The Role of Database Aware Flash Technologies in Accelerating Mission-Critical Databases

Gurmeet Goindi

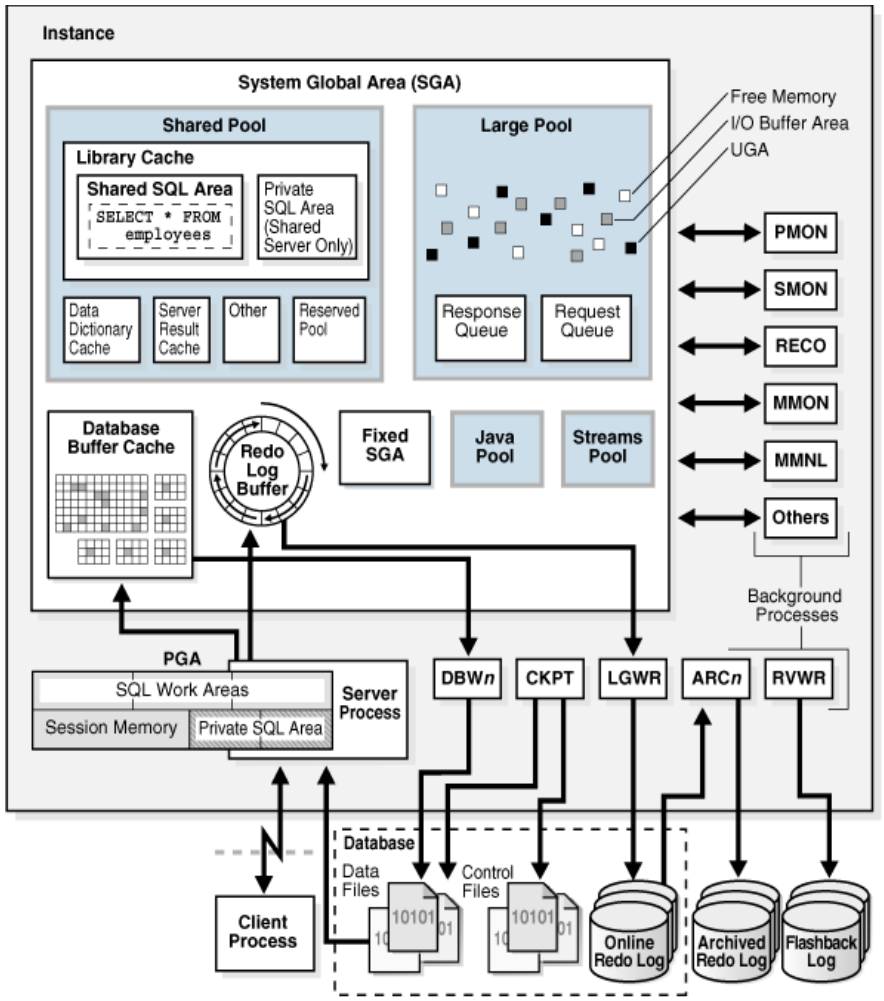
Principal Product Manager

Oracle

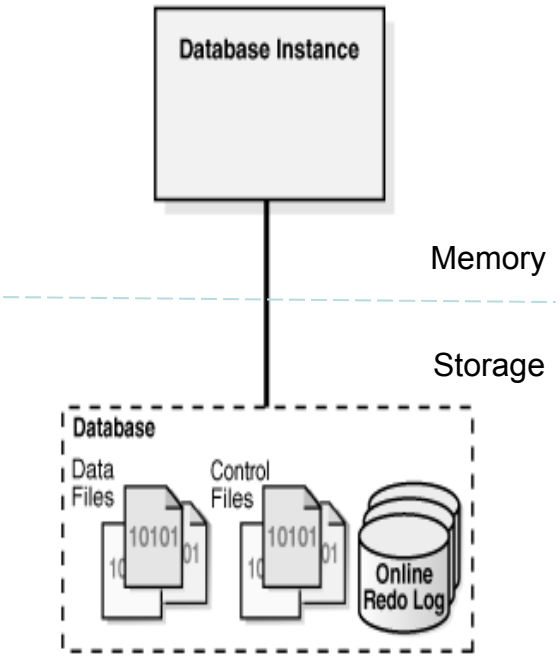
Agenda

- Relational Database – Architecture & Performance Characteristics
- The Promise of Flash and Relational Databases
- Conclusion

