

DBMS Data Loading: An Analysis on Modern Hardware

Adam Dzedzic, **Manos Karpathiotakis***,
Ioannis Alagiannis, Raja Appuswamy,
Anastasia Ailamaki

Data loading: A necessary evil

✘ Volume => Expensive
40 zettabytes by 2020*



✘ Velocity => Continuous
Fresh data = Interesting data[†]

✓ Top query performance

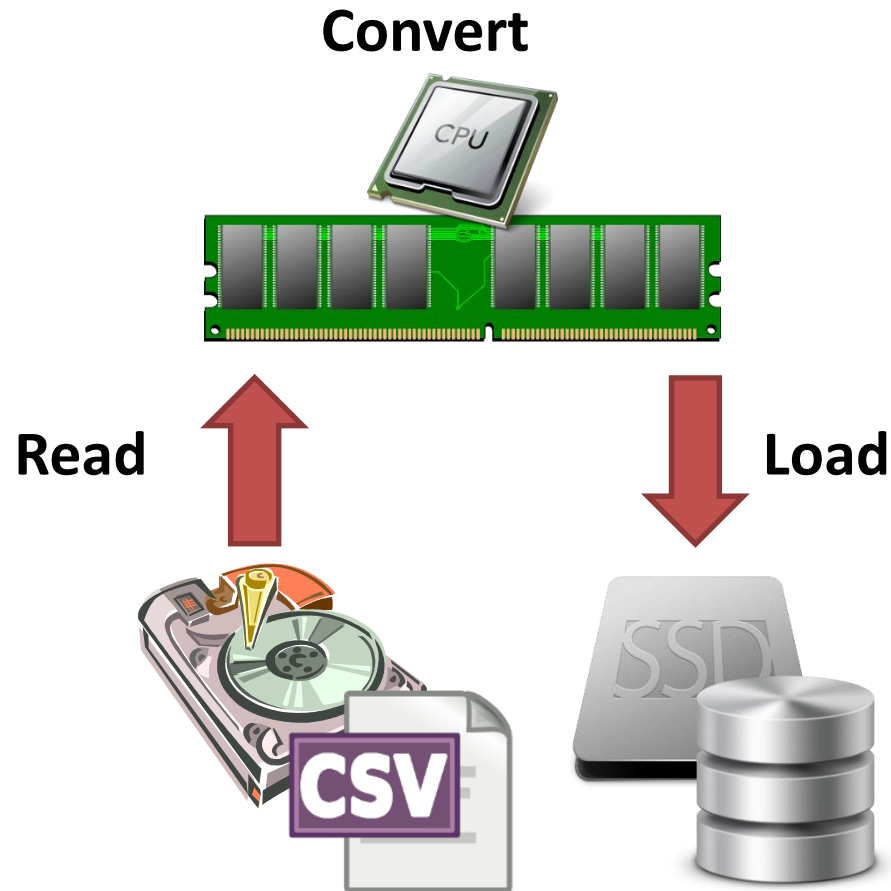
✓ ACID guarantees

* [IDC12]

† Abad [IISWC12]

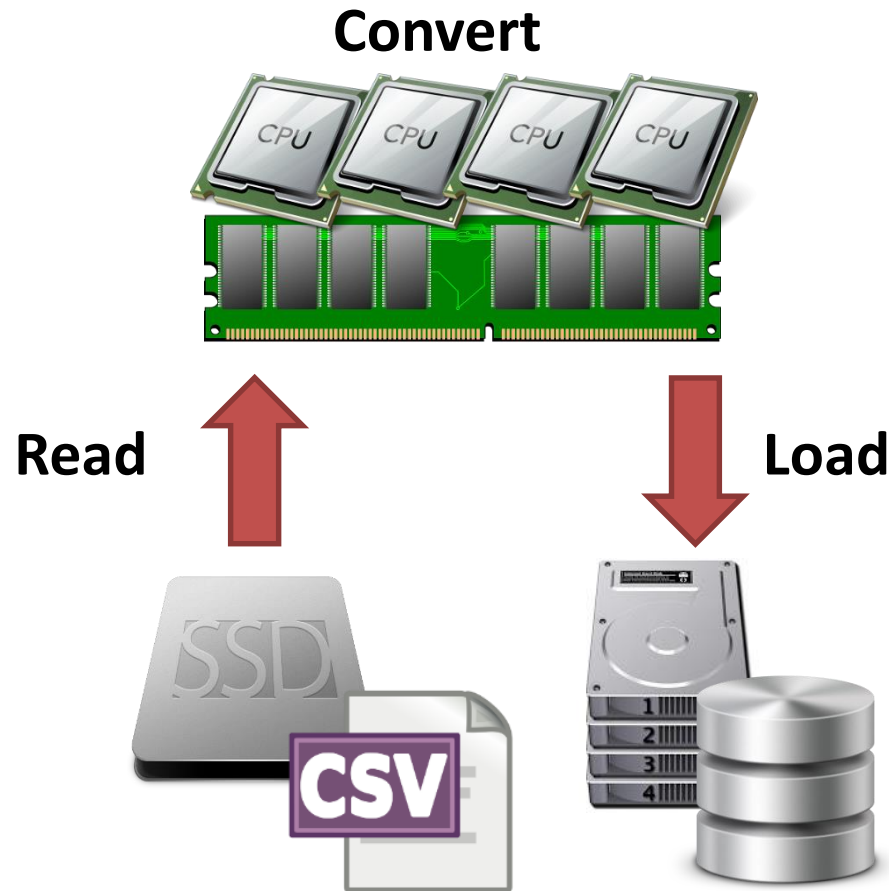
Data loading is a persistent analysis bottleneck

Loading a DBMS



How does hardware affect loading?

Loading a DBMS



How does hardware affect loading?

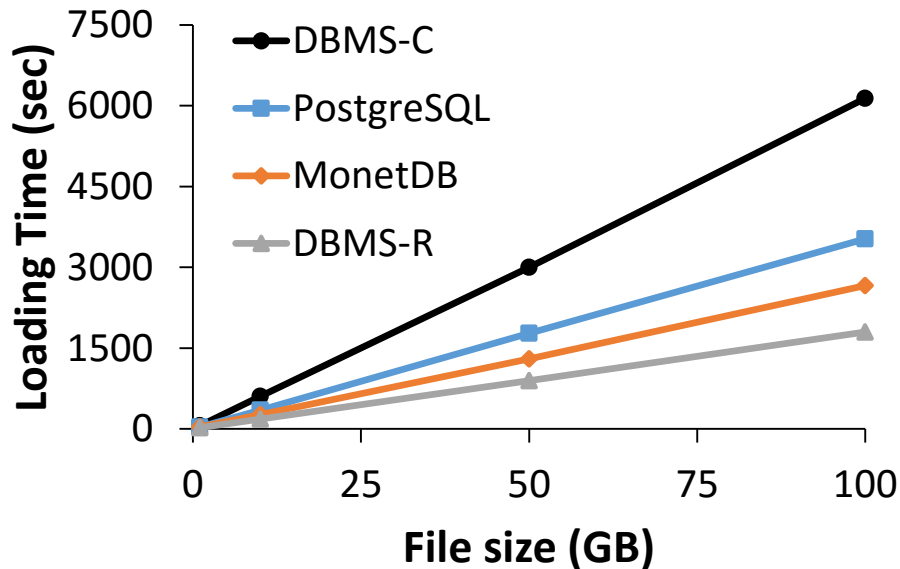
Experimental setup

- Hardware
 - Dual socket 8 cores Intel(R) Xeon(R) CPU E5-2640
 - 64 GB RAM
 - HDD: 4 x 500 GB 7.5k RPM SATA disks
 - SSD: 3 x 200GB SSD disks
 - DAS: 24 x 500 GB 7.5k RPM SATA disks
- Software
 - PostgreSQL, DBMS-R
 - MonetDB, DBMS-C
 - PostgreSQL parallel external loader (“**PCOPY**”)
- Benchmarks & Real-world Datasets

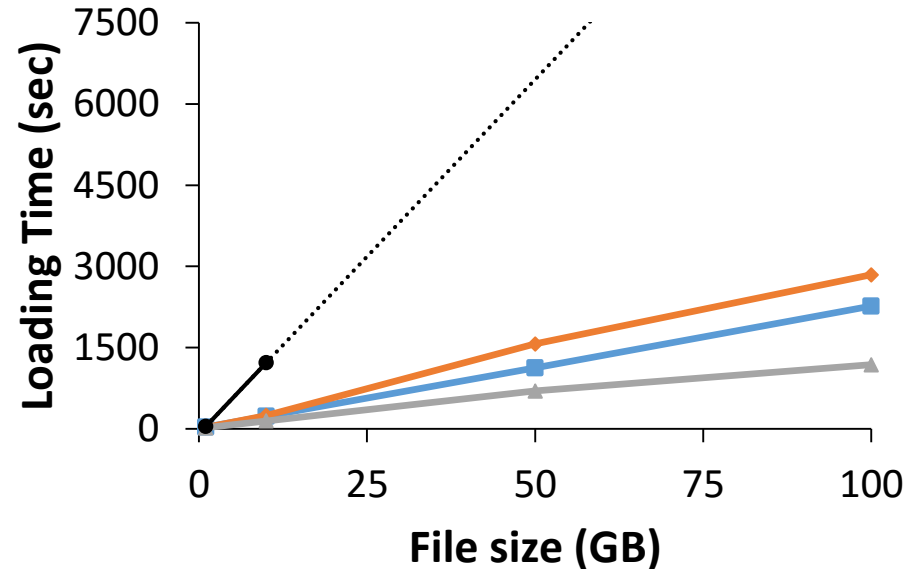
Single-threaded data loading

[Input storage: HDD
Destination storage: DAS]

