Improving Analytics: Experiences Testing Oracle Database 12c In-Memory Feature

Wayne Martin
Technology Manager, IBM CORPORATION

Ralf Schmidt-Dannert,
Executive IT Specialist, IBM CORPORATION

Agenda

- IBM Oracle Relationship
- Project Objective
- Oracle Database In-Memory Overview
- Test Environment
- In-Memory Compression Effectiveness
- Test Results
- System Resource Considerations
- Oracle Database In-Memory & Oracle Real Application Clusters
- Why POWER8 for In-Memory
- Conclusions, Thank you and Acknowledgments

IBM and Oracle, a shared commitment to Client Value

Sustained Collaboration for Clients

Oracle 1986, PeopleSoft1988, JD Edwards 1976, Siebel 1998

Mutual Executive Commitment

Dedicated, Executive-led Alliance teams

Regular Senior executive reviews and functional Executive interlocks

More than 100K joint Technology Clients

Hardware and Software support via in-depth certification process

Award-winning Oracle Services Practice– Diamond Partner

More than 6,500 successful joint services projects,

>14,000 skilled Oracle resources worldwide

Vibrant technology Collaboration

Continued joint development delivering Oracle software optimized for IBM hardware, significant skills/resources investment, dedicated International Competency Centers

Cooperative Client Support Process

Dedicated resources and significant program investments

Project Objective:
Evaluate Oracle Database In-Memory

Oracle's stated benefits of Database In-Memory Include*

- Faster scanning of large number of rows and applying filters that use operators such as =,<,>, and IN
- Faster querying of a subset of columns in a table, for example, selecting 5 of 100 columns
- Enhance join performance - convert predicates on small dimension tables to filters on a large fact table
- Efficient aggregation by using VECTOR GROUP BY transformation
- Reductions in indexes, materialized views, and OLAP cubes can produce:
  - Reduced Storage Space requirements
  - Reduced SAN bandwidth demand
  - More efficient use of processor power

This project focused on evaluating those benefits by:

- Creating a workload that exercises the listed benefits.
- Executing the workload on a well optimized server and storage systems
  - Base line created with row based execution
  - Rerunning with the DB In-Memory feature activated
- Evaluating scalability of the solution with concurrent execution
  - Queries run with 1, 2, 4 and 8 concurrent users.
  - Use of Single Instance and Real Application Cluster (RAC)

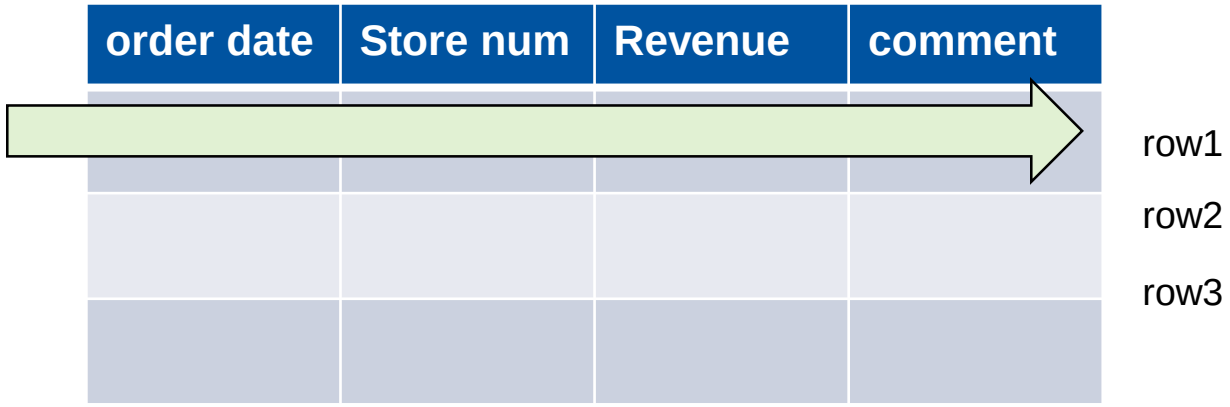
* According to the Oracle 12c Release 1 Database Data Warehousing Guide, https://docs.oracle.com/database/121/

Oracle In-Memory - miscellaneous

- Available since November, 2014 on AIX with Oracle 12c release 12.1.0.2
- Minimum supported AIX releases are:
  - 6.1 TL7 SP3, 7.1 TL1 SP3 and 7.2 TL0 SP1
- Chargeable feature with list price of **$23k per POWER core**
- Support for single instance and clustered (RAC) Oracle databases.
- Note: Mirroring of In-Memory data on multiple nodes in a RAC configuration is only supported in Oracle Exadata servers.
- In-Memory is also supported on Linux on IBM z Systems.
- Oracle Database In-Memory license also enables another feature called “Vector Group By Aggregation”.

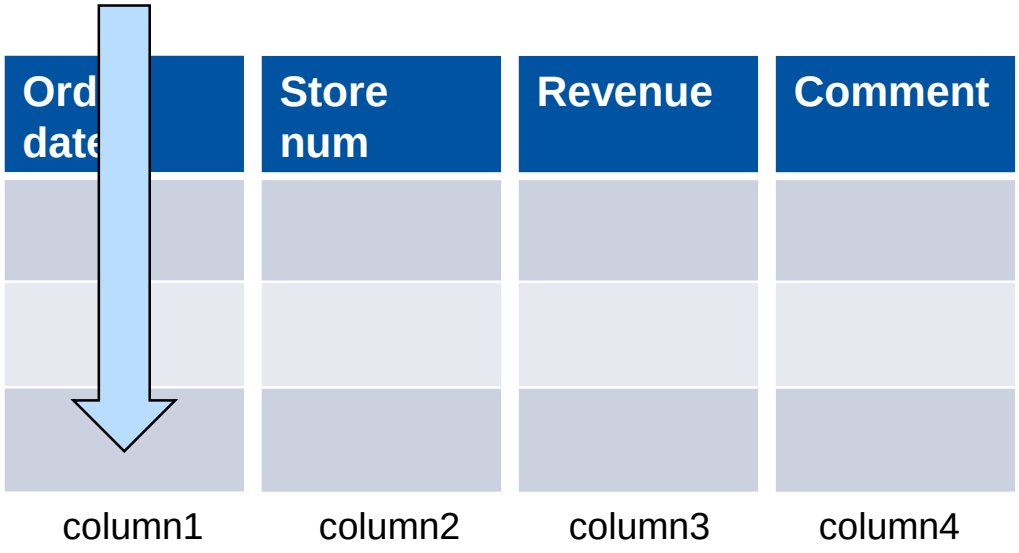
Sources: Exadata pricing guide and Oracle technology price list (as of March 3, 2015)