RDBMS Database Architecture



Logical View

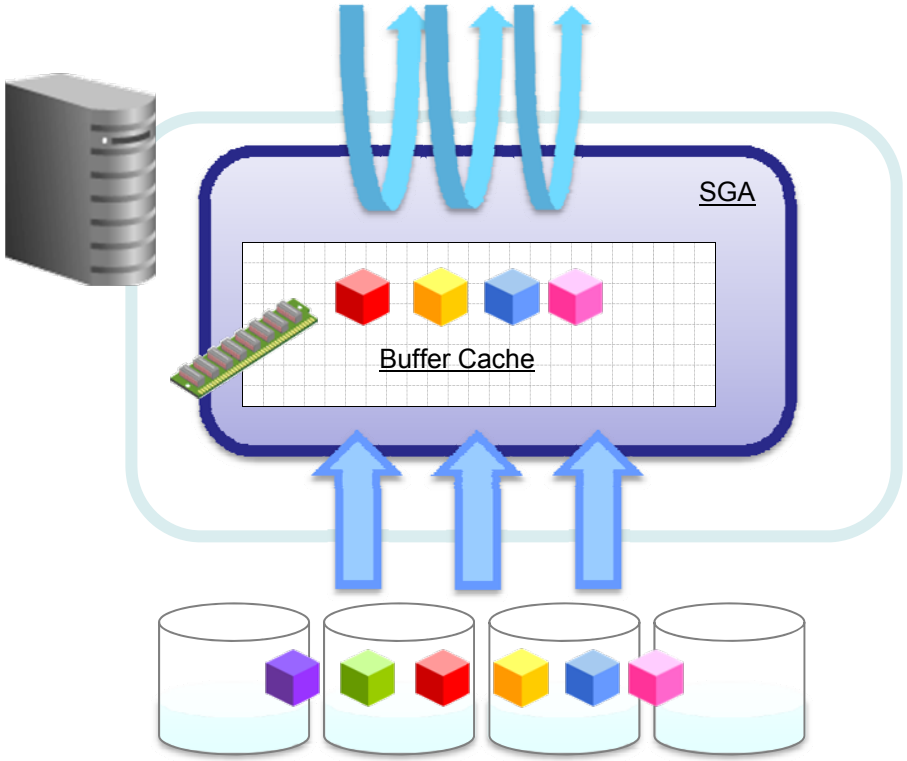


Physical

Databases and Storage

- Database Objects
  - Reads and Writes
  - Meta data associated with these Objects
- Database Logging
  - Mechanism to deliver transactional consistency
  - Critical functionality for database operation
  - Availability features

