DBMS workloads in online services

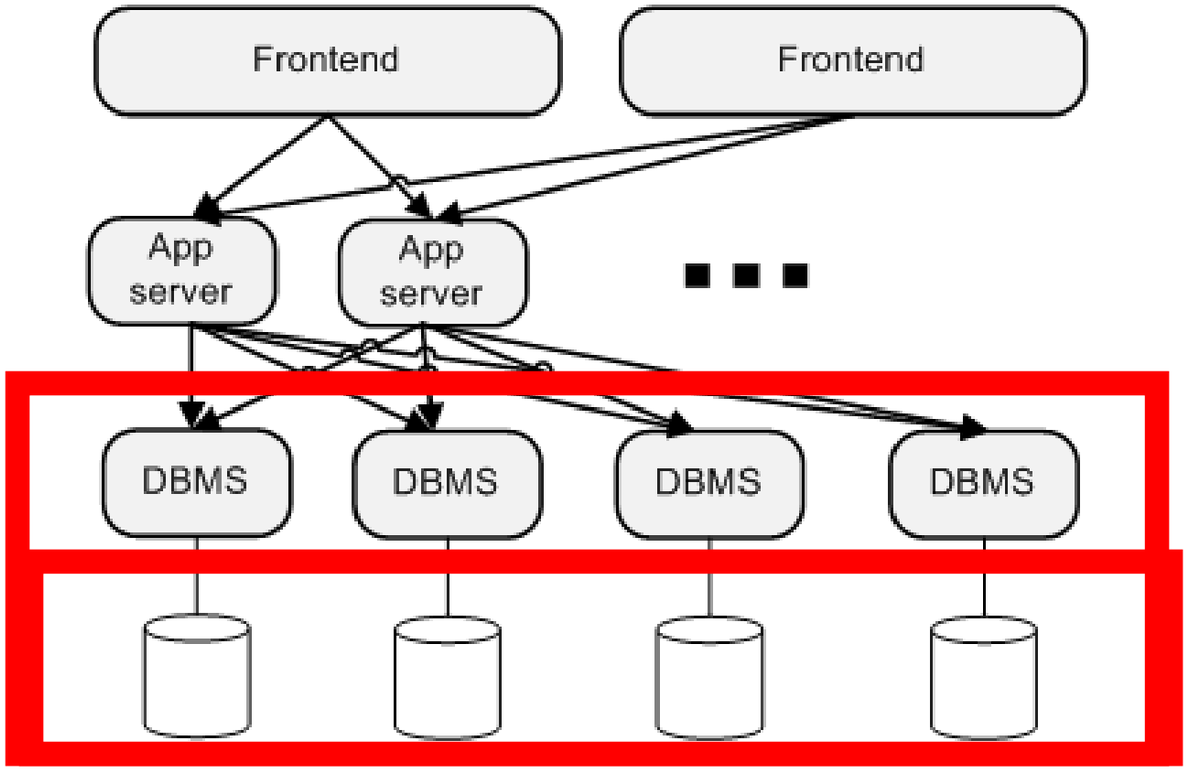
Swaroop Kavalanekar, **Dushyanth Narayanan**, Sriram Sankar,
Eno Thereska, Kushagra Vaid, and Bruce Worthington

Microsoft Redmond and Microsoft Research Cambridge

Large-scale online services

- 1000s of servers
- Millions of users
- In mega-scale data centers
  - Each hosting many such services
- Server, infrastructure costs dominate
- Rightsizing is key
  - pick the right #servers

Large-scale online services



Structured storage tier I/O

Load variation over time

- User-facing services show diurnal pattern
  - "Pacific Ocean trough"
- Important to understand
  - Consolidate un/anti correlated workloads
  - Schedule background tasks intelligently
  - Power down resources at low load
- Potentially big $$$ at mega-DC scale

Challenges

- Rightsizing
  - How many servers, and what hardware?
  - How much disk space v. IOPS v. CPU ...
- Consolidation
  - Which workloads are un(anti)correlated
- Power-proportionality
  - Maximize work done / Joule

In this talk

- Analyze I/O traces of real workloads
  - from structured storage in online services
- Characterize the workloads
  - Using a variety of metrics
- Compare with standard TPC benchmarks
  - How well do they match?

Outline

- Motivation
- Online workload analysis
- Conclusion

Workloads studied

- IM-DB
  - Messenger user profiles, buddy lists
- MSN-DB:
  - Web content for online portal
- EMAIL-DB
  - E-mail service metadata
- BLOB-DB
  - Metadata for blob store (blobs = photos, videos, ...)