Traditional databases are either row or column oriented



Row format is best for OLTP workloads:

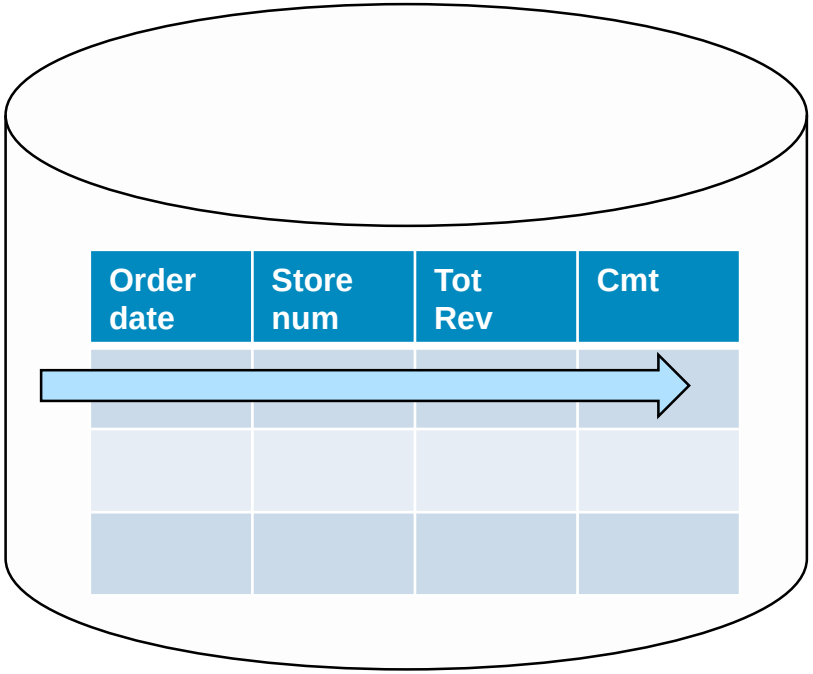
- Data insertion, update
- Selecting many columns but few rows



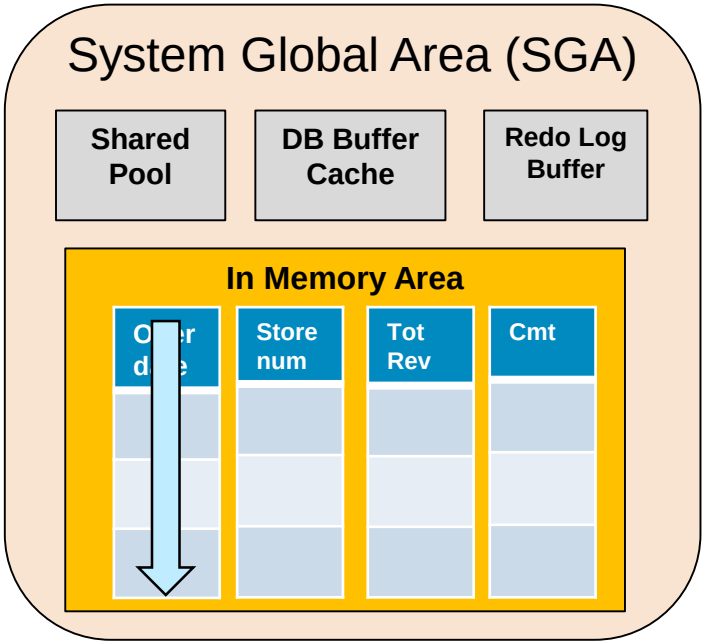
Column format is best for analytics:

- Mostly read data
- Selecting few columns but many rows

Oracle Database In-Memory feature offers both row (required) and column (optional) format