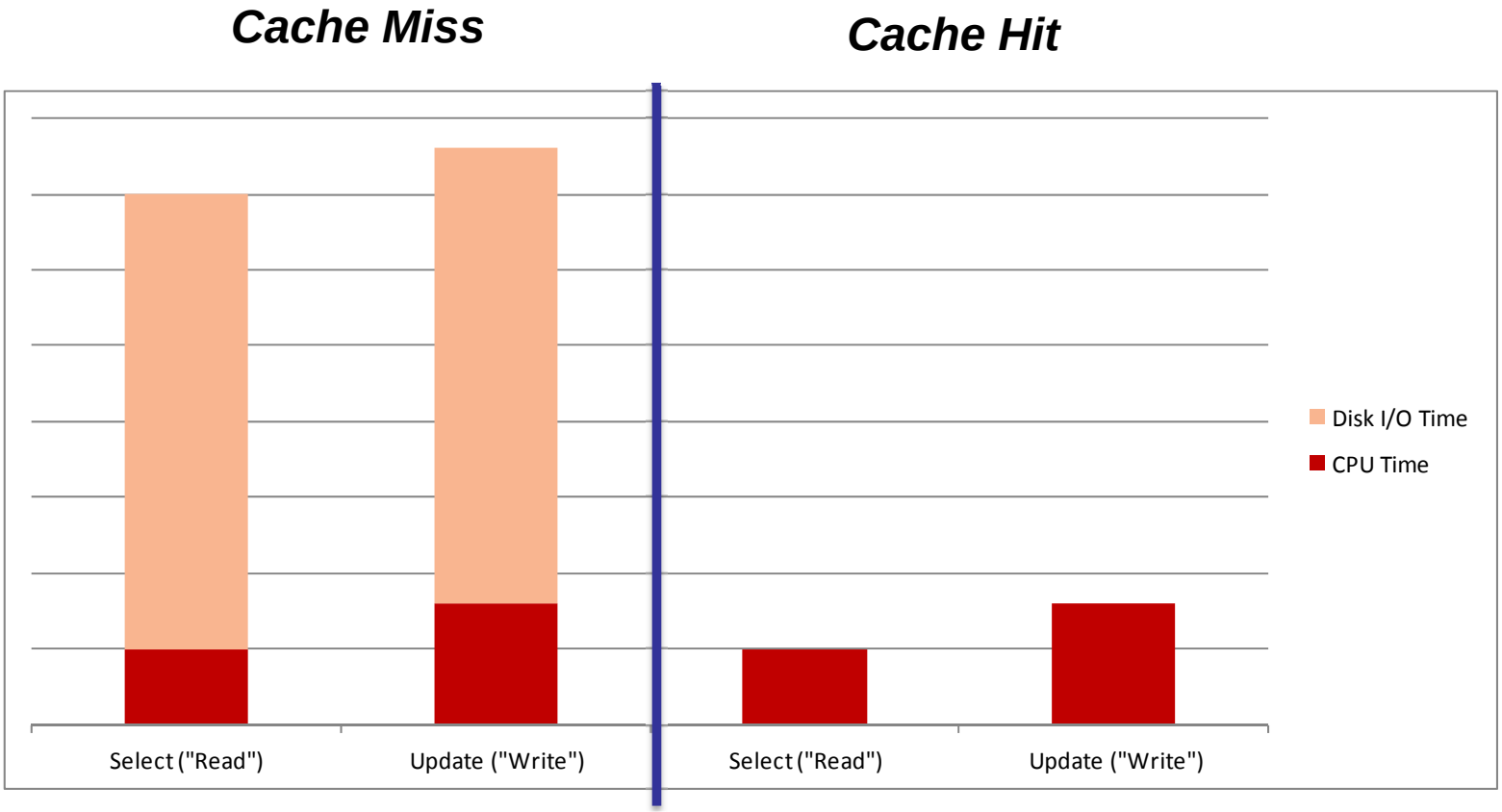
Why cache matters



100% cache hit ratio is ideal for OLTP workloads

- Data cached in memory yields faster query response
- Memory size has big impact on database performance

Why cache matters (ctd.)