Production server tracing

- Gives a very realistic picture of workload
- Low-overhead tracing infrastructure
  - Event Tracing for Windows (since Win 2000)
- Not trivial to setup (but worth it)
  - Avoid operations impact
  - Anonymize PII
  - Build trust with stakeholders

Production server tracing

- 4 services, 1 representative server each
- Traced every block-level read and write
  - Device number, offset, size, timestamp
- Trace periods vary (25 min – 24 hrs)
- Below the buffer cache
- Also traced runs of TPC-C, TPC-E, TPC-H

Workload traces

Workload	Trace length	Storage arrays	Total disks
IM-DB	25 min	5 x RAID-10	34
MSN-DB	24 hrs	10 x RAID-10	46
EMAIL-DB	2 hrs	4 x RAID-10	34
BLOB-DB	24 hrs	10 x RAID-10	46
TPC-C	6 min	14 x RAID-0	392
TPC-E	17 min	12 x RAID-0	336
TPC-H	1.5 hrs	4 x RAID-0	36

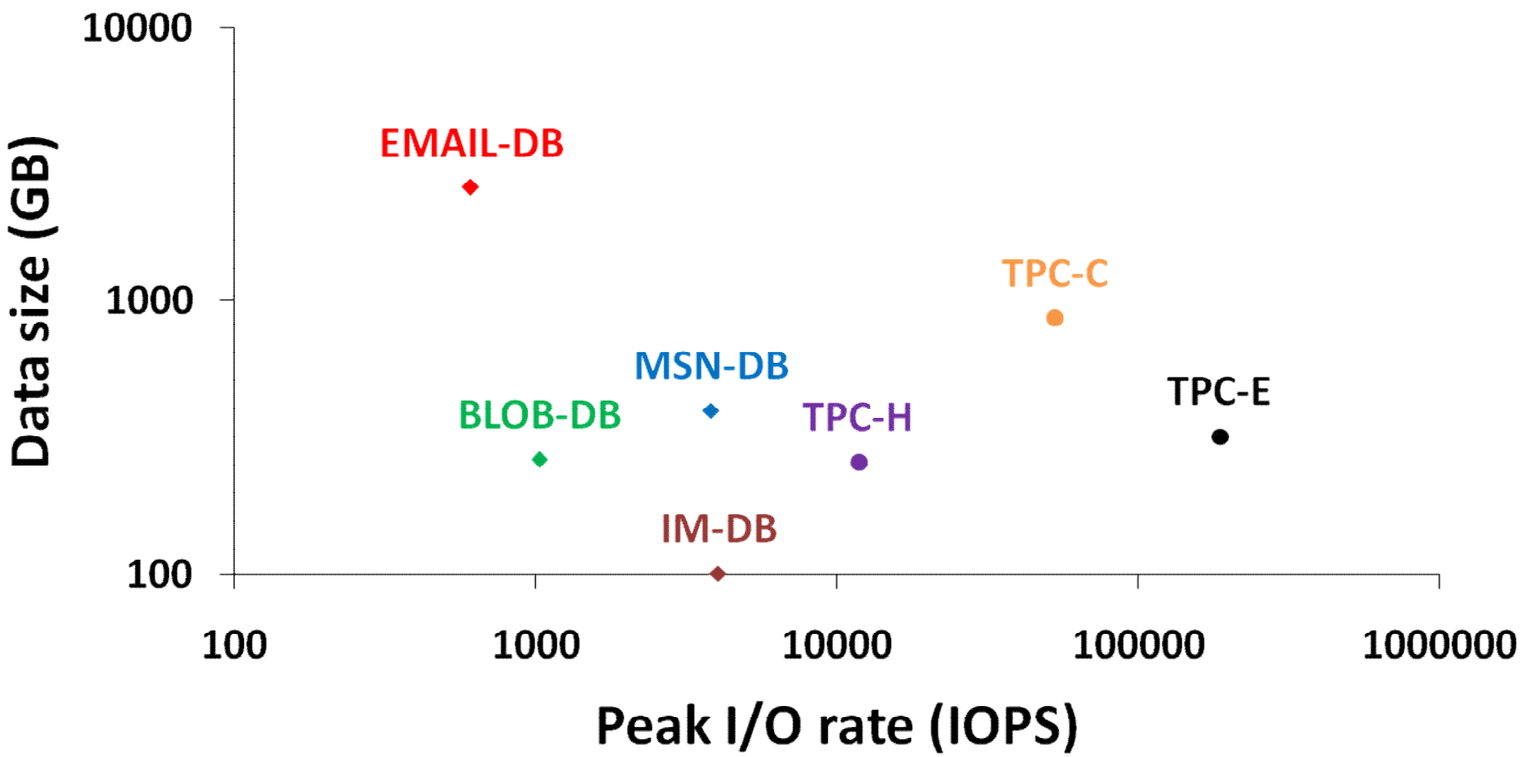
Workload trace observations

- Data file I/Os dominate
  - Log traffic is 11-12% for BLOB-DB, MSN-DB
  - < 2% for others
- Traced servers provisioned differently
  - 34 – 392 spindles
- Need to normalize load “per unit storage”
  - We normalize by data size, e.g. IOPS/GB