TPC-H Loading Time



Symantec Loading Time

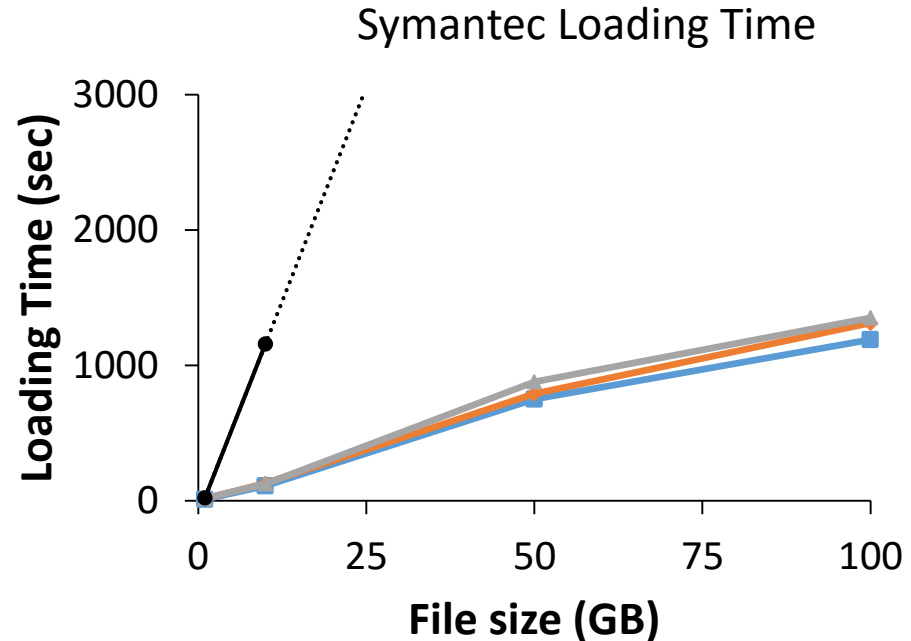
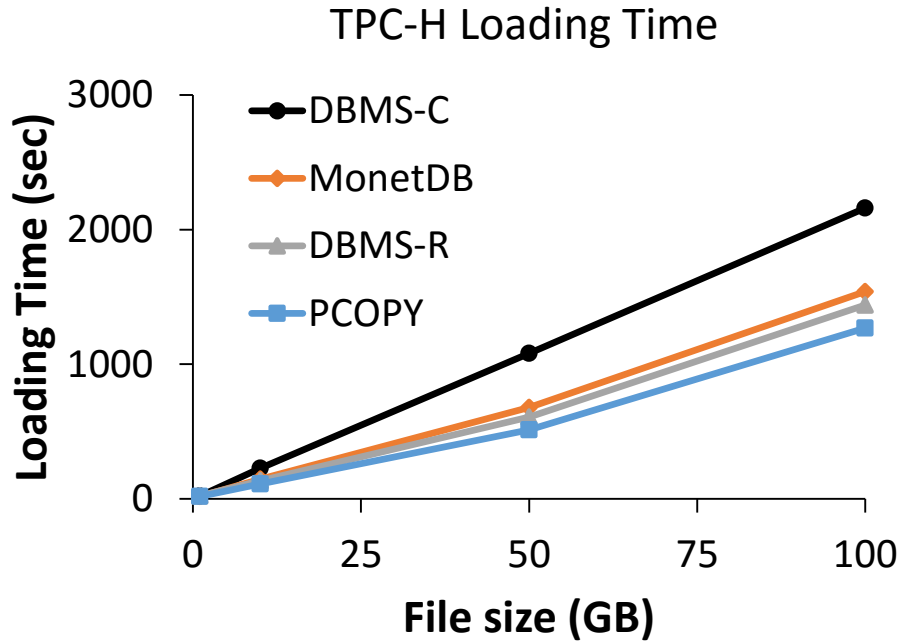


Dataset characteristics matter

Effect of compression

Parallel data loading

Input storage: HDD - Destination storage: DAS
16 threads

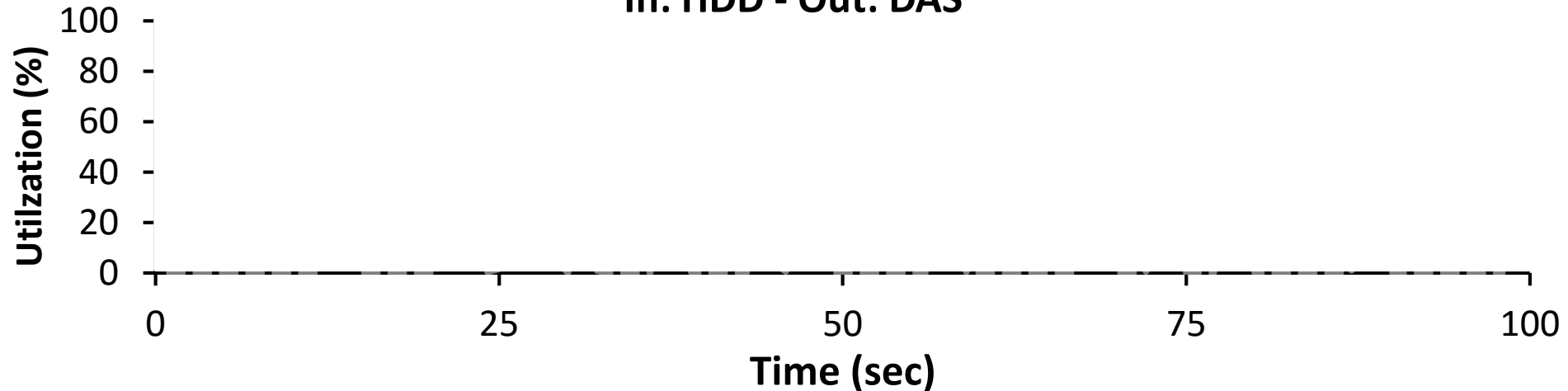


| Speedup – 16 threads | DBMS-R | PCOPY | MonetDB | DBMS-C |
|----------------------|-------------|-------|---------|--------|
| TPC-H 100GB | 1.25 | 2.77 | 1.72 | 2.84 |
| Symantec 100GB | 0.87 | 1.9 | 2.1 | - |