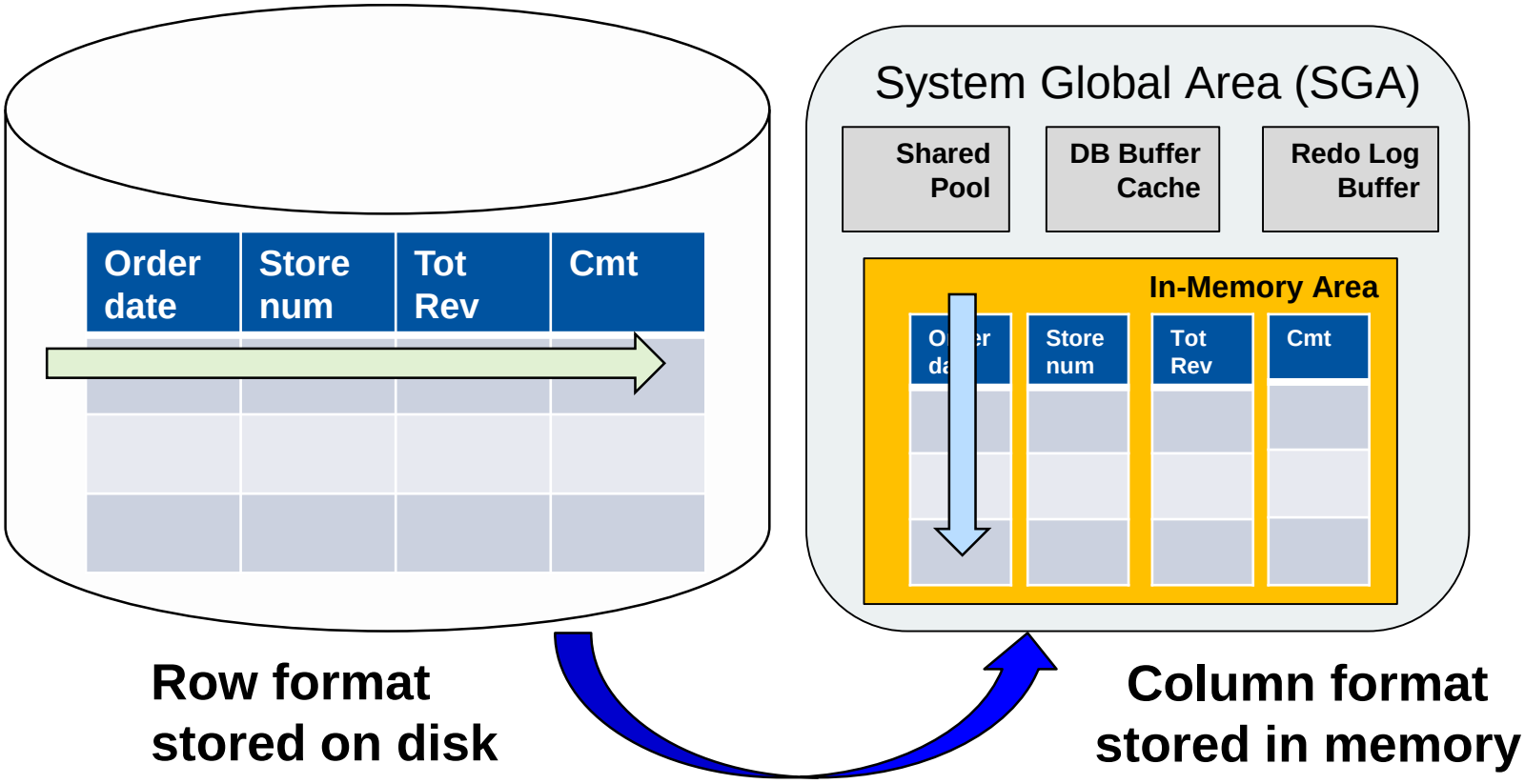
Row format stored on disk and buffered in the SGA



Column format stored only in memory

- Does not require the entire database or object to be stored in column format in memory
- Column format must be specified manually and be specified for a tablespace, table, partition, subpartition, materialized view, or just a column of a table or partition
- These objects are permitted to exceed the space allocated for the In-Memory area
- Also enables “Vector Group By Aggregation” which supports both data formats, row and column

How the column format is created



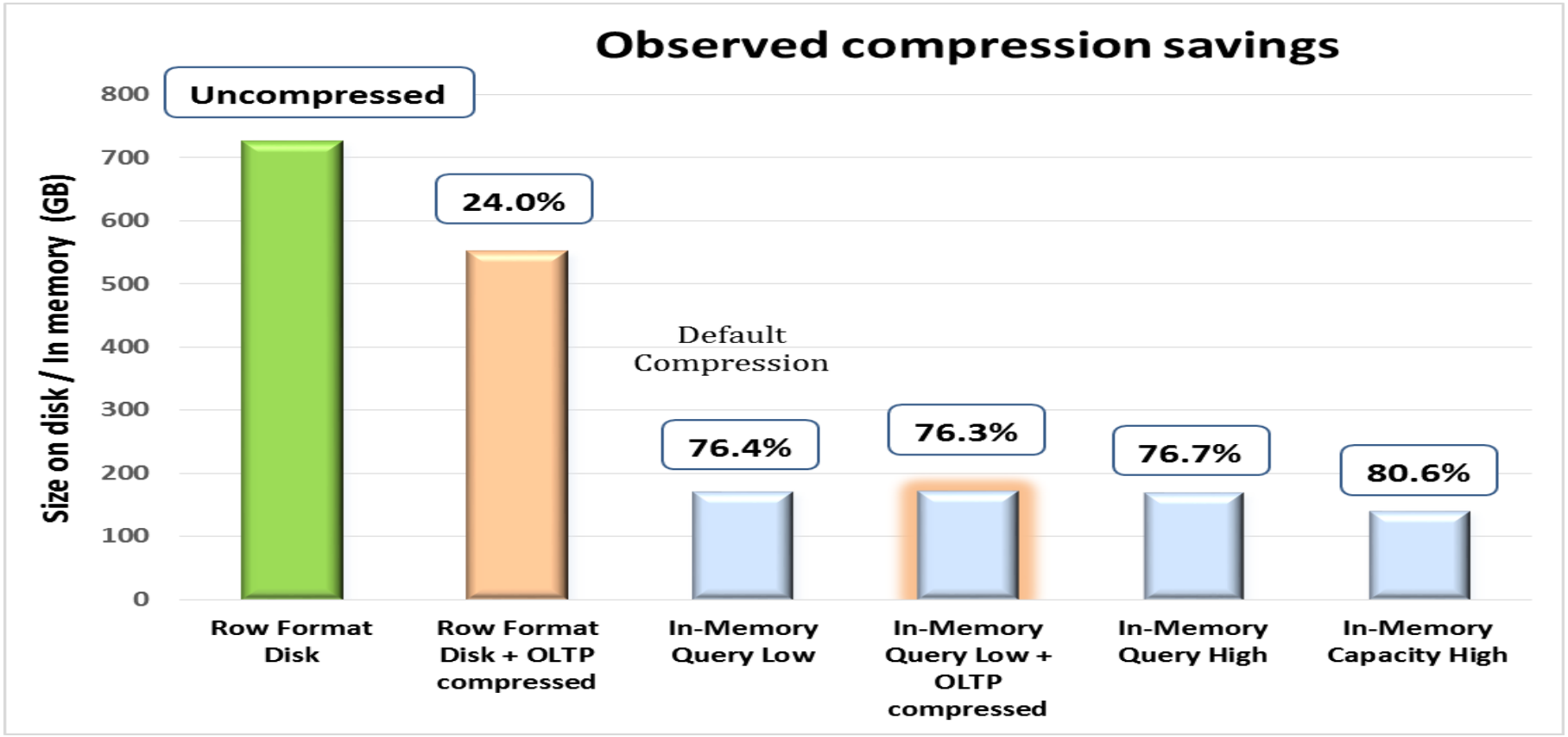
SQL> alter table sales_fact **inmemory**;

