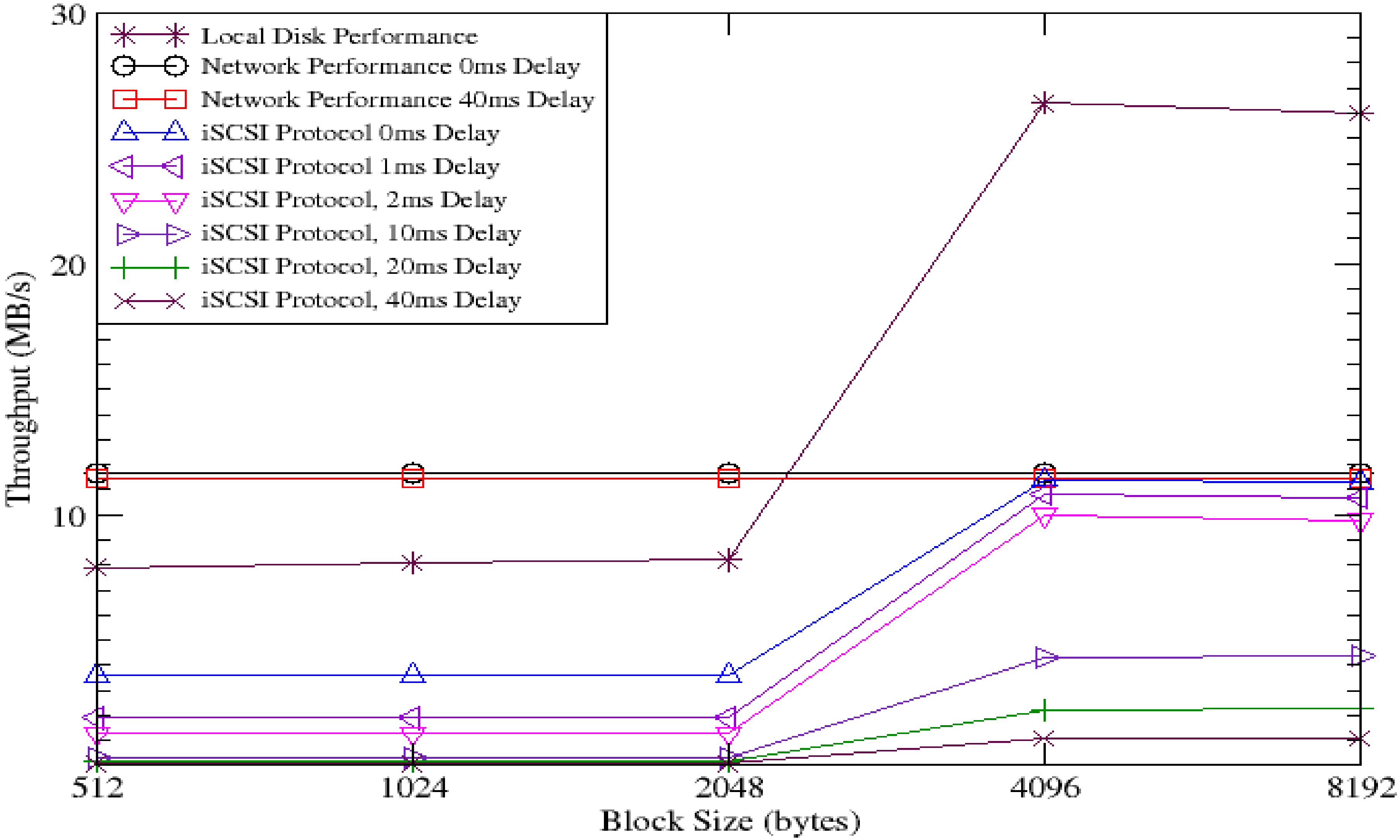
Software WAN – Experimental Setup



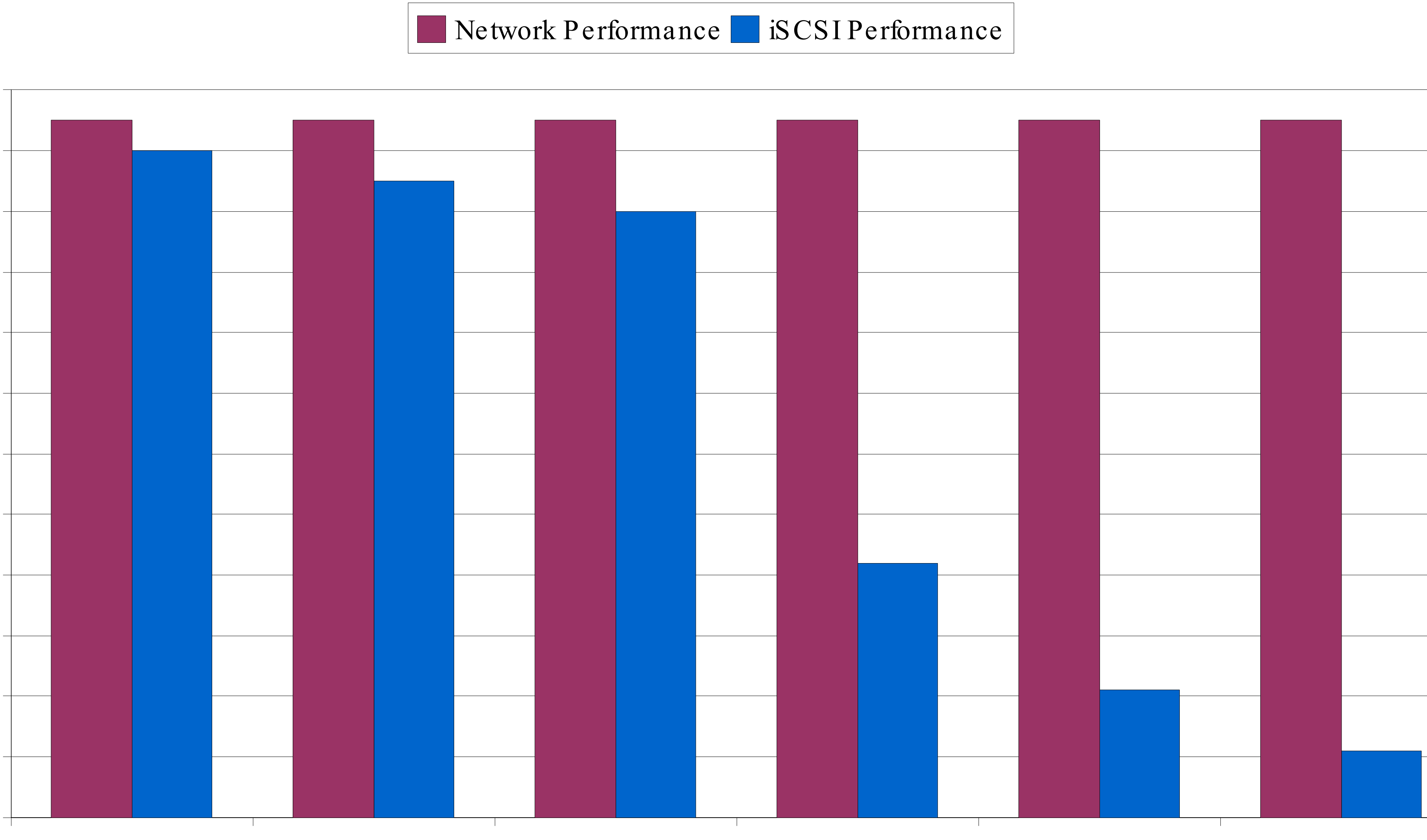
Software WAN – 8K Block Reads, Increasing Delay



