Birds of a Feather Session: Best Practices for TimesTen on Exalytics

Chris Jenkins
Senior Director, In-Memory Technology, Oracle
Antony Heljula
Technical Director, Peak Indicators Ltd.
Mark Rittman
CTO, Rittman Mead
Program Agenda

- Introduction
- Presentation / Discussion
- Q & A
Introduction

BoF: Best Practices for TimesTen on Exalytics

- This is a ‘Birds of a Feather’ session
  - An opportunity for sharing experiences and discussion rather than just presentations
- Let’s be interactive!
- Please ask questions and share your thoughts and experiences
  - Both during the presentations and afterwards
Antony Heljula
Technical Director, Peak Indicators Ltd.

- Widely regarded as one of Europe’s top Oracle BI architects
- Over 10 years delivery experience with OBIEE and OBI Applications
- Over 15 years experience with Oracle Data Warehousing products
- Also specializes in SOA, maps/spatial and performance tuning
Mark Rittman
CTO, Rittman Mead

- CTO and co-founder of Rittman Mead
  - Oracle Gold Partner with offices in UK, USA, India and Australia
- Over 15 years experience with Oracle BI, data warehousing and database technologies
- Oracle ACE Director and author of *Oracle Business Intelligence Developer’s Guide* (Oracle press)
- Writes regularly for the Rittman mead blog and *Oracle Magazine*
  [http://www.rittmanmead.com/blog](http://www.rittmanmead.com/blog)
Program Agenda

- Introduction
- Presentation / Discussion
- Q & A
What is Oracle TimesTen for Exalytics?

• An in-memory relational database, acquired by Oracle in 2005
• Data held entirely in-memory, with recovery and persistence to disk
• Fully-featured RDBMS with Oracle DB integration and compatibility
• Used within the BI context in two main roles
  ‣ As the in-memory aggregate cache within Oracle Exalytics
  ‣ As a way of holding “hot data” to speed-up detail-level reporting
• Licensed separately to OBIEE and Essbase within Exalytics, but a prerequisite for Exalytics deployments
• The “secret sauce” behind Exalytics’ “speed-of-thought” analysis
Common Deployment Scenarios

1) “Aggregate Persistence”

- Oracle BI comes with a unique feature where it automatically builds and deploys in-memory aggregates into TimesTen

- Exalytics customers benefit from the “Summary Advisor” which suggests aggregates to build (based on previous user activity)

- The use of aggregates is transparent:
  - OBI directs summary queries to TimesTen
  - A more detailed query will go to the underlying Data Warehouse
Common Deployment Scenarios

2) Operational Data Store (ODS)

- TimesTen can be used as an ODS where data is cached into memory for optimising real-time / operational reports

- The TimesTen database can be loaded via various means including:
  - DAC
  - Oracle Data Integrator (ODI)
  - ttImportFromOracle
  - Custom scripts / application code
  - Oracle GoldenGate
Common Deployment Scenarios

3) *Fragmentation*

- Oracle BI comes with a “Fragmentation” feature where it can effectively union data from different data sources together.

- The end user sees it as a single data source:
  - In this example, we have cached “Current” data into TimesTen.

- Depending on the query, Oracle BI is intelligent enough to only use the data sources that are required:
  - A query for “Current” data will only use TimesTen.
  - A query for “Historical” data will only use Oracle DB.
  - A query across all data will issue concurrent requests to both databases.
Configuration Best Practices

• Use “Large Pages” even if your database is < 256GB
  • More efficient memory utilisation
  • Needs OS and TimesTen to be configured to use them

• Understand the difference between RANGE and HASH indexes (as explained later by Mark)
  • OBI Aggregate Persistence only creates RANGE indexes, so you may need to create as HASH

• Use the appropriate TimesTen data types – especially with Numbers/Integers e.g. use TT_TINYINT/TT_SMALLINT/TT_INTEGER/TT_BIGINT instead of NUMBER
  • Use the ttImportFromOracle tool to help with this

• Use the Index Advisor and Explain Plan tools – they are really good!
Configuration Best Practices

*Continued...*

- **Optimise your I/O throughput**
  - Store your TimesTen database on Flash
  - Set LogBufMB = 1024 to increase log buffer size (for optimising data loads etc.)

- **Keep up to date with TimesTen releases**
  - Significant performance improvements come in every release
  - Everyone should be on 11.2.2.5.1 now

- **Set TTC_TIMEOUT=0 on your clients so there is no timeout**
  - Especially important with Oracle BI “Aggregate Persistence”

- **Make sure your tables are sufficiently analysed**
  - call ttOptUpdateStats('owner.name', 1); after changes in data volume or adding new indexes
Configuration Best Practices

Continued...

• Set “RAM Policy” to manual to ensure data is always loaded in-memory
  • Avoids long startup time
  • But remember to explicitly startup the database after e.g. a reboot

• You rarely need to set TEMPSIZE = PERMSIZE
  • 400GB / 400GB for Exalytics is not mandatory
  • Usually TEMPSIZE << PERMSIZE
  • Use “dssize” command in tIsql to find high-water mark for sizing TEMPSIZE
Loading and Refreshing TimesTen Databases

• TimesTen database loaded, and refreshed, from main data source
  ▶ Data source is typically a data warehouse
  ▶ Or detail-level application databases

• Exalytics’ Summary Advisor automates the load / refresh process for aggregates
  ▶ Uses BI Server (Aggregate Persistence) as the ETL tool
    - Supports all OBIEE data sources
    - RPD integration
  ▶ Manages aggregate lifecycle
  ▶ ‘Incremental’ refreshes using Summary Advisor can be inefficient
Loading and Refreshing TimesTen Databases (cont)

• Various methods for manually loading TimesTen databases
  ▶ Including the `ttImportFromOracle` utility

