Characterization of Storage Workload Traces from Production Windows Servers

Swaroop Kavalanekar, Bruce Worthington,
Qi Zhang, Vishal Sharda

Microsoft Corporation

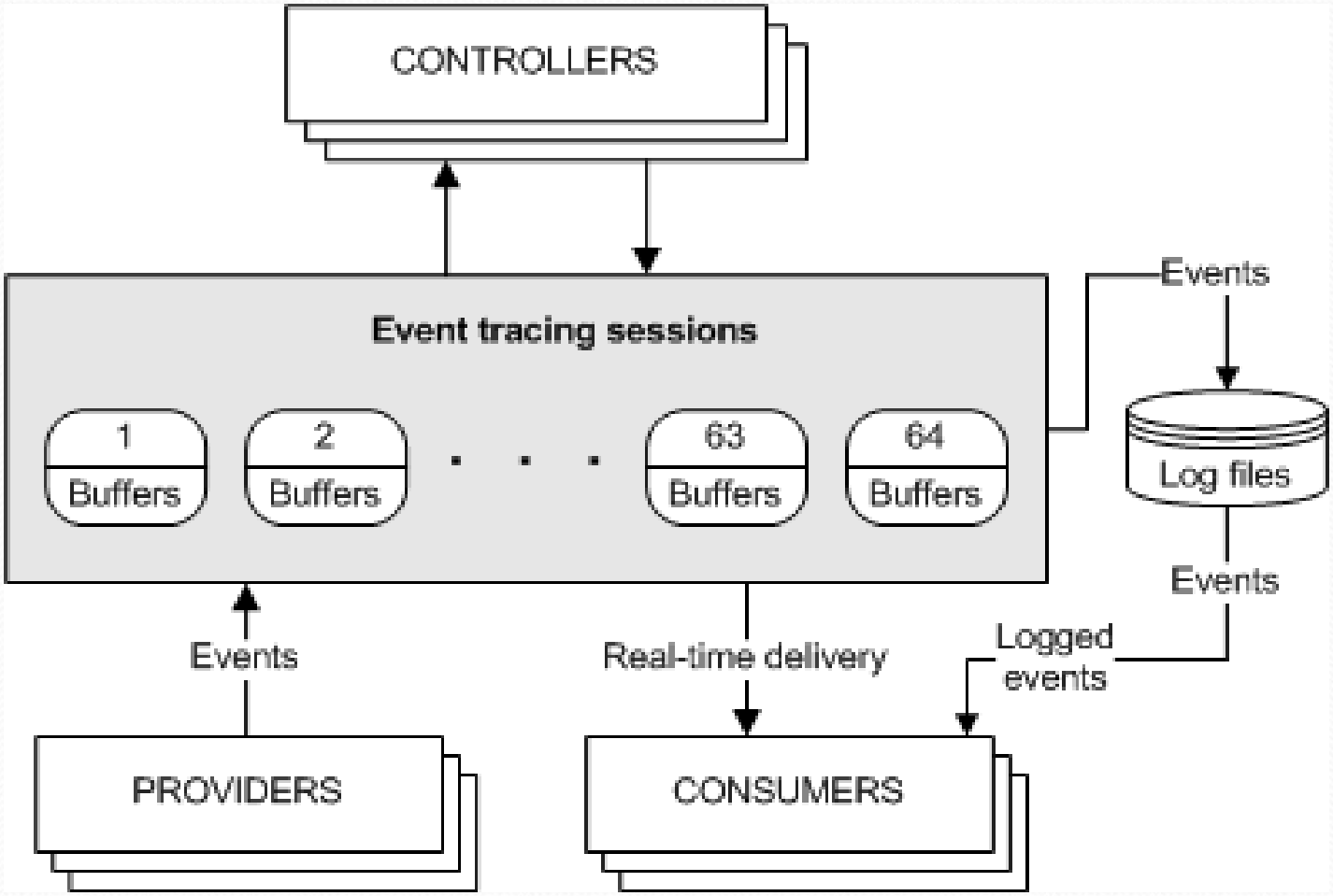
Motivation

- Scarcity of publicly available storage workload traces of production servers
- Tracing storage workloads on live production servers enables higher precision performance analyses, capacity planning and diagnosis
- Over the past 5 years, Microsoft has made it easy to capture increasingly more detailed storage workload traces on Windows systems (along with many other types of traces and profiles)

Outline

- Motivation
- **Trace and analysis tools**
  - Event Tracing for Windows (ETW)
  - Windows Performance Tools kit (WPT)
- Storage workload metrics and characterization
- Newly published traces
- Analysis
- Summary

ETW Architecture



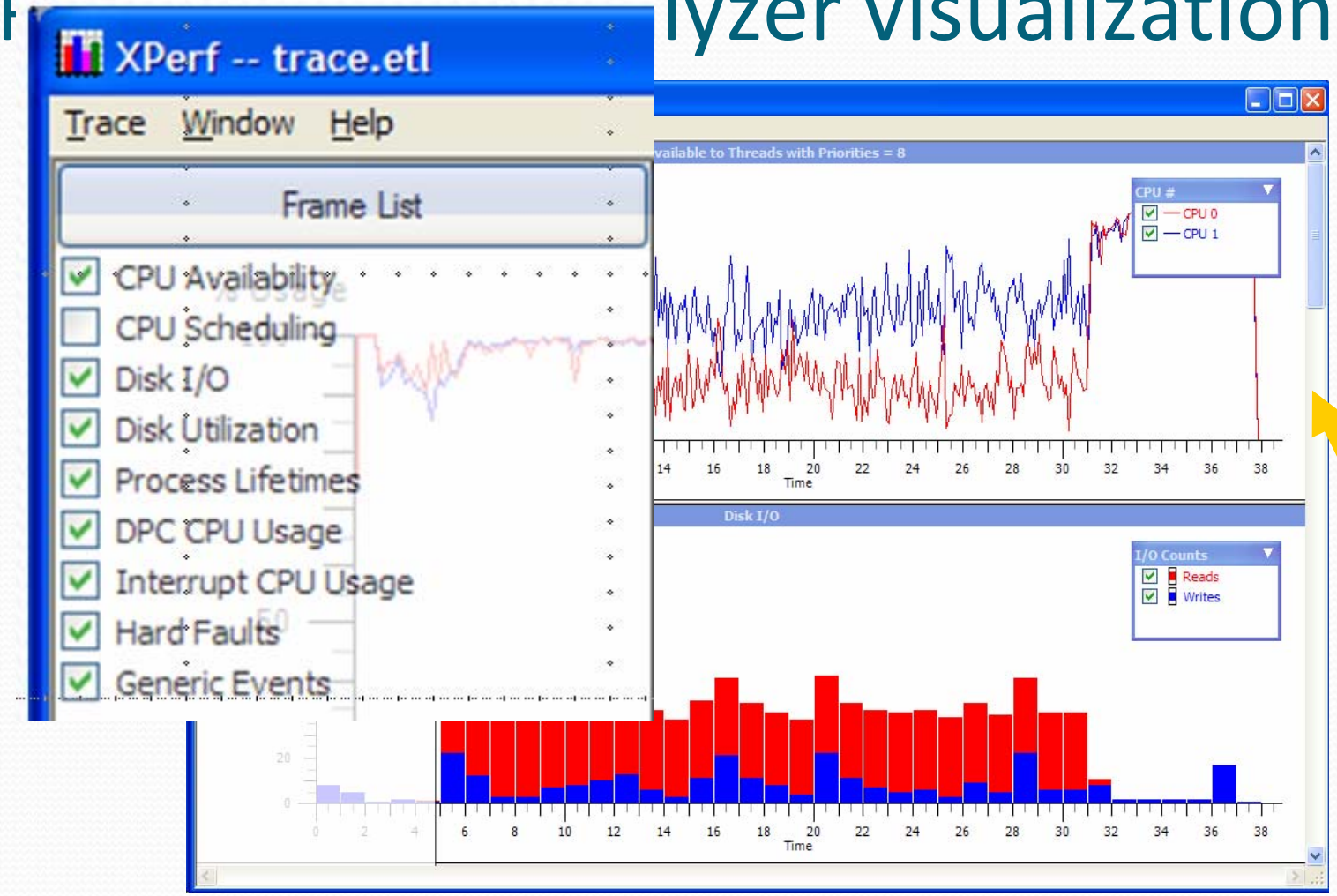
Event Tracing for Windows (ETW)

- Main Windows tracing component since Windows 2000
- Many Windows components, including the kernel, produce numerous events describing their behavior
- Typical events are discrete time-stamped trace points, but sampling and statistical data capture are also possible
- **High performance; low overhead; highly scalable**
  - Efficient buffering and non-blocking logging mechanisms using per-CPU buffers written to disk by a separate thread
- Since 2003, tracing can be enabled/disabled dynamically without requiring system reboots or application restarts

