

Oracle Database In-Memory

Fast Analytics in Real-Time

North Central Oracle Apps User Group
August 1, 2019

Andy Rivenes
Database In-Memory Product Management
Oracle Corporation

Email: andy.rivenes@oracle.com
Twitter: [@TheInMemoryGuy](https://twitter.com/TheInMemoryGuy)
Blog: blogs.oracle.com/in-memory



ORACLE

Safe Harbor Statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Why Use Database In-Memory



Improved Reporting Performance

Faster Reports – No Application Changes

- Organizations can use Oracle reporting/analytical applications or existing 3rd party reporting tools
 - No application or data format changes required
- Improves performance (**10x typical**) of reporting applications with existing data warehouse and/or data marts
- Improves performance to ensure SLA's continue to be maintained
- Increases capacity of mixed workload environments to enable additional growth and performance



- Using Database In-Memory resulted in:
 - **Triple the volume of Data**
 - **No changes required to Business Objects reports**
 - **50X performance improvement on reports**
 - Reports that took days now return in less than 1 hour

Real-Time Analytics

Use Operational Data for Real-Time Analytics

- Real-time analytics on operational data directly -- without the time delay of moving data for reporting
- Enables real-time business intelligence at the point of contact
 - Delivers real-time insight, visibility and agility for critical business operations and decisions
- Enables real-time ad-hoc reporting /analysis and iterative drill-downs on operational data
- No application or data format changes required



Pricerite 實惠

- Using Database In-Memory resulted in:
 - **Analytic queries up to 5x faster**
 - **Real-time analytics dashboard**

Reduced Overhead

Faster Analytics -- Less Storage Overhead

- Analytic indexes can slow down the performance of transactional applications
 - Requires significantly more database storage (on costly tier 1 storage)
 - Increases overhead due to index maintenance
- Database In-Memory allows users to eliminate analytic reporting indexes – without impacting performance
- Removing the need for analytic reporting indexes greatly simplifies tuning and reduces ongoing administration

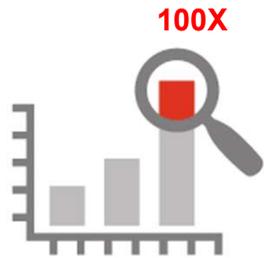


Walgreens

- Using Database In-Memory resulted in:
 - **Performance Gains: 1.8X to 12X**
 - **Space savings and reduced contention on DML by dropping analytic indexes**

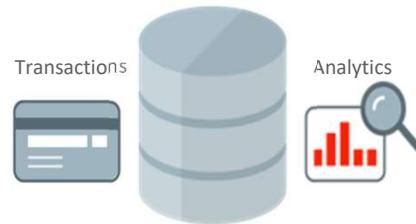
Oracle Database In-Memory

Real-Time Analytics



Enable Real-Time Business Decisions

Accelerate Mixed Workload



Run analytics on Operational Systems

Risk-Free



Proven Scale-Out, Availability, Security

Trivial to Implement



No Application Changes Not Limited by Memory

Oracle In-Memory: Simple to Implement

1. Configure Memory Capacity

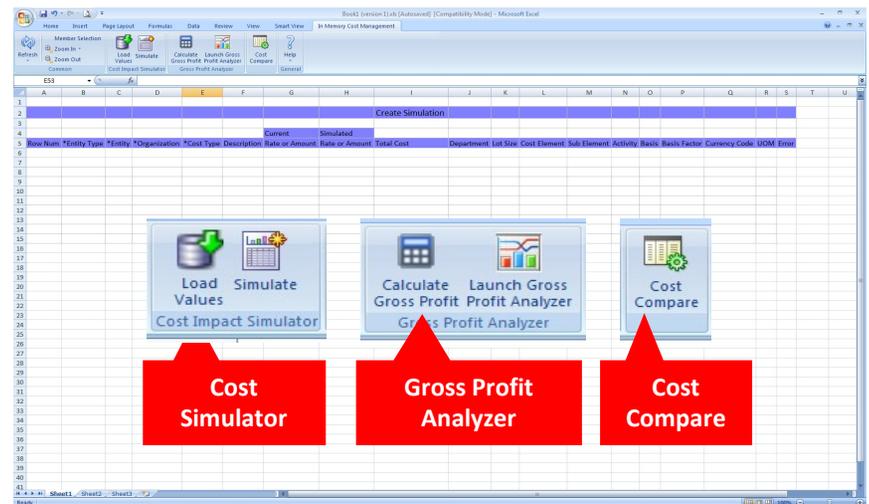
- `inmemory_size = XXX GB`

2. Configure tables or partitions to be in memory

- `alter table | partition ... inmemory;`

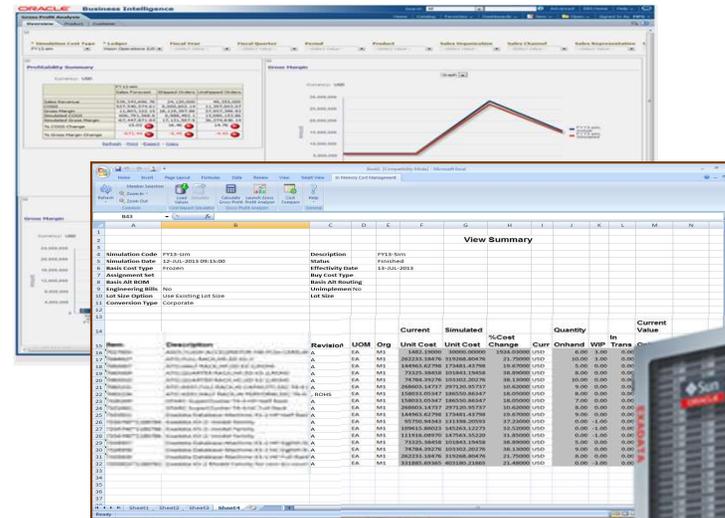
3. Later drop analytic indexes to speed up OLTP

E-Business Suite In-Memory Cost Management



In-Memory Cost Management

- Cost and Profitability analysis in Real-time
- Supports Process and Discrete industries
- Provides real time analysis
 - Optimize costs and working capital
 - Evaluate COGS and valuations
 - Maximizing margins and gross profits
- Spread sheet based UI
- Powered by Database In-Memory & Oracle Engineered Systems
 - 12c Database In-Memory option
 - Heap memory, Flash cache, write back flash cache
 - Hybrid columnar storage and Data compression
 - Process large volumes of cost data in real-time

