A Principled Approach for Selecting Block I/O Traces

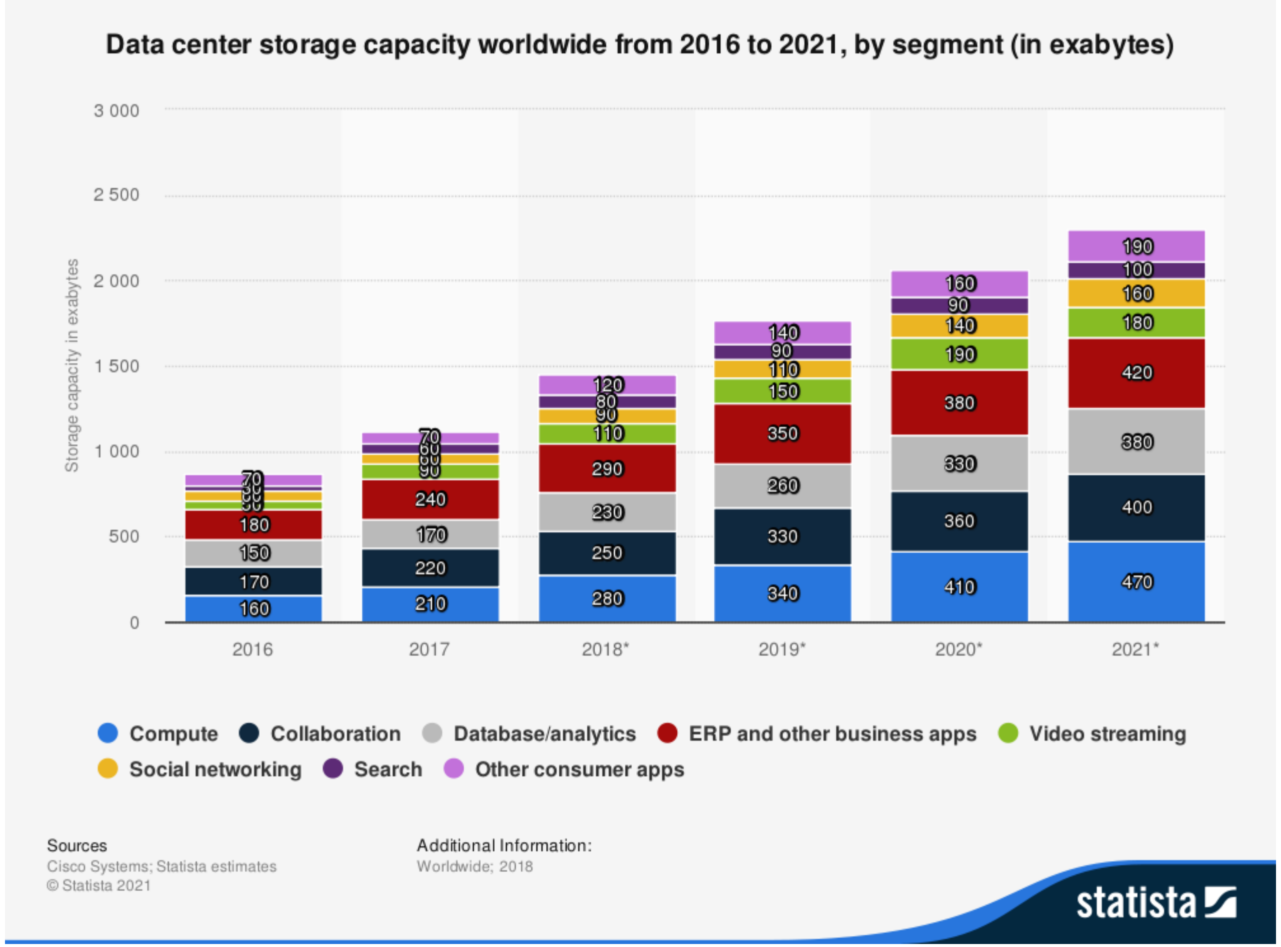
Omkar Desai (Syracuse University)
Seungmin Shin (Soongsil University)
Eunji Lee (Soongsil University)
Bryan S. Kim (Syracuse University)

Overview

1. I/O traces in storage systems
2. System design
3. Evaluation methodology & results
4. Conclusion

Why are traces important

- Our storage stack was imagined and built more than 25 years ago
- Changing dynamics of data requires a reimagination of the storage stack
- Traces are important to understand this and adapt systems to cater to new needs