Windows Performance Tools Kit

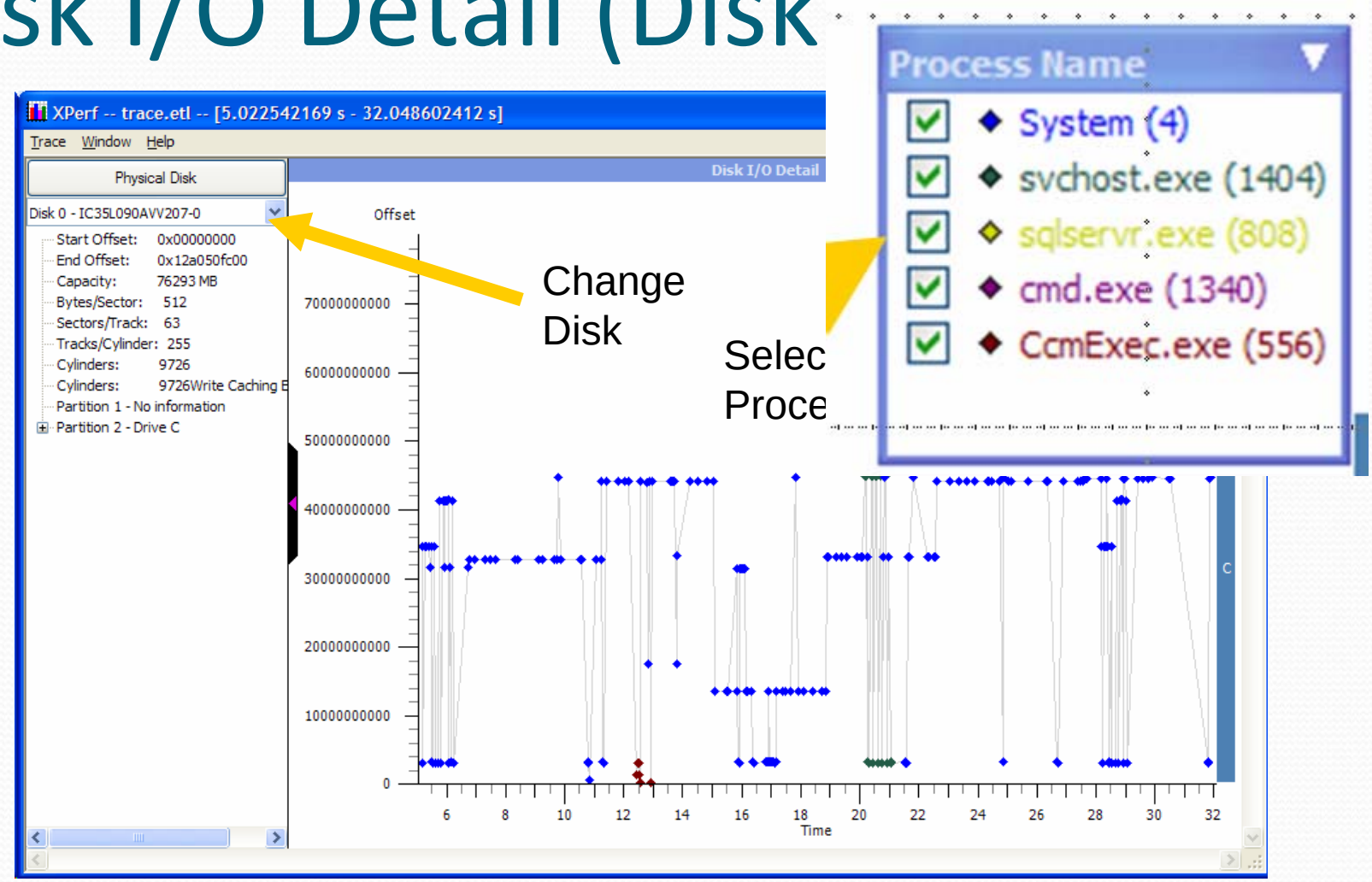
- Extensible performance analysis toolset
- High-level control and decoding of ETW traces
  - Emphasis on kernel events and system-wide resource usage
  - Support for 3rd-party events, primarily in conjunction with kernel events
- Officially supported on Windows Server 2008 and Vista
- Cross-architecture (x86, x64, ia64)
- Capture-anywhere, process-anywhere