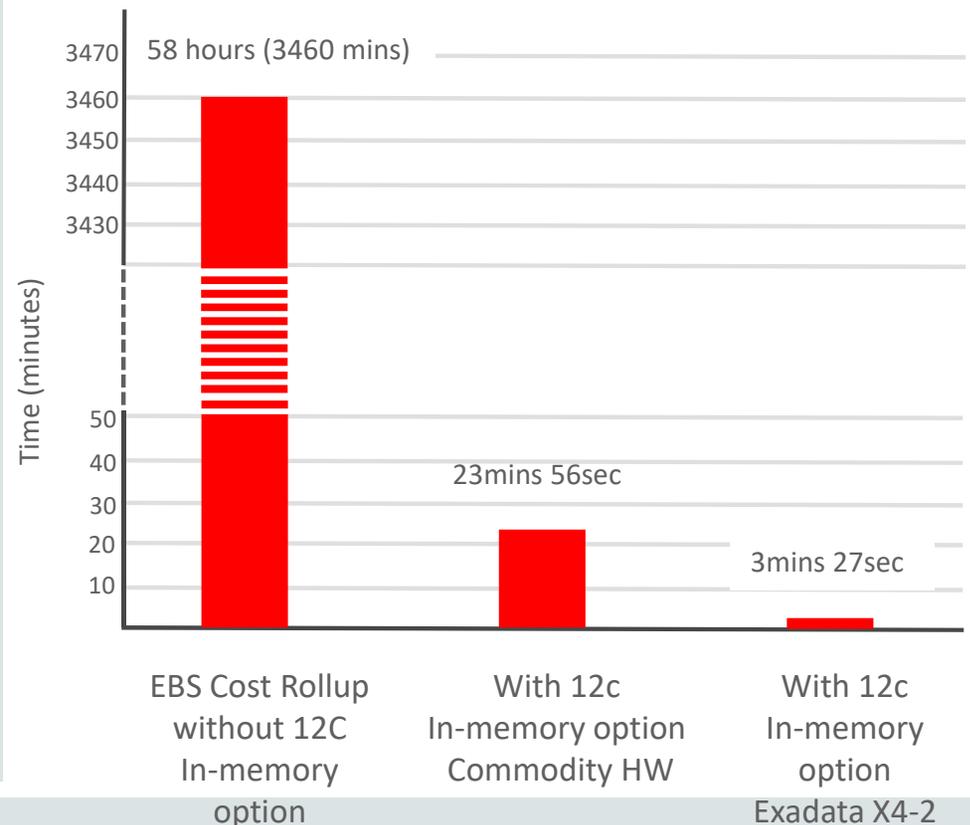


In-Memory Cost Management Performance Benchmarks

Key Benefits

- Significant performance gains with Oracle 12c Database In-Memory option
- Achieved 145x faster performance with 12c Database In-Memory option on commodity hardware
- Achieved over 1003x faster performance with Oracle 12c Database In-Memory option on Exadata X4-2

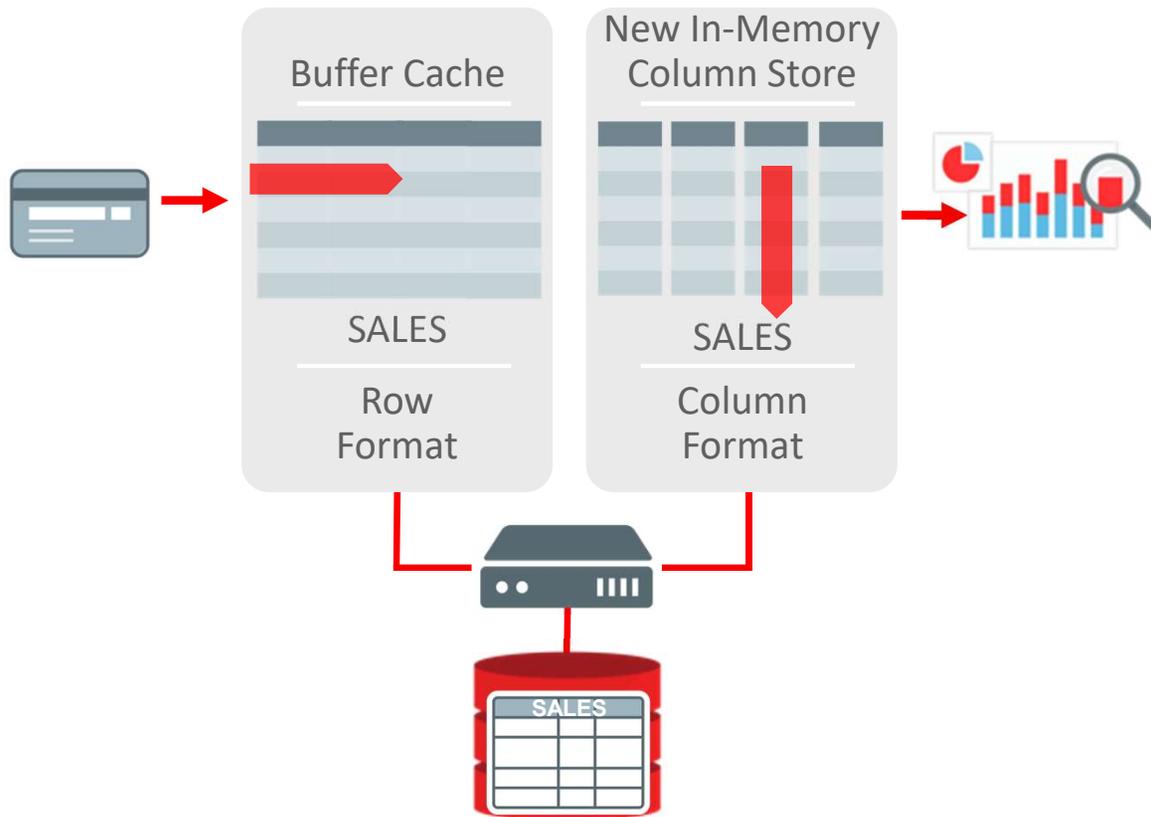
From 58hrs to 3mins 27sec



What Is Database In-Memory



Breakthrough: Dual Format Database



- **BOTH** row and column formats for same table
- Simultaneously active and transactionally consistent
- Analytics & reporting use new in-memory Column format
- OLTP uses proven row format

Why Columnar Formats?

- Only scan the columns involved in the query
- Columnar formats enable better compression
- Columnar data is vector oriented – takes advantage of SIMD
- Can skip portions of the data if outside value ranges - In-Memory storage indexes
- Oracle Database is capable of scanning billions of rows per second per core
- But don't forget, it does not accelerate DML – that's why we have both formats

Why In-Memory?