Sublinear speedup for 16 threads

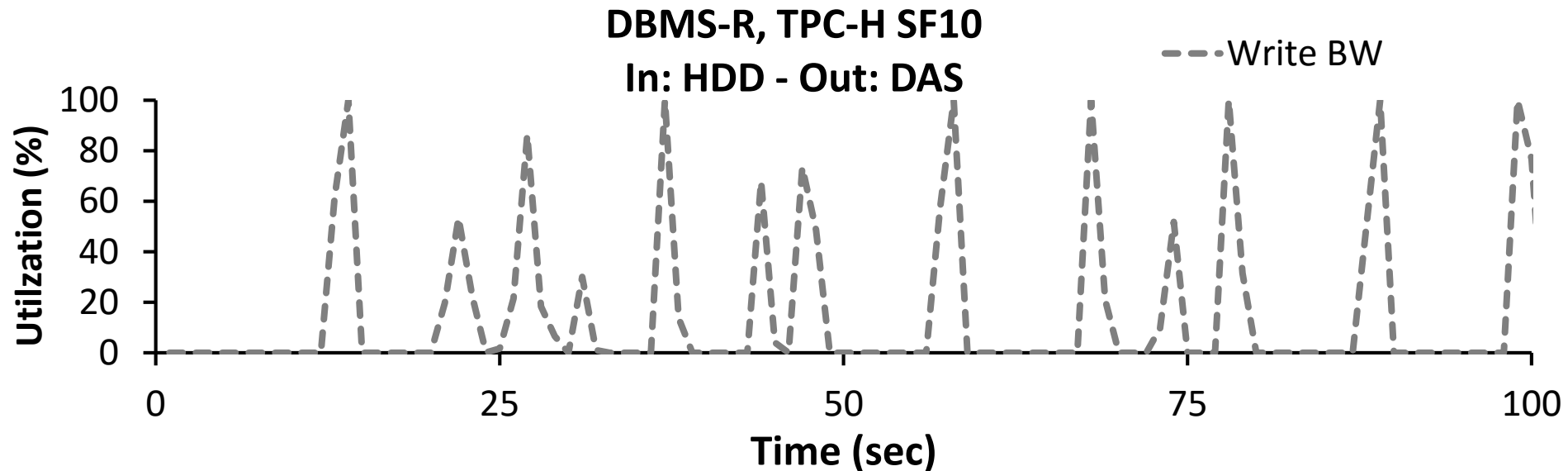
Resource Utilization

DBMS-R, TPC-H SF10
In: HDD - Out: DAS



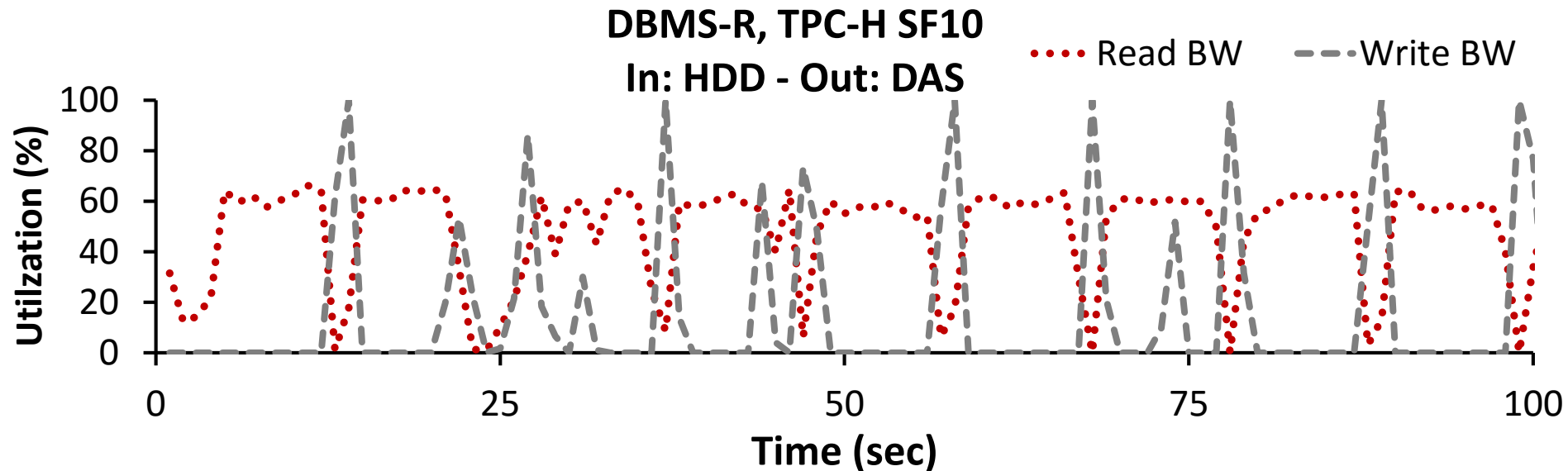
Unable to saturate resources

Resource Utilization



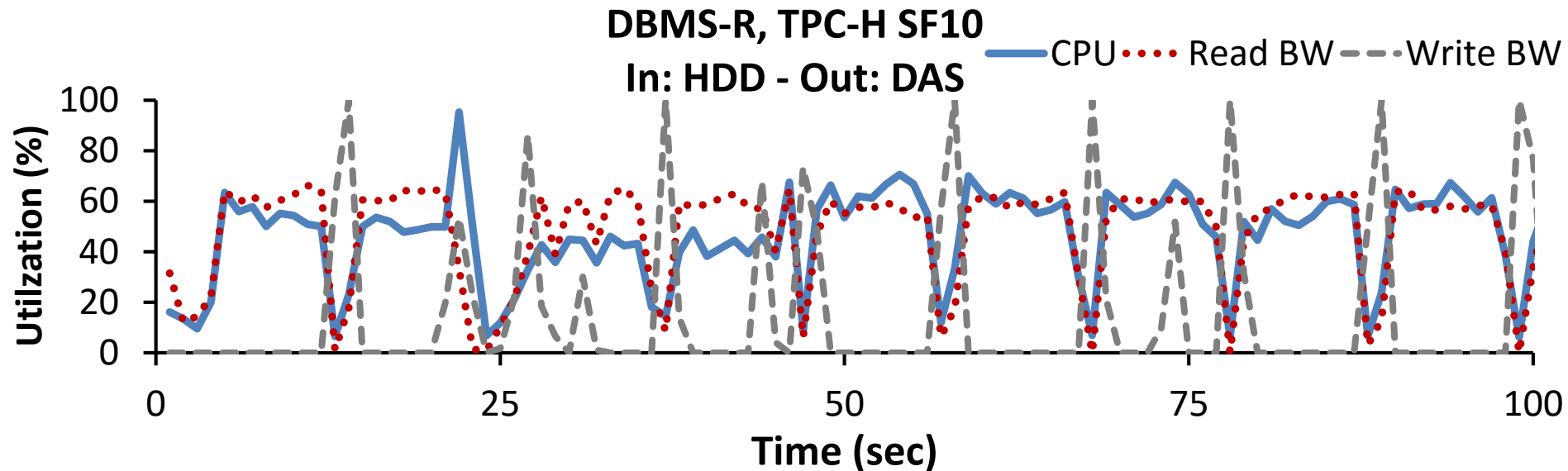
Unable to saturate resources

Resource Utilization



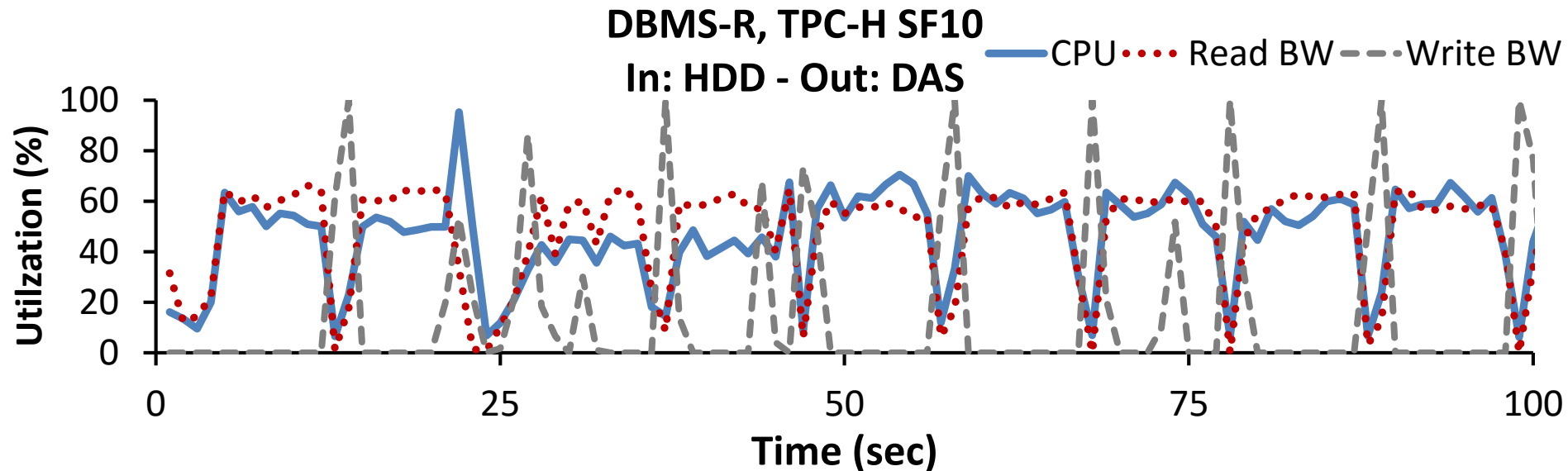
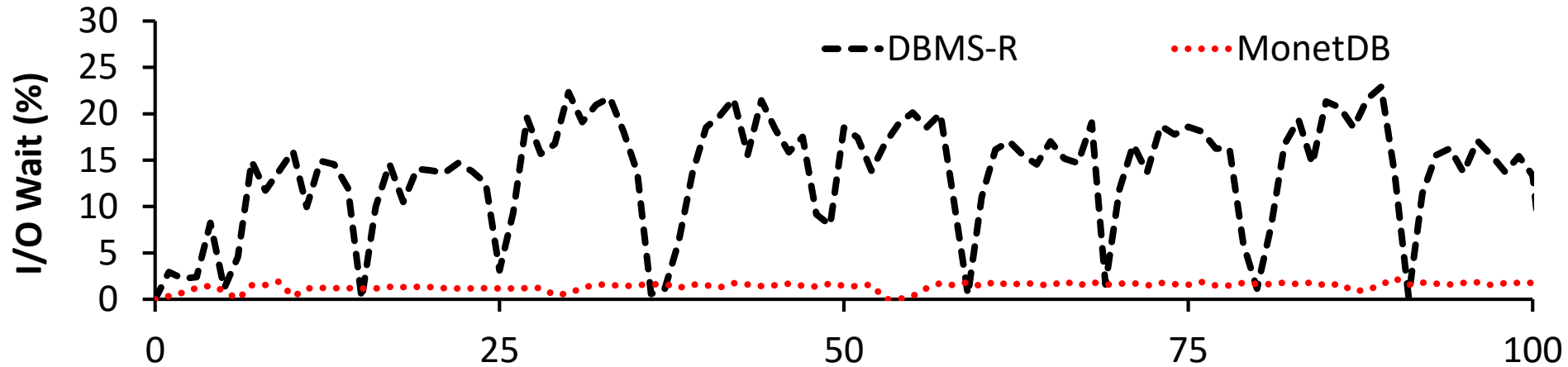
Unable to saturate resources

Resource Utilization



Unable to saturate resources

Resource Utilization

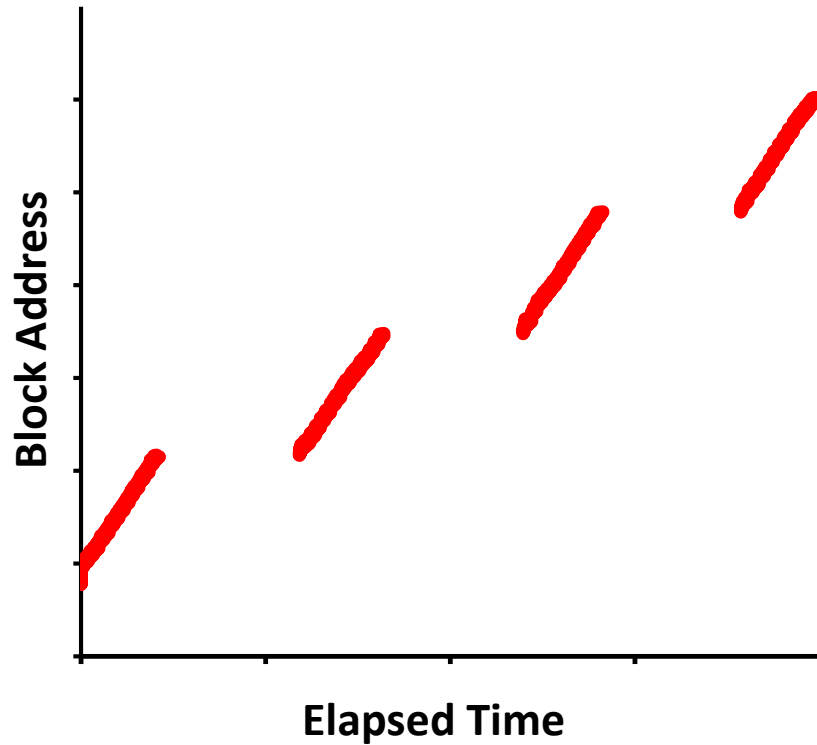


Unable to saturate resources

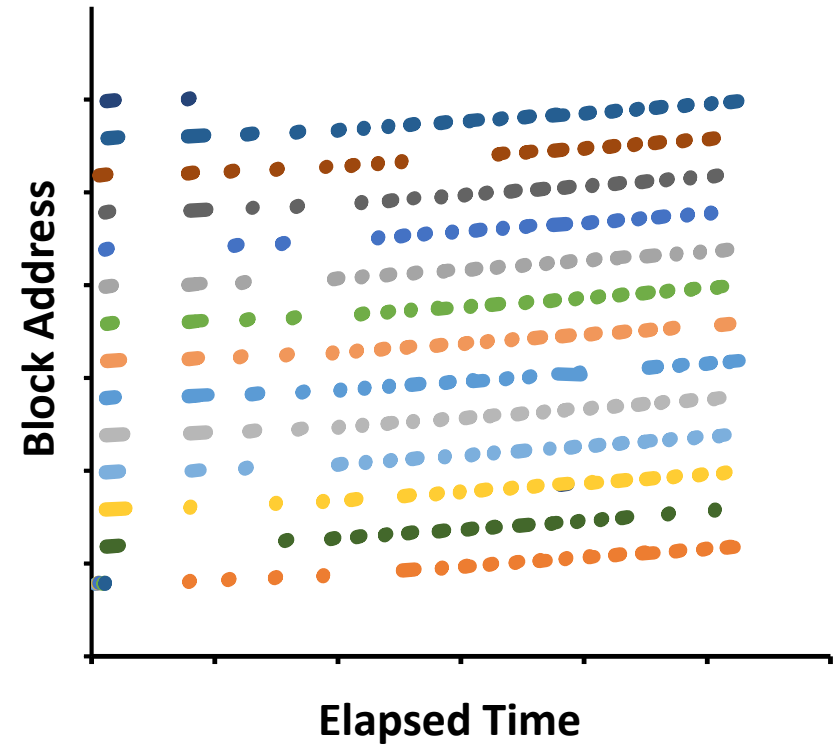
Read patterns

[TPC-H SF10
Input storage: HDD
Destination storage: DAS]