- Parameter ***inmemory_max_populate_servers*** controlls the # of background populate servers to use for In-Memory (to load data into the SGA)
- Parameter ***inmemory_trickle_repopulate_servers_percent*** controls the amount of time the background in-memory populate servers can use to update the in-memory data with data changes

Test workload and environment

- Internally developed database
  - Scalable, tests utilized 1 TB and 10 TB database size
  - Single range-hash partitioned sales fact table with 9 billion and 90 billion rows in 1152 partitions
- Queries based on representative business needs
  - Set of 23 queries with a range of complexity
  - Work load scaled by execution with 1, 2, 4 and 8 concurrent users
  - Single instance and Oracle RAC configurations
- Additional “simple queries”
  - Designed to specifically benefit from In-Memory features
- Systems
  - Utilized IBM FlashSystem 840 storage
  - IBM Power E880 for non-RAC testing
  - 2x IBM Power E850 for two-node RAC testing

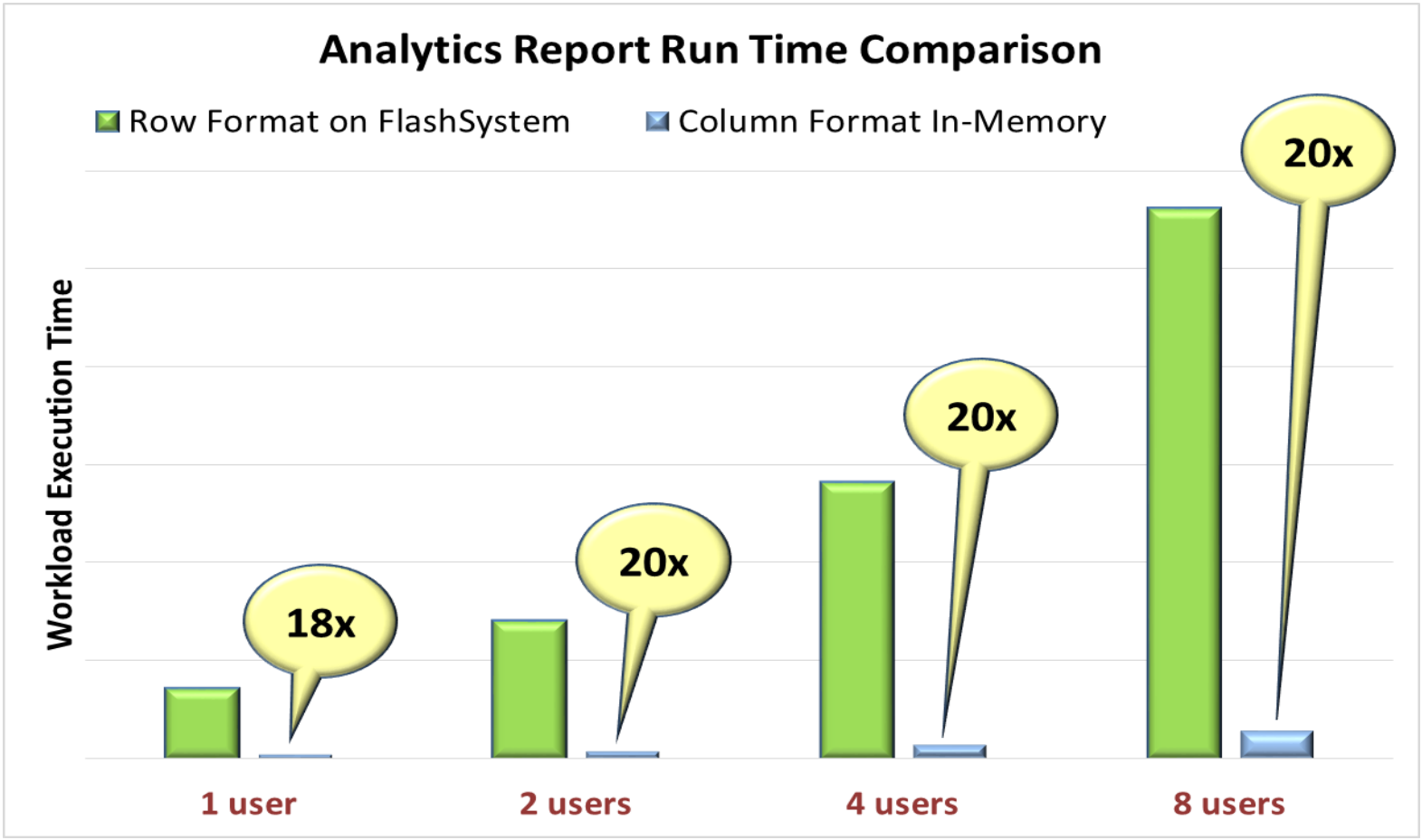
In-Memory compression effectiveness – a comparison



The test data consisted of a 726GB range-hash partitioned fact table, with 16 numeric columns. Achievable compression ratios are data dependent and your results may vary.

Note: This test required to set inmemory_size to 200GB to completely store the 171.4GB of compressed actual data in memory. The remaining ~28.6GB were used for the additional metadata required.