- Memory is an enabler, allowing the fastest scanning possible
 - Populating columnar data in-memory means not having to wait for I/O
- However, columnar formatted data can now be placed on any storage tier:
 - DRAM – In Oracle Database SGA
 - Flash – In Exadata flash cache
 - On-disk – Engineered systems Hybrid Columnar Compression (HCC)

Where Is It Available



Database In-Memory

- Database In-Memory is an option for Oracle Database Enterprise Edition
- Database In-Memory was included in the first patchset (12.1.0.2) for 12.1 and all subsequent Oracle Database releases
- Available:
 - Database Cloud Service – Virtual Machines: **Extreme Performance**
 - Database Cloud Service – Bare Metal: **Extreme Performance**
 - Exadata Cloud Service
 - Exadata Cloud at Customer
 - Autonomous Data Warehouse (Flash only)
 - On-premises

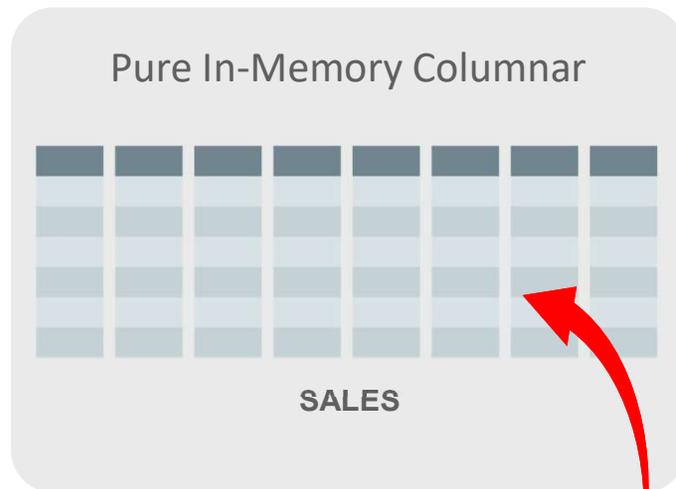


Note: Database In-Memory is **not** enabled by default

How Does Database In-Memory Work

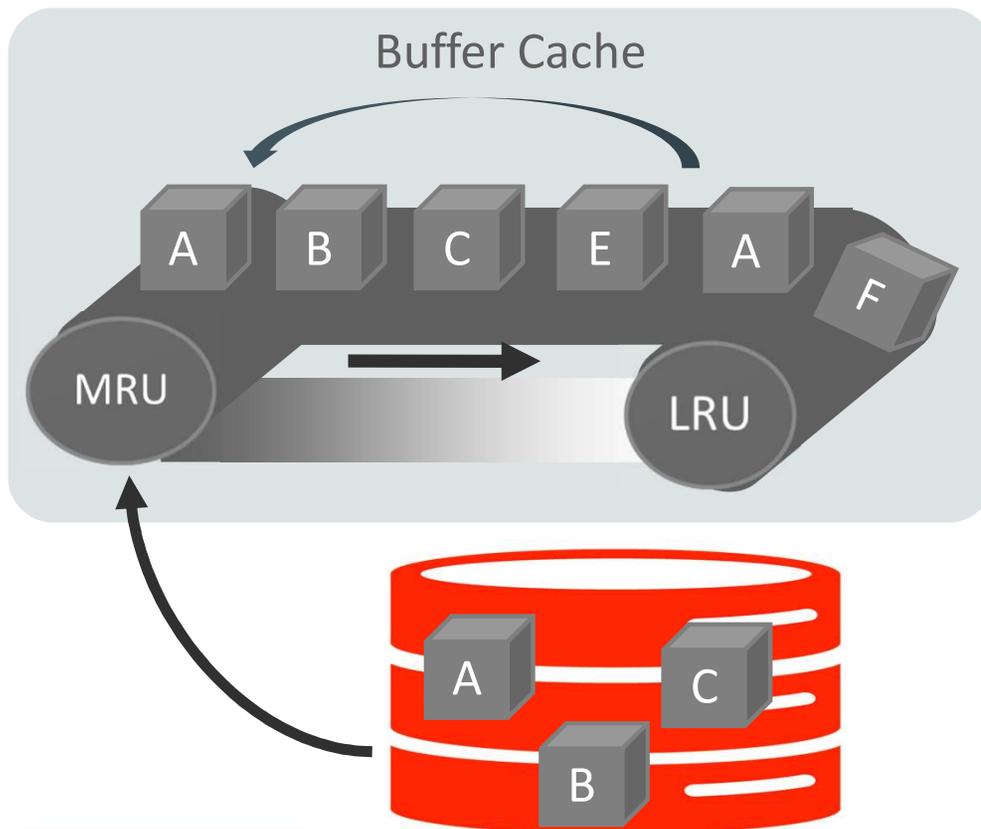


Oracle In-Memory Columnar Technology



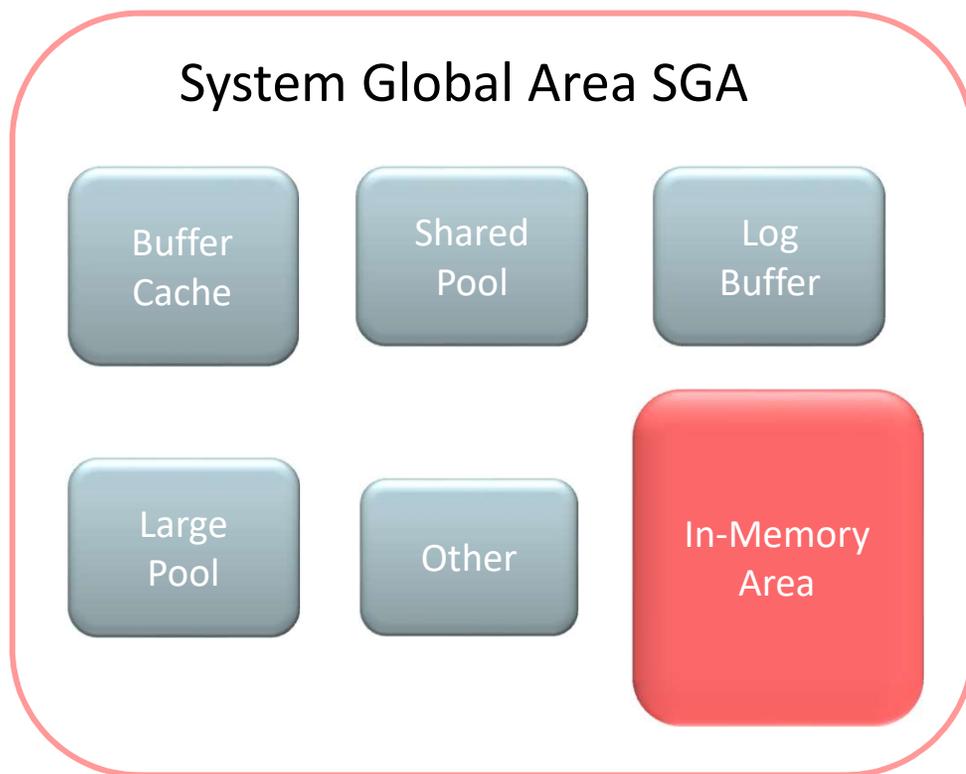
- Pure in-memory columnar format
 - Not persistent, so no undo or redo is generated
- Can be enabled for table, partition, subpartition or materialized view
- 2x to 20x compression typical
- Available on all hardware platforms