MonetDB



DBMS-R



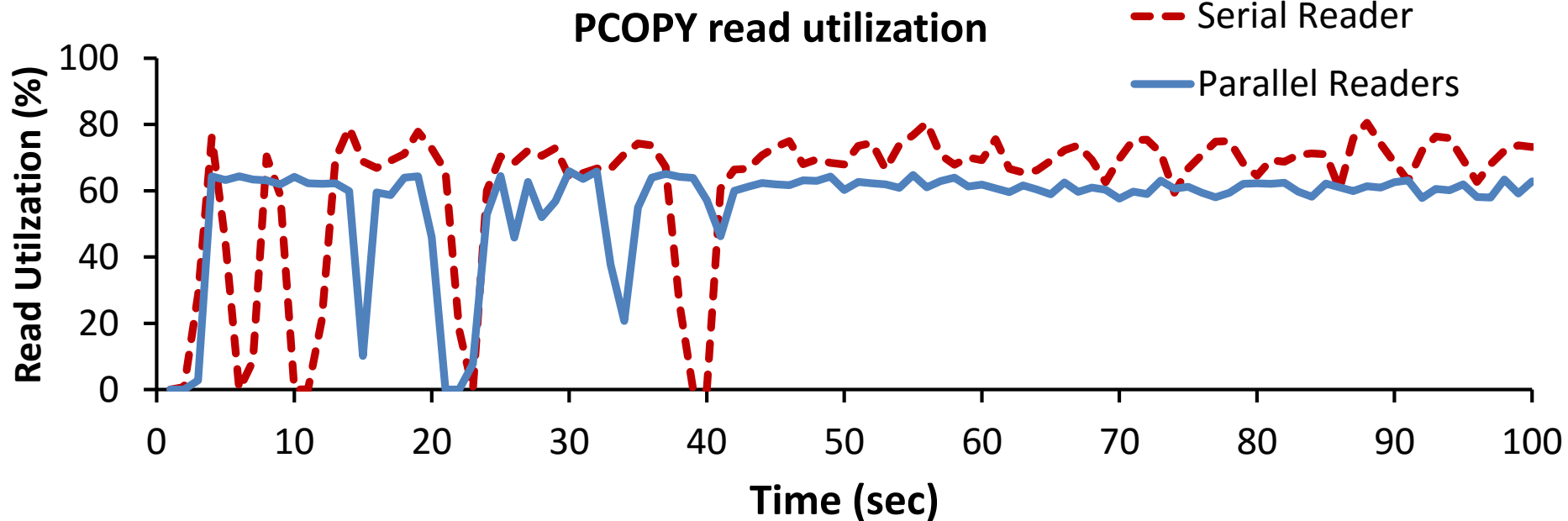
Random I/O causes underutilization

Serial reader vs. Parallel readers

[TPC-H SF10

Input storage: HDD

Destination storage: DAS]



Serial reader improves read utilization

readers depends on input device speed

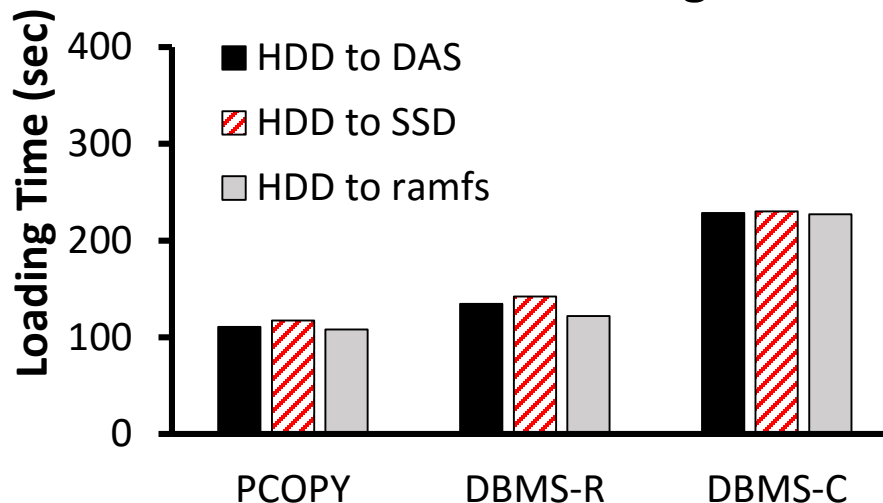
Impact of storage

Impact of storage

[TPC-H SF10]

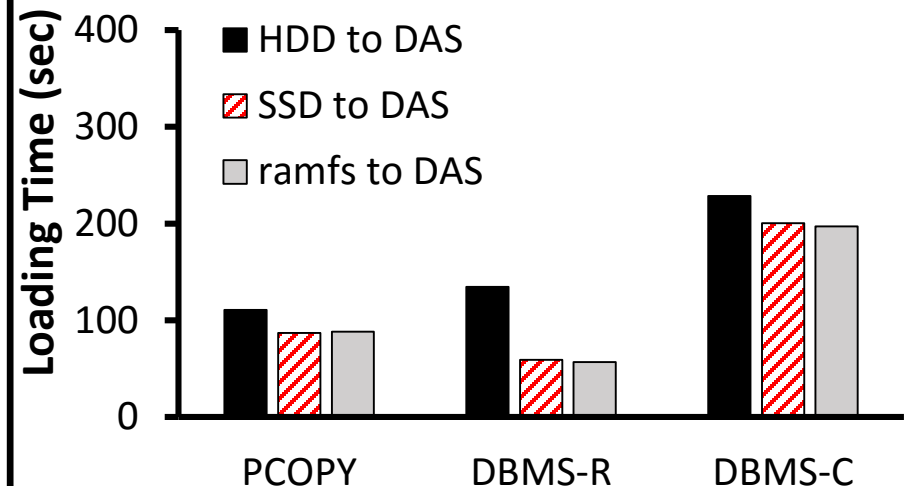
Slow input storage

HDD Source Storage



Varying input storage

DAS Destination Storage



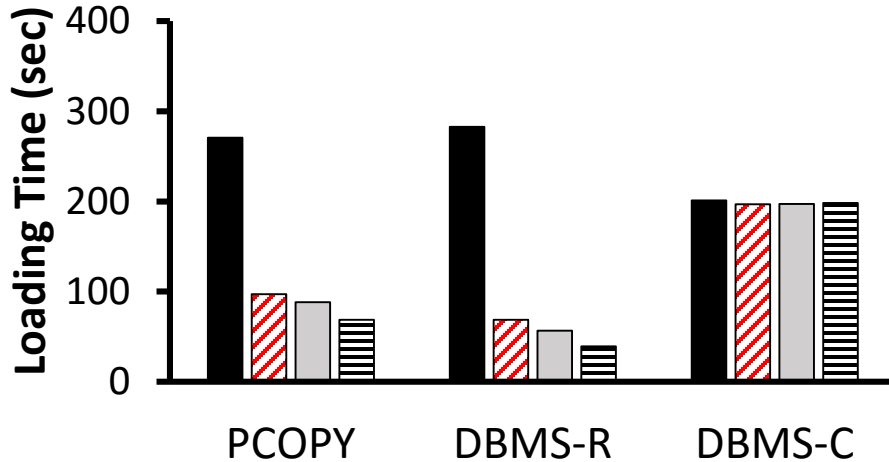
Slow source storage bottlenecks all systems
Write bottleneck when source storage is fast

Best-case storage scenario

[TPC-H SF10]