Performance Analyzer visualization tool

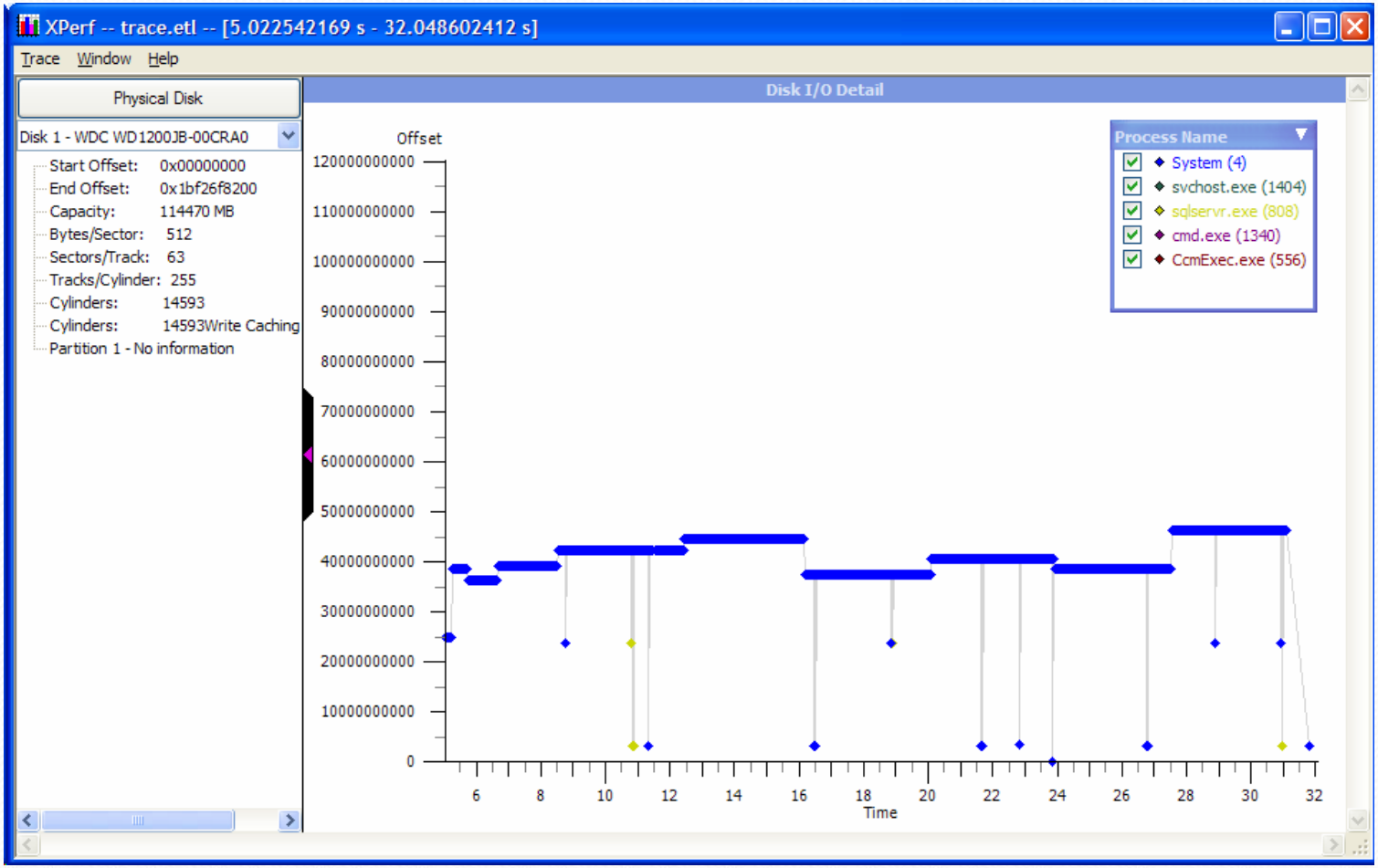


Frame Scrollbar

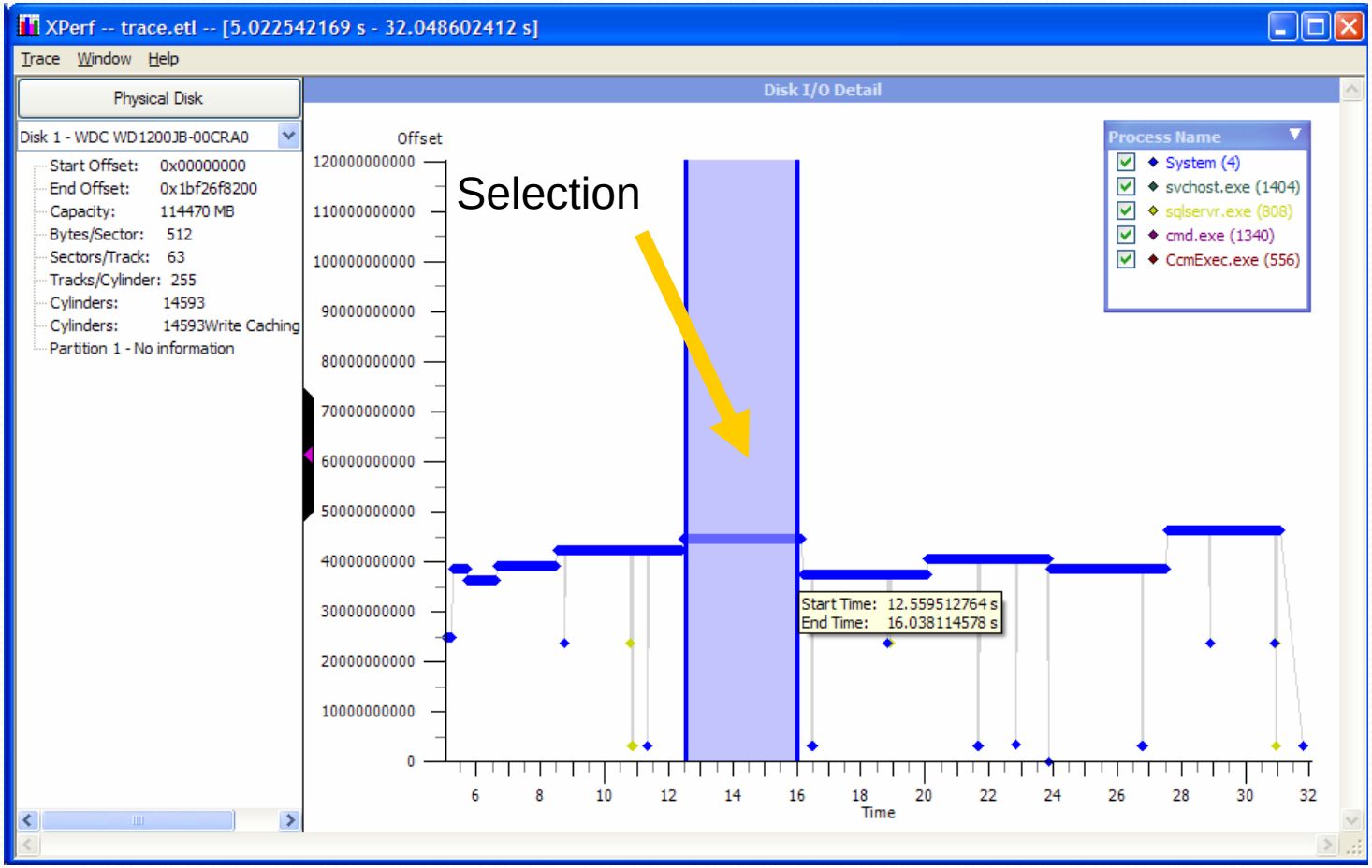
Disk I/O Detail (Disk #0)



Disk I/O Detail (Disk #1)



Disk I/O Detail (Disk #1)



Disk I/O Detail Summary Table

Default sort field

XPerf -- trace.etl -- [12.559512764 s - 16.038114578 s]

Trace Window Help

IO Type	Complete Time (ms)	IO Time (us)	Disk Service Time (us)	Pri	Queue Depth	IO Size	Byte Offset	Process	Thread ID	Path Name
Read	16033.187166	28489.439	28489.439	0x0	0	0x10000	0xa65325e00	System (4)	28	\V.vhd
Read	16004.536275	29539.162	29539.162	0x0	0	0x10000	0xa65315e00	System (4)	28	\V.vhd
Read	15974.626718	31048.878	31048.878	0x0	0	0x10000	0xa65305e00	System (4)	28	\V.vhd
Read	15943.310959	29520.195	29520.195	0x0	0	0x10000	0xa652f5e00	System (4)	28	\V.vhd
Read	15913.546337	29731.956	29731.956	0x0	0	0x10000	0xa652e5e00	System (4)	28	\V.vhd
Read	15883.622195	32557.419	32557.419	0x0	0	0x10000	0xa652d5e00	System (4)	28	\V.vhd
Read	15850.786823	29029.656	29029.656	0x0	0	0x10000	0xa652c5e00	System (4)	28	\V.vhd
Read	15821.627628	28413.707	28413.707	0x0	0	0x10000	0xa652b5e00	System (4)	28	\V.vhd
Read	15793.051066	29087.263	29087.263	0x0	0	0x10000	0xa652a5e00	System (4)	28	\V.vhd
Read	15763.816194	28135.411	28135.411	0x0	0	0x10000	0xa65295e00	System (4)	28	\V.vhd
Read	15735.490115	28547.529	28547.529	0x0	0	0x10000	0xa65285e00	System (4)	28	\V.vhd
Read	15706.831889	29251.315	29251.315	0x0	0	0x10000	0xa65275e00	System (4)	28	\V.vhd
Read	15677.286228	28559.273	28559.273	0x0	0	0x10000	0xa65265e00	System (4)	28	\V.vhd
Read	15648.587271	28430.138	28430.138	0x0	0	0x10000	0xa65255e00	System (4)	28	\V.vhd
Read	15619.960260	28967.825	28967.825	0x0	0	0x10000	0xa65245e00	System (4)	28	\V.vhd
Read	15590.847513	28538.754	28538.754	0x0	0	0x10000	0xa65235e00	System (4)	28	\V.vhd
Read	15562.117744	29982.388	29982.388	0x0	0	0x10000	0xa65225e00	System (4)	28	\V.vhd
Read	15532.027057	29190.540	29190.540	0x0	0	0x10000	0xa65215e00	System (4)	28	\V.vhd
Read	15502.686998	30607.684	30607.684	0x0	0	0x10000	0xa65205e00	System (4)	28	\V.vhd
Read	15471.959094	29325.852	29325.852	0x0	0	0x10000	0xa651f5e00	System (4)	28	\V.vhd
Read	15442.243124	28234.282	28234.282	0x0	0	0x10000	0xa651e5e00	System (4)	28	\V.vhd
Read	15413.893983	28247.493	28247.493	0x0	0	0x10000	0xa651d5e00	System (4)	28	\V.vhd
Read	15385.494070	28927.786	28927.786	0x0	0	0x10000	0xa651c5e00	System (4)	28	\V.vhd
Read	15356.441462	28362.464	28362.464	0x0	0	0x10000	0xa651b5e00	System (4)	28	\V.vhd
Read	15327.922054	29539.559	29539.559	0x0	0	0x10000	0xa651a5e00	System (4)	28	\V.vhd
Read	15298.255123	28353.890	28353.890	0x0	0	0x10000	0xa65195e00	System (4)	28	\V.vhd
Read	15269.774424	28626.728	28626.728	0x0	0	0x10000	0xa65185e00	System (4)	28	\V.vhd
Read	15241.005628	29193.402	29193.402	0x0	0	0x10000	0xa65175e00	System (4)	28	\V.vhd
Read	15211.567032	28550.502	28550.502	0x0	0	0x10000	0xa65165e00	System (4)	28	\V.vhd
Read	15182.818628	27961.457	27961.457	0x0	0	0x10000	0xa65155e00	System (4)	28	\V.vhd
Read	15154.713683	28401.088	28401.088	0x0	0	0x10000	0xa65145e00	System (4)	28	\V.vhd

In this interval: 120 reads totalling 7864.320 kB, 0 writes totalling 0.000 kB

Outline

- Motivation
- Trace and analysis tools
- **Storage workload metric and characterization**
- Newly published traces
- Analysis
- Summary

Storage Workload Metrics

- Read/write ratio
- Dominant request sizes
- Spatial locality
  - Sequential
    - interleaved streams
  - Random
  - LBN heat
- Temporal locality
  - Overwrites / block lifetimes
  - Rereads
- Boundary alignment
  - Array stripe units
- File heat
- Interarrival times
  - At subsystem
  - At disk
- Queue lengths
  - On initiation
  - Over time
- Idle time
- Self-similarity
  - Time
  - Space
- *Response time*

Storage Workload Characterization

- Compute mean, mode, standard deviation to get a first-cut rough approximation
- Trace events can be filtered to extract sub-workloads and identify trends
  - Disk # / Partition / File system
  - Reads / Writes
  - Random / Sequential / Streams
  - Request size
  - LBN range
    - File
  - Process / Thread
  - *Interarrival time*
  - *Queue length*

Outline

- Motivation
- Trace and analysis tools
- Storage workload metrics and characterization
- **Newly published traces**
  - Trace sanitization
  - Trace descriptions
  - Sample statistics
- Analysis
- Summary

Trace Sanitization

- Traces can be sanitized to desired levels
  - Files
  - Directories
  - Process names
  - Execution (command line) strings
  - Image checksums
- Currently sanitization occurs in text dump files
  - Future version of WPT should have options for sanitization of binary (.etl) files

Newly Published Traces

- Exchange server
- Live Maps front-end server
- Live Maps back-end server
- Display Ads Platform data server
- Display Ads Platform payload server
- Windows build server
- MSN storage metadata server
- MSN storage file servers
- Developer tools release server
- RADIUS authentication server
- RADIUS SQL Back-end server
- *Database benchmarks: TPC-C and TPC-E*

Sample Workload Statistics

Trace Statistics		Live Maps Front End	Live Maps Back End	Display Ads Data Server	Display Ads Payload
Total IOs (Millions)		0.342	44.77	1.53	1.09
	R	0.001	35.31	1.39	0.61
	W	0.341	9.46	0.15	0.48
Total GB Transferred		8.92	2344	42.63	80.3
Avg IO/s		3.956	517.4	17.7	12.6
Req Size Modes (KB)		4, 64	64	32	64
Avg Q Length on Initiation		7.123	1.94	0.082	0.043
Avg Response Time (ms)		11.32	1.96	3.42	0.77
Avg System InterArrival (ms)		253	1.93	56.5	79.6
% Seq IOs Initiated		32.62	70.43	0.77	61.65
Unique Files Accessed		818	15097	3647	4338
Hot Files (80% of total reqs)		16	219	1	191