In-Memory A Store – **Not A Cache**



- What is a cache?
- A pool of memory
- Data automatically brought into memory based on access
- Data automatically aged out
- Good example:
Oracle Database Buffer Cache

In-Memory Area: **New Static Area within SGA**



- Contains data in the new In-Memory Column Format
- Controlled by INMEMORY_SIZE parameter
 - Minimum size of 100MB
- Can dynamically grow larger (12.2)
- SGA_TARGET must be large enough to accommodate this area



Note: Don't steal Memory from other components

Population

- Order in which objects are populated controlled by PRIORITY subclause:

```
ALTER TABLE sales  
INMEMORY PRIORITY HIGH;
```

- Levels:
 - CRITICAL > HIGH > MEDIUM > LOW
 - Controls order (not speed) of populate
- Default PRIORITY is NONE
 - Populate only on first access

- Population completed by new set of background processes

ora_w001_orcl

- Number of processes controlled by parameter:

INMEMORY_MAX_POPULATE_SERVERS

```
oracle@srv80101:~/In_Memory_Beta/lesson4  
top - 15:32:09 up 7 days, 23:45, 7 users, load average: 14.72, 4.14, 1.55  
Tasks: 622 total, 36 running, 586 sleeping, 0 stopped, 0 zombie  
Cpu(s): 96.7%us, 1.9%sy, 0.0%ni, 0.0%id, 1.2%wa, 0.0%hi, 0.1%si, 0.0%st  
Mem: 148834648k total, 146686500k used, 2148148k free, 187748k buffers  
Swap: 2096440k total, 92k used, 2096348k free, 131648316k cached  
PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND  
24673 oracle 20 0 120g 1.7g 1.6g R 79.0 1.2 6:13.27 ora_w014_orcl  
24569 oracle 20 0 120g 2.5g 2.4g R 76.7 1.7 9:15.98 ora_w003_orcl  
24663 oracle 20 0 120g 1.7g 1.7g R 74.4 1.2 6:32.98 ora_w00z_orcl  
24627 oracle 20 0 120g 2.0g 1.9g R 73.1 1.4 7:57.44 ora_w00n_orcl  
24625 oracle 20 0 120g 2.2g 2.1g R 72.4 1.5 8:42.75 ora_w00n_orcl  
24667 oracle 20 0 120g 2.0g 1.9g R 72.1 1.4 7:31.26 ora_w011_orcl  
24571 oracle 20 0 120g 2.5g 2.3g R 71.8 1.8 9:32.78 ora_w004_orcl  
24657 oracle 20 0 120g 1.8g 1.7g R 71.1 1.3 6:41.06 ora_w00w_orcl  
24669 oracle 20 0 120g 2.2g 2.1g R 70.8 1.6 8:56.33 ora_w012_orcl  
24683 oracle 20 0 120g 1.7g 1.7g R 70.5 1.2 6:46.73 ora_w018_orcl  
24621 oracle 20 0 120g 2.0g 1.9g R 70.1 1.4 8:12.00 ora_w001_orcl  
24687 oracle 20 0 120g 1.9g 1.8g R 70.1 1.4 7:58.64 ora_w019_orcl  
24611 oracle 20 0 120g 2.3g 2.0g R 69.8 1.6 8:13.25 ora_w00g_orcl  
24619 oracle 20 0 120g 1.9g 1.8g R 68.5 1.3 6:36.23 ora_w00k_orcl  
24671 oracle 20 0 120g 1.9g 1.8g R 68.2 1.3 7:13.82 ora_w013_orcl  
24675 oracle 20 0 120g 1.7g 1.6g R 67.5 1.2 6:42.18 ora_w015_orcl  
24659 oracle 20 0 120g 1.9g 1.8g R 67.2 1.3 6:53.53 ora_w00x_orcl  
24631 oracle 20 0 120g 2.3g 2.3g R 66.9 1.6 9:28.48 ora_w00p_orcl  
24654 oracle 20 0 120g 1.8g 1.7g R 66.9 1.3 6:57.75 ora_w00v_orcl
```

Database In-Memory Technology

Scanning and filtering data more efficiently

Columnar Format



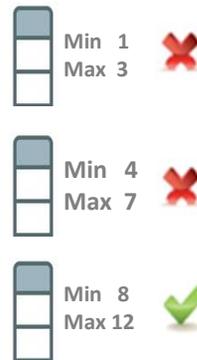
Access only the columns you need

Compression



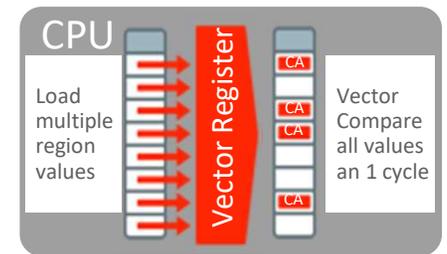
Scan & filter data in compressed format

Storage Indexes



Prune out any unnecessary data from the column

SIMD Vector Processing



Process multiple column values in a single CPU instruction

Optimizer Enhancements

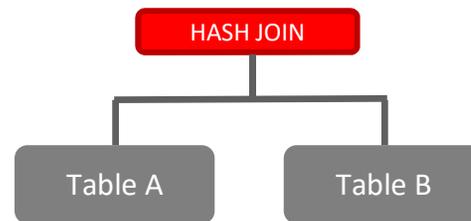
Improves all aspects of analytic queries

Data Scans



- Speed of memory
- Scan and Filter only the needed Columns
- Vector Instructions

Joins



- Convert Star Joins into 10X Faster Column Scans
- Search large table for values that match small table

In-Memory Aggregation



- Create In-Memory Report Outline that is Populated during Fast Scan
- Runs Reports Instantly

How Much Memory Do You Need



Oracle In-Memory Advisor

Workload Database Usage

Total Database Time (Seconds)	Analytics Processing Time (Seconds)	Analytics Processing Percentage
2990	2640	88%