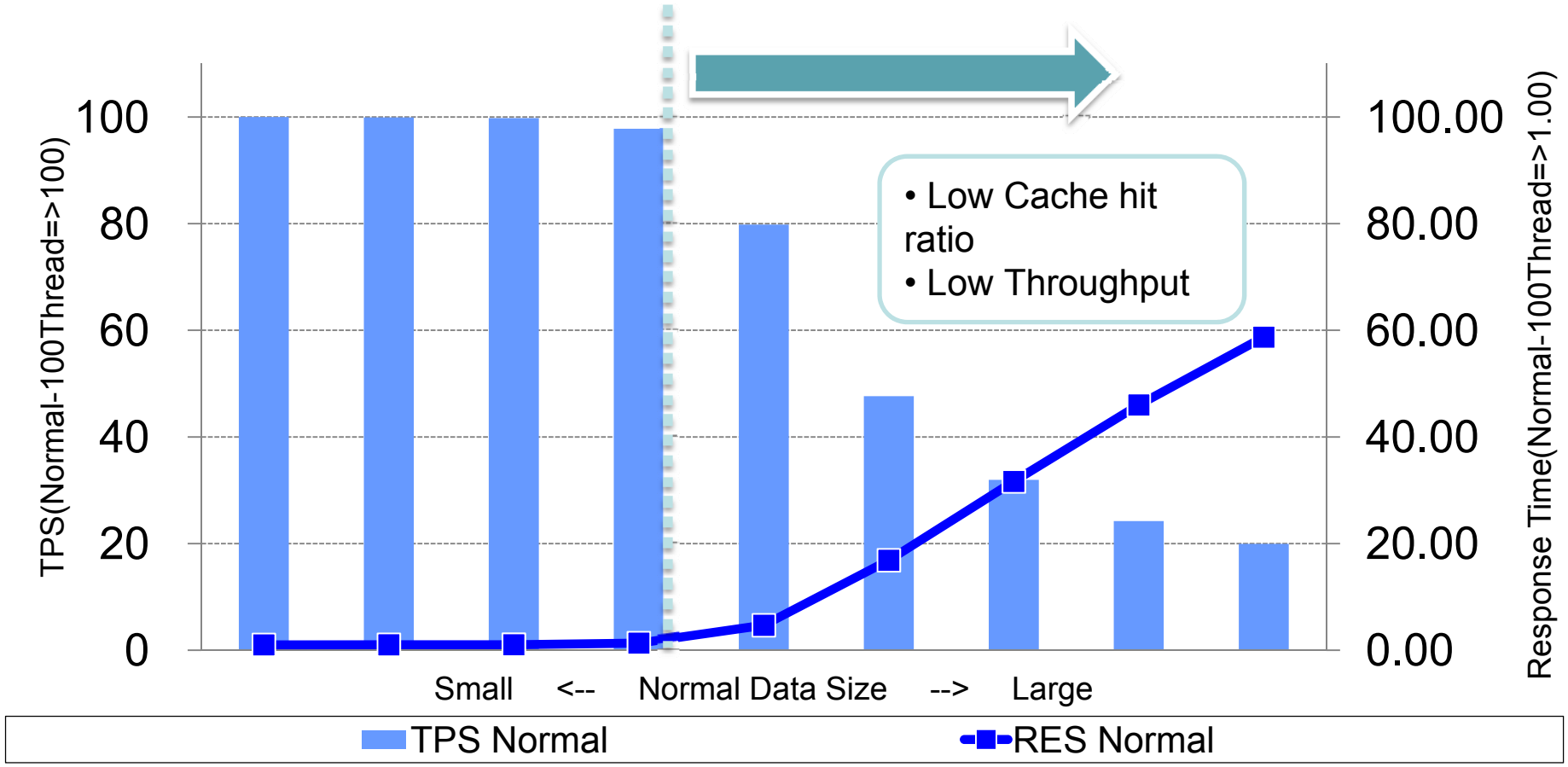


CPU time is mainly spent waiting for I/O

Performance degrades when

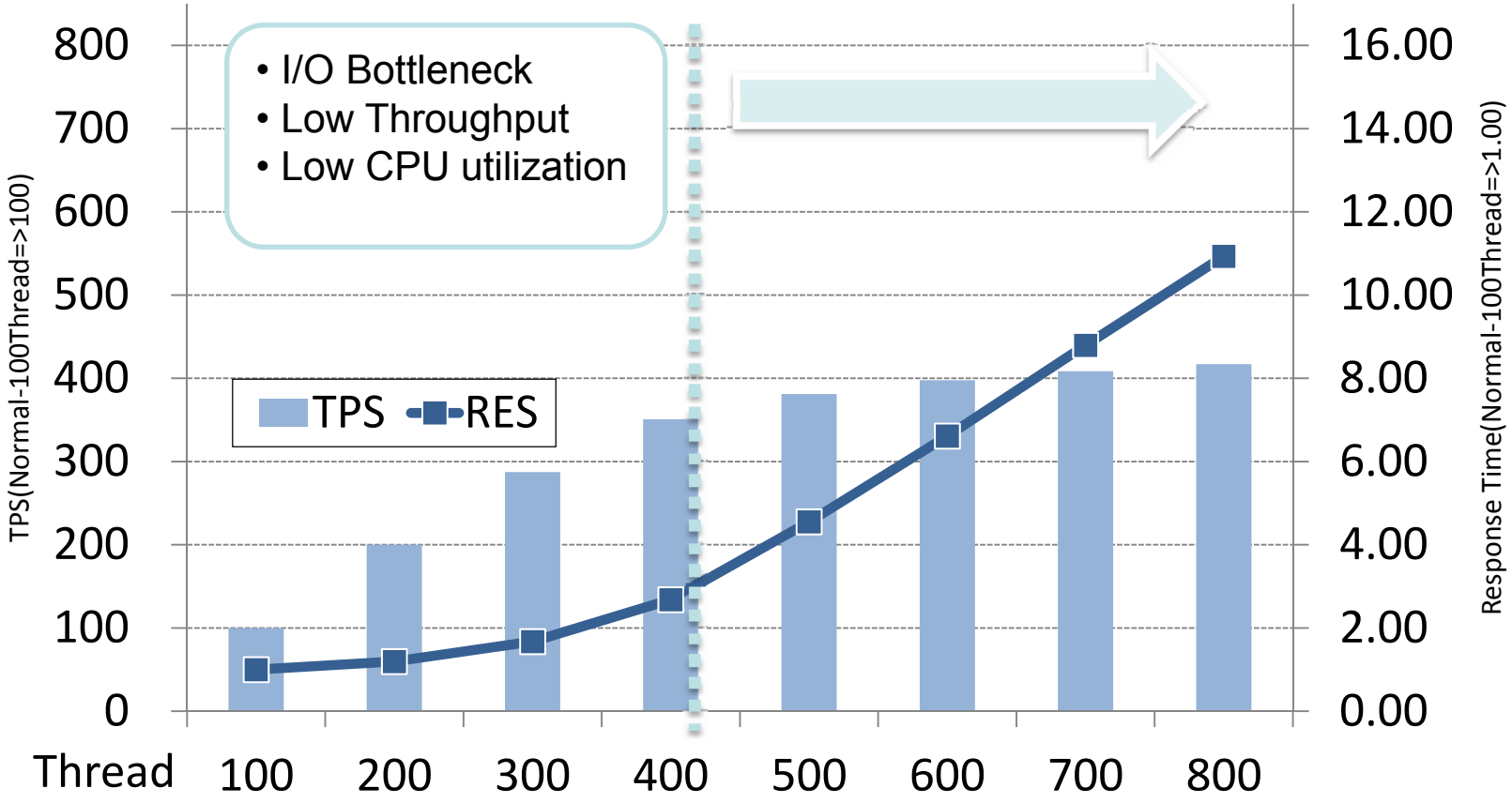
- Data size increases relative to cache size
  - Reduces the number of transactions that can be processed per unit time
  - Increases the response time per transaction
- Number of active users increases
  - Reduces the benefits of caching as different users work on different sets of data
  - Performance decreases as maximum user parallelism is reached