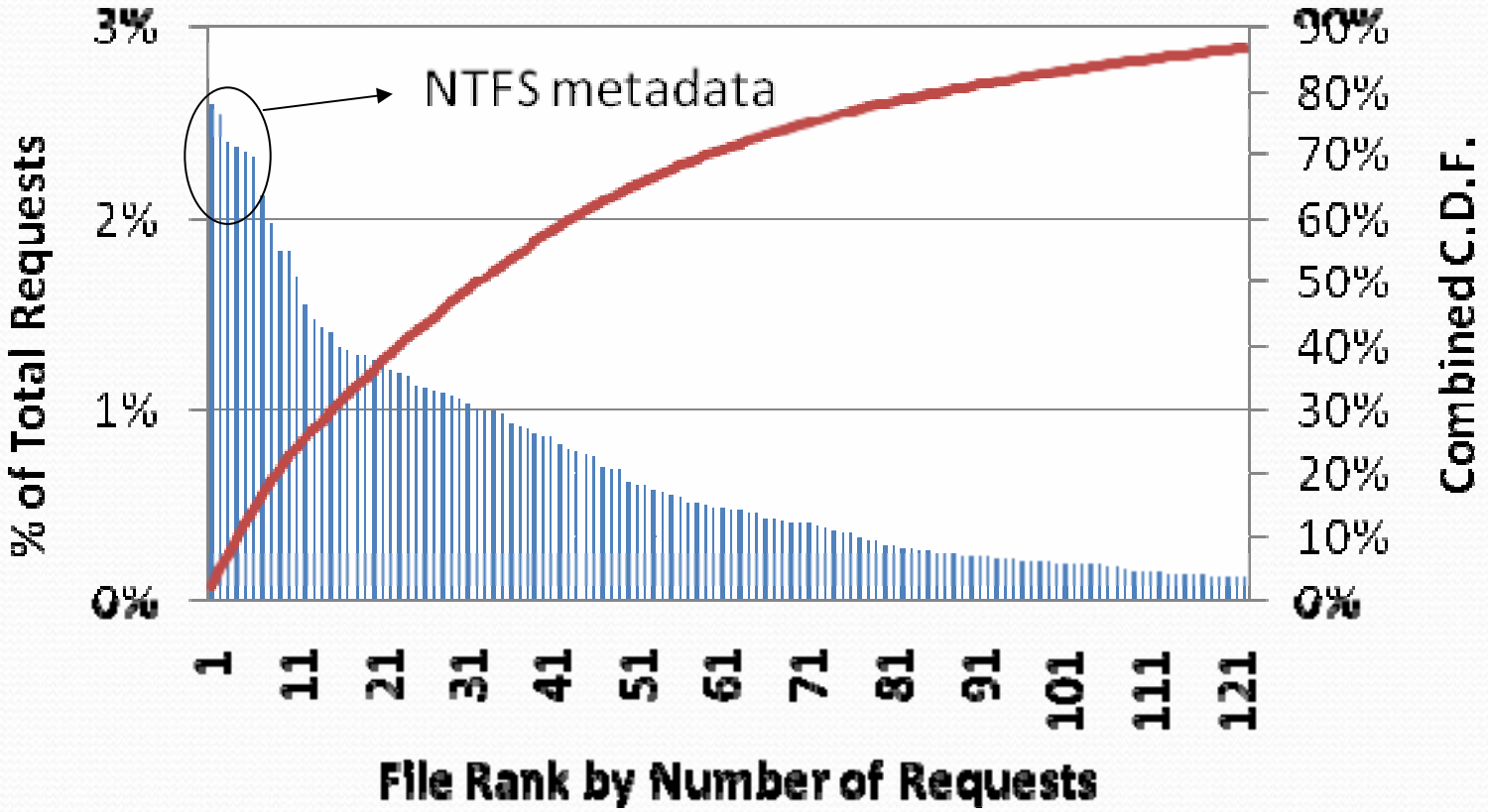
Outline

- Motivation
- Trace and analysis tools
- Storage workload metrics and characterization
- Newly published traces
- **Analysis**
  - Exchange Server
  - Sample visualizations
  - Unpublished workload
  - Self-similarity
- Summary

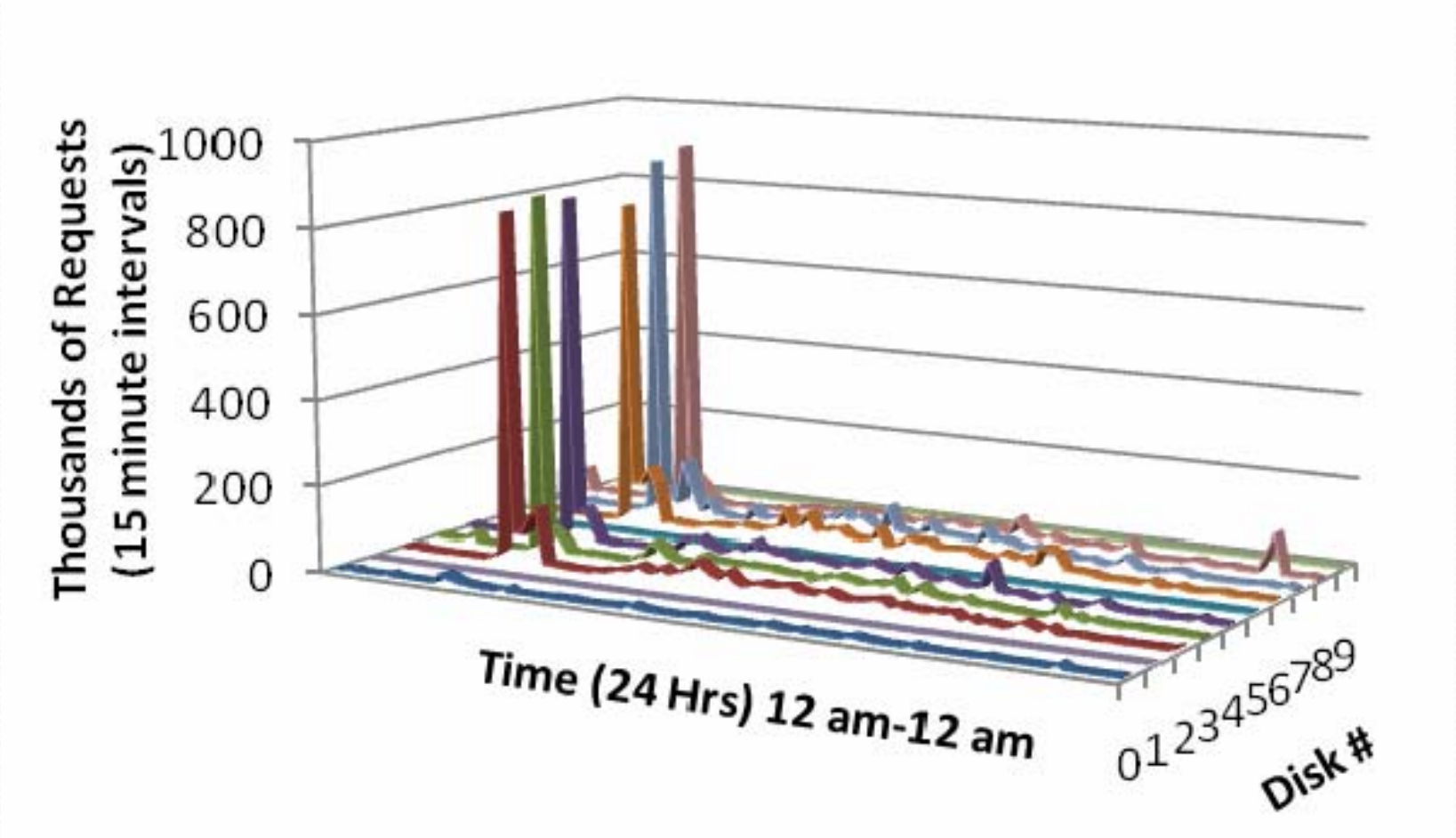
5000-User Exchange Server

- 20% disk accesses were for NTFS metadata updates
  - Can be reduced by setting appropriate registry key
- Replication or backup activity causes peak disk I/Os
- Partition alignment has an impact on disk activity
  - Windows Server 2008 and beyond does this automatically
- NTFS cluster size misalignment
  - Setting NTFS cluster size to 8 KB (Exchange Server default) would avoid misalignment
- Other activities such as antivirus can skew expected data access patterns