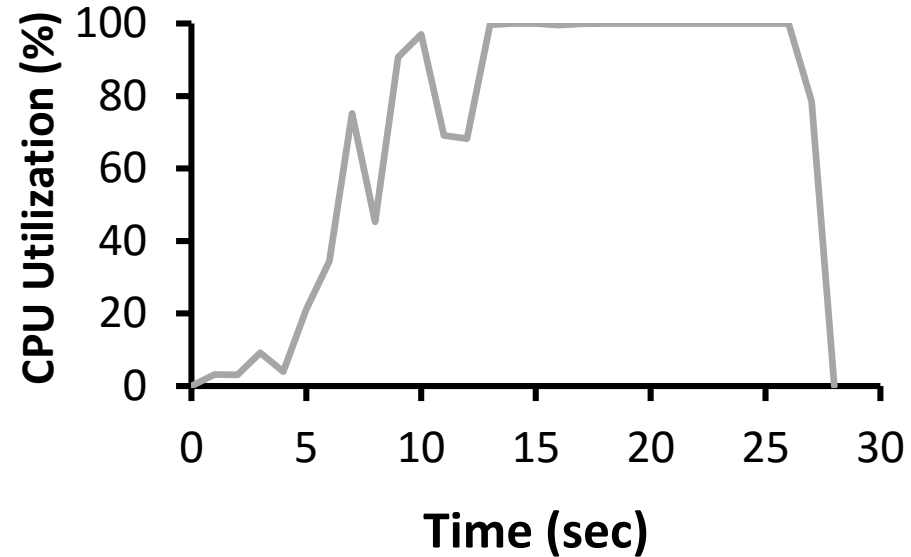
ramfs Source Storage

- ramfs to HDD
- ▨ ramfs to DAS
- ▤ ramfs to SSD
- ▧ ramfs to ramfs



DBMS-R

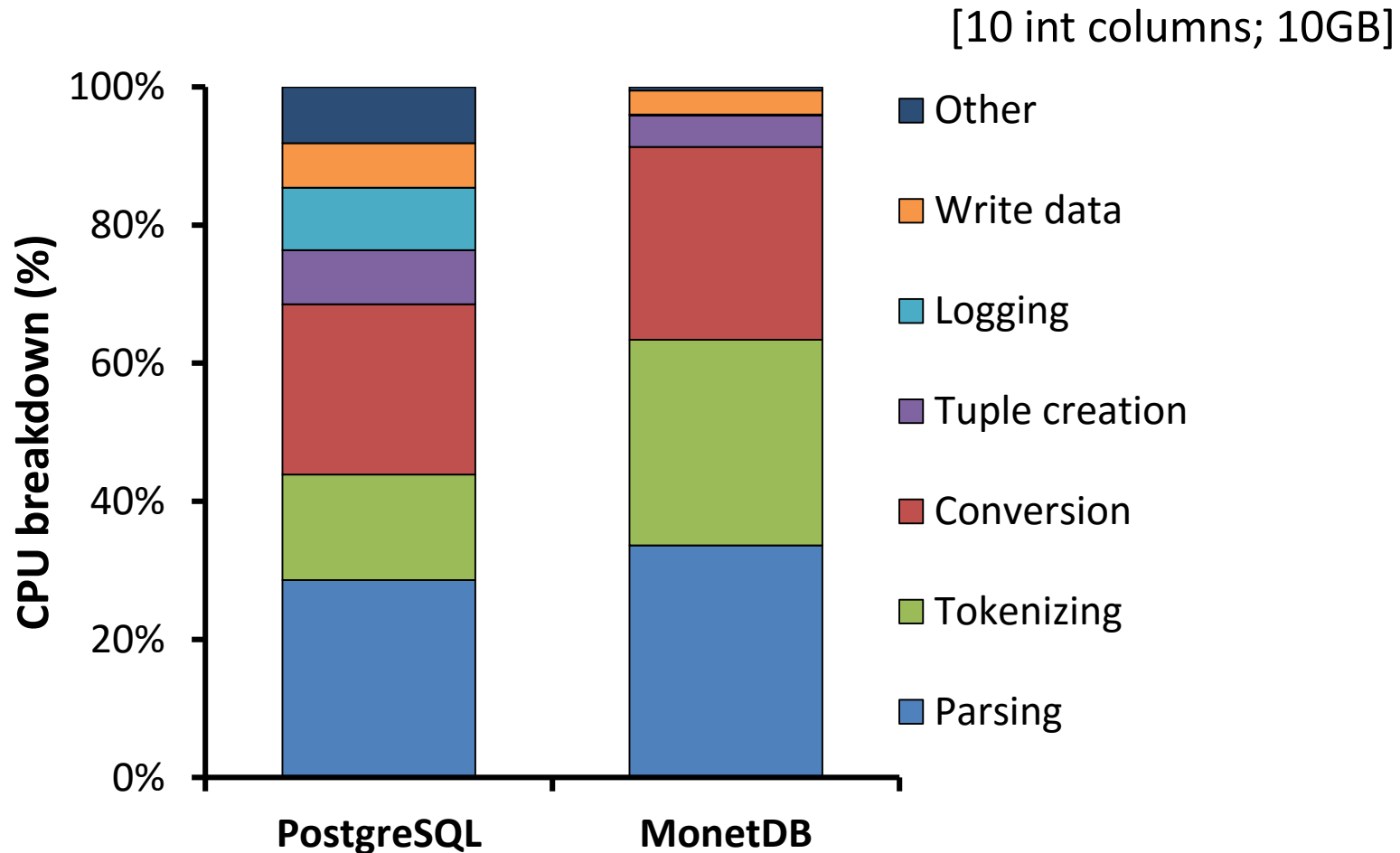
In: ramfs – Out: ramfs



Device Bandwidth: 12.8 GB/sec
Read Rate: 250 MB/sec

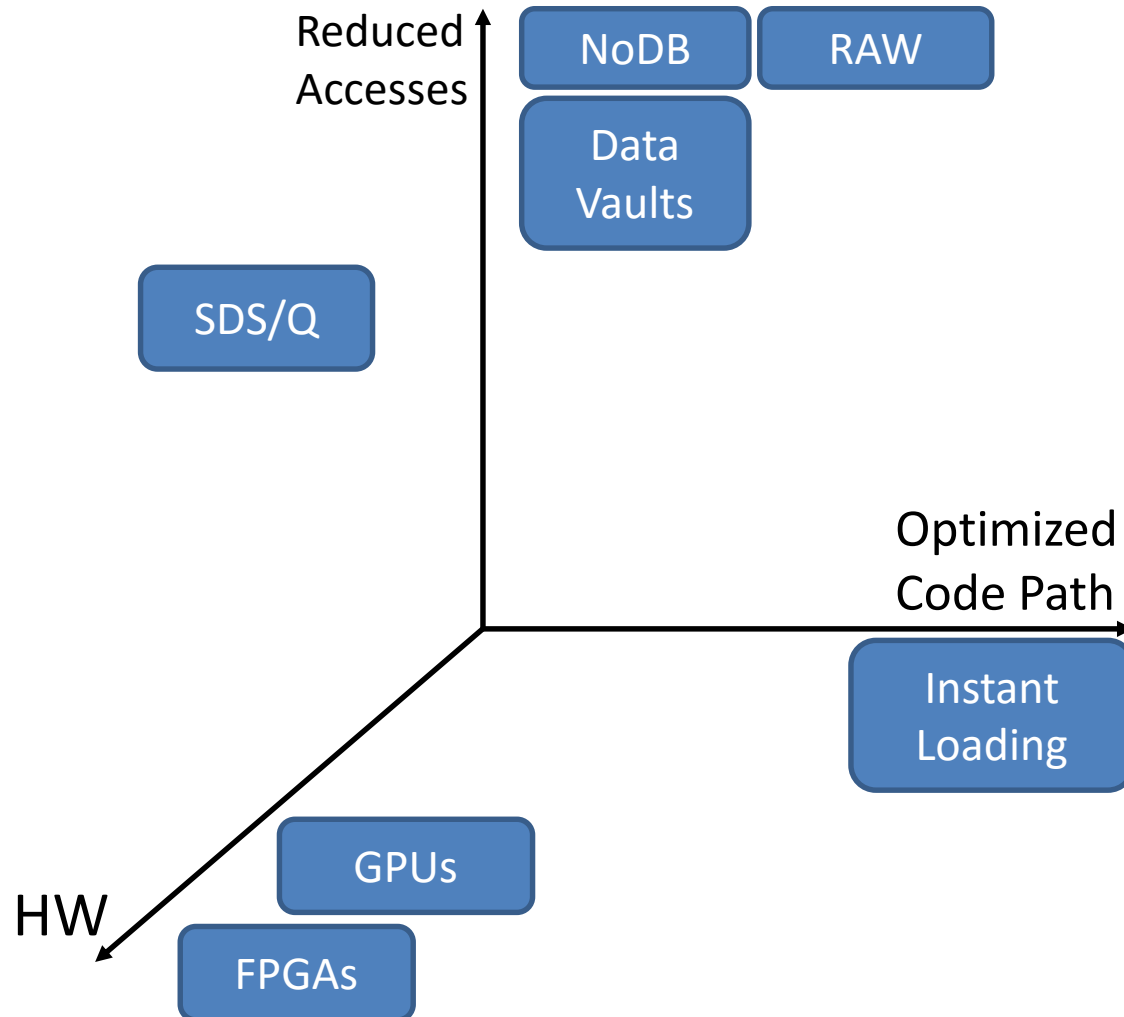
100% CPU utilization, yet B/W still underutilized 12

Data loading: Where does time go?



Parsing, conversion, tokenization hotspots

Reducing data loading overheads



Bulk loading on modern hardware

- General case: Resource under-utilization
- Slow destination storage matters
- Complex code paths bound max speed

Bulk loading on modern hardware

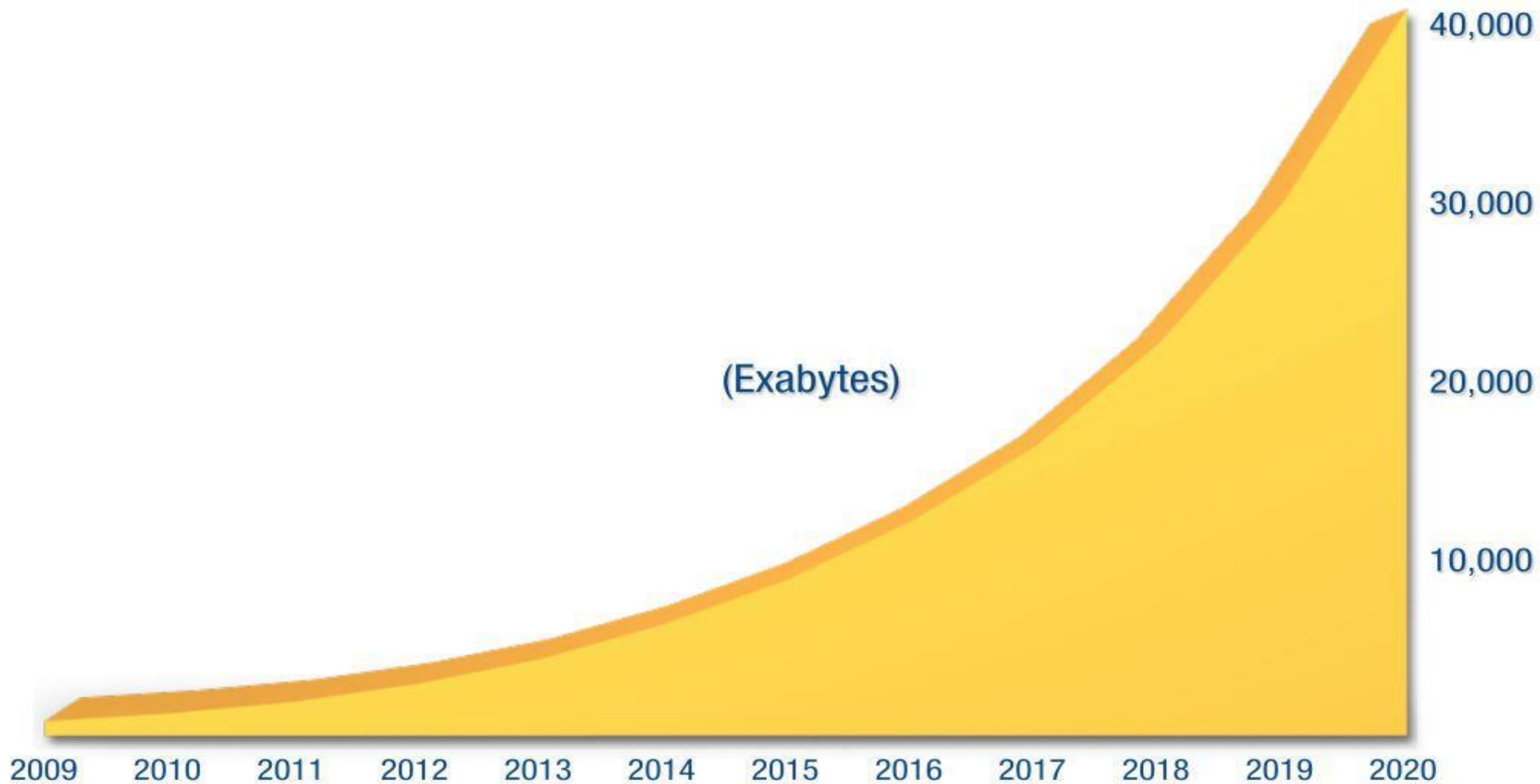
- General case: Resource under-utilization
- Slow destination storage matters
- Complex code paths bound max speed

Thank You!
Questions?