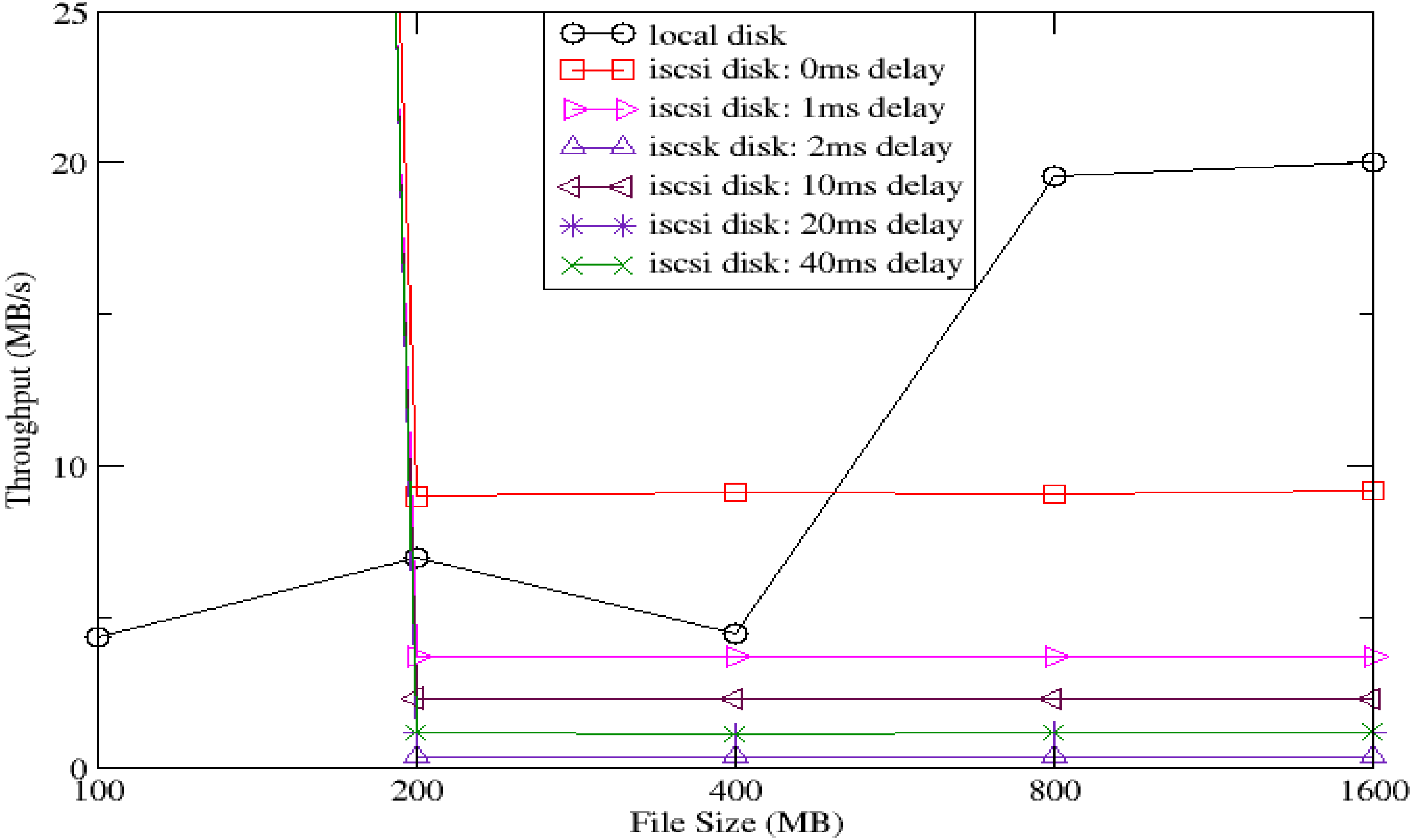
Software WAN – Block Writes, Increasing Delay