Data center storage capacity worldwide from 2016 to 2021, by segment (in exabytes)

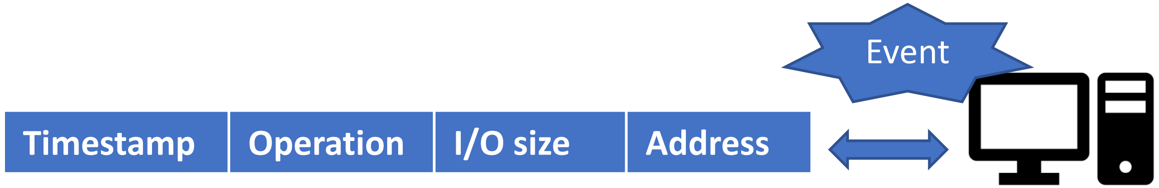
Sources
Cisco Systems; Statista estimates
© Statista 2021

Additional Information:
Worldwide; 2018

https://www.statista.com/

Traces

- Block I/O traces – a record of a sequence of I/O commands issued to a storage system.
- Typical fields:
  - Timestamp
  - Operation
  - I/O size
  - Address
- Use cases:
  - Understand behavior of storage systems in production
  - Evaluate and build better storage systems



Timestamp	Operation	I/O size	Address

Timestamp	Operation	I/O size	Address
0	read	1024	306

Challenges with block I/O traces

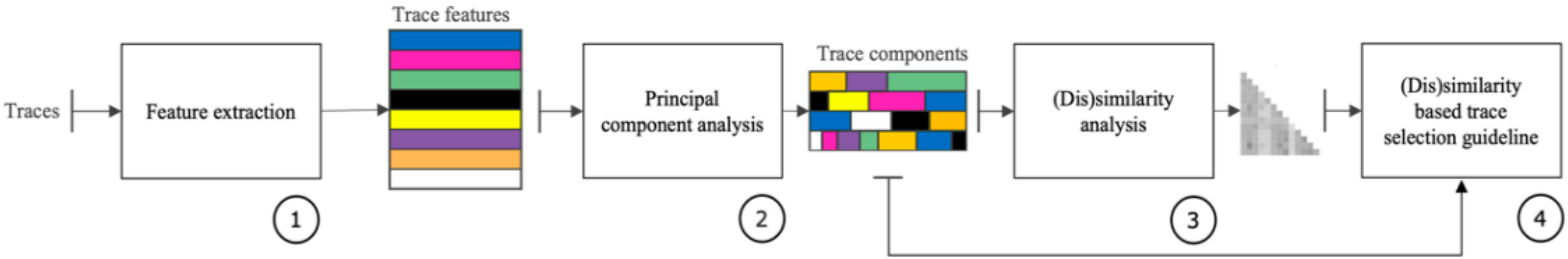
- Large in size
- Increasing number of traces
- Absence of inherent (dis)similarities with other traces
- Difficulty in distribution and replay

Trace	# of files	# of I/Os	Total time
YCSB RocksDB	27	352 M	0.4 Days
Virtual Desktop	2694	4.3 B	103.3 Days
Nexus 5	31	410 K	23.3 Mins
Slacker	57	274.2 K	13.9 Mins
MS Prod	297	1 B	120 Days
MS Ent	116	2.6 B	120 Days
MSR C	36	434 M	8 Days
Total	3258	8.7 B	441 Days