OLTP System Performance



Drastic increase in data impacting DB performance

OLTP System Performance

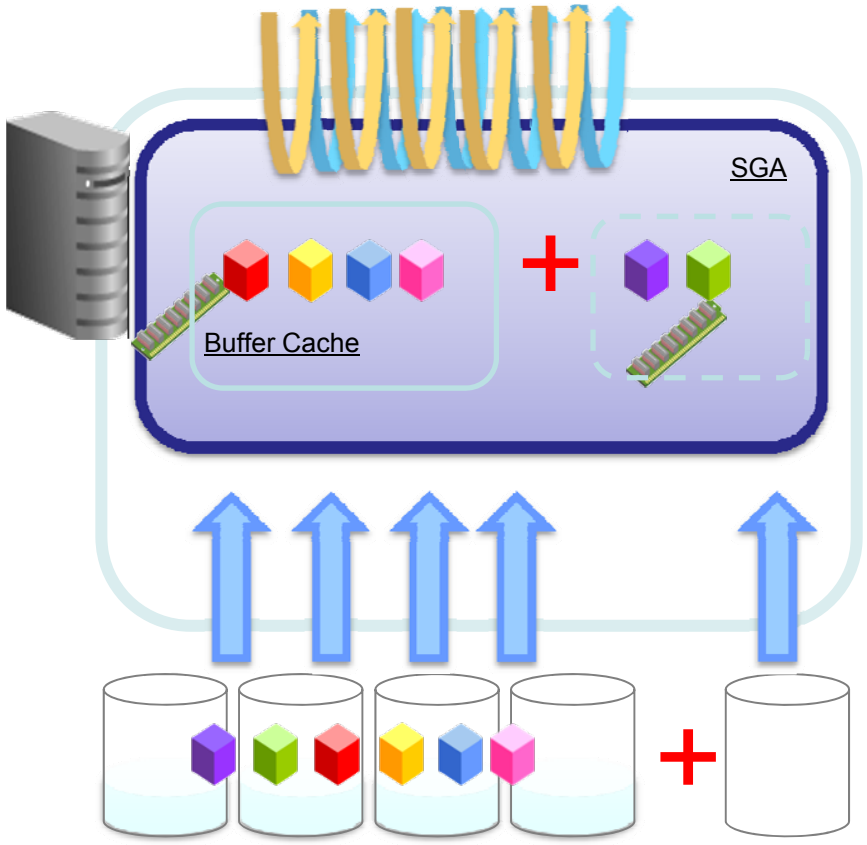


Huge user volume impacting DB performance

Why fast logging matters

- Log writing to permanent storage guarantees durability and availability of data
- Each committed transaction depends on log records being flushed to storage
  - Database frequently send logs to storage
  - Transaction response time depends on how fast the logs can be flushed to storage
  - How fast storage processes log writes has a direct impact on database performance and availability

Addressing performance issues



- Add Memory (DRAM)
- Add More disk
- Add Faster Disk
- Add Compute Nodes

… or:

- Introduce Flash in to the architecture

Agenda

- Relational Database – Architecture & Performance Characteristics
- The Promise of Flash and Relational Databases
- Conclusion

Applications of Flash in Databases

- As an additional storage layer (caching)
  - Stage active Database objects in flash
  - Accelerate reads and writes to these objects
- For Database Logging
  - Host critical logging functions of the database
  - Improves user transaction response time
  - Increases overall throughput for IO intensive workloads

Flash as a Cache

- Two flavors
  - Write Through
  - Write Back
- Boost database performance as currently active data resides on a much faster media
- Usually software controlling the Flash layer decides what data to cache
- Flash layer caches based on observed access patterns, not on application needs
  - aka “content agnostic” caching

To Cache or not to Cache

- Random reads against tables and indexes
  - Cached: more likely to have subsequent reads
- Sequential read tables, or Scans
  - Not Cached: sequentially accessed data is unlikely to be followed by reads of the same data
- Backups, mirrored copies of the block
  - Not Cached: Why?

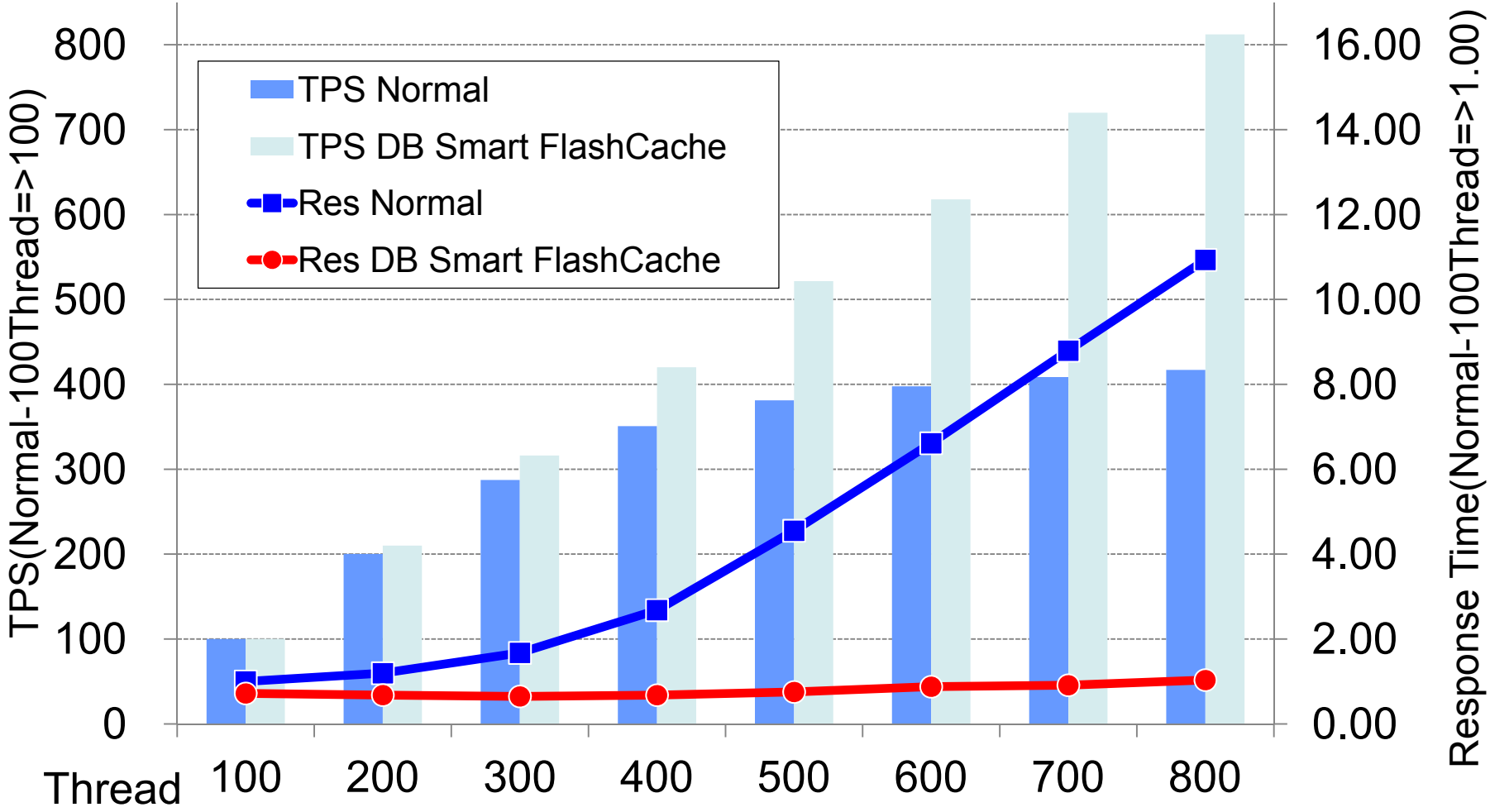
But most general purpose flash solutions are database agnostic and cache all the above workloads