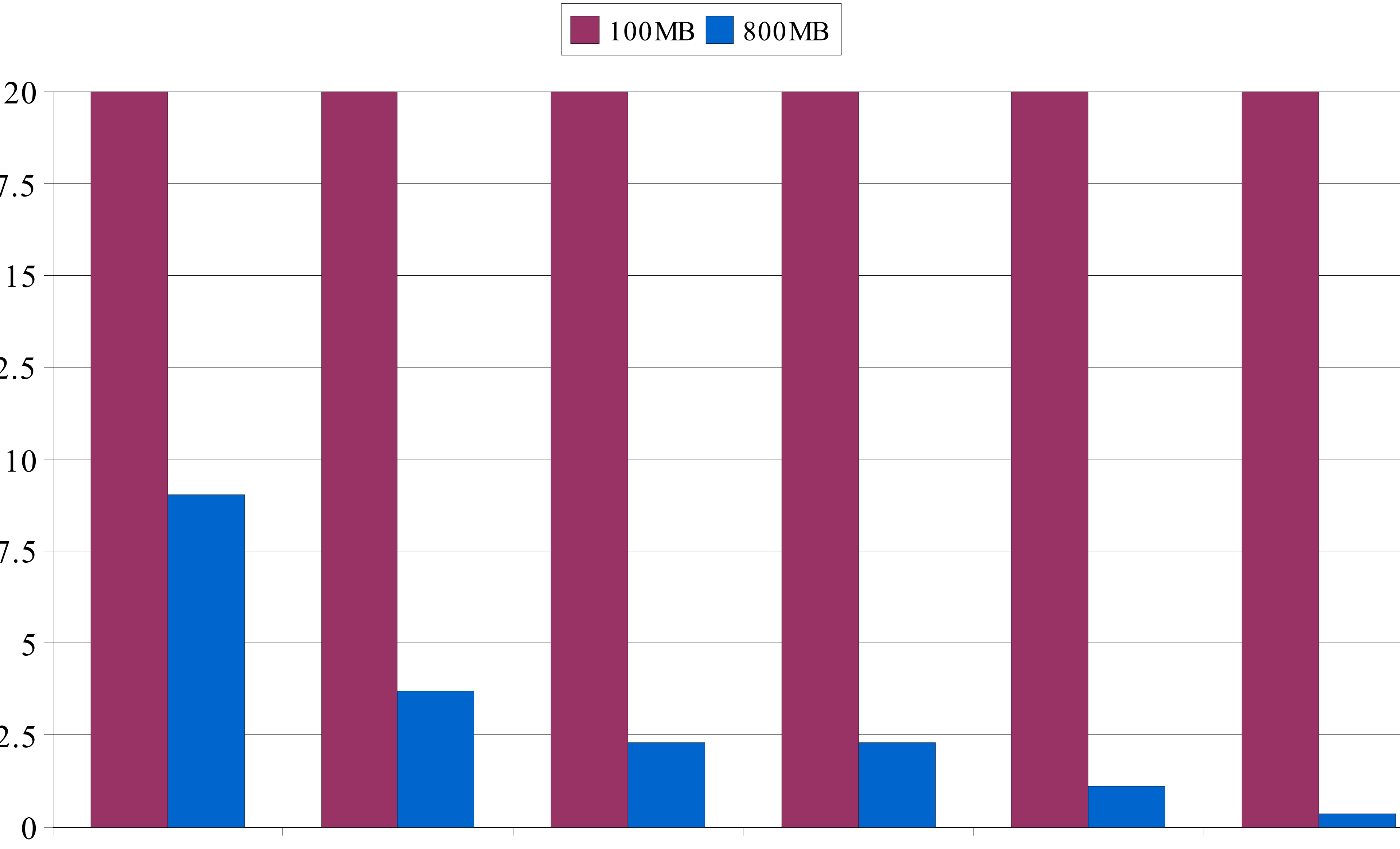
Software WAN – Block Writes, Increasing Delay