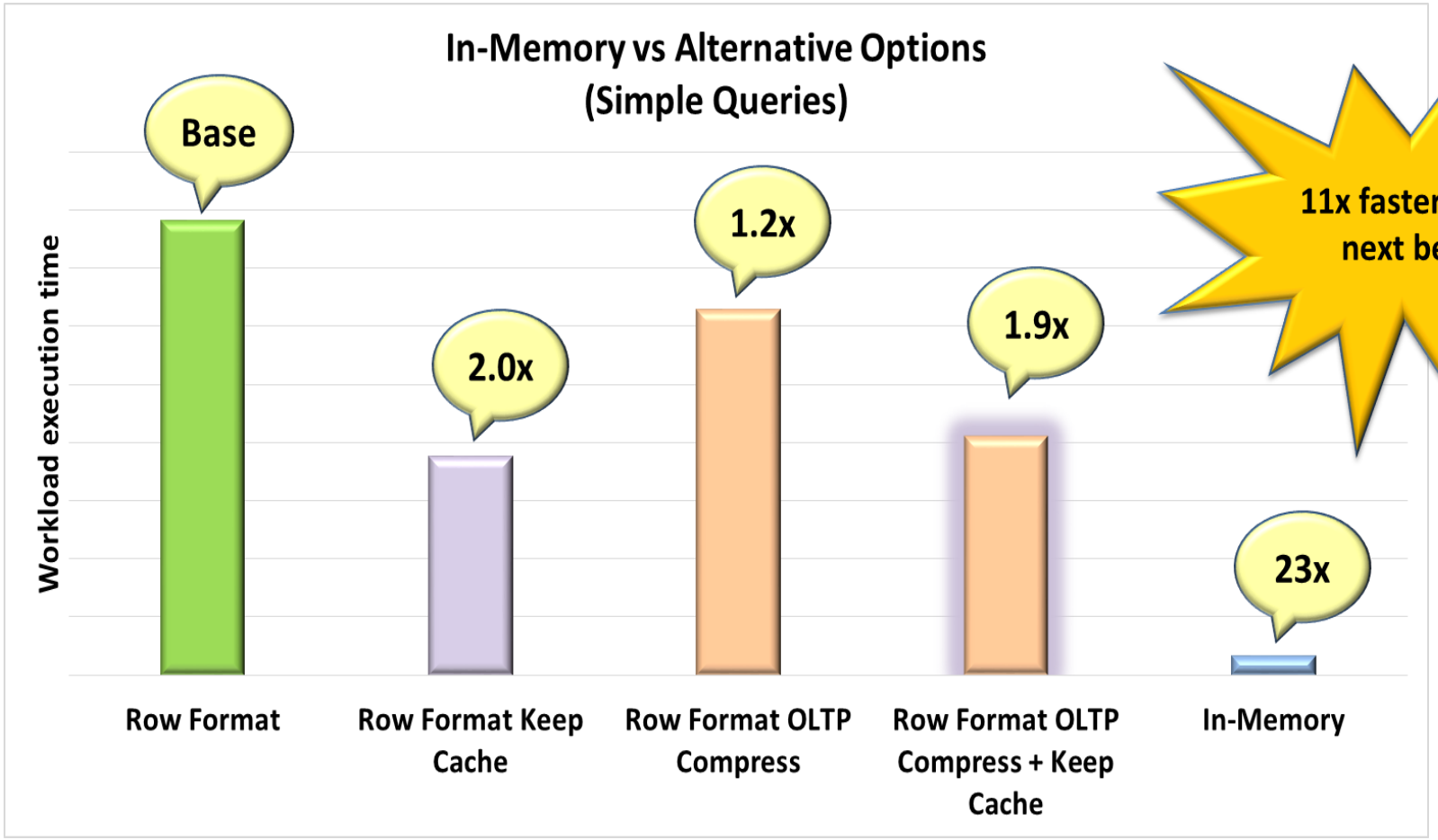
“Real World” complex Analytics Reporting with Oracle Database In-Memory



- The table queried had 9 Billion rows with 16 numeric columns and was loaded into the In-Memory area with compression set to “Query Low”.
- The execution time was measured from the start of the first report until the last report had completed for all users.