Exchange – File Heat



Exchange – Distribution over time



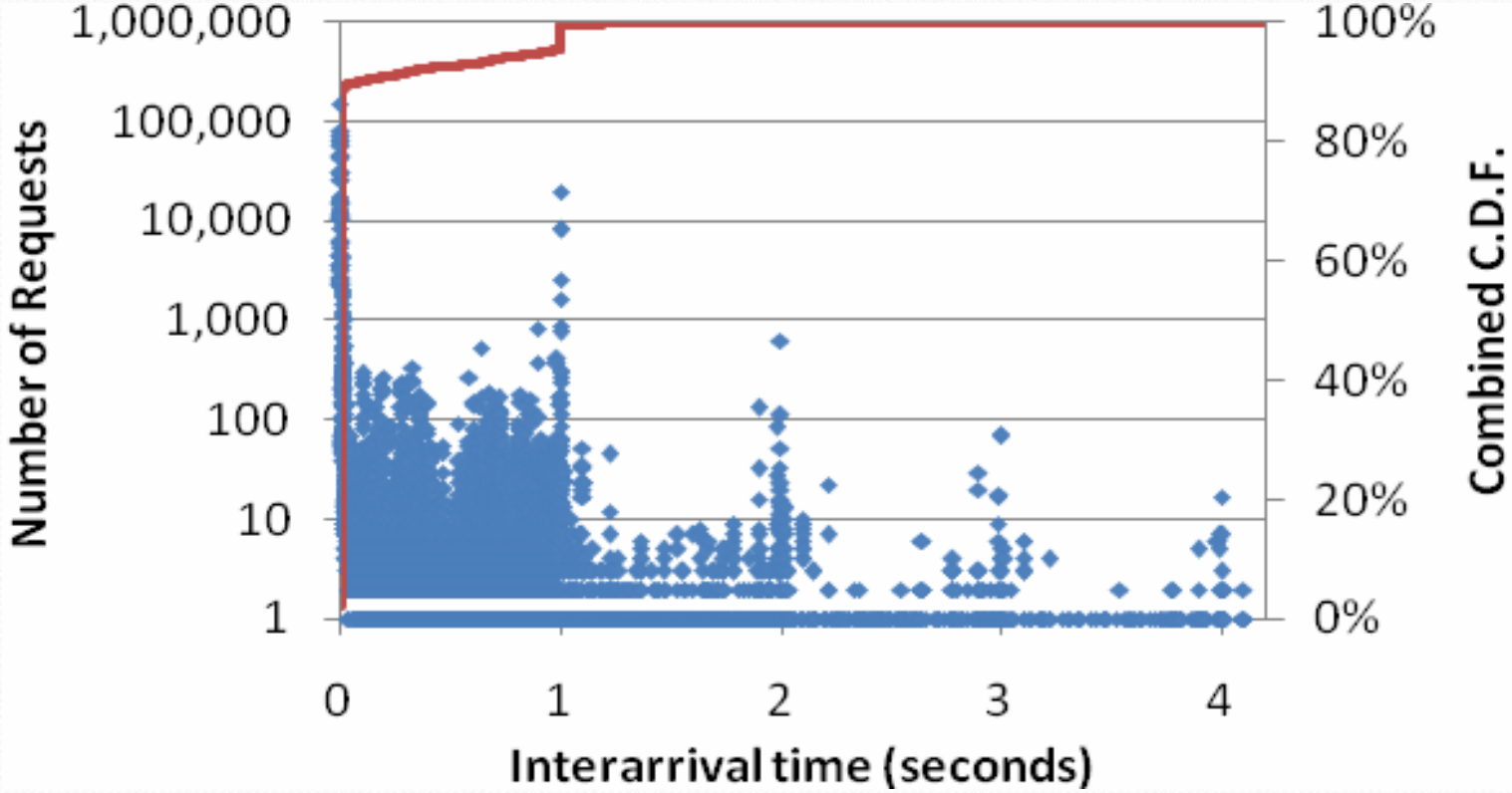
5000-User Exchange Server

- 20% disk accesses were for NTFS metadata updates
  - Can be reduced by setting appropriate registry key
- Replication or backup activity causes peak disk Ios
- Partition alignment has an impact on disk activity
  - Windows Server 2008 and beyond does this automatically
- NTFS cluster size misalignment
  - Setting NTFS cluster size to 8 KB (Exchange Server default) would avoid misalignment
- Other activities such as antivirus can skew expected data access patterns

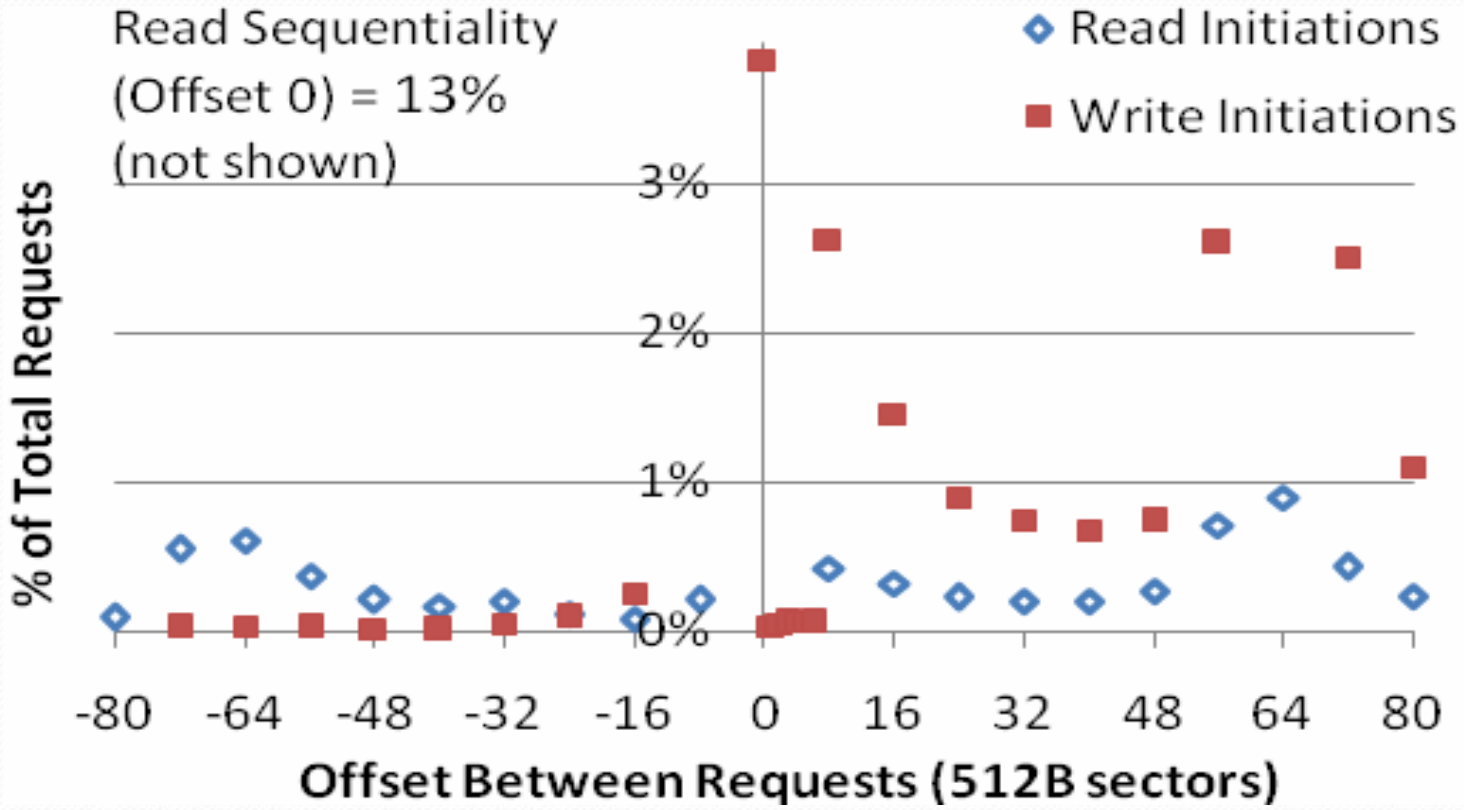
Sample Visualizations

- Interarrival Time
- Spatial locality
  - Interleaved streams
- File Heat
- LBN Heat
- Queue lengths