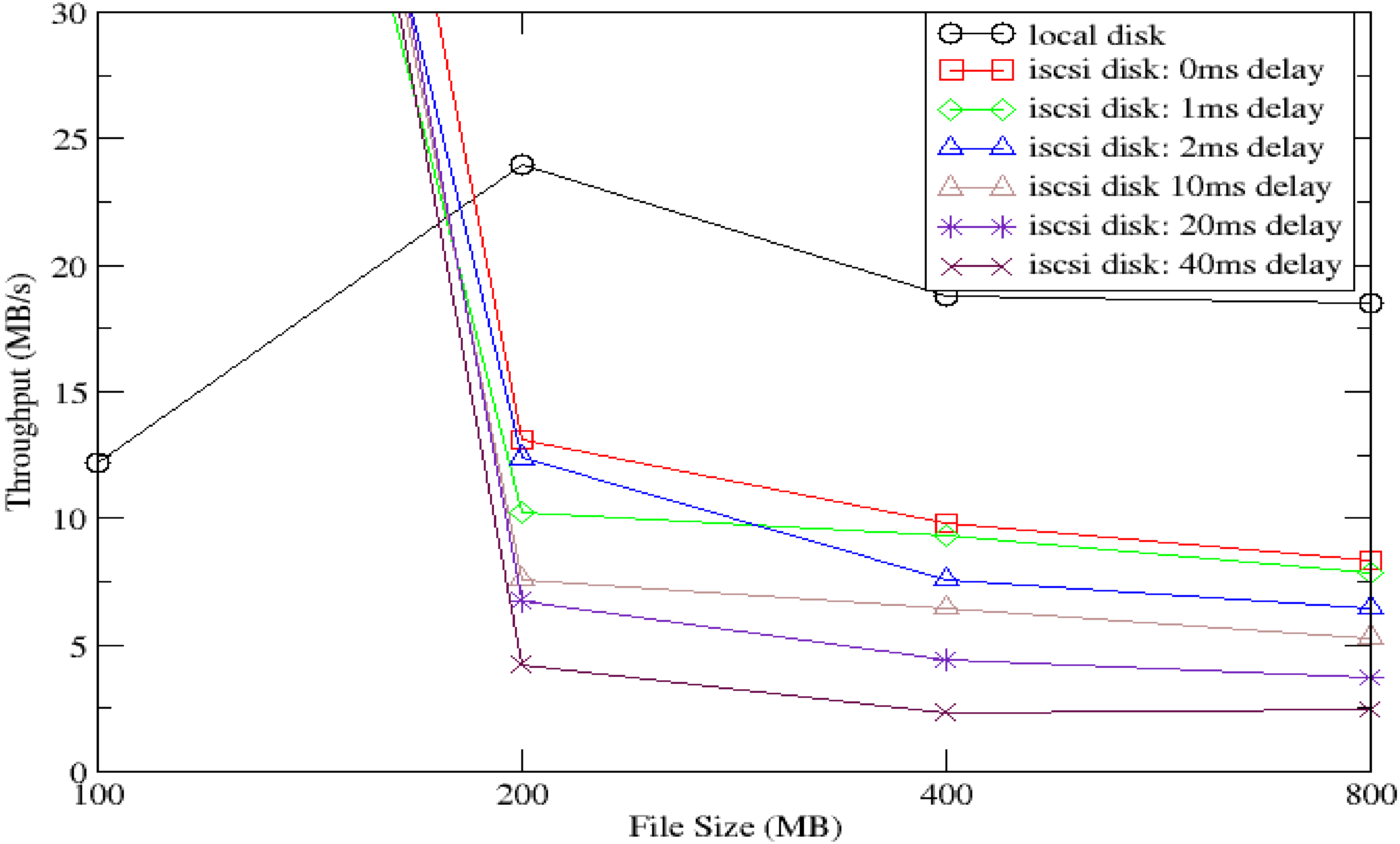
Software WAN – Filesystem Reads, Increasing Delay



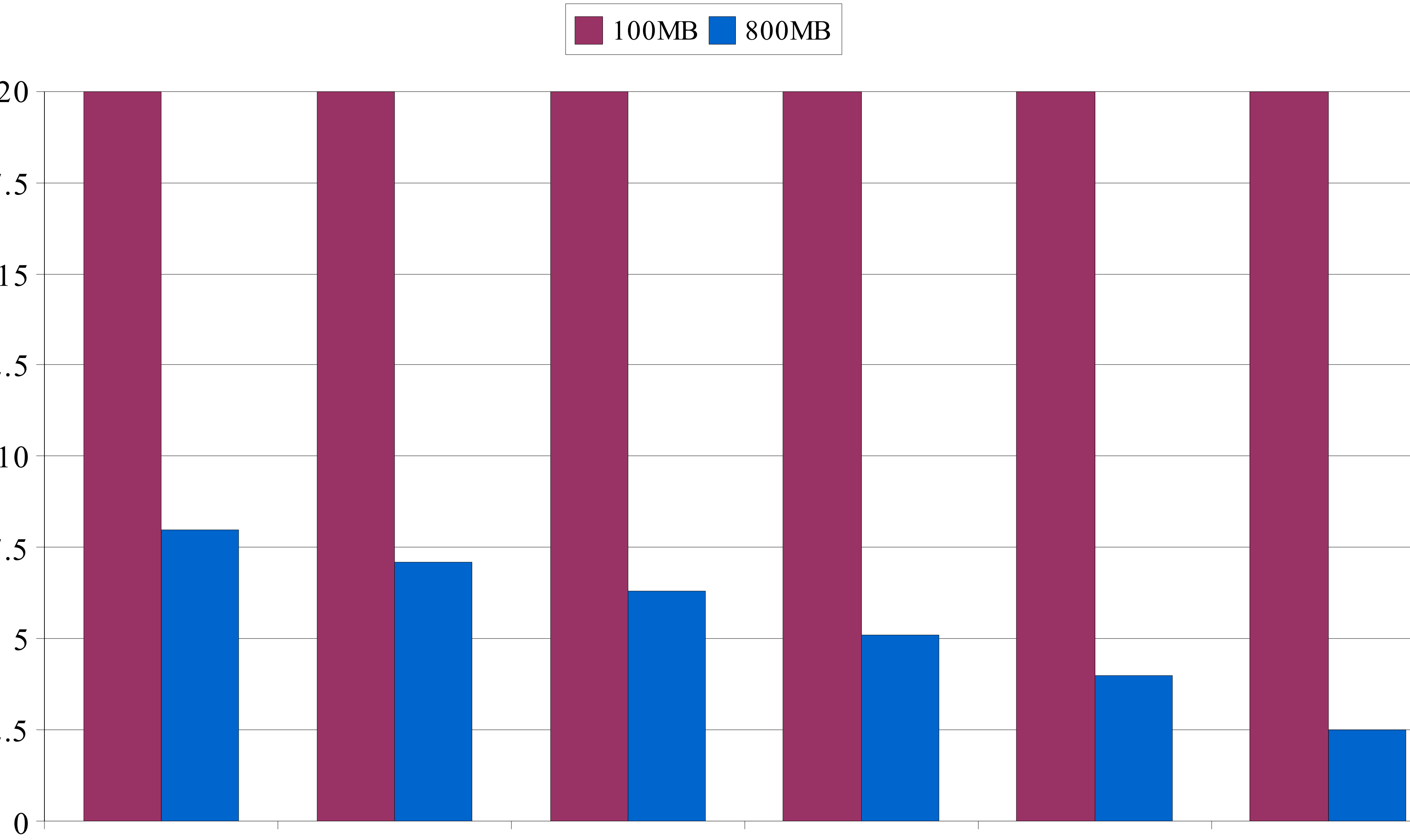
Software WAN – Filesystem Reads, Increasing Delay