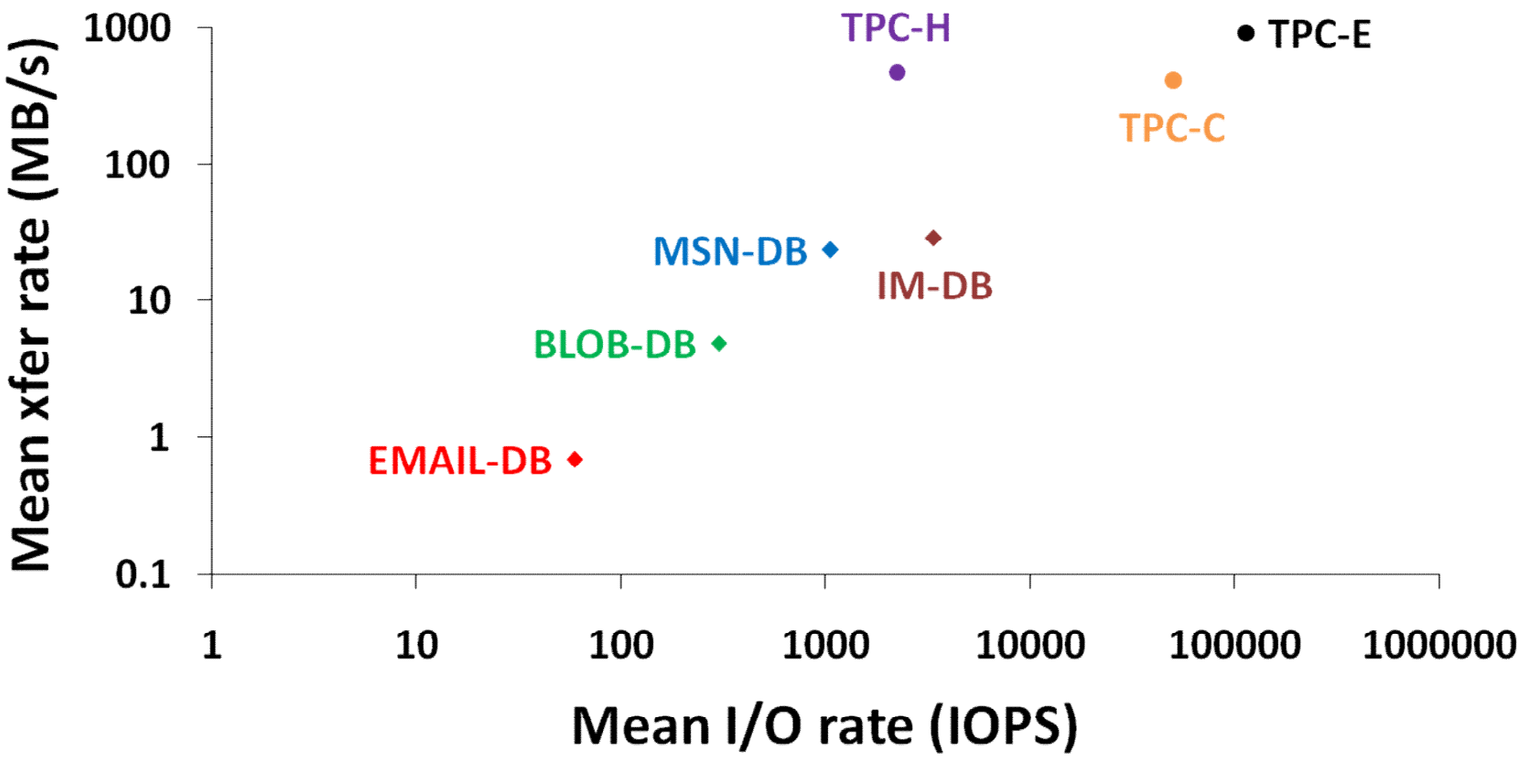
Workload metrics extracted

- Peak non-sequential request rate (IOPS)
- Peak sequential transfer rate (MB/s)
- Peak-to-mean ratios (for IOPS, MB/s)
- Data set size (GB)
  - Based on highest LBN accessed in trace
- Sequential fraction of I/Os
- Read/write ratio

Peak IOPS v data size (log-log)