Overview

1. I/O traces in storage systems
2. System design
3. Evaluation methodology & results
4. Conclusion

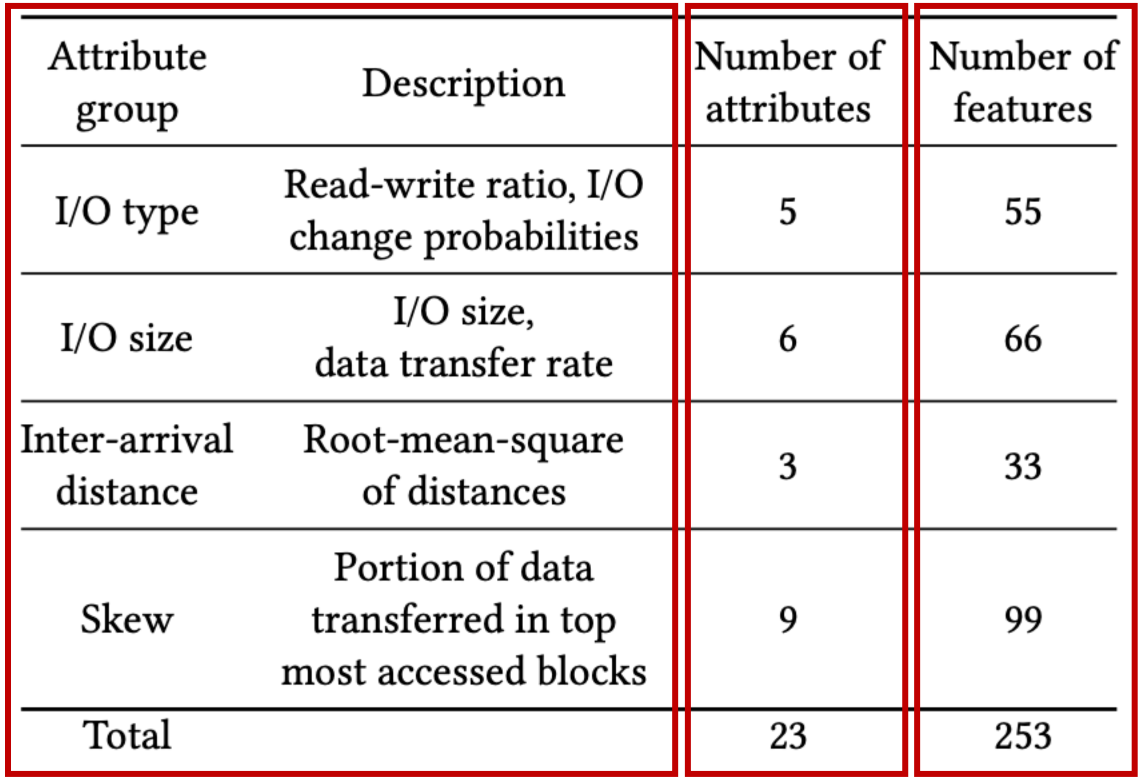
IOTap: I/O trace analysis and profiling



1. Extracts 253 features from each trace
2. Reduces its dimensionality through PCA (Principal Component Analysis)
3. Computes the (dis)similarity between traces based on their distance in the PC dimension
4. Sets forth a guideline for selecting traces based on clustering similar traces

Feature selection

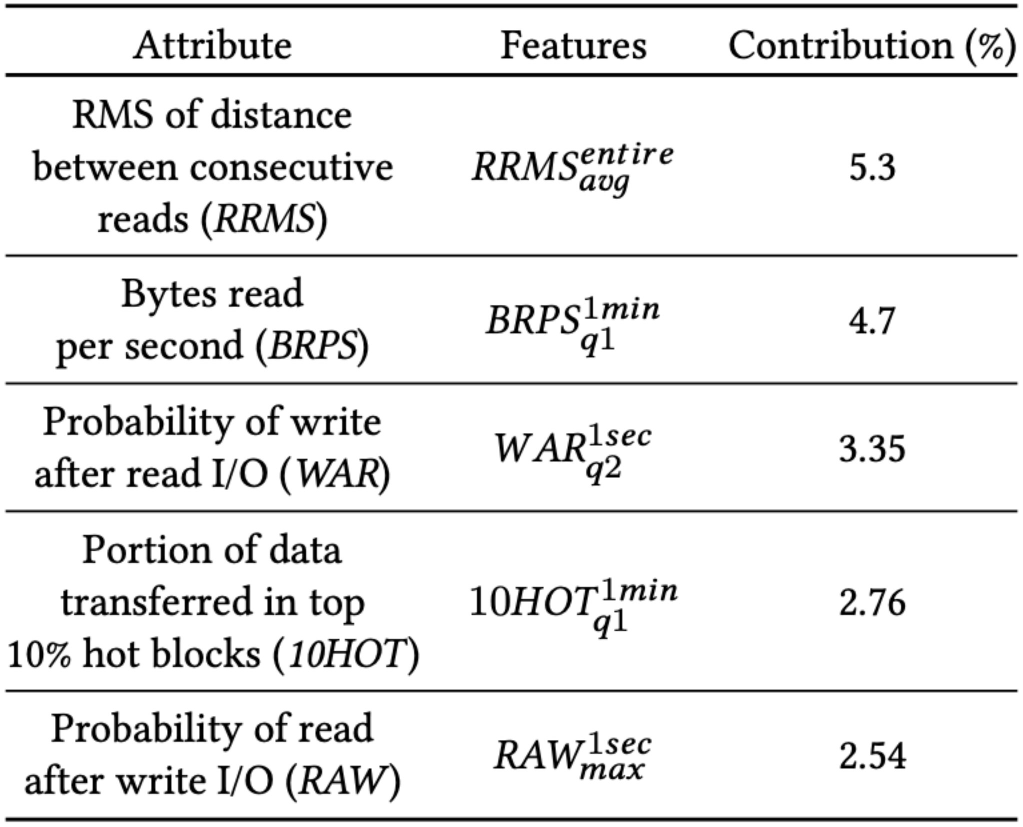
- Incorporate all major indicators impacting I/O performance
- Attributes extracted for reads, writes and combined operations
- Attributes at various chunking intervals capture changing dynamics of workloads