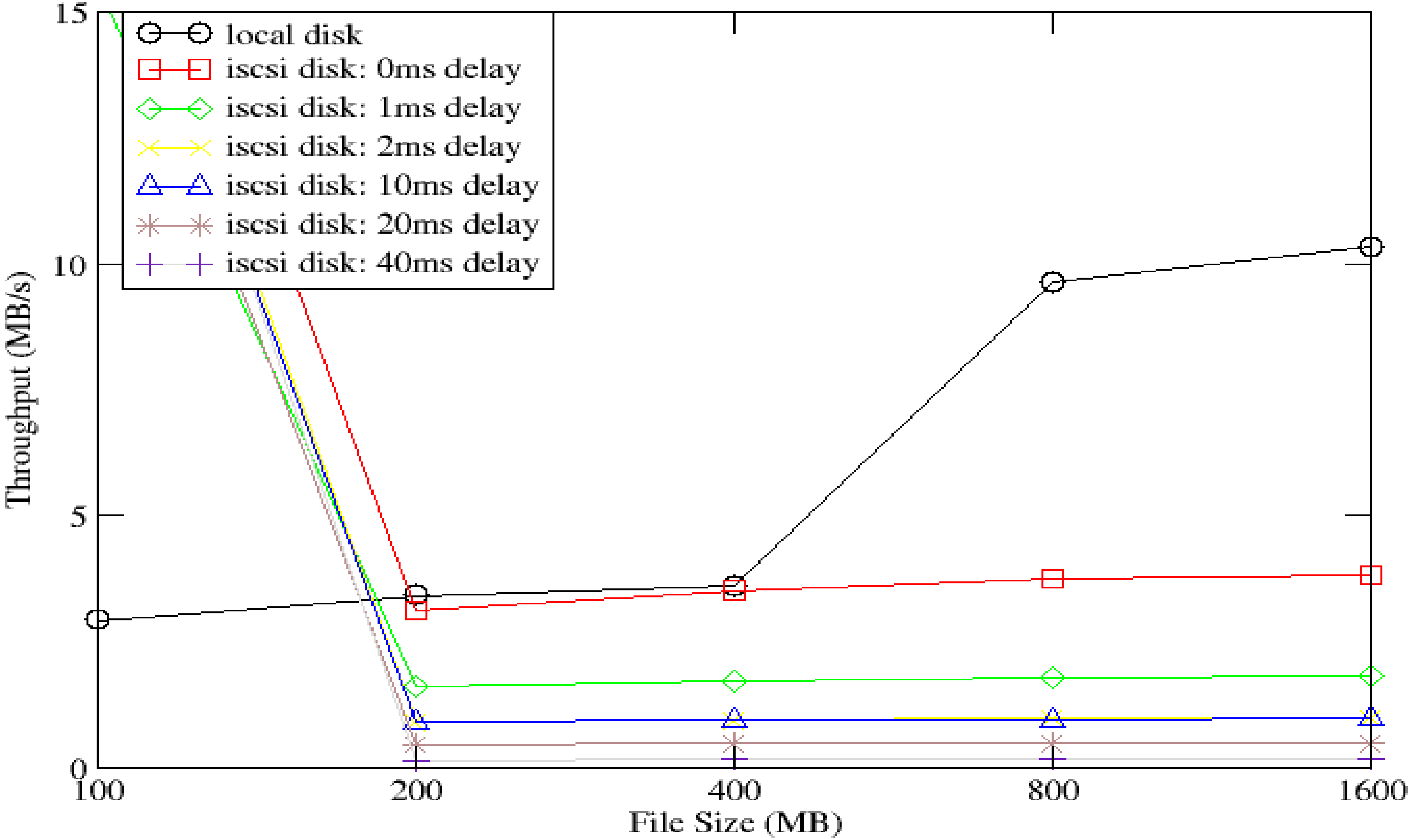
Software WAN – Filesystem Writes, Increasing Delay