Disclaimer: Mileage may vary by workload

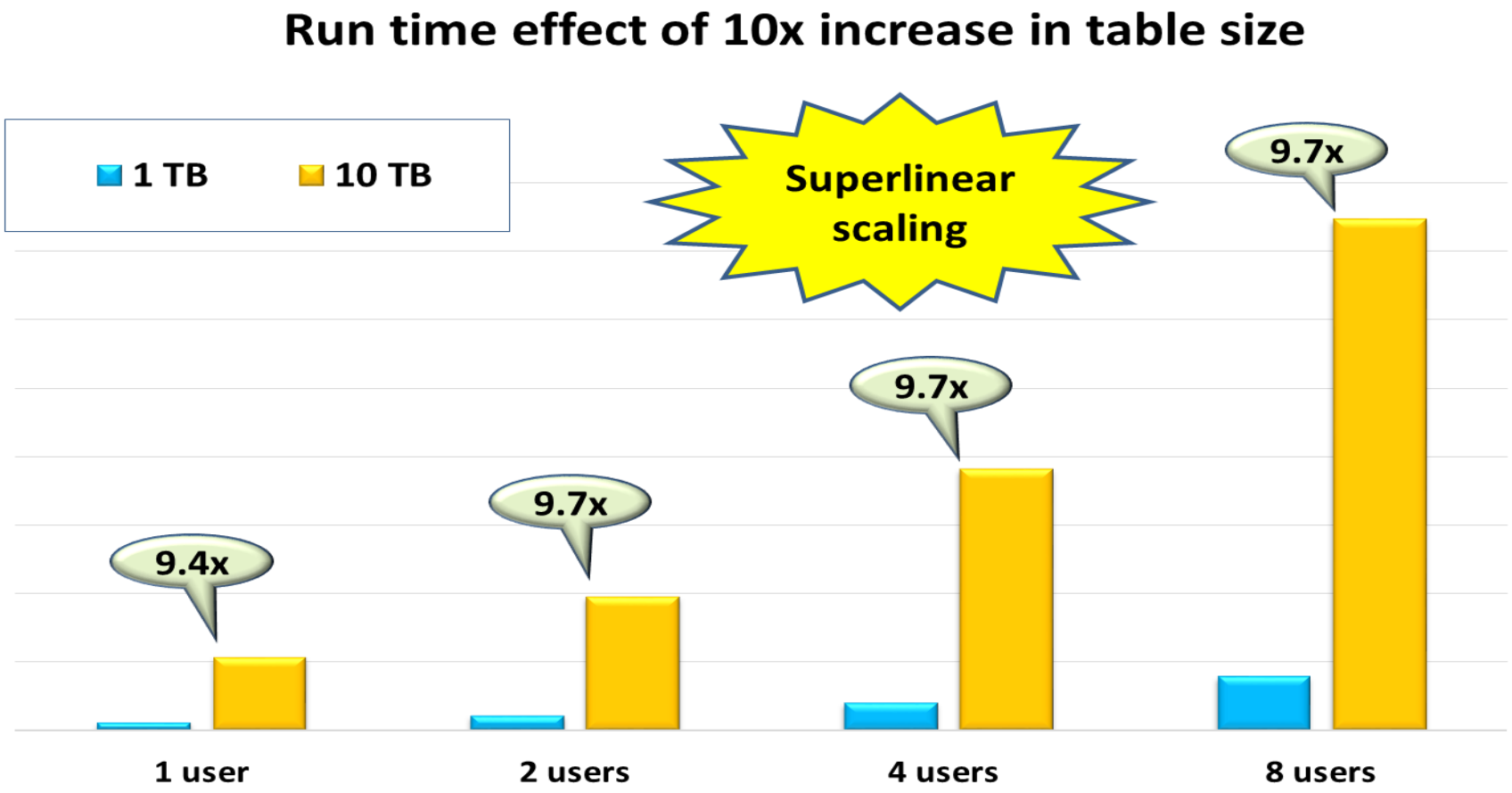
Comparing Oracle Database In-Memory execution times with alternatives using “simple queries”



Results with Oracle DB In-Memory feature are 11x faster than the next best alternative (Row Format Keep Cache), while requiring ~ 3x (550G) less memory!