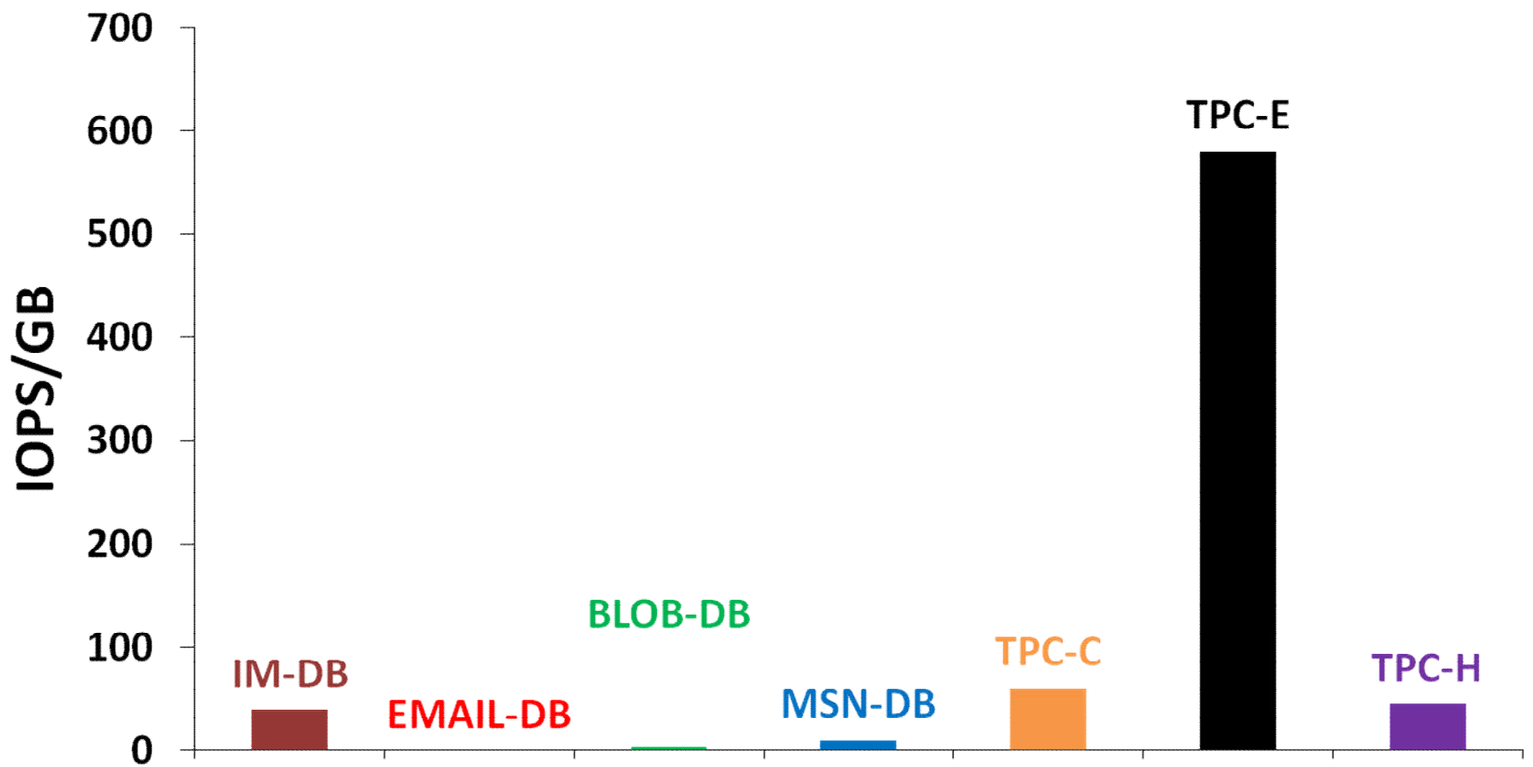
I/O rate v transfer rate (log-log)