In-Memory Size	Percentage of Maximum SGA Size (100.0GB)	Estimated Analytics Processing Time Reduction (Seconds)	Estimated Analytics Processing Performance Improvement Factor
9.141GB	9%	2102	4.9X
8.684GB	9%	2101	4.9X
8.226GB	8%	2101	4.9X
7.769GB	8%	2100	4.9X

- In-Memory Advisor – free download available on OTN for 11.2.0.3+ DBs
- Analyzes existing DB workload via AWR & ASH repositories
- Provides list of objects that would benefit most from being populated into IM column



Note: Database Tuning Pack license required

Oracle Compression Advisor And In-Memory

```
DECLARE
  l_blkcnt cmp          PLS_INTEGER;
  l_blkcnt uncmp       PLS_INTEGER;
  l_row cmp           PLS_INTEGER;
  l_row uncmp         PLS_INTEGER;
  cmp_ratio           PLS_INTEGER;
  l_comptype str      VARCHAR2(100);
  comp_ratio allows NUMBER := -1;
BEGIN
  dbms_compression.Get_compression_ratio (
    scratchtbsname => 'TS_DATA',
    ownname        => 'SSB',
    objname        => 'LINEORDER',
    subobjname     => NULL,
    comptype       => dbms_compression.comp_inmemory_query_low,
    blkcnt cmp     => l_blkcnt cmp,
    blkcnt uncmp   => l_blkcnt uncmp,
    row cmp        => l_row cmp,
    row uncmp      => l_row uncmp,
    cmp_ratio      => cmp_ratio,
    comptype str   => l_comptype str,
    subset numRows => dbms_compression.comp_ratio allows);
  dbms_output.Put_line('The IM compression ratio is ' || cmp_ratio);
END;
```

- Easy way to determine memory requirements
- Use DBMS_COMPRESSION
- Applies MEMCOMPRESS to sample set of data from a table
- Returns estimated compression ratio

What If You Don't Have Enough Memory



Compression

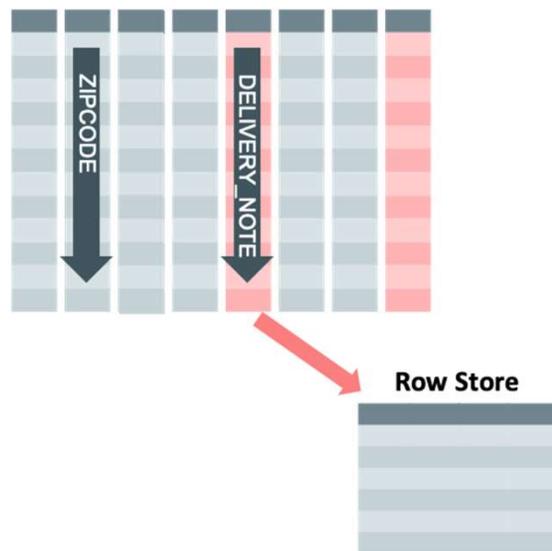
```
ALTER MATERIALIZED VIEW mv1  
INMEMORY  
MEMCOMPRESS FOR QUERY LOW;
```

```
CREATE TABLE trades  
  (Name varchar(20) ,  
   Desc varchar(200))  
INMEMORY  
MEMCOMPRESS FOR DML(desc) ;
```

- Objects compressed during population
- New compression techniques
 - Focused on scan performance
- 2x to 20x compression typical
- Multiple levels of compression
 - FOR DML
 - FOR QUERY LOW/HIGH
 - FOR CAPACITY LOW/HIGH
- Possible to use a different level for different partitions in a table

Columns Can Be Excluded

```
ALTER TABLE sales INMEMORY  
NO INMEMORY (delivery_note) ;
```



- You don't have to populate all columns
 - But, if excluded columns are accessed then the query will run against the row-store
- Two phase approach
 1. INMEMORY attribute on table automatically inherited by columns
 2. Need to remove attribute from the columns you don't want populated

Why Not Just Cache The Table In The Buffer Cache



Compare Column-store to Row-store

```
SQL> -- In-Memory Column Store query
SQL>
SQL> select max(lo_ordtotalprice) most_expensive_order From LINEORDER;
```

MOST_EXPENSIVE_ORDER

57346348

Elapsed: 00:00:00.01

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT				5401 (100)	
1	SORT AGGREGATE		1	6		
2	TABLE ACCESS INMEMORY FULL	LINEORDER	59M	343M	5401 (16)	00:00:01

```
SQL> -- Buffer Cache query with the column store disabled via NO_INMEMORY hint
SQL>
SQL> select /*+ NO_INMEMORY */ max(lo_ordtotalprice) most_expensive_order From LINEORDER;
```

MOST_EXPENSIVE_ORDER

57346348

Elapsed: 00:00:08.38

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT				123K(100)	
1	SORT AGGREGATE		1	6		
2	TABLE ACCESS FULL	LINEORDER	59M	343M	123K (1)	00:00:05

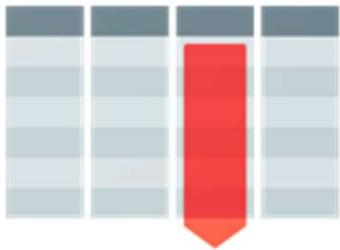
How Do You Tell If It's Working



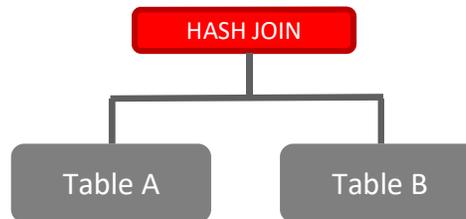
Which Queries Benefit From Database In-Memory?

For a non-trivial amount of rows and execution time, when a significant amount of time ...