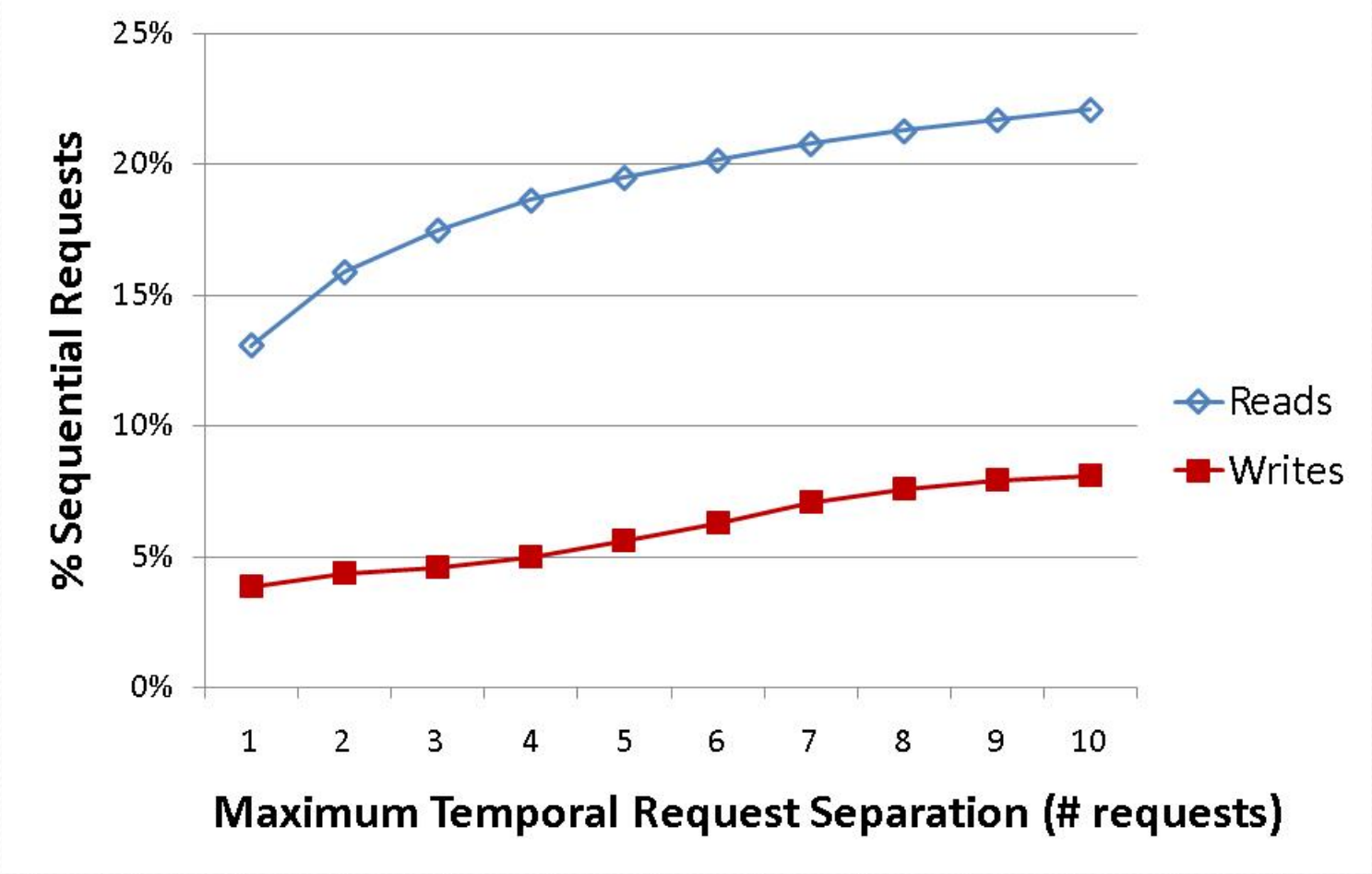
Interarrival Time (DAP-PS)



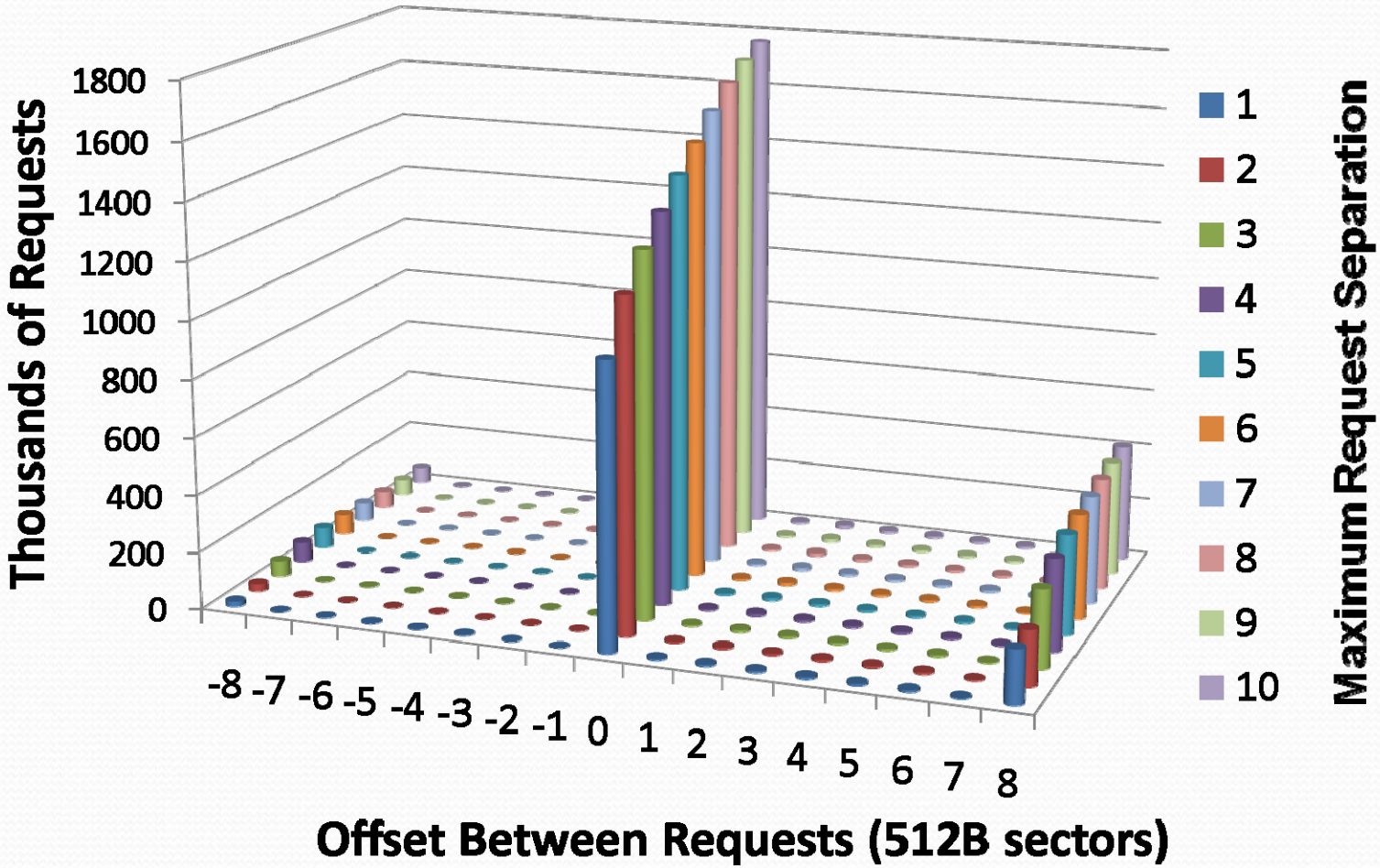
Tight Spatial Locality (WBS)



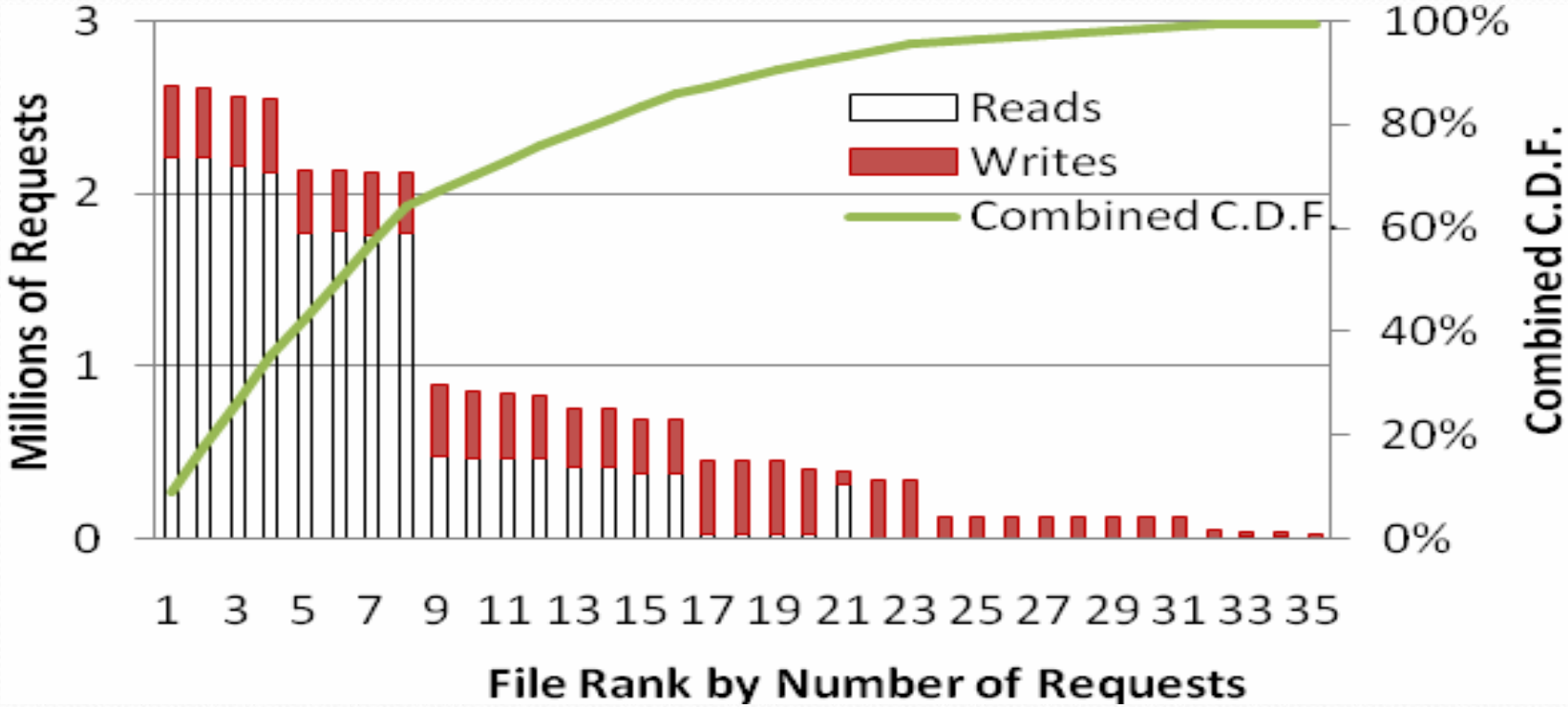
Interleaved Streams (WBS)