Attribute group	Description	Number of attributes	Number of features
I/O type	Read-write ratio, I/O change probabilities	5	55
I/O size	I/O size, data transfer rate	6	66
Inter-arrival distance	Root-mean-square of distances	3	33
Skew	Portion of data transferred in top most accessed blocks	9	99
Total		23	253

Identifying the important attributes

- Identified the top features used by PCA for dimensionality reduction
- RMS of distance between reads has the highest contribution
- Randomness in accesses and the probability of change in I/O type are discerning attributes in our analysis



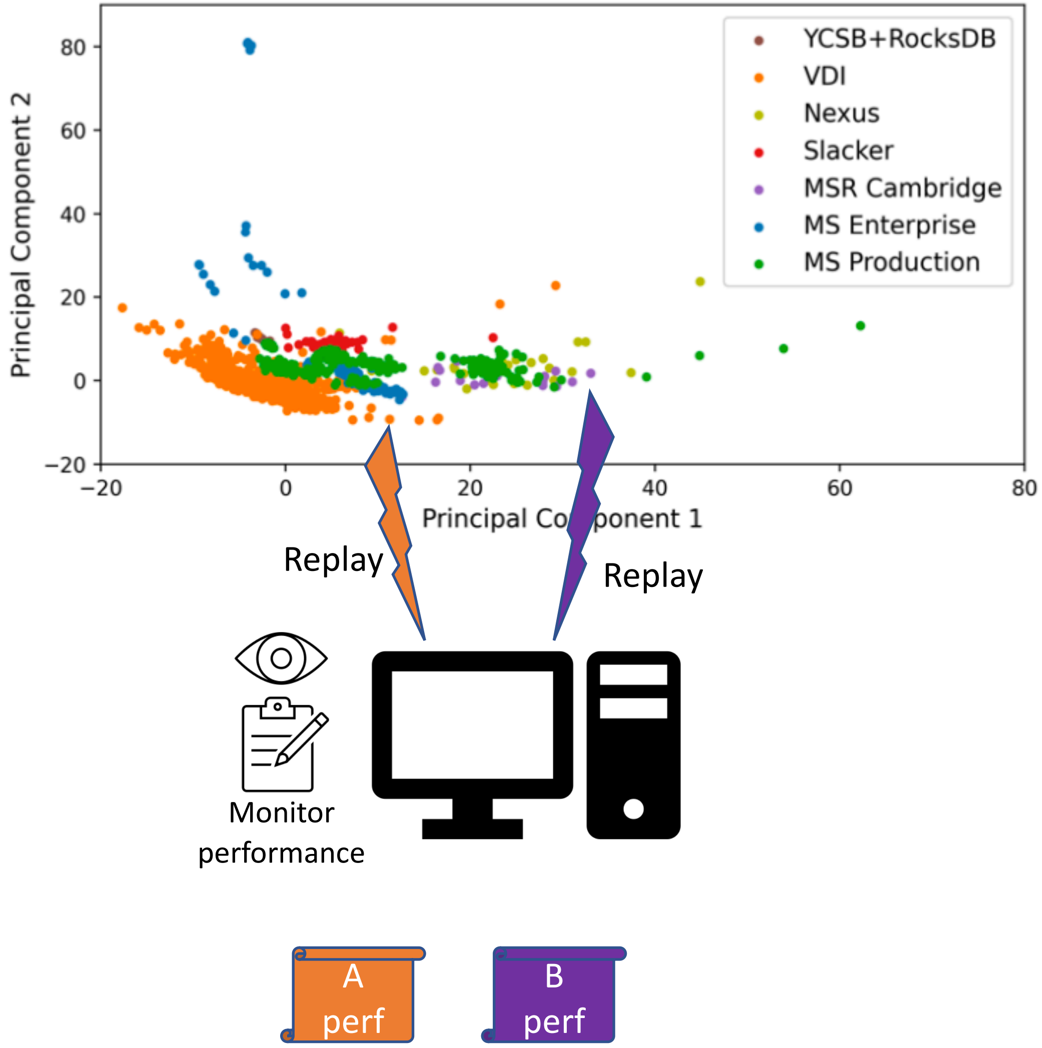
Attribute	Features	Contribution (%)
RMS of distance between consecutive reads (*RRMS*)	$RRMS_{avg}^{entire}$	5.3
Bytes read per second (*BRPS*)	$BRPS_{q1}^{1min}$	4.7
Probability of write after read I/O (*WAR*)	WAR_{q2}^{1sec}	3.35
Portion of data transferred in top 10% hot blocks (*10HOT*)	$10HOT_{q1}^{1min}$	2.76
Probability of read after write I/O (*RAW*)	RAW_{max}^{1sec}	2.54