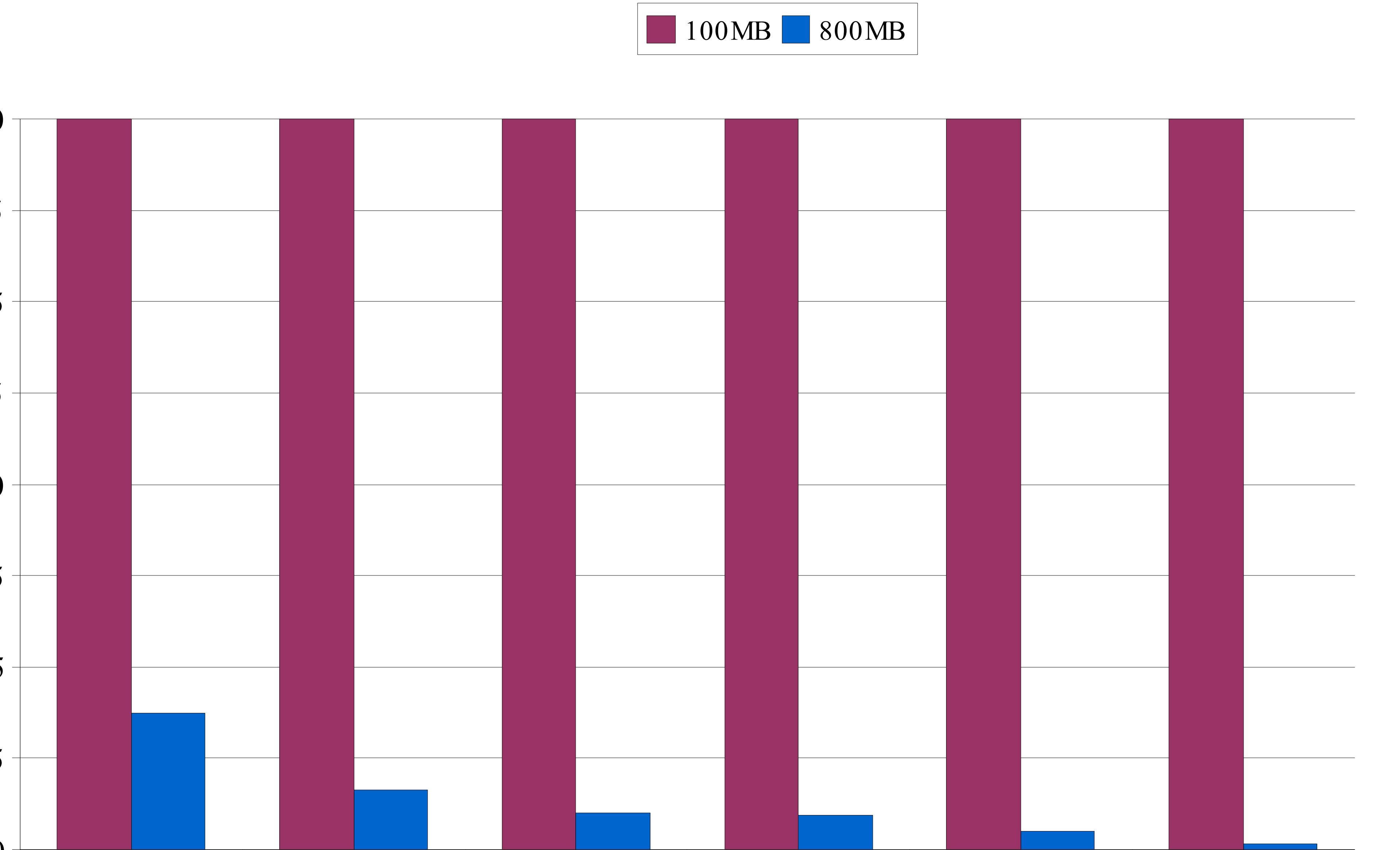
Software WAN – Filesystem Writes, Increasing Delay



Software WAN – Filesystem Rewrite, Increasing Delay



Software WAN – Filesystem Rewrite, Increasing Delay



Software WAN – Network Performance With Pathologies

- Pathologies introduced
  - Packet loss : 2.7%
  - Packet corruption : 0.02%
  - Packet reordering : 2.0%

Socket Buffer Size	Payload Size					
	512	1K	2K	4K	8K	16K
Standard	3.17 +/-	4.38 +/-(	4.28 +/-(	4.53 +/-(	4.22 +/-(	3.89 +/-1.07
Maximum	4.37 +/-(	4.24 +/-(	4.52 +/-(	4.12 +/-(	3.36 +/-	3.68 +/-2.85

Network performance was so poor that it was decided to forego iSCSI testing.

Software WAN – Summary

- Performance degrades rapidly as delay is increased
  - Due to the synchronous nature of the iSCSI implementation
    - Tag command queuing
- System caching improves performance filesystem access for small files
- Packet loss and other pathologies greatly hinder performance overall

Conclusions

- iSCSI is comprable to fibre channel for large block sizes
- Deeper tag command queuing will most likely increase the performance for networks with large delay
- System caching greatly improves small I/O transaction throughput while hindering large I/O transaction throughput