Disclaimer: Mileage may vary by workload

Run time scaling with 10x larger fact table: 1TB -> 10TB

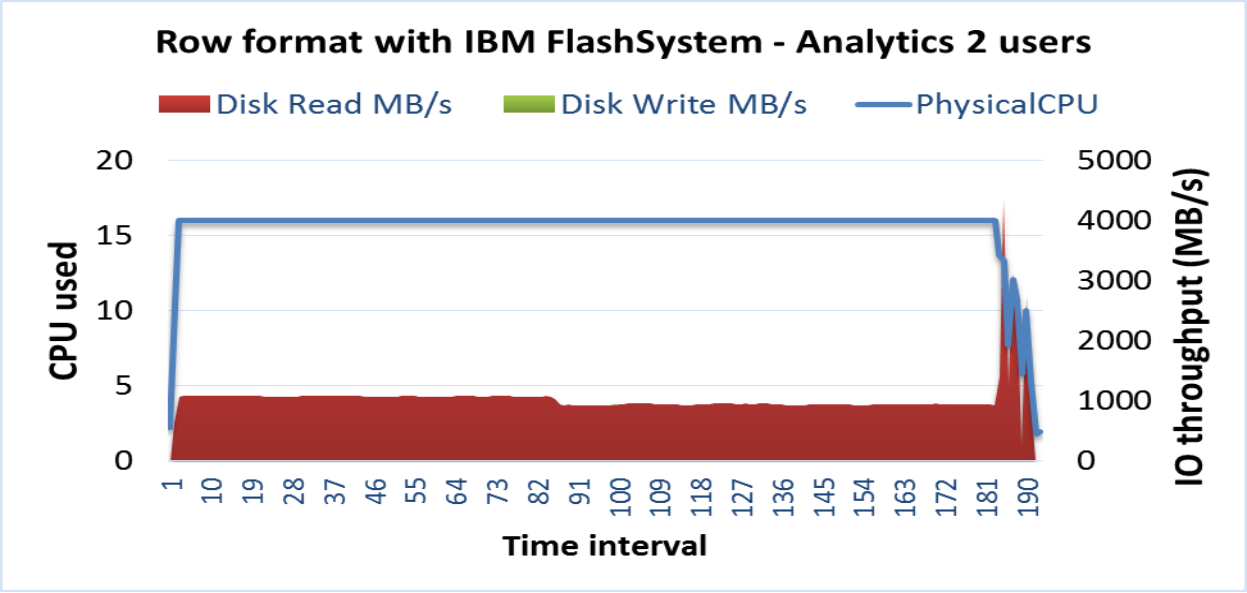


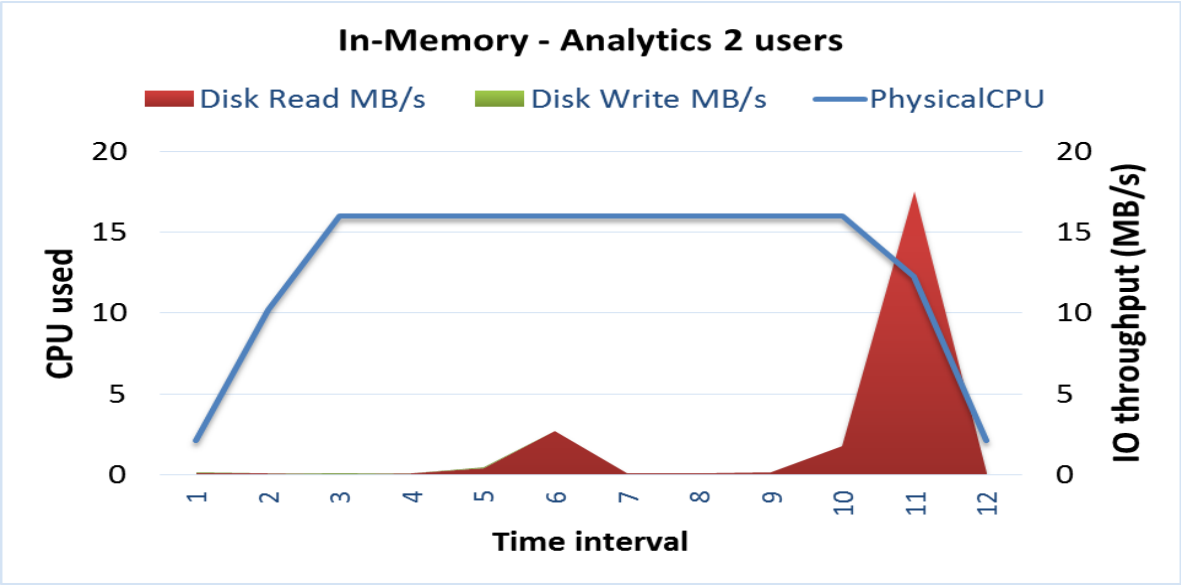
10x larger fact table (7.26TB uncompressed, 2TB compressed with 90 billion rows), resulted in run time increase of only **9.7x**.

Disclaimer: Mileage may vary by workload

System resource considerations

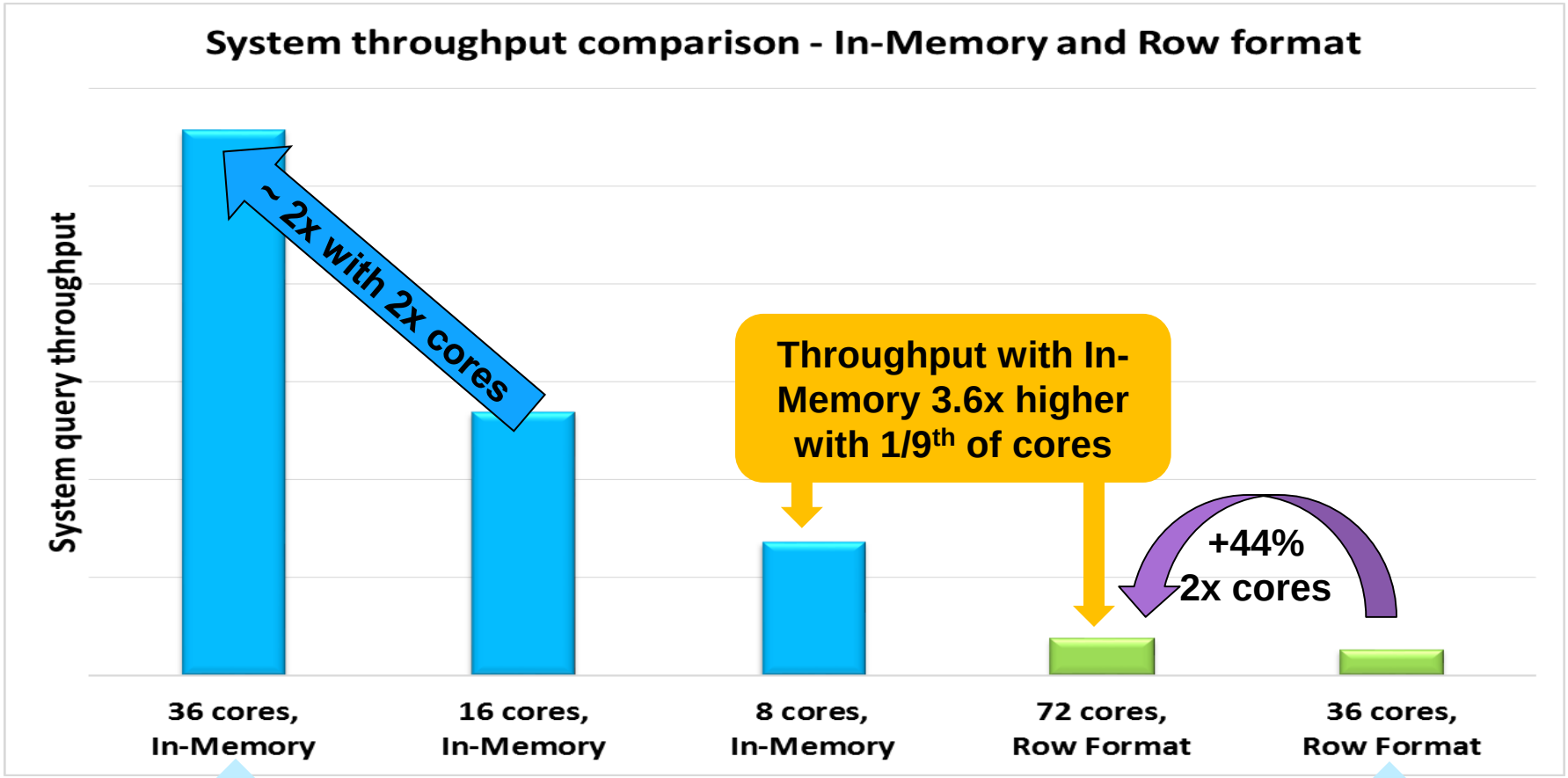
- 726GB fact table
- CPU bound
- Peak 4.3GB/s read
- Sustained > 1GB/s