Overview

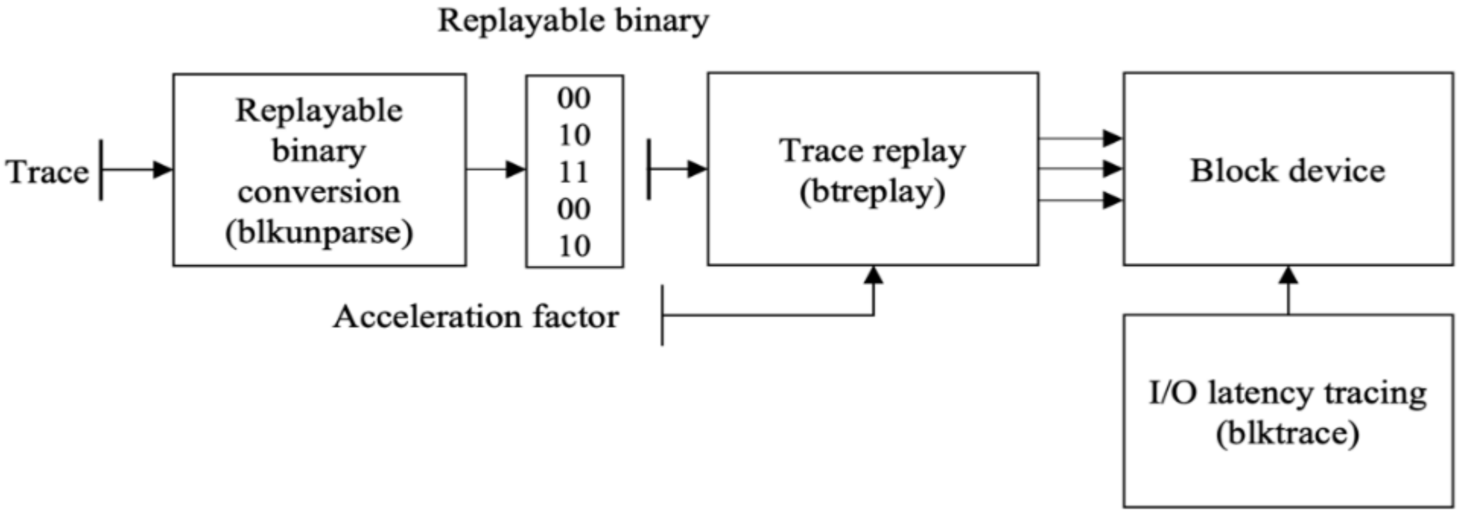
1. I/O traces in storage systems
2. System design
3. Evaluation methodology & results
4. Conclusion

Evaluation methodology

- Hypothesis: Traces that are close together when analyzed using IOTap will have a similar performance compared to traces that are further apart.
- Testing of hypothesis: Monitor performance of a device (SSD) while subjecting it to the same workload.
- Quantification: Measure similarity in performance distribution using Kolmogorov-Smirnov (KS) test