is spent accessing data



is spent joining data



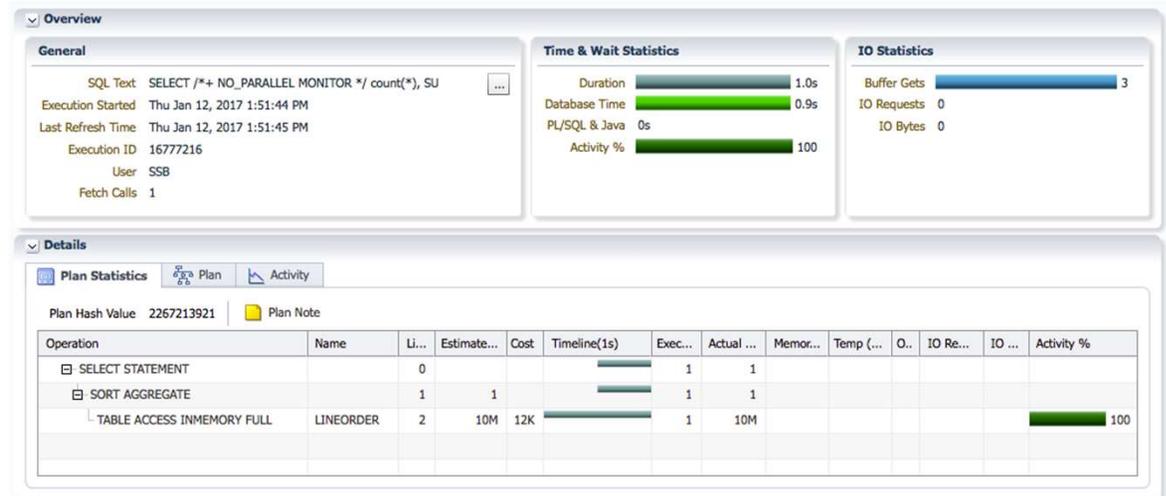
is spent aggregating data



Use Time Based Analysis Techniques To Evaluate Benefit

SQL Monitor Active Reports

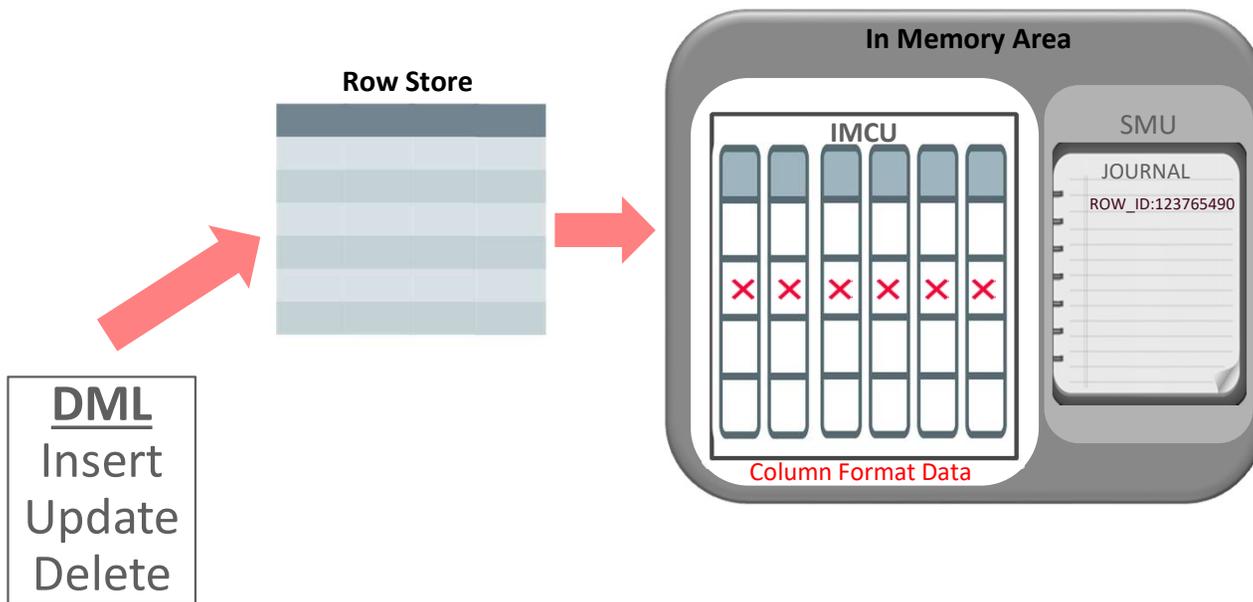
- Shows how SQL was executed and where **time was spent**
- See blogs.oracle.com/In-Memory for a technical brief on creating SQL Monitor active reports



How Does It Handle Data Changes

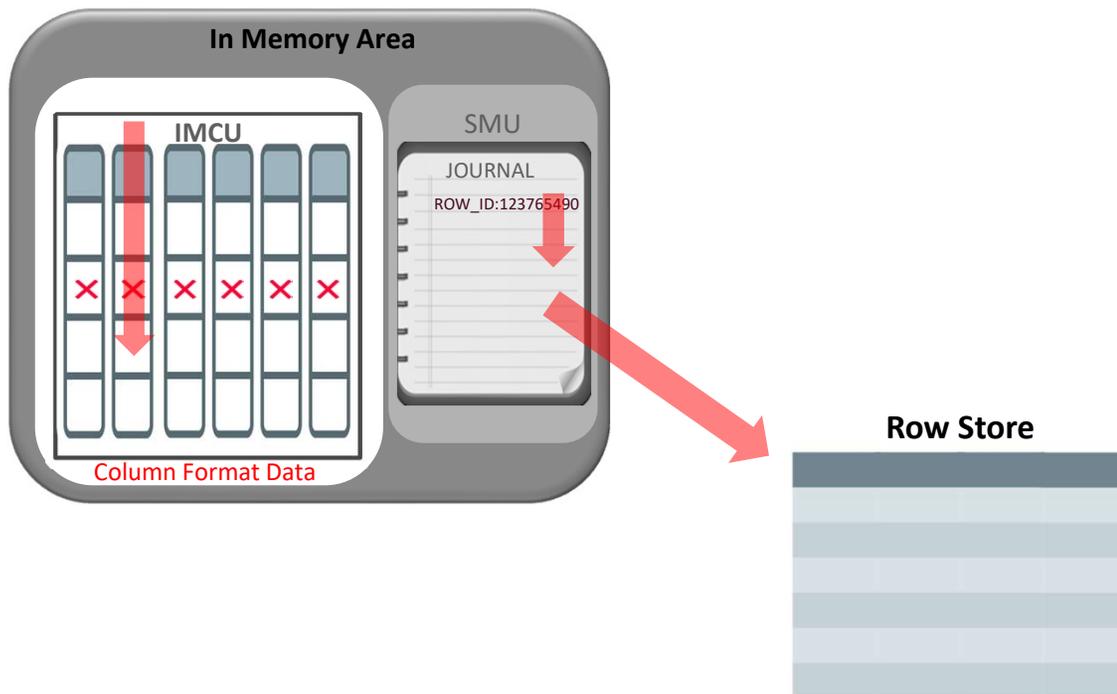


DML and the In-Memory Column Store



- DML operations processed in row store just as they are today
- Corresponding entry in column store marked stale as of SCN
- ROWID of row stored in the Transaction Journal

DML and the In-Memory Column Store



- In-Memory Column Store is never out of date
- Read-consistency is achieved by merging contents of column, the transaction journal and buffer cache
- Pure In-Memory format means no undo/redo to repopulate IMCUs