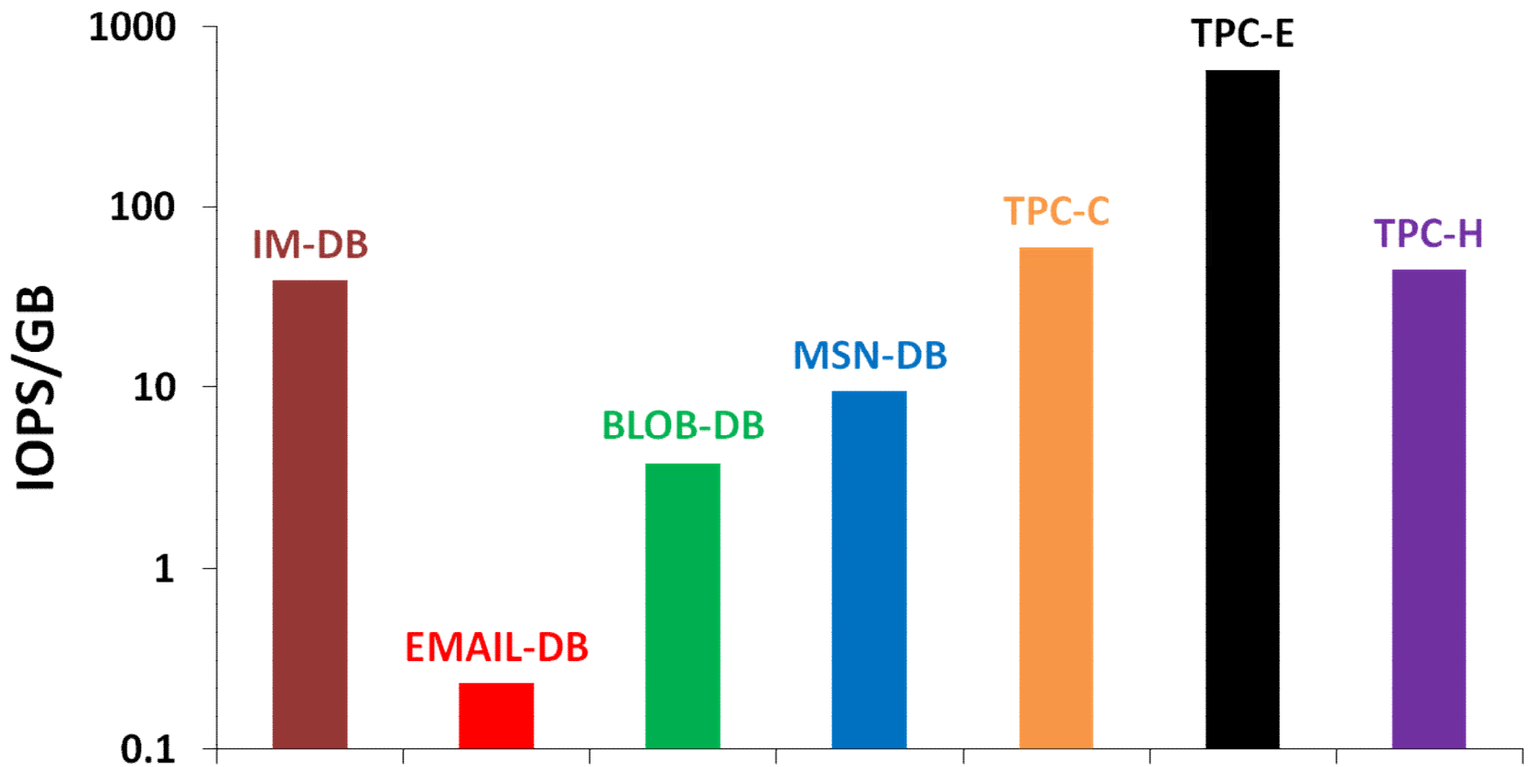
IOPS v data size

- Order-of-magnitude differences
  - Between all workloads (online & TPC)
- But, servers provisioned differently
  - TPC-C had 10x the spindles of EMAIL-DB