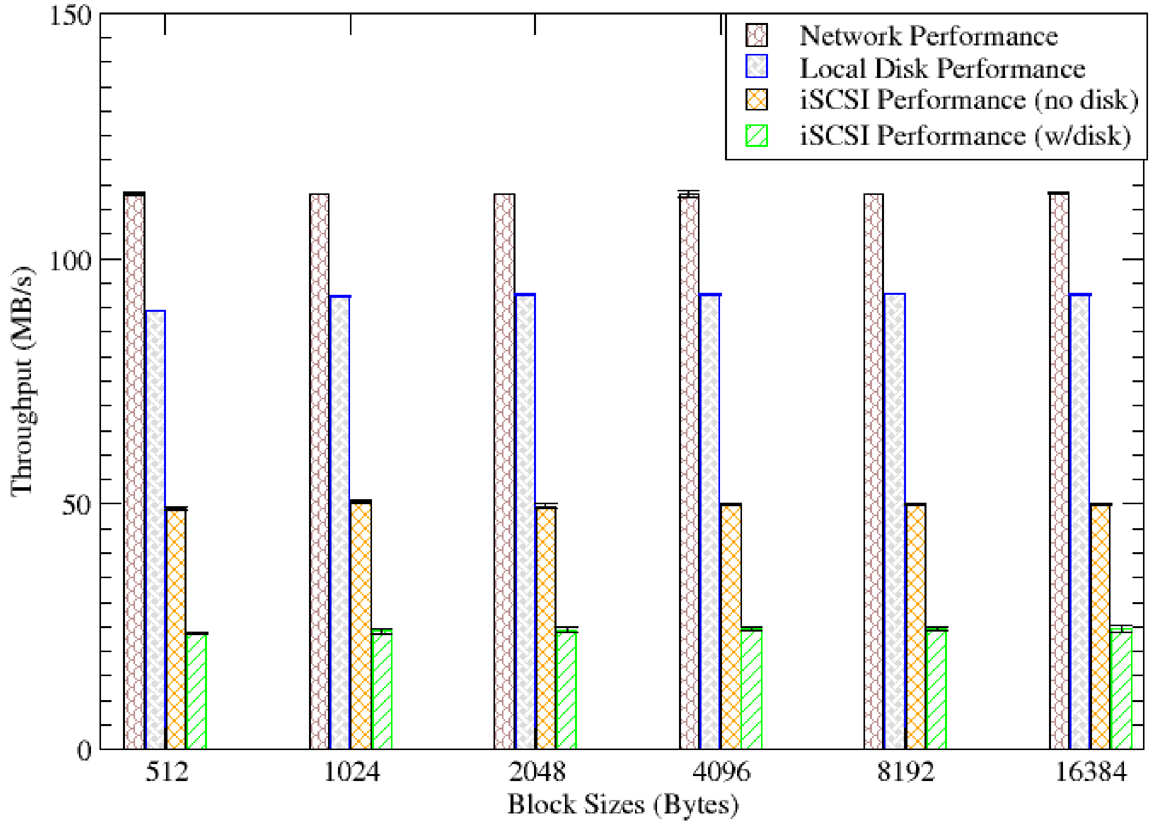
Commercial Deployment – Hardware

- Server – IBM x-series 360
  - Dual 1.5GHz Xeon processors with hyperthreading
  - 2GB RAM
  - 64bit/133MHz PCI bus
- Intel Pro1000F gigabit ethernet card
- Intel Pro1000T iSCSI HBA adaptor
- Emulex LP9002 fibre channel HBA
- Two CISCO SN-5428 storage routers
- Sphereon 4500 fibre channel switch
- Dell 5224 gigabit ethernet switch

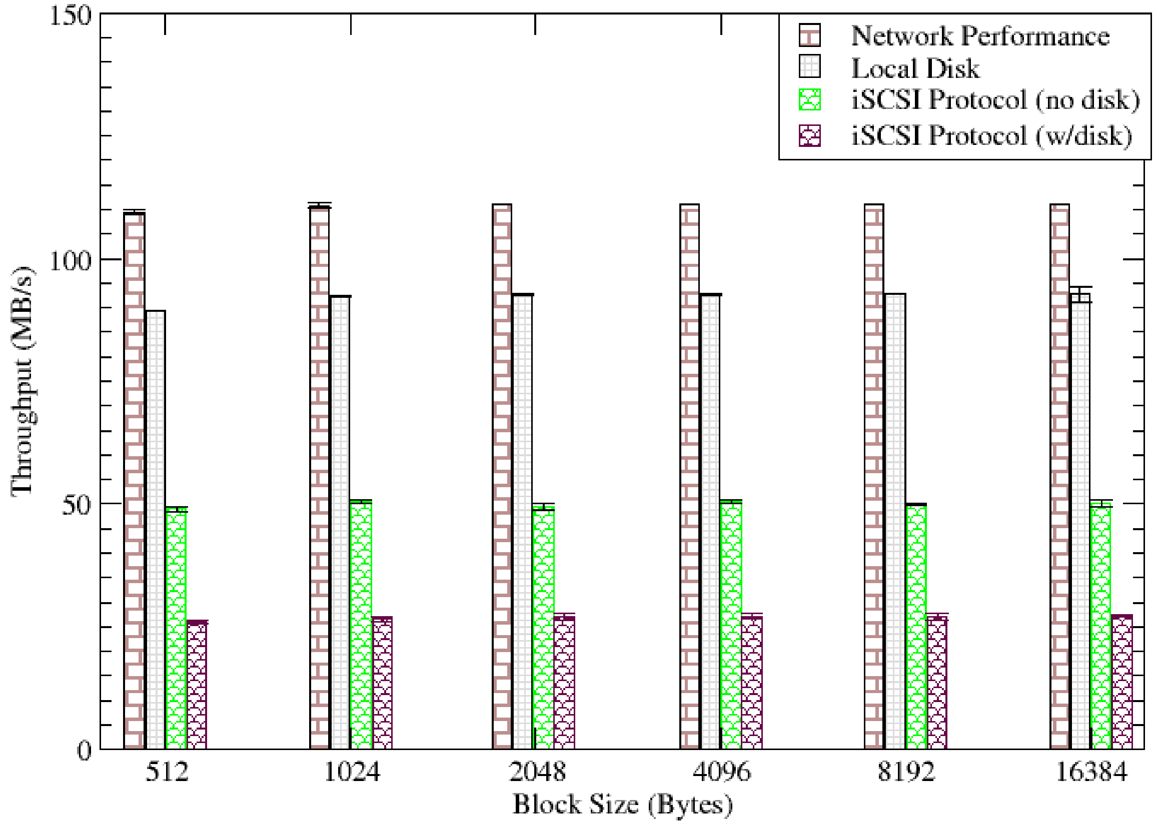
Software SAN – Hardware

- Target
  - Intel 1.4 Ghz Pentium4
  - 256MB RAM
  - 880GB striped array
  - D-Link DL2K gigabit ethernet NIC
- Initiator
  - Intel 860MHz PentiumIII
  - 256MB RAM
  - D-Link DL2K gigabit ethernet NIC