Backup Slides

50x data growth from 2010 to 2020

[IDC2012]



Can DBMS keep up with data growth?

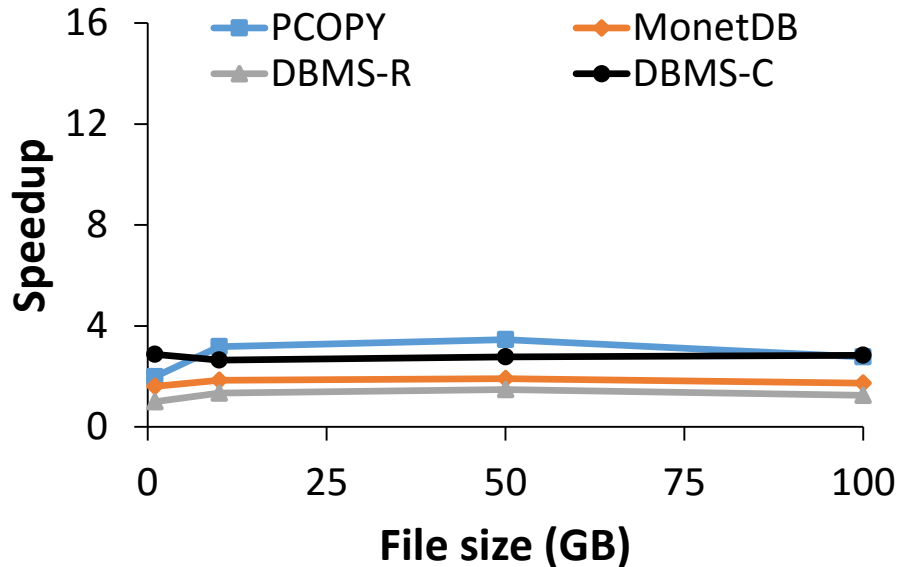
Storage Characteristics

| Name | Capacity | Configuration | Read Speed | Write Speed | RPM |
|------|----------|-------------------|------------|-------------|------|
| HDD | 2TB | 4 x HDD (RAID-0) | 170 MB/s | 160 MB/s | 7.5K |
| DAS | 12TB | 24 x HDD (RAID-0) | 1100 MB/s | 330 MB/s | 7.5K |
| SSD | 600GB | 3 x SSD (RAID-0) | 565 MB/s | 268 MB/s | n/a |

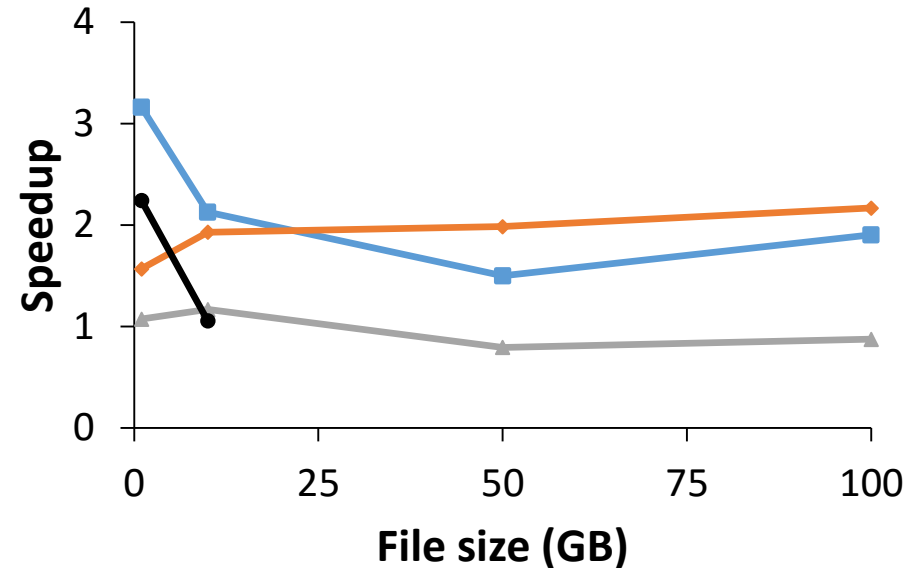
Parallel data loading – 16 threads

[Input storage: HDD
Destination storage: DAS]

TPC-H



Symantec

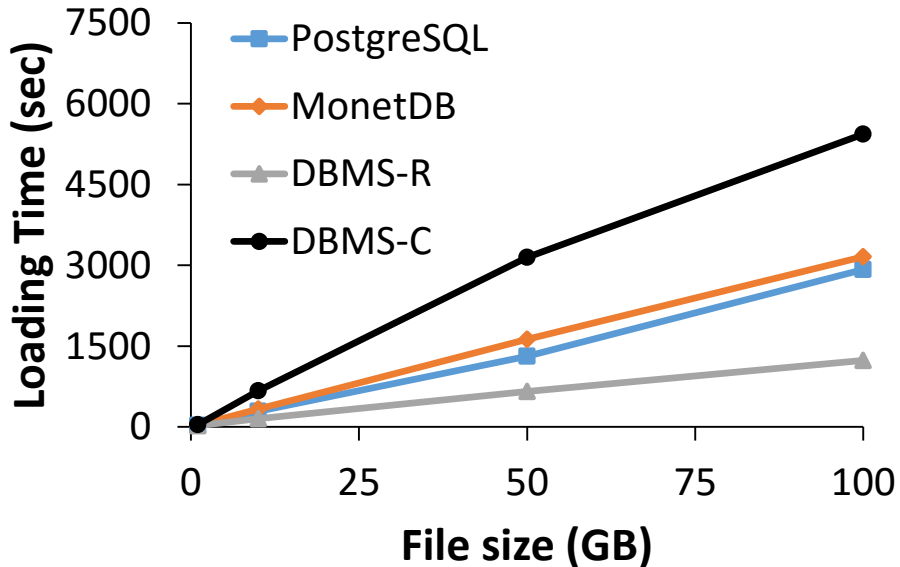


Sublinear speedup for 16x DoP

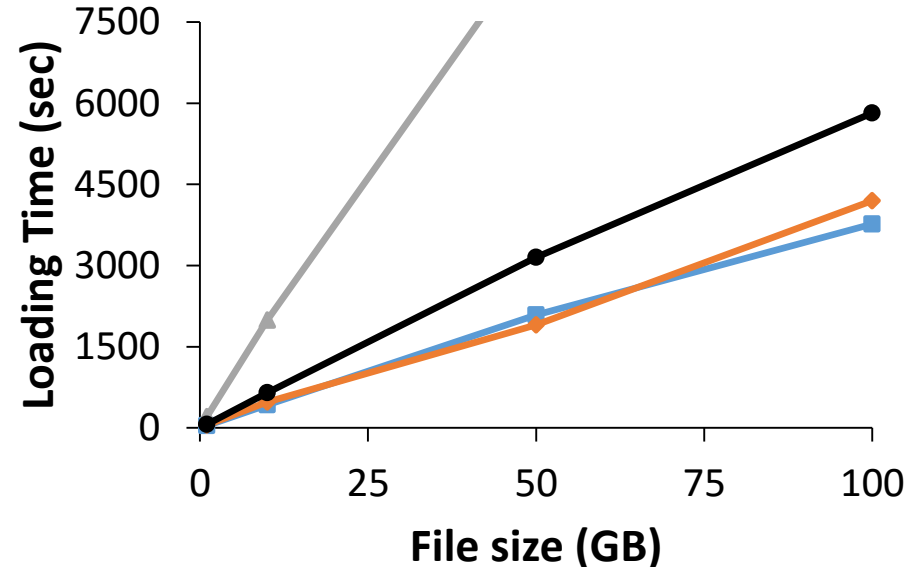
Single-threaded loading – Extra datasets