Oracle's Approach: Make Flash Database Aware

- Database Smart Flash Cache
  - Extends the buffer cache and is not an alternative to primary storage
  - Cache data that is aged out of buffer cache
  - Consistency maintained when used in a clustered environment (Oracle RAC)
- Software understands Oracle block structure and the corresponding operations
  - Skips Caching: I/O Mirrors to copies, backups, data extraction (Datapump) & table space formatting
  - Caches: Control files reads and writes, data blocks and index blocks

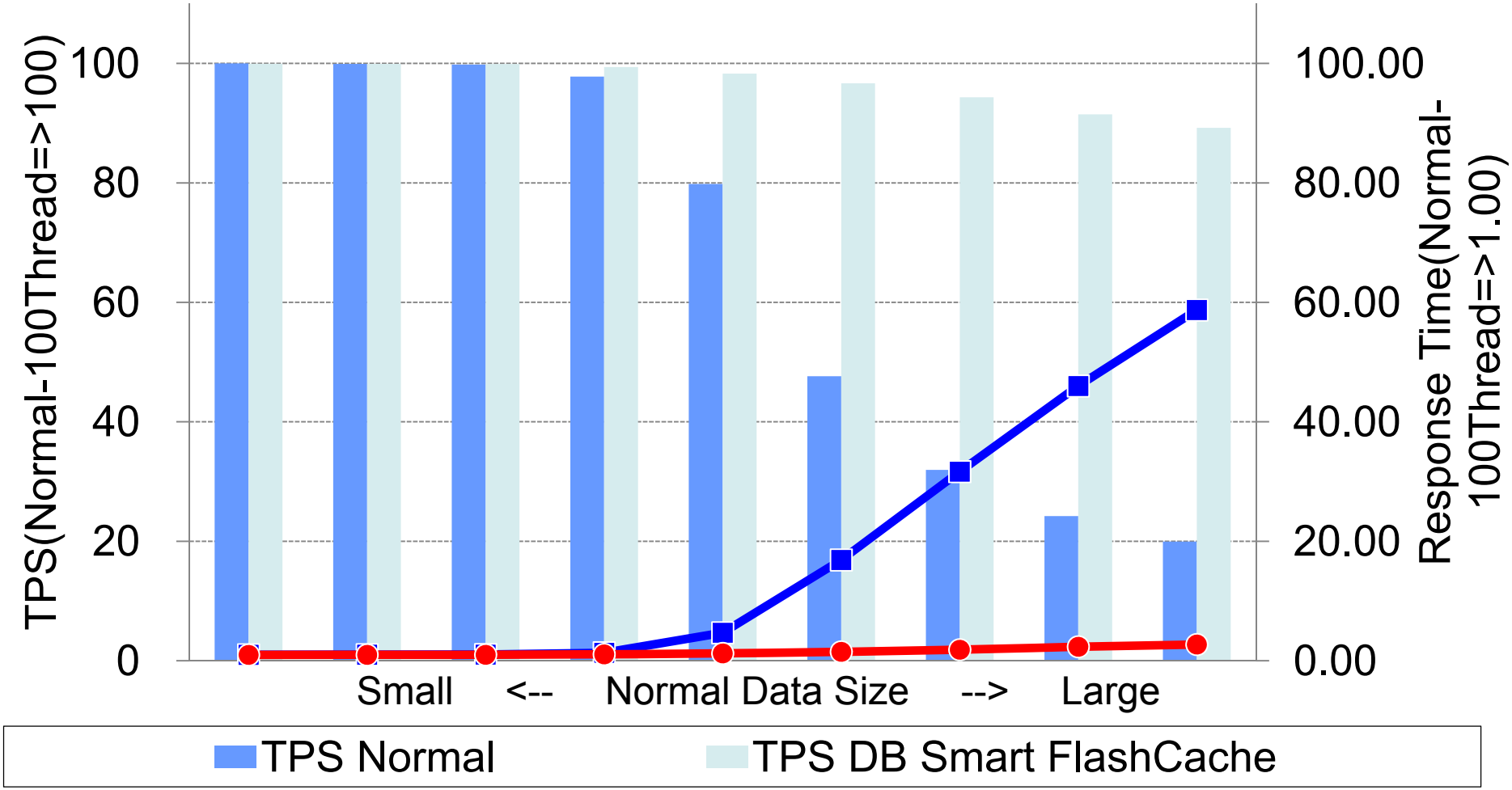
Database Smart Flash Cache

Performance(TPS): Increasing number of users



Database Smart Flash Cache

Throughput : Increasing data size

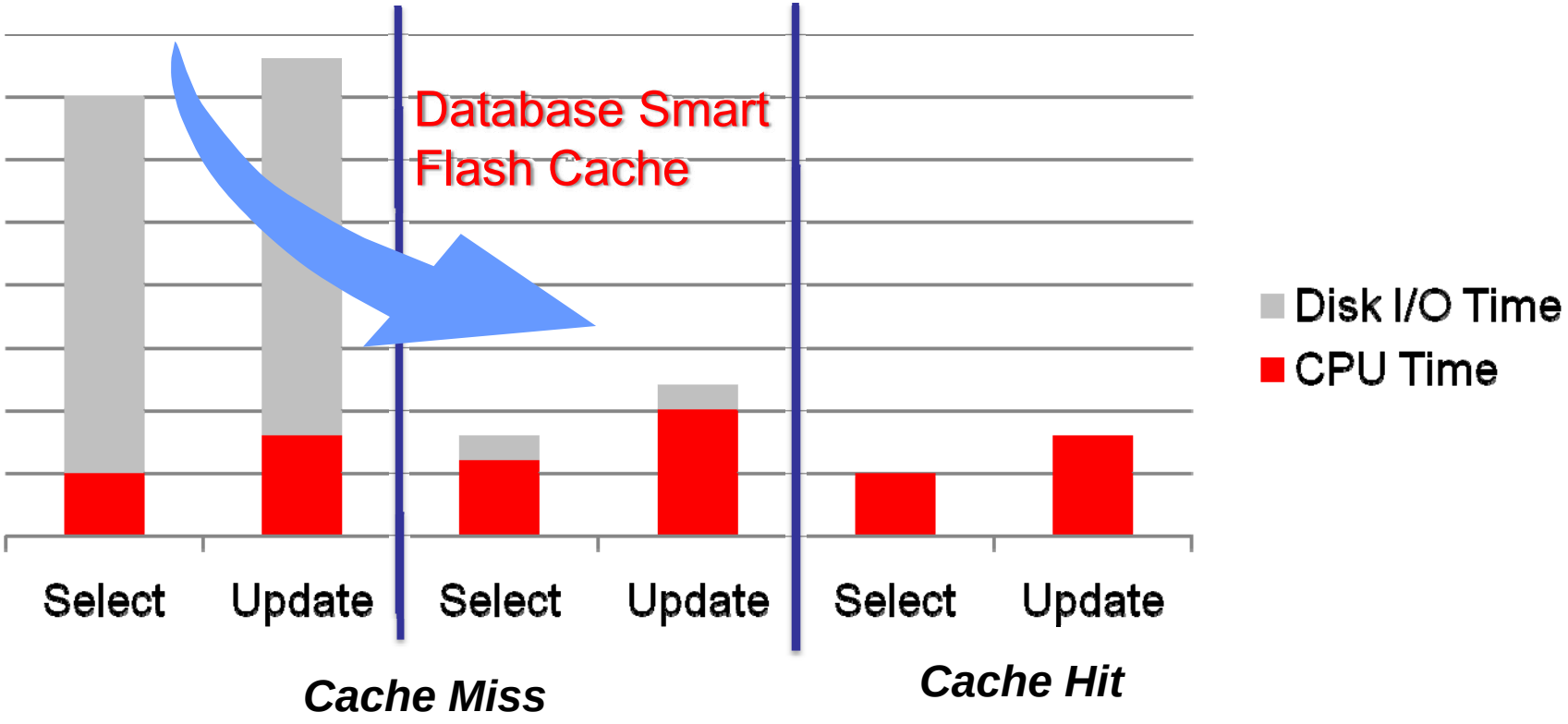


Database Smart Flash Cache

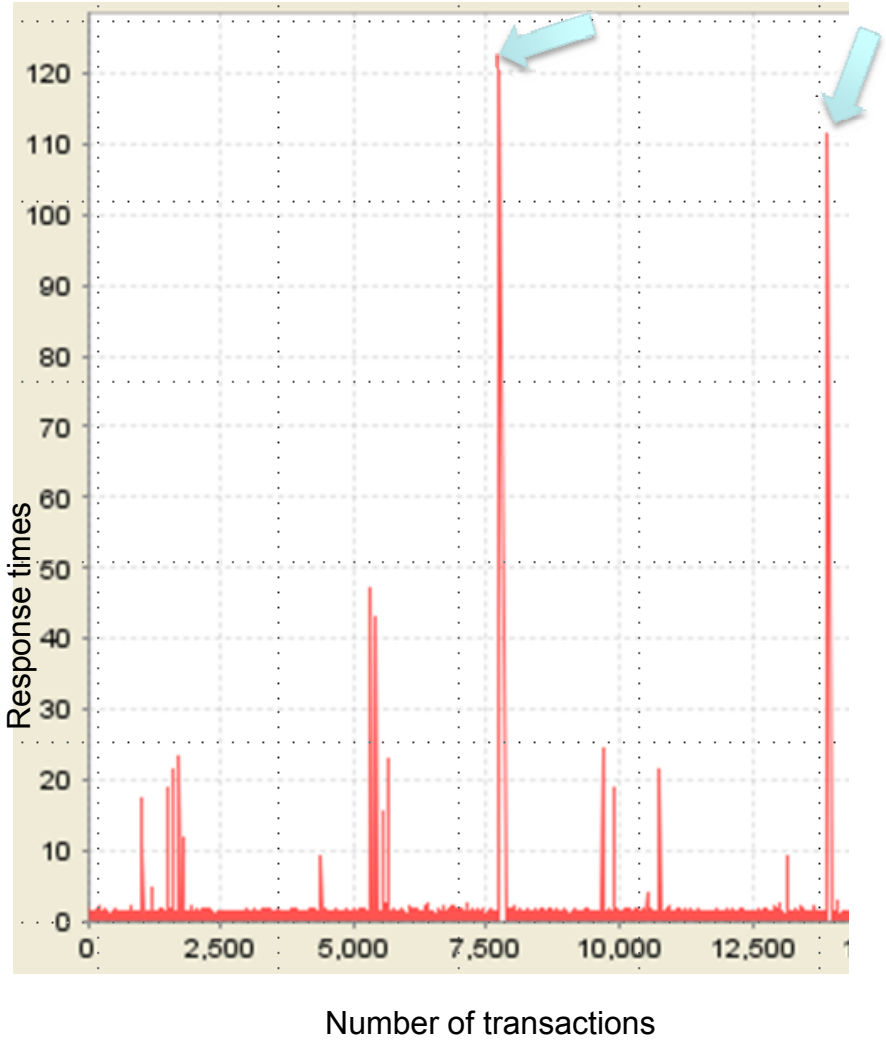
Time consumed while executing SQL

- Reduce I/O wait time on “cache miss”

→ Same performance with “cache hit”



Flash And Database Logs



- Flash has very good *average* write latency
- Greatly improves user transaction response time
- Flash occasional outliers, one or two orders of magnitude slower
- OLTP workloads dislike such large variations