- We should look at load *per unit storage*
  - IOPS/GB, not IOPS/traced server
- IOPS and MB/s highly correlated
  - SQL Server uses mostly 8KB requests

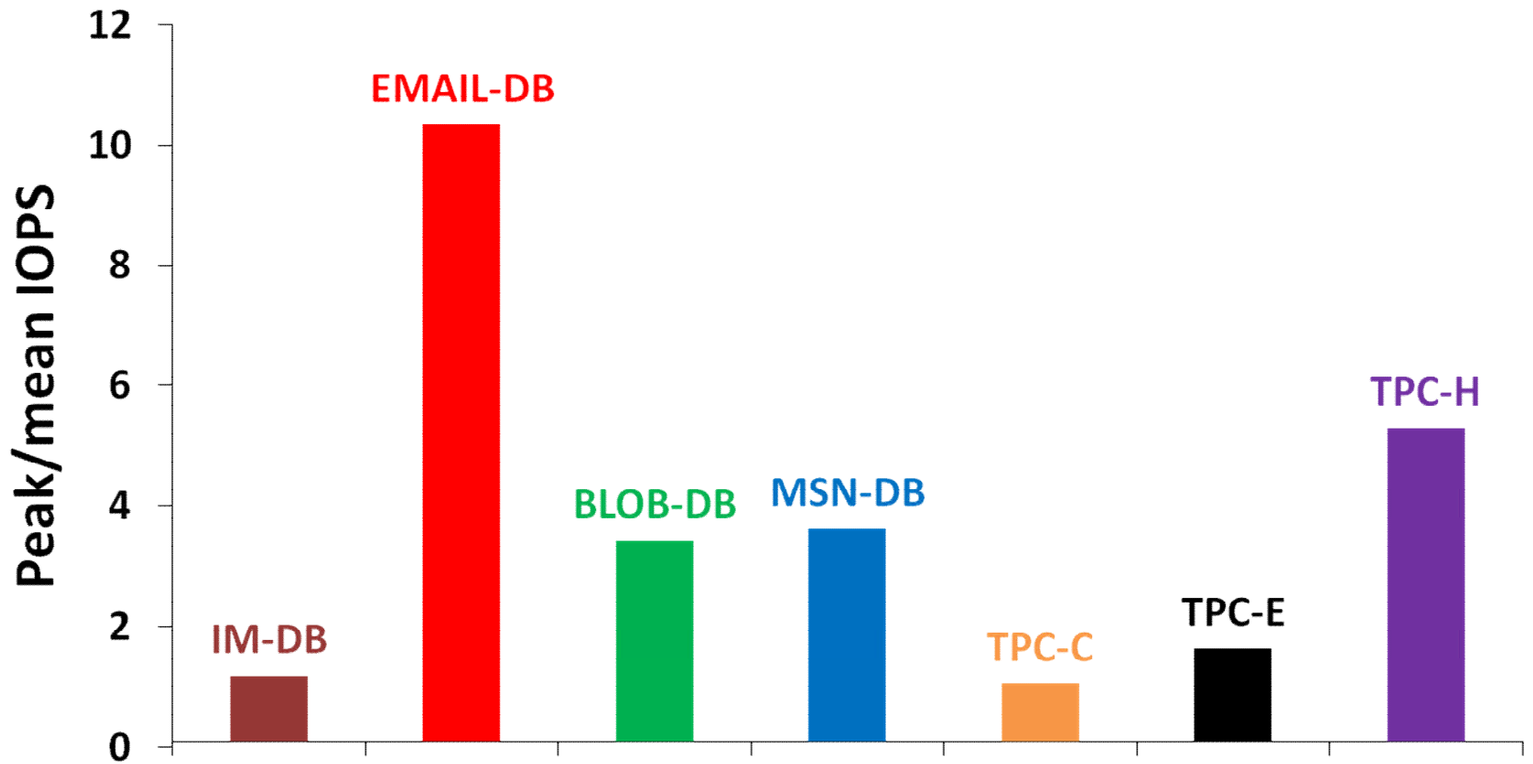
IOPS/GB (peak IOPS)