SAN – Raw Input, Standard Socket Buffer Sizes



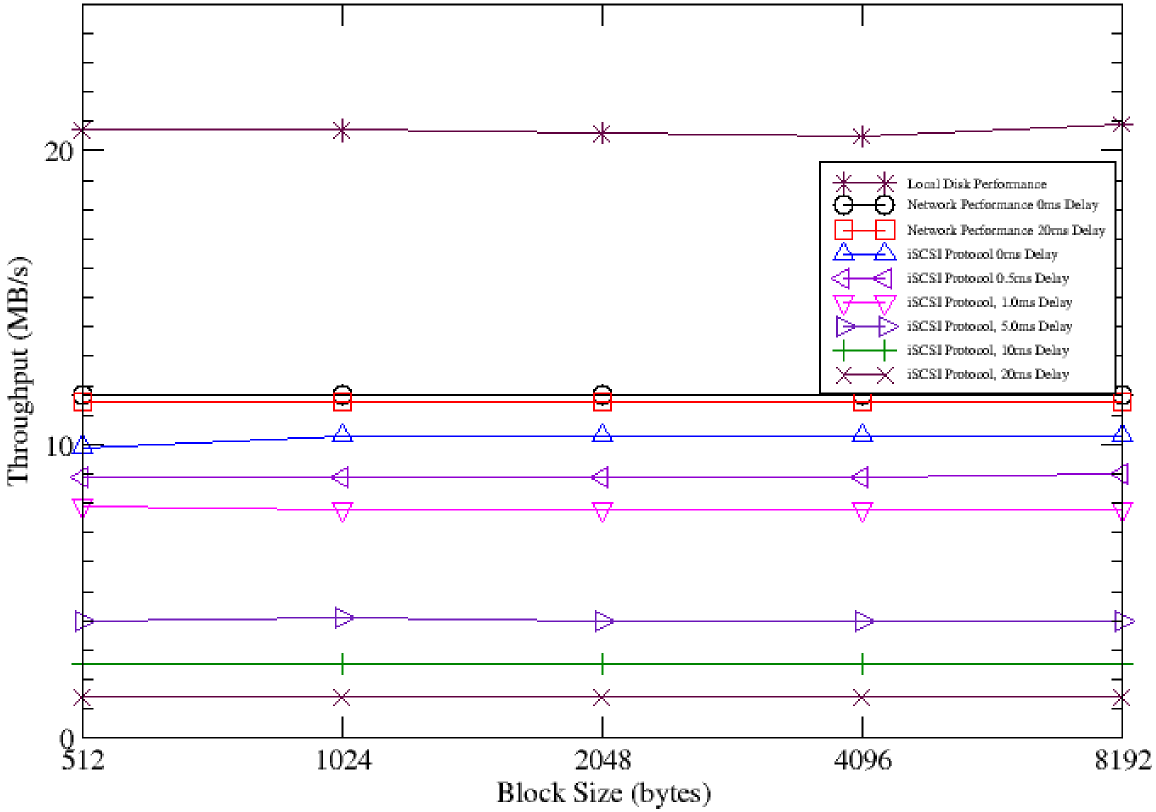
SAN – Raw Input, Maximum Socket Buffer Sizes



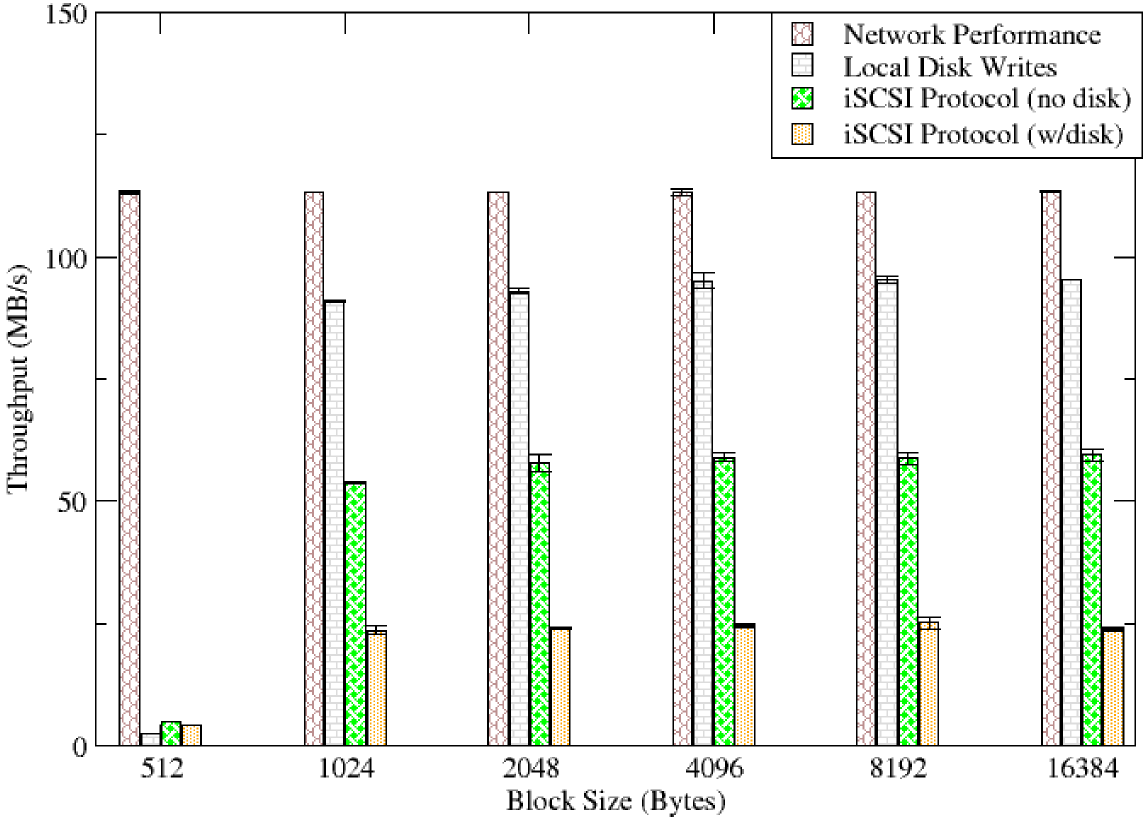
Software WAN – Hardware

- Target, Initiator
  - Intel 860MHz PentiumIII
  - 256MB RAM
  - 20GB hard drive
  - Intel Ethernet Express Pro100 100Mbps NIC
- Router
  - Intel 860MHz PentiumIII
  - 256MB RAM
  - 20GB hard drive
  - 4 port Tulip 100Mbps NIC

Software WAN – Block Reads, Increasing Delay



Software SAN – Block Writes, Standard Socket Buffer Sizes



Software SAN – Block Writes, Maximum Socket Buffer Sizes