- *Oracle's Approach:* Write to Flash and the DRAM cache in the disk controller simultaneously to even out the impact of outliers
  - the first to complete "wins" so that outliers are avoided (on either medium)

Where To Introduce Flash ?

- Direct Attached – PCI or SSD
  - Extremely fast, improves performance for currently active data
  - Limited scale out, clustering and high availability features and capacity
- Tiered or Hybrid Storage
  - More resilient and scalable architecture compared to server attached flash. Better resource utilization as well
  - Tier-ing algorithms rebalance the data between various layers, every few hours, based on IO usage patterns
  - Most Storage Controllers are IO bound
- All Flash Arrays
  - Best in class performance, expensive but getting cheaper
  - Lack the maturity and functionality that traditional arrays have added over two decades
  - Storage controllers are still IO bound

Most solutions are driven by application usage patterns and not by application needs

Where To Introduce Flash (contd) ?

- Flash devices in application tier (server attached flash) lacks enterprise class scalability and high availability
- Flash devices in traditional storage arrays are not efficient as storage controllers don't respond quickly enough to workload changes and are IO bound
- *Oracle's Approach:* Move database functions to storage tier to get the best of both solutions - *Exadata Smart Flash Cache and Intelligent Storage Grid*

Oracle's Approach: Database Aware Flash Architecture

- Database Aware Flash Cache
  - Intelligent Caching
  - Adaptive to workload changes
  - Fine grain cache sizes
- Extend Database's high availability and clustering features to the Flash Layer
- Pure Scale-out Architecture: Non-blocking low latency Inifiniband Fabric
- Offload some Database processing to the storage servers

Agenda

- Relational Database – Architecture & Performance Characteristics
- The Promise of Flash and Relational Databases
- Conclusion

Conclusion

- Flash generally improves the performance of the database, BUT
- To realize the full benefits of a flash solution for the database, it needs to be carefully introduced at every layer of storage hierarchy
- At each layer there are tradeoffs for Availability, Scalability and Performance of the database

Thank You