Will Response Time Suffer

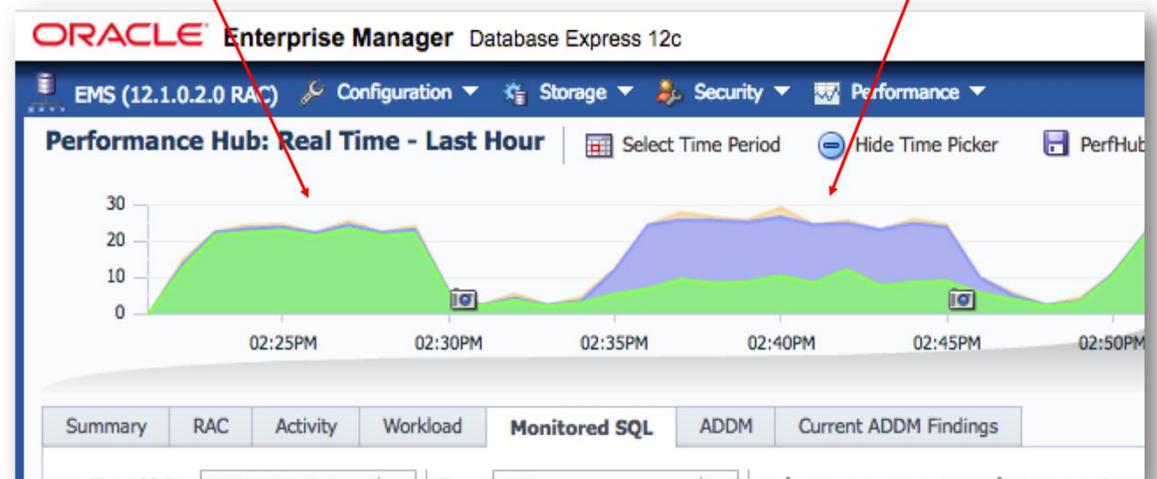


Real World Application Demo – [DBIM YouTube Channel](#)

- Database In-Memory enables real time analytics
- OLTP transaction workload with Database In-Memory
- **Workload is not allowed to exceed the capacity of the machine**
 - **OLTP workload** (simulates 6400 users each running a transaction every 10 seconds)
 - **OLAP workload** (20 connections running a stream of 100 randomly selected reports with no think time)

OLAP (IM Enabled) + OLTP

OLAP (IM Disabled) + OLTP



DBIM enabled – Maximum throughput, no OLTP degradation

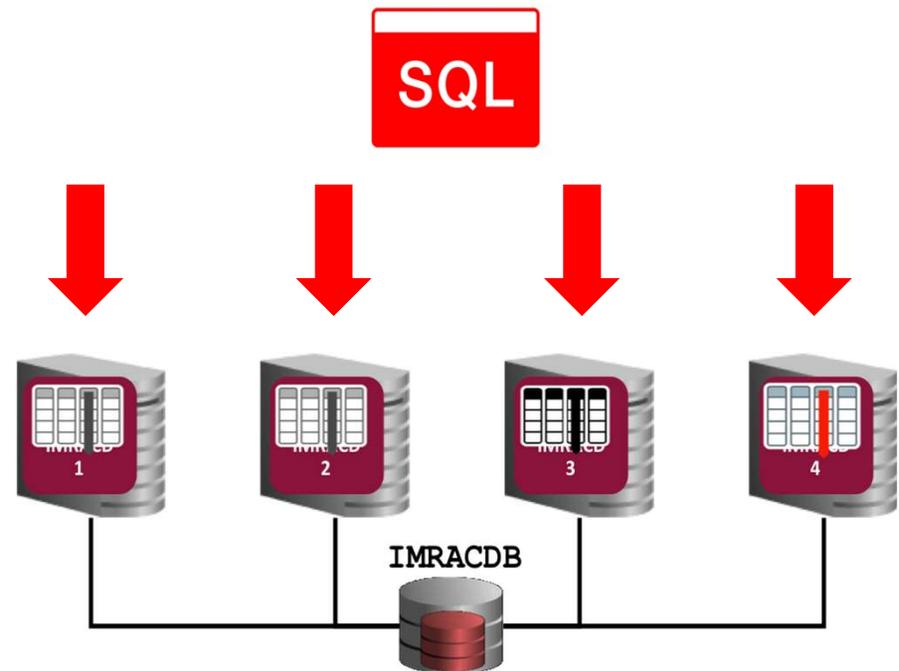
DBIM disabled – I/O required, fewer reports, OLTP impacted

How does Database In-Memory Work With RAC



RAC : Scale-Out In-Memory Database to Any Size

- Scale-Out across servers to grow memory and CPUs
- Shared nothing architecture
- IMCUs not shipped across interconnect – cache fusion is not in play!
- In-Memory **queries are parallelized** across servers to access local column data



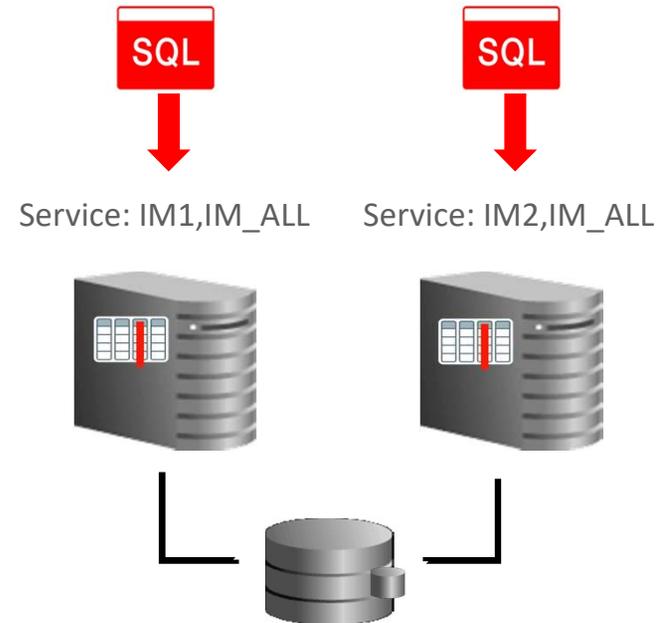
Scale Out: **Distribute For Service**

- Used for Active Data Guard environments to control placement based on where the service is active
 - Honors service location when a role change or switchover occurs
- Also useful for populating to a subset of nodes in a RAC environment

Independent Column Stores

Controlling Object Placement

- Services allow simplified object placement
- Both connections and PQ honor service connection
- Using the FOR SERVICE subclause and service based connections, object access can be limited to the services defined