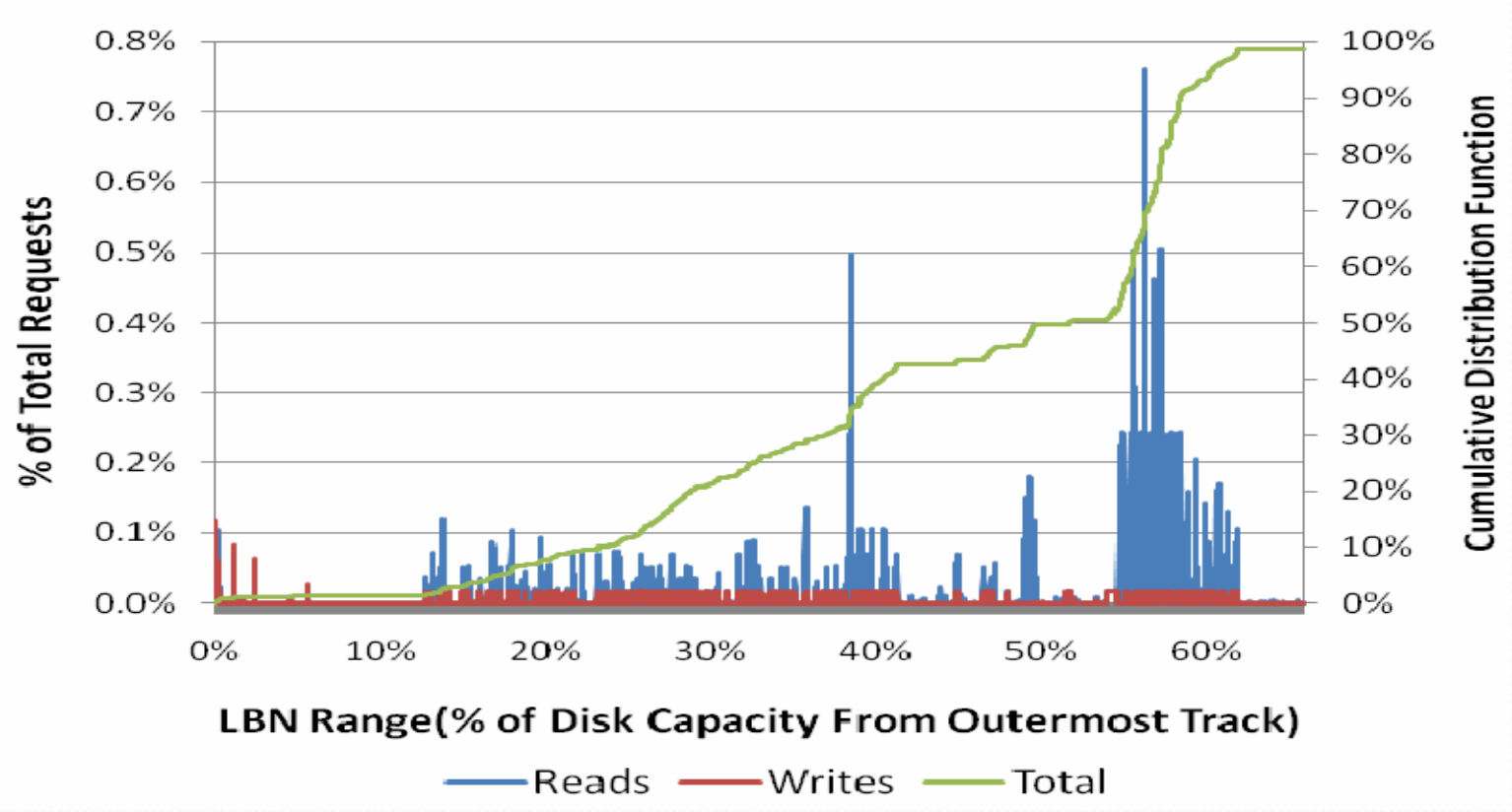
Interleaved Streams (WBS)



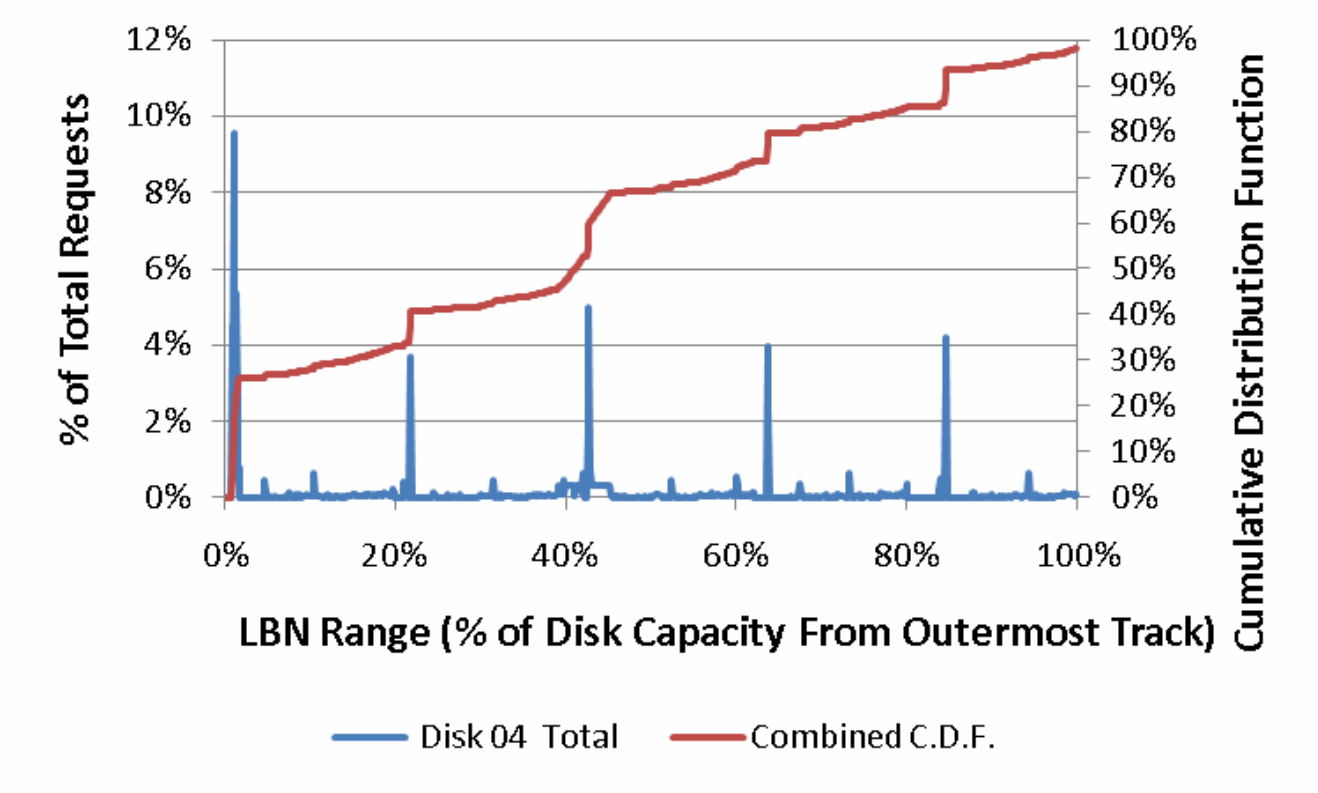
File Heat (MSN-BEFS)