Experimental setup

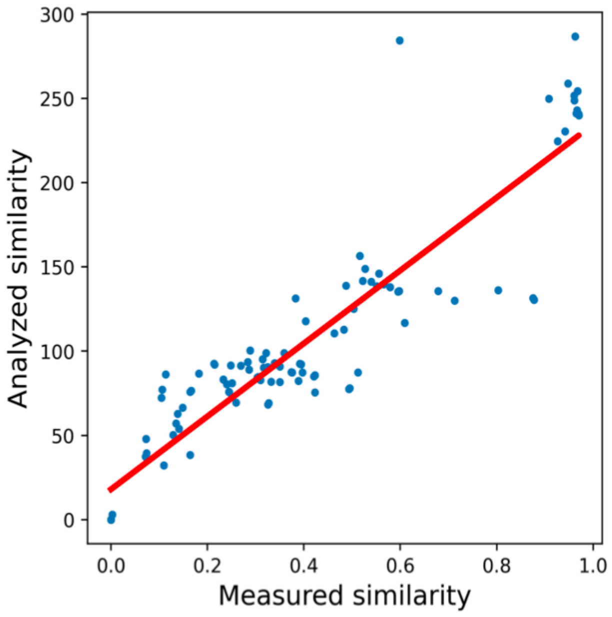


1. Blkunparse takes a record of block I/O activity and converts it into a replayable binary
   - Blkunparse is open sourced and available for use at github.com/swiftomkar/IOTap
2. Btreplay is used to replay the trace with an acceleration factor
   - Accel. factor is is determined based on the target device's performance profile
3. Blktrace is used to trace the performance of the device under replay load and latencies are interpreted from the traced data

https://git.kernel.org/pub/scm/linux/kernel/git/axboe/blktrace.git/

Results

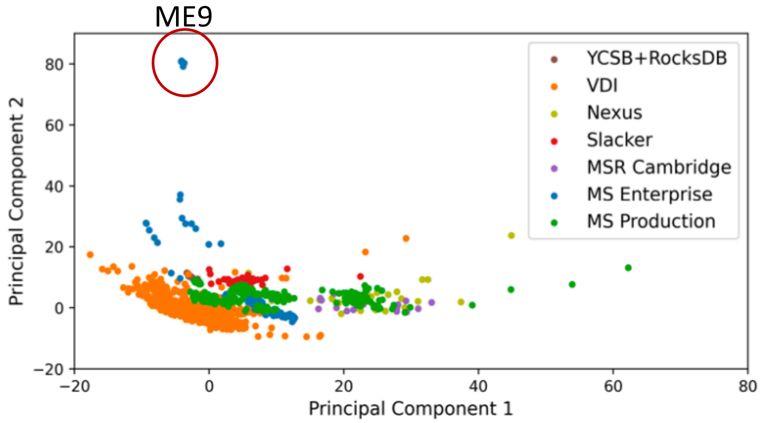
- Figure alongside show the accuracy of our work
- The line in red shows the regression fit
- Experiments were conducted on SSDs
- In future HDDs, RAID arrays and more will be evaluated