Input storage: HDD
Destination storage: DAS

TPC-C Loading Time



SDSS Loading Time

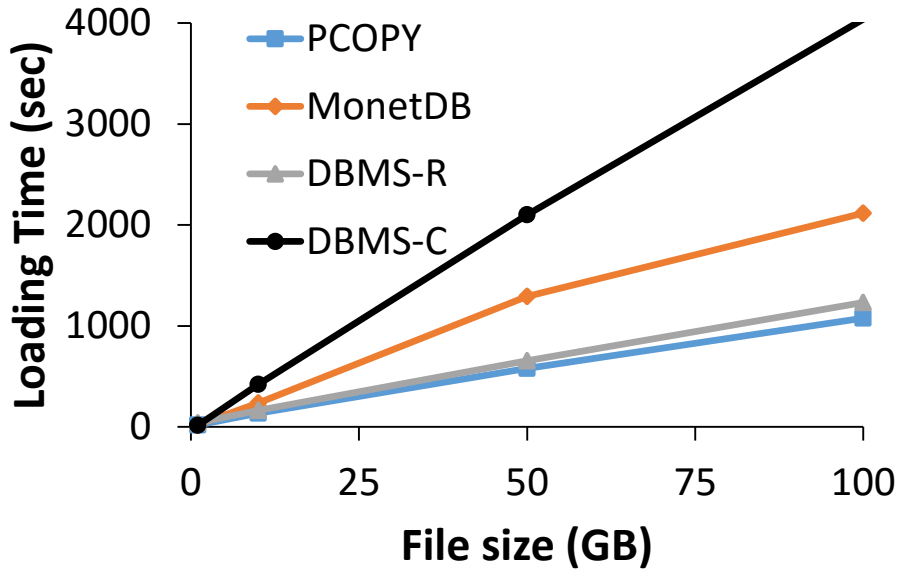


Column stores invest in compression

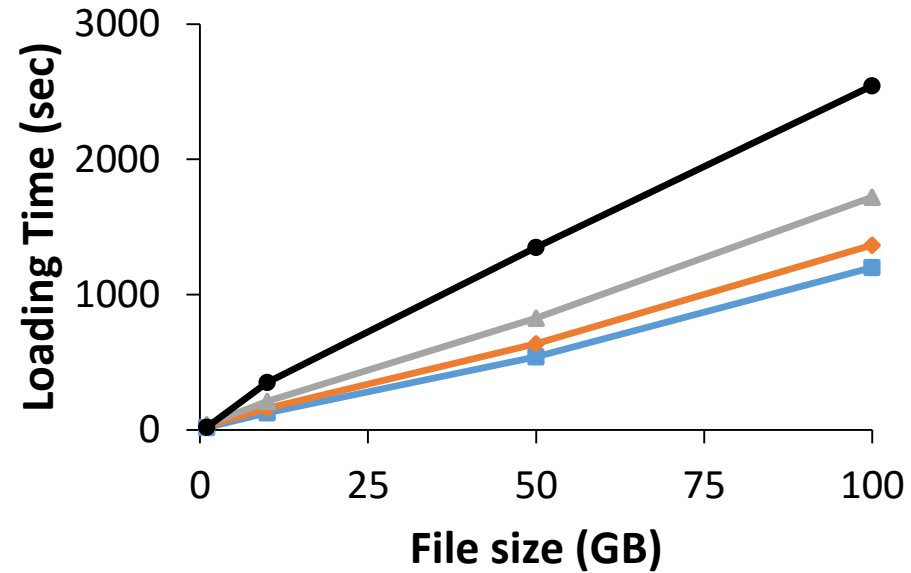
Parallel data loading – Extra datasets

Input storage: HDD
 Destination storage: DAS

TPC-C Loading Time



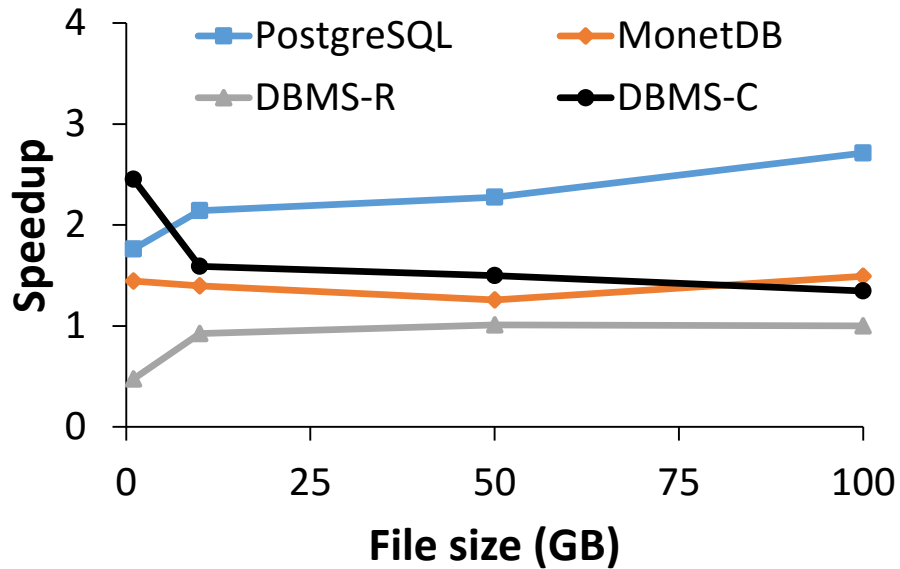
SDSS Loading Time



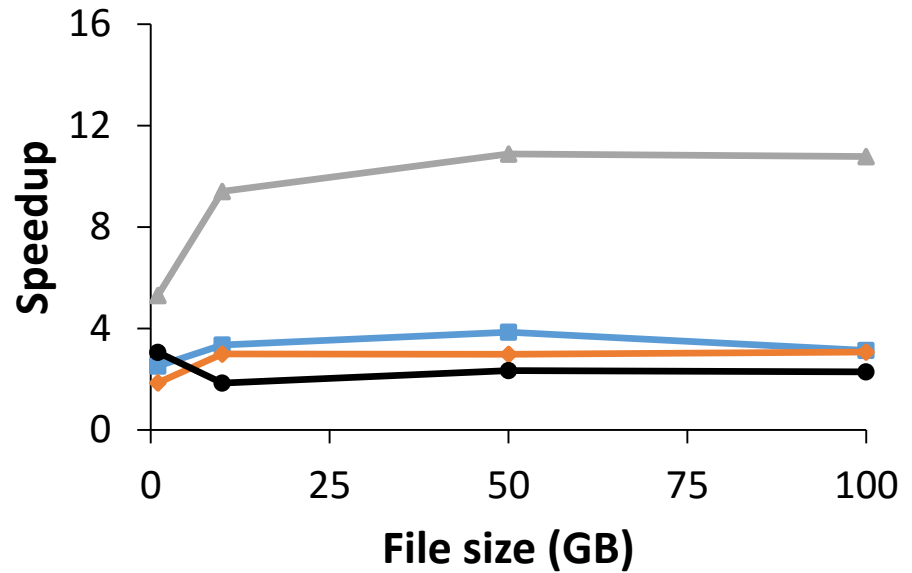
Parallel data loading – Extra datasets

Input storage: HDD
Destination storage: DAS

TPC-C



SDSS



The effect of compression

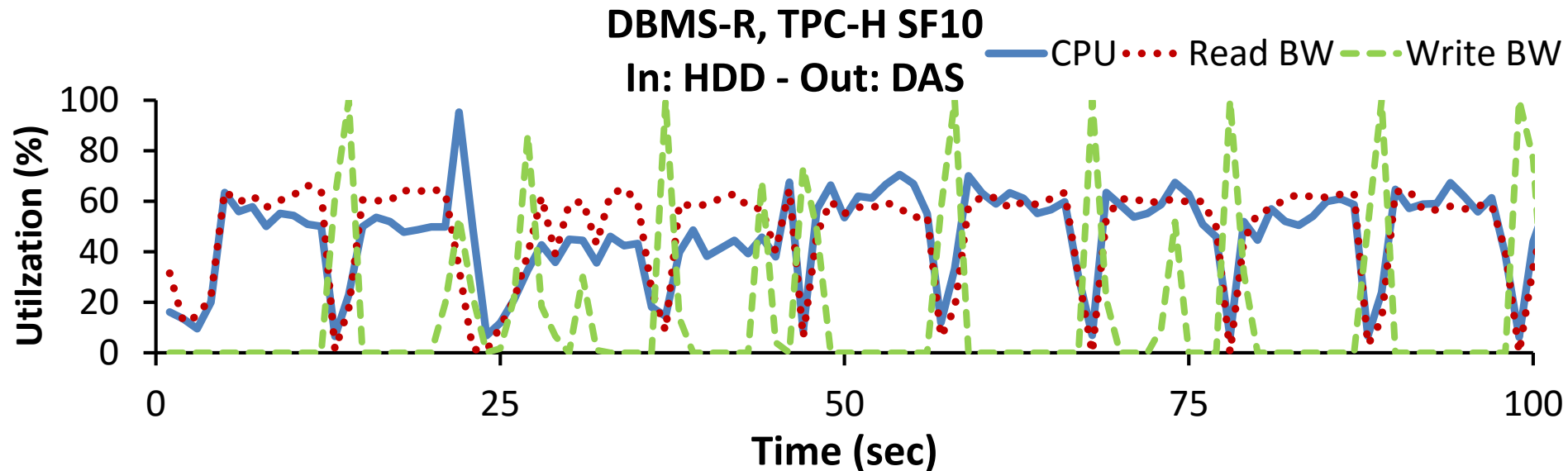
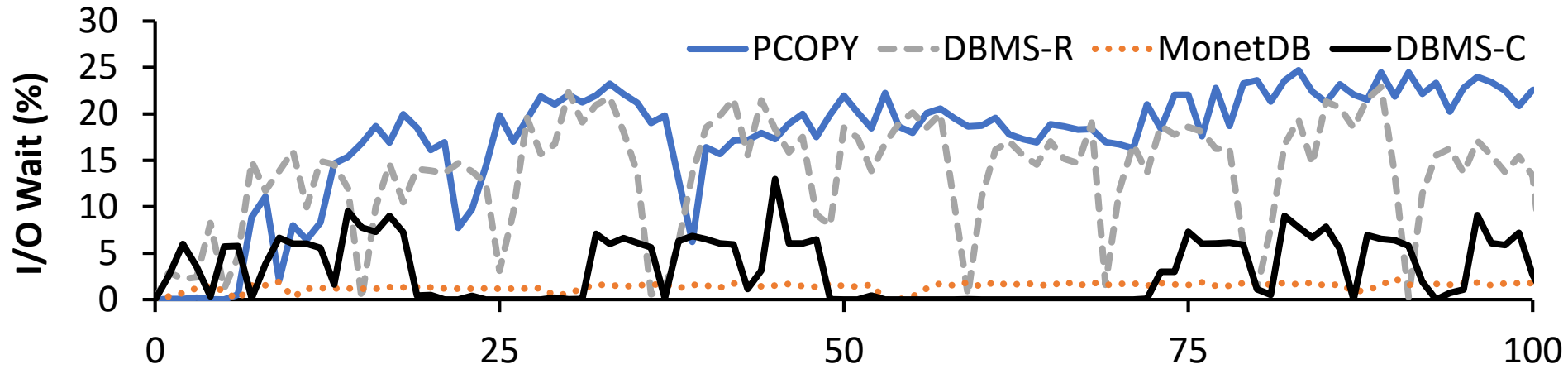
[10GB]

DB size / input file

| Name | TPC-H | TPC-C | SDSS | Symantec |
|------------|-------|-------|------|----------|
| DBMS-R | 1.5 | 1.3 | 1.5 | 1.5 |
| PostgreSQL | 1.4 | 1.4 | 1.4 | 1.1 |
| DBMS-C | 0.27 | 0.82 | 0.18 | 0.25 |
| MonetDB | 1.1 | 1.4 | 1.0 | 0.92 |