IOPS/GB (log scale)



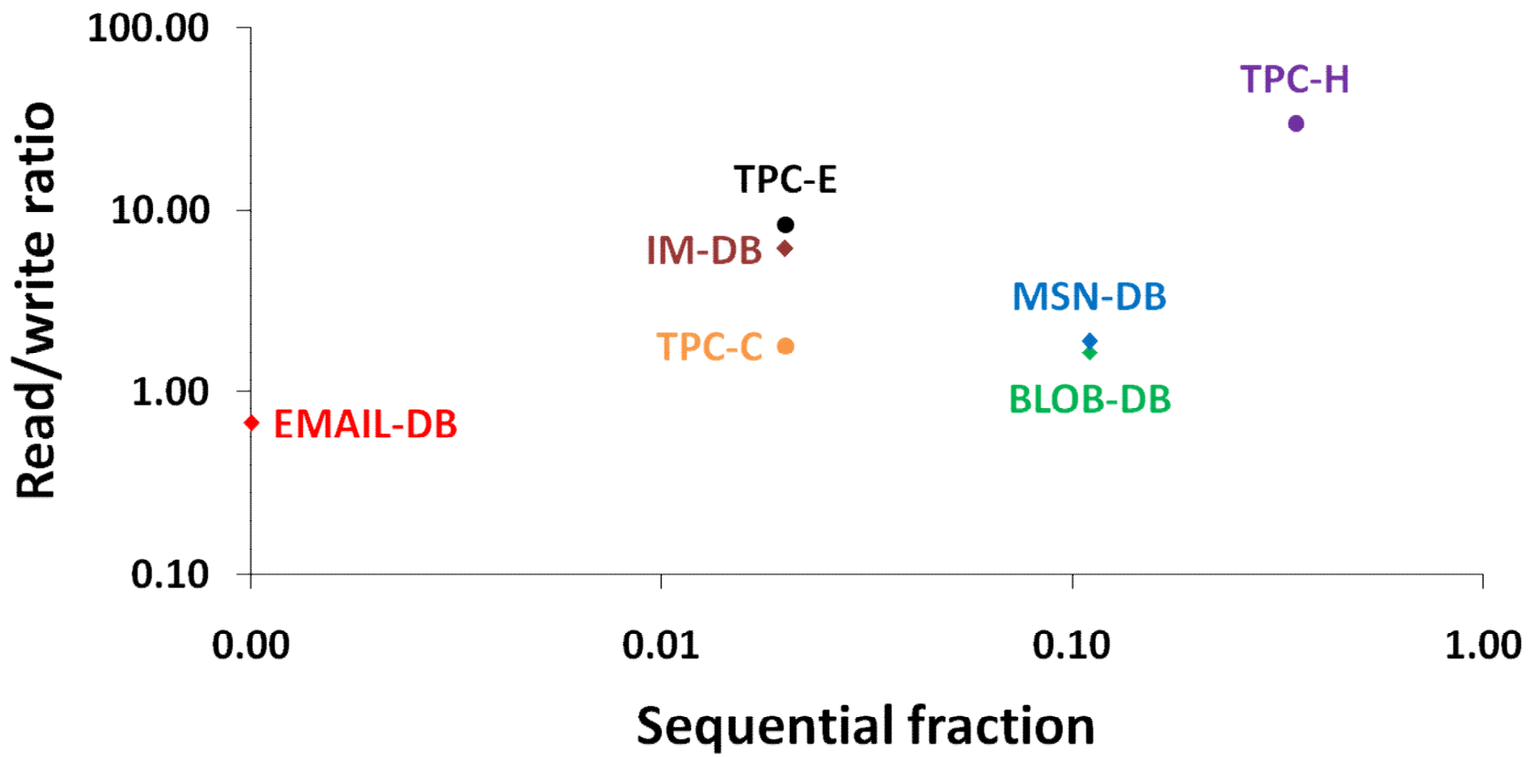
Peak-to-mean load ratios



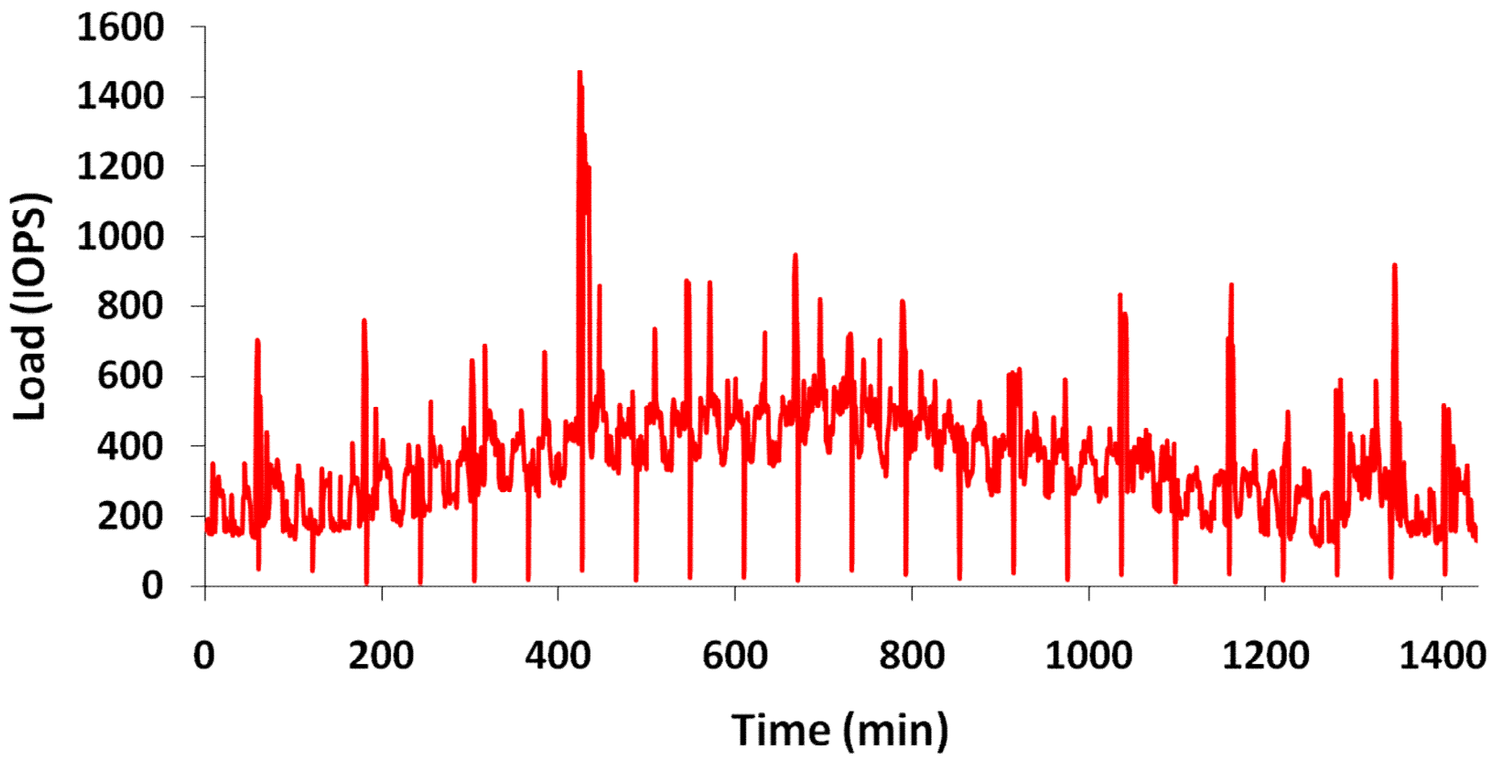
Online workloads have ...

- much lower IOPS/GB than TPC
  - Even when considering peak IOPS
  - Except IM-DB: roughly same as TPC-C
- higher peak/mean ratios than TPC-C,E
  - Except IM-DB
  - TPC-H comparable to BLOB-DB, MSN-DB
    - But for different reasons (TPC-H has phases)
  - EMAIL-DB has very high peak/mean ratio