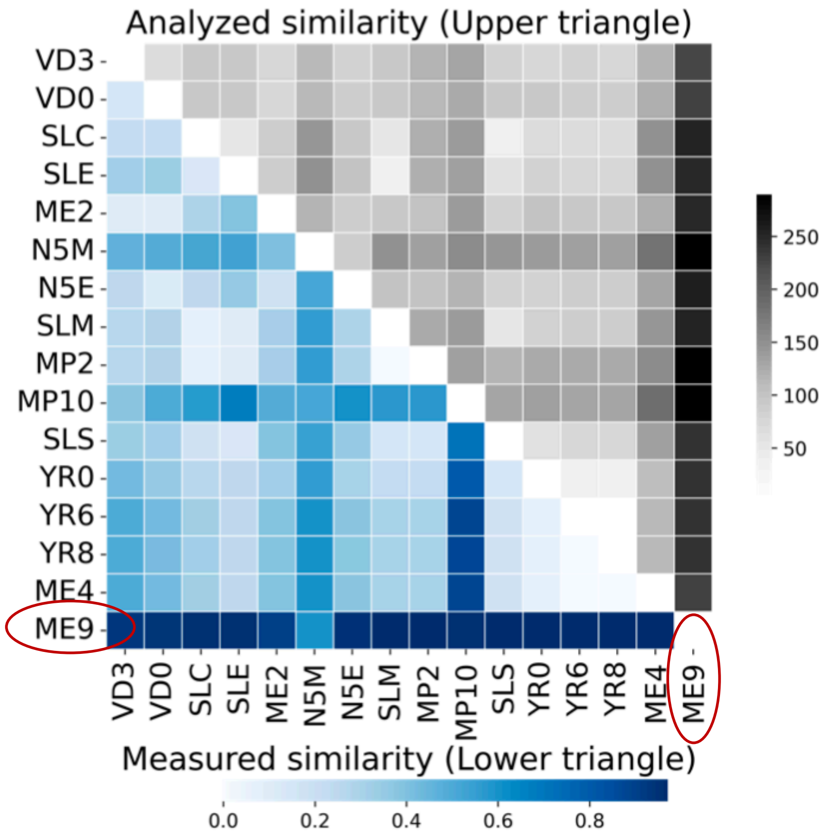


NVMe SSD
0.92

Fine grain analysis

- Similarity matrices show the accuracy of analysis with every evaluated trace
- A greater degree of diagonal mirroring means a higher correlation between the analytical and empirical results.
- A simple look at ME9 shows that measured and analyzed results are consistent

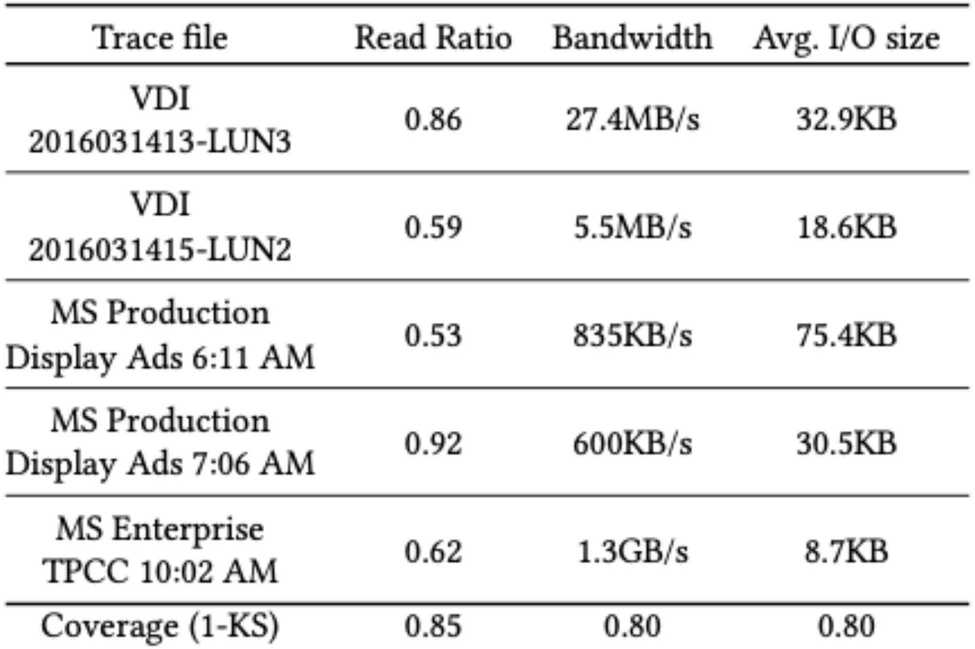




(a) Similarity matrix.

Trace sampling

- Our trace sampling module clusters similar traces together
- A representative trace is chosen from each cluster as part of a sampled set
- An example of a set 5 traces cover at least 80% of analyzed I/O spectrum



Trace file	Read Ratio	Bandwidth	Avg. I/O size
VDI 2016031413-LUN3	0.86	27.4MB/s	32.9KB
VDI 2016031415-LUN2	0.59	5.5MB/s	18.6KB
MS Production Display Ads 6:11 AM	0.53	835KB/s	75.4KB
MS Production Display Ads 7:06 AM	0.92	600KB/s	30.5KB
MS Enterprise TPCC 10:02 AM	0.62	1.3GB/s	8.7KB
Coverage (1-KS)	0.85	0.80	0.80

Conclusion

- 2 seemingly different traces may have the same effect on a storage device
- IOTap is able to analytically compute the dis-similarities between different traces
- It can be used to provide a selection guideline

IOTap’s accuracy is 92%

Q & A

odesai@syr.edu