- 171GB compressed In-Memory fact table
- CPU bound
- Peak 17MB/s read

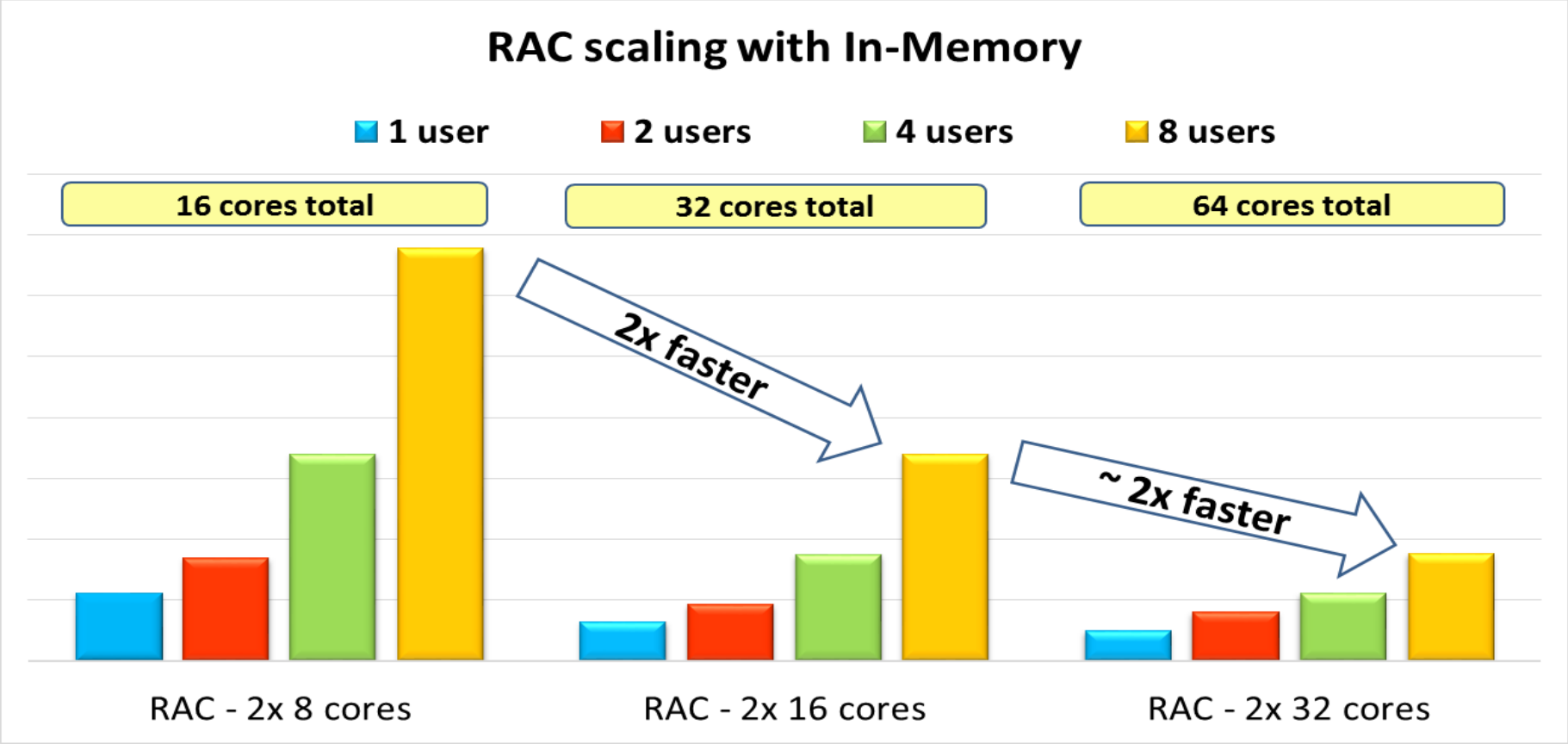
System resource considerations – example with 10TB analytics workload



Same 36 core count delivers 22x higher throughput with In-Memory

Disclaimer: Mileage may vary by workload

In-Memory with Oracle Real Application Cluster



- Two-node RAC cluster with IBM Power E850
- The range-hash partitioned table queried had 9 Billion rows in 1152 partitions and was loaded into the In-Memory area with “Query Low” compression.
- In-Memory data was distributed using the default distribution algorithm based on partitions.

Disclaimer: Mileage may vary by workload

Designed & Optimized for Big Data & Analytics performance

Power Systems

IBM

4X

threads per core vs. x86
(up to 1536 threads per system)

4X

memory bandwidth vs. x86
(up to 16TB of memory)

5X

more cache vs. x86
(up to 224MB cache per socket)

Processors
flexible, fast execution of analytics algorithms

Memory
large, fast workspace to maximize business insight

Cache
ensure continuous data load for fast responses

Optimized for a broad range of big data & analytics

UNSTRUCTURED	IN-MEMORY	STRUCTURED

IBM Systems and Storage leverage Oracle Software

Power Systems

IBM

FlashSystem 900

Power E880

Enhanced

DS8880

Storwize V7000

Power E870

Enhanced

Power Systems

for Business

Linux

AIX

Power E850

Power S824

Power S814

Power S822

POWER8

PowerKVM PowerVP PowerVM
PowerHA PowerVC PowerSC

Conclusion

- Oracle Database In-Memory provides strong benefits
  - Greatly reduced run time
    - Up to 20X improvement in execution time
  - More efficient use of processor resource
    - Same number of processors used for a smaller period of time
- IBM Power provides benefits running DB In-Memory
  - Large low latency memory system
    - Provides scalability for concurrent users and larger data
  - Powerful multi-threaded POWER8 processor architecture
    - Leverages the parallelism the Oracle DB In-Memory enables
  - IBM FlashSystem storage system
    - Provides superb performance for mixed OLTP and Analytics
    - The DB In-Memory feature reduces sequential I/O demand on system and SAN

Acknowledgments

The authors graciously thank:

- Ravisankar Shanmugam, for configuring the system, storage, Oracle 12c and executing the workload in many different configurations.
- Rebecca Ballough, for database expertise, identifying the workloads used in the project and establishing process for running the queries.
- Nilesh Shah, for his tireless efforts in setting up a POWER8 system to start the project.

Power Systems

IBM

Thank You!

Questions?

Reach out to the

IBM Oracle International Competency Center at

ibmoracle@us.ibm.cm