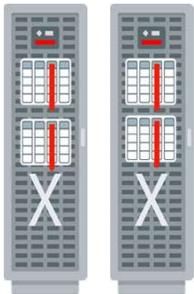
```
srvctl add service -db orcl -s IM1 -preferred "rac1"  
srvctl add service -db orcl -s IM2 -preferred "rac2"  
srvctl add service -db orcl -s IM_ALL -preferred "rac1,rac2"  
  
srvctl start service -db orcl -service "IM1,IM2,IM_ALL"
```

Does It Work With Other Oracle Database Features



Database In-Memory: Other Features

Scale-Out



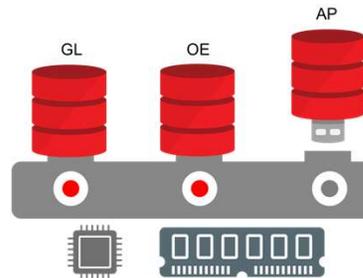
- Scale-Out Across Servers to Grow Memory and CPUs
- In-Memory Queries Parallelized Across Servers

Scale-Up



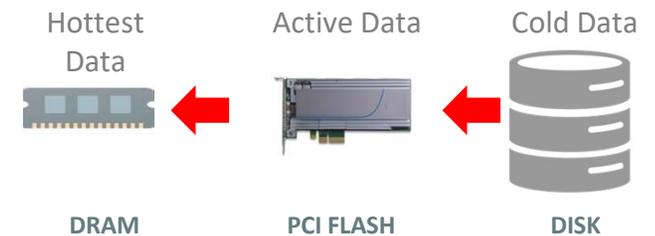
- Scale-Up on large SMPs
- NUMA Optimized

Consolidation



- Frees up memory and CPU
- Shares memory and background processes
- Column store defined at CDB level

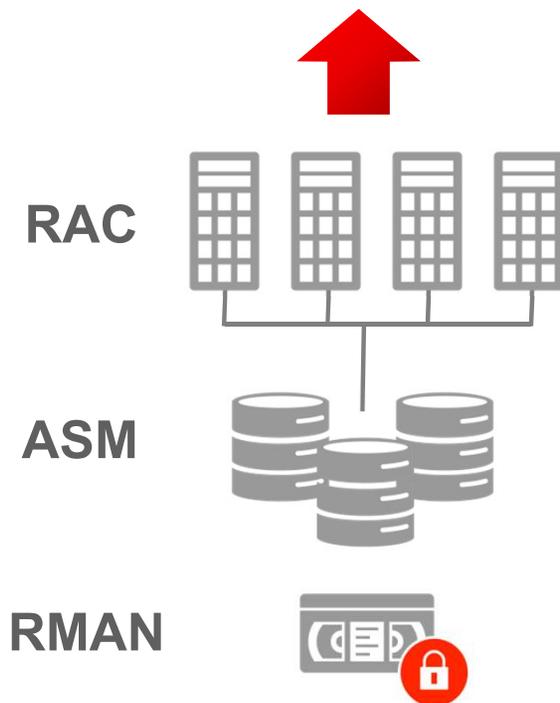
Combine with Flash and Disk



- Easily place data on most cost effective tier
- Simultaneously Achieve:
 - **Speed** of DRAM
 - **I/Os** of Flash
 - **Cost** of Disk

Database In-Memory: **Industrial Strength Availability**

Data Guard & GoldenGate



- Pure In-Memory format does not change Oracle's storage format, logging, backup, recovery, etc.
- All Oracle's proven availability technologies work transparently
- **Protection from all failures**
 - Node, site, corruption, human error, etc.

What's New

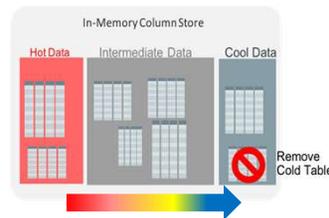


Database In-Memory New Features



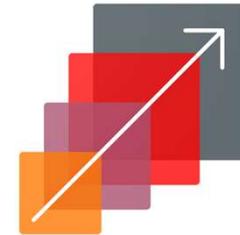
Performance

- In-Memory Expressions
- Join Groups
- In-Memory Dynamic Scans
- In-Memory Optimized Arithmetic



Managability

- Automatic Data Optimization
- Automatic In-Memory



Expanded Capacity

- Exadata Flash
- Active Data Guard
- External Tables

How Do I Get Started



https://blogs.oracle.com/in-memory/dbim-resources

Oracle Database In-Memory

Enabling Real-Time Analytics with Database In-Memory

Try Oracle Cloud Platform For Free

FEATURED POST

Database In-Memory Columnar Format in Exadata Flash →
Exadata Flash Cache with DBIM

Visit the Oracle Developers Portal
Get the latest technical articles, resources, and more.

Recent Posts

```
o_extendedprice * lo_discount) reve  
rder l,  
dim d
```

Join Processing with Database In-Memory →
Join processing with Database In-Memory

```
orderdate = d.d_datekey  
.lo_discount BETWEEN 2 AND 3  
.lo_quantity < 24  
.d_date = 'December 24, 1995';
```

IOUG Webinar Series Continues →
Third IOUG Webinar on Identifying the Right Workload for Database In-Memory

Oracle Database In-Memory Blog

Enterprise Data Architecture

Application Information

- Oracle Database In-Memory with Oracle's JD Edwards EnterpriseOne
- Performance Characterization of Oracle JD Edwards EnterpriseOne with Oracle Database In-Memory
- Using Oracle Database In-Memory with Oracle E-Business Suite

Videos

- Tirthankar Lahiri presents In-Memory Database Architectures at the IMC Summit
- Database In-Memory Real-Time Analytics Demo
- OOW 2017 DBIM Temperature Demo
- Oracle Database In-Memory: What's new in 12.2
- Customers discuss the benefits of Oracle Database In-Memory
- Database In-Memory Introduction
- Juan Loaiza explains everything you need to know about Oracle Database In-Memory using the Wikipedia demo
- Juan Loaiza explains how Software-in-Silicon will optimize Oracle Database In-Memory at Oracle Openworld 2014
- Larry Ellison Launches Database In-Memory
- Oracle ACE Directors Talk about Oracle Database In-Memory
- Oracle Application teams explain how Database In-Memory will improve their performance
- Panel discussion on Oracle Database In-Memory hosted by Juan Loaiza

ORACLE

Additional Resources



Join the Conversation

- https://twitter.com/db_inmemory
- <https://twitter.com/TheInMemoryGuy>
- <http://www.oracle.com/goto/dbim.html>

Database In-Memory Information

[Database In-Memory Blog](#)

[oracle.com – Database In-Memory](#)

[Database In-Memory YouTube Channel](#)

[Ask TOM Database In-Memory Office Hours](#)

[Database In-Memory Guide \(Documentation\)](#)