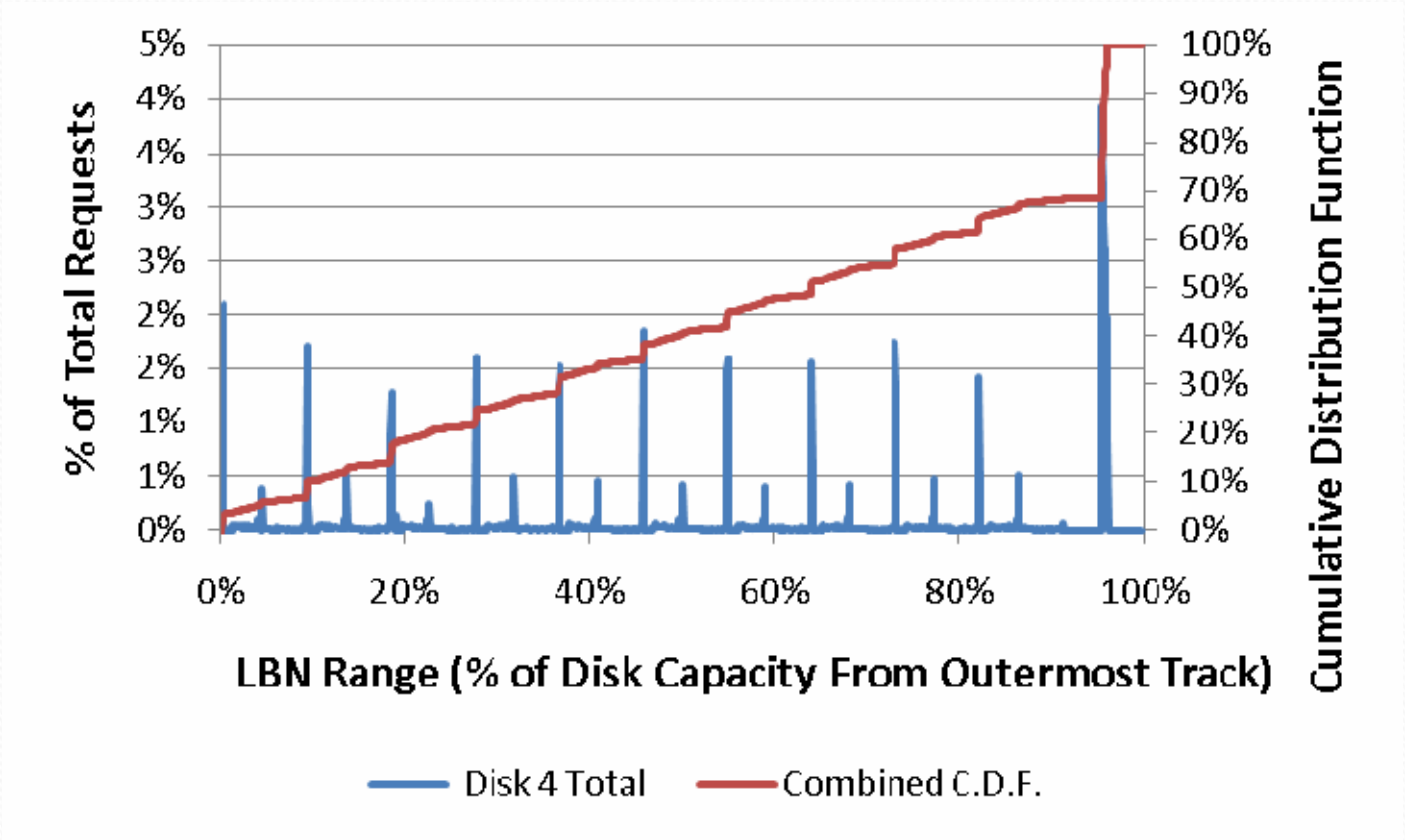
LBN Heat (LM-TBE)



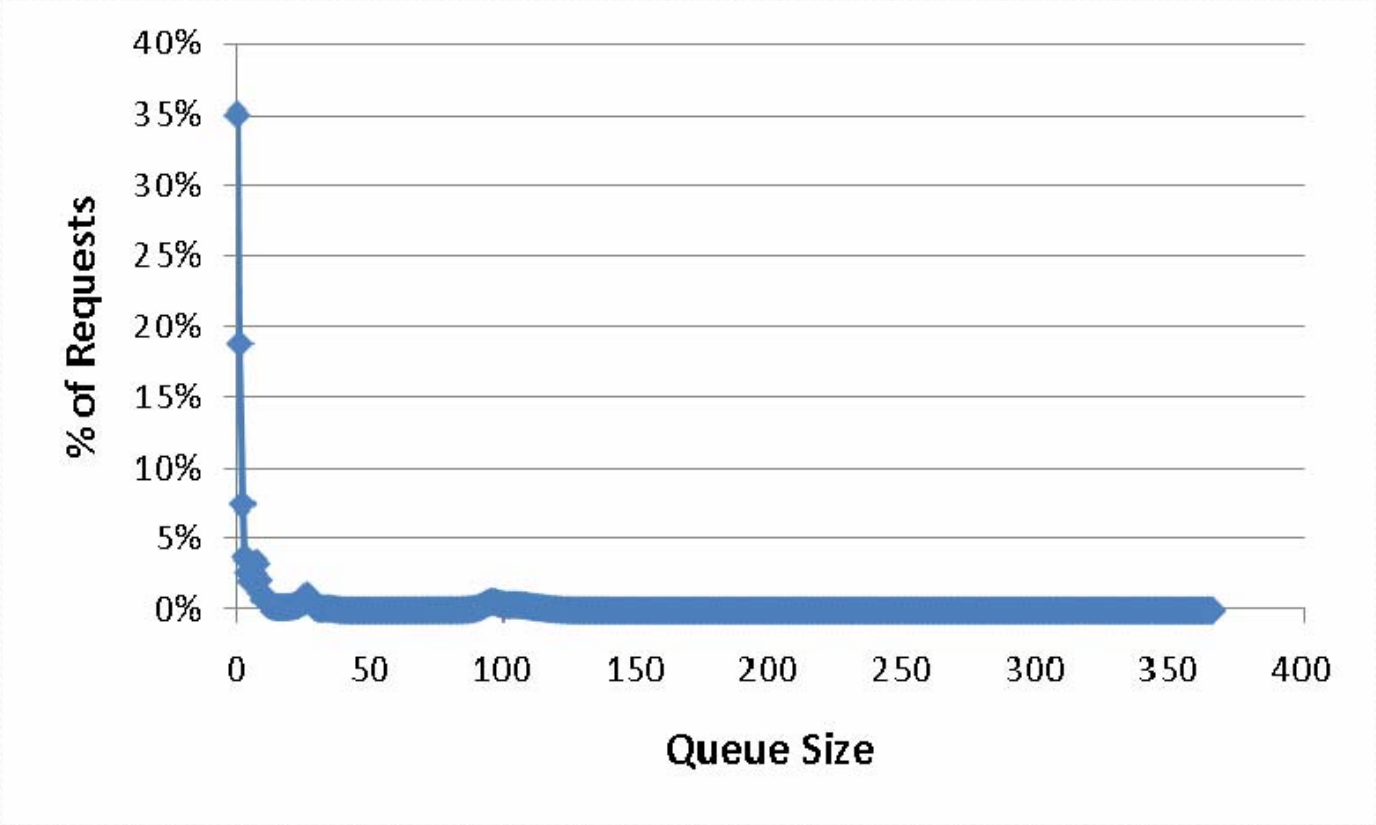
LBN Heat (Unpublished Workload)



LBN Heat (Unpublished Workload)



Initiation Queue Length (Unpublished Workload)



Self-similarity

- Methodology
  - Hurst parameter estimated using variance-time plot
- Traces
  - DAP-DS, DAP-PS, WBS, and MSN-BEFS
- Two dimensions
  - In time: # I/O requests per second over time
  - In space: # I/O requests per starting block bucket over logical disk blocks
- Observation
  - Both read and write I/O requests display strong self-similarity in time and space

Hurst Parameters

Temporal Self-similarity

Trace	R	W	Total
DAP-DS	0.966	0.832	0.938
DAP-PS	0.461	0.705	0.548
WBS	0.945	0.939	0.958
MSN-BEFS	0.878	0.744	0.872
Disko	N/A	0.686	0.676
Disk1	0.844	0.883	0.893
Disk4	0.875	0.688	0.865
Disk5	0.878	0.688	0.869

Spatial Self-similarity

Trace	R	W	Total
DAP-DS	0.983	0.616	0.940
DAP-PS	0.963	0.564	0.581
WBS	0.827	0.700	0.792
MSN-BEFS			
Disko	N/A	0.938	0.940
Disk1	0.967	0.931	0.934
Disk4	0.931	0.847	0.919
Disk5	0.936	0.857	0.925

- Self-similarity exists when $H>0.5$
- Higher H, stronger self-similarity

Summary: Trace Capture and Analysis

- Event Tracing for Windows (ETW) = the engine
  - Extensive instrumentation built into retail Windows
  - NT Kernel Provider covers most kernel-level activity
- WPT/xperf = interactive browser and command line tool
  - An ETW trace controller and decoder
  - Detailed visualization via high-level graphs, summary tables, and individual events
  - Exports human-readable decoding of trace events

Summary: Next Steps

- Publish additional traces through SNIA IOTTA
- Publish longer traces
- Publish scripts and Excel macros
  - SNIA has supposedly resolved their legal issues
- Devise new visualizations to aid analysis
- Automate sub-workload extraction
- Capture multi-week/month traces

References

- Event Tracing for Windows on MSDN
  http://msdn2.microsoft.com/en-us/library/aa363787.aspx
- Windows Performance Tools Kit
  http://www.microsoft.com/whdc/system/sysperf/perftools.mspx
- SNIA IO Trace Repository
  http://iotta.snia.org
- "Windows Internals 4th edition" by Russinovich and Solomon
  - 5th edition (Windows Server 2008) being written

Q & A