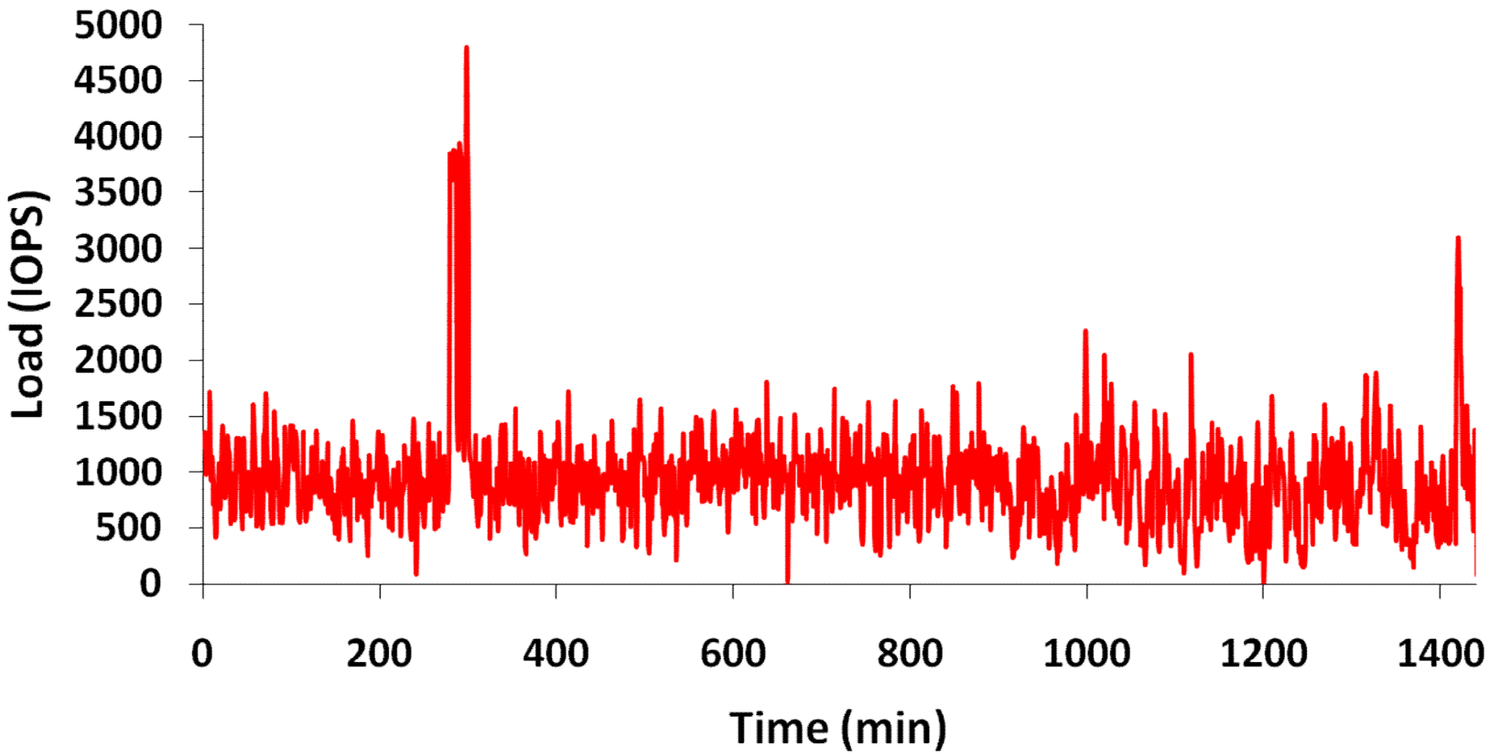
R/w ratio and sequentiality



Time variation: BLOB-DB



Time variation (MSN-DB)



Summary

- R/w ratio, sequentiality vary hugely
  - Some workloads close to TPC benchmarks
    - But differ on other metrics (like IOPS/GB)
- Online workloads have time variation
  - Periodic (diurnal, hourly)
  - Noise (high-frequency variation)
  - Load spikes
- TPC benchmarks do not have this notion

Outline

- Motivation
- Online workload analysis
- Conclusion

Analysis summary

- Online workloads vary widely
  - Differ from TPC benchmarks and each other
  - IM-DB is the most “TPC-like”
    - Sometimes like TPC-C, sometimes like TCP-E
    - Still not a great match
- Low IOPS/GB ratio even at peak
- High peak-to-mean ratios
- Time variation in load

How do we measure perf?

- Current benchmarks not representative
  - For these workloads
- Devise new benchmarks?
  - Workloads also vary widely among each other
  - Would need one benchmark per service
- Measure using I/O trace replay?
  - Effective, but has its limitations

Trace replay advantages

- Captures properties of real workload
- We used traces to drive many evaluations
  - Disk spin-down → depends on idle times
  - Burst absorbtion → depends on burstiness
  - SSD v disk → depends on IOPS/GB
- Benchmarks would not have worked here

Trace replay limitations

- Trace replay captures real workload
- But has limitations vis-a-vis benchmarks
  - I/O trace replay only measures disk resources
  - “Open loop” problems
  - Hard to scale (up or down)
  - Not standardized for comparison of systems

Future directions

- End-to-end tracing
  - All resources (CPU, network, user think time)
- Parameterize the benchmarks
  - Set IOPS/GB, r/w ratio, ... to measured values
  - Need to allow orders of magnitude variation
  - Need to model/express “time variation”
- Trace repository a la IOTTA
  - Maybe TPC can help set this up?