Column stores: Reduced footprint favors OLAP

Resource Utilization



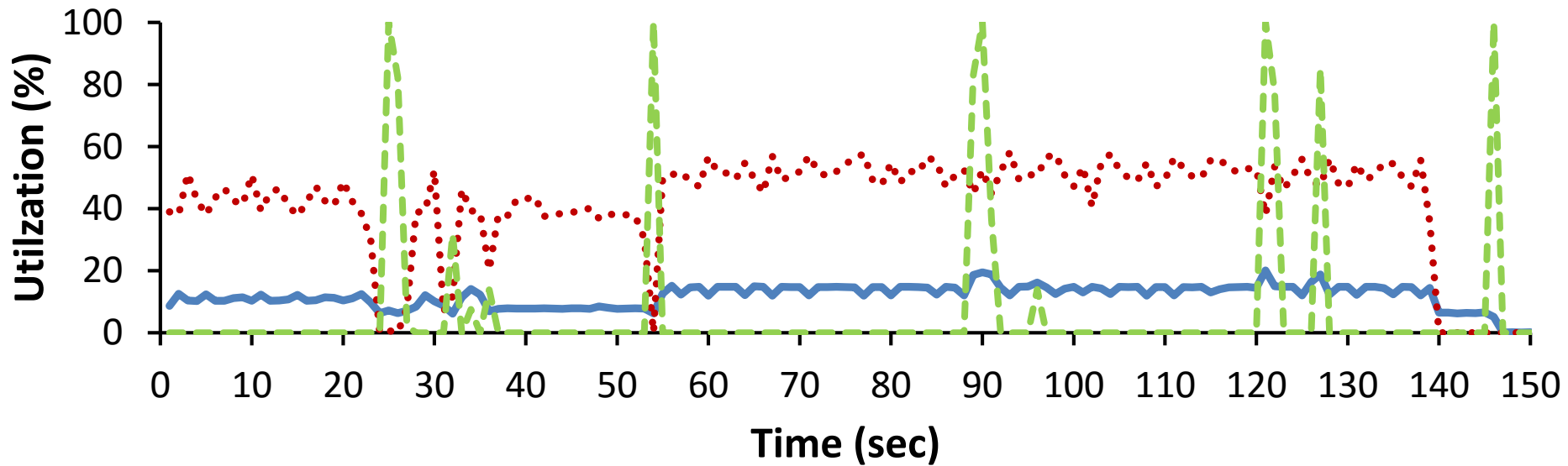
Unable to saturate resources

MonetDB utilization

[Data: TPCH – SF10

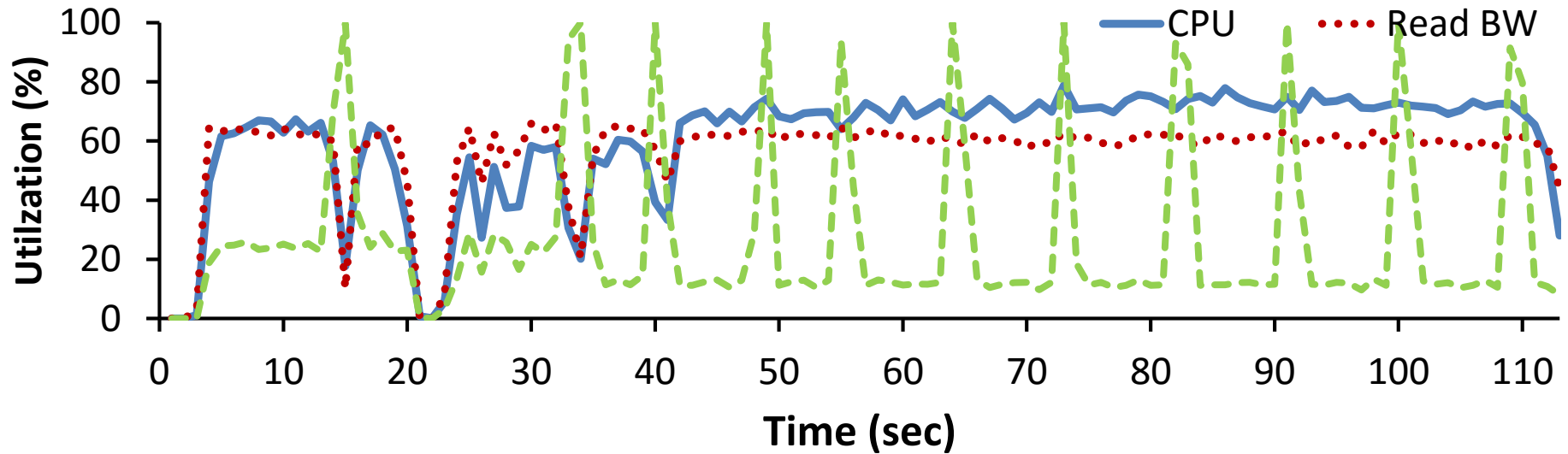
Input storage: HDD

Destination storage: DAS]



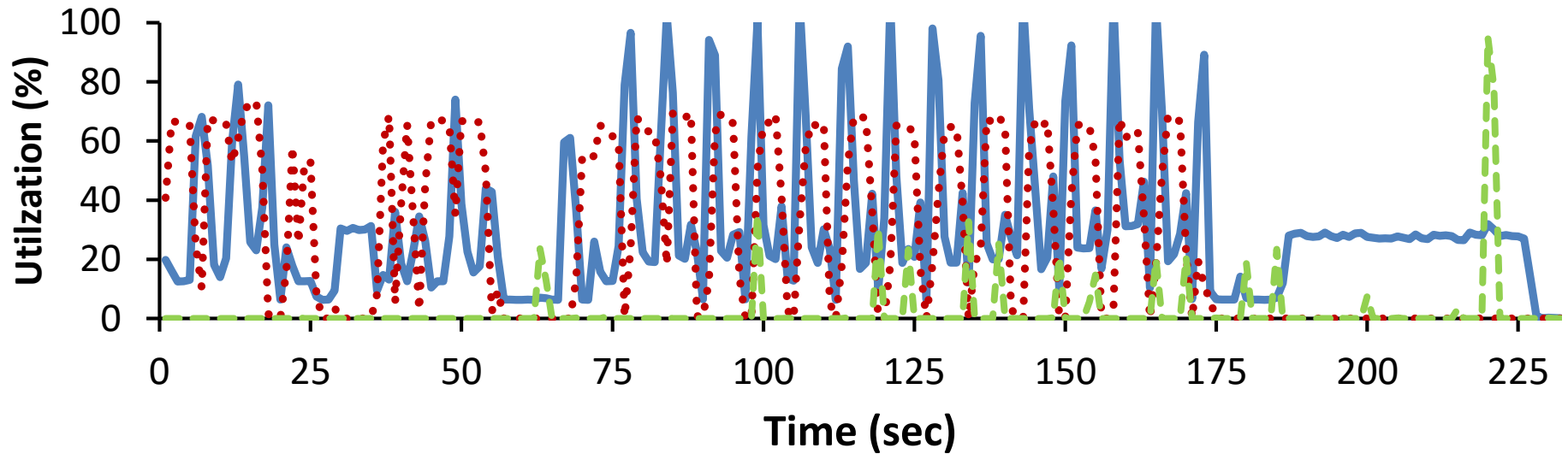
PCOPY utilization

[Data: TPCH – SF10
Input storage: HDD
Destination storage: DAS]

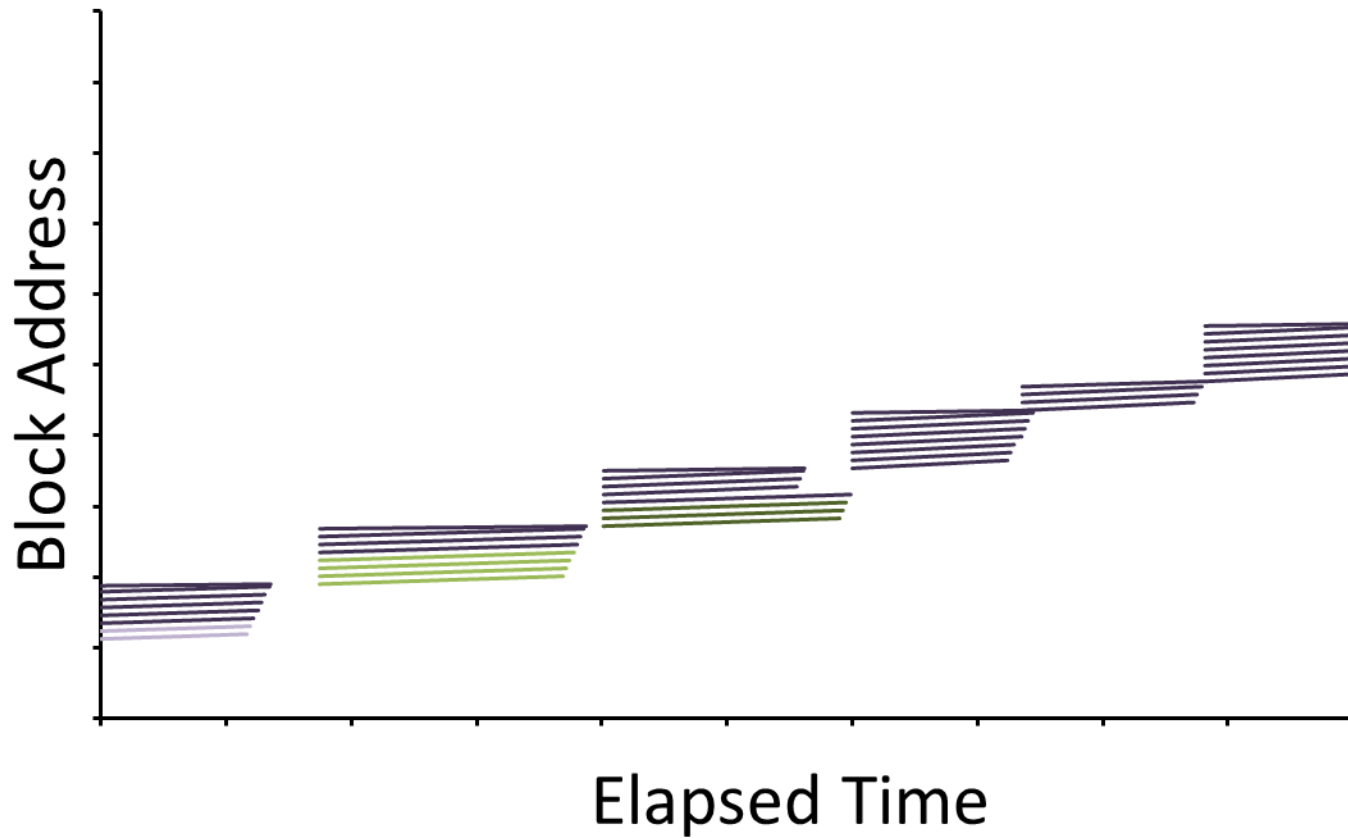


DBMS-C utilization

[Data: TPCH – SF10
Input storage: HDD
Destination storage: DAS]



DBMS-C read patterns



Reducing data loading overheads

- In situ querying [SIGMOD12, VLDB14]
- Data Vaults: Exploit metadata [Ivanova12, Kargin15]
- Instant Loading: SIMD & Code gen. [Muehlbauer13]
- Accelerators (FPGAs, GPUs)