• Need to consider various factors
  ▶ How “hands-off” you want the process to be
  ▶ The available time window for the refresh
  ▶ What tools you want to use
    - E.g. Oracle Data Integrator, DAC
  ▶ Whether the data source is Oracle, or non-Oracle
  ▶ Is TimesTen being used with BI Applications
TimesTen for Exalytics Data Loading Best Practices

• For Summary Advisor aggregates
  › Table creation (DDL) and data load (DML) through SA
  › Enable parallel index creation
    - opmn.xml config file

• Subsequent refreshes of the same tables
  › Incremental load using POPULATE and INACTIVE_SCHEMAS logical SQL clauses
    - Still uses BI Server ETL, but no RPD editing + partial refresh
TimesTen for Exalytics Data Loading Best Practices (cont)

• If loading TimesTen within the context of the BI Apps
  ‣ Consider using DAC 11g’s Data Copy feature
  ‣ Replicates whole BI Apps DW tables into TimesTen
    - recommends optimal datatypes, compression etc.

• For detail-level, “hot data” replication from Oracle into TimesTen
  ‣ Use the `ttImportFromOracle` utility
    - Recommends optimal datatypes and compression clauses
    - Parallel data load of regular and compressed tables
  ‣ **Important** - plan for additional index analysis step after the data load based on actual query patterns
    - `ttImportFromOracle` replicates existing Oracle table indexes, may not reflect actual user queries

• For non-Exalytics non-Oracle source data, use `ttIsql`, `ttBulkCp` or similar
  ‣ Or a full ETL tool such as Oracle Data Integrator - additional license cost
TimesTen Table Indexing

• Just like regular Oracle RDBMS tables, TimesTen tables benefit from indexing
• The Summary Advisor, and `ttImportFromOracle`, both create indexes on the tables they create
  ‣ `ttImportFromOracle` creates indexes based on the existing ones on the Oracle source
  ‣ Summary Advisor creates indexes based on hierarchy keys and attributes
• In most cases, you’ll need to optimize these indexes to reflect real-world query patterns
  ‣ Can improve query performance significantly
  ‣ Avoids on-the-fly index creation
  ‣ TmpHashScan and TmpRangeScan operations in query plans
    [link: http://www.rittmanmead.com/2013/08/optimizing-timesten-for-exalytics-queries-using-the-timestens-index-advisor/]
TimesTen Indexing Best Practices

• Built-in utility for index optimization: TimesTen Index Advisor
  › Start with the default created indexes
  › Use Index Advisor to capture a representative workload
  › Generate new index recommendations, and implement
  › Update optimizer statistics
  › Repeat until no new recommendations are generated

• Consider creating HASH indexes rather than default RANGE index for columns used in equality (=) joins
  › Alter DDL to **CREATE HASH INDEX**...
  › Can be 30%-40% faster than RANGE indexes for equi-joins and equality lookups

```sql
Command> call ttindexAdviceCaptureStart(1,0);
Command> call ttindexAdviceCaptureEnd(1);
Command> call ttindexAdviceCaptureOutput(1);
< 33, create hash index SALES_i6 on SH.SALES(CHANNEL_ID); >
< 2, create hash index SALES_i10 on SH.SALES(PROD_ID); >
< 1, create unique hash index SALES_i12 on SH.SALES(TIME_ID); >
< 1, create hash index SALES_i12 on SH.SALES(TIME_ID); >
< 16, create hash index PRODUCTS_i7 on SH.PRODUCTS(PROD_ID,PROD_CATEGORY); >
< 1, create hash index PRODUCTS_i13 on SH.PRODUCTS(PROD_ID,PROD_SUBCATEGORY); >
< 35, create hash index TIMES_i8 on SH.TIMES(TIME_ID,CALENDAR_YEAR); >
< 1, create index TIMES_i11 on SH.TIMES(CALENDAR_YEAR); >
< 8, create unique hash index CUSTOMERS_i9 on SH.CUSTOMERS(CUST_ID); >
< 1, create hash index CUSTOMERS_i9 on SH.CUSTOMERS(CUST_ID); >
```
Compression

Know when and when not to use it!

- Think of "Compression" as a space/memory saving feature which you may also give performance improvement

- The Compression feature is actually "data de-duplication"
  - There is no point using Compression on primary keys
  - Never do Compression on decimal point numbers

- By default you will get a 4-byte "pointer size"
  - Use the "MAXVALUES" parameter to specify number of distinct values – you might get a smaller 1-byte or 2-byte pointer size (will be more efficient)

- Compression is best suited for larger columns containing >4 bytes of text. If columns contain less than 4 bytes of text then Compression is not worthwhile

- Use the `ttImportFromOracle` tool to generate the optimal scheme
Program Agenda

- Introduction
- Presentation / Discussion
- Q & A
Other TimesTen In-Memory Database Sessions

<table>
<thead>
<tr>
<th>Session / Time / Location</th>
<th>Session / Time / Location</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Tuesday</strong></td>
<td><strong>Tuesday</strong></td>
</tr>
<tr>
<td>Application-Tier In-Memory Database for Performance and High Availability</td>
<td>Developing Applications to Take Advantage of In-Memory Database Technology</td>
</tr>
<tr>
<td>▪ 12:00 PM – 1:00 PM</td>
<td>▪ 12:00 PM – 1:00 PM</td>
</tr>
<tr>
<td>▪ Moscone North 131</td>
<td>▪ Marriott Marquis - Salon 8</td>
</tr>
<tr>
<td><strong>Wednesday</strong></td>
<td><strong>Wednesday</strong></td>
</tr>
<tr>
<td>Real-Life Stories: Extreme Performance with In-Memory Database Technology</td>
<td></td>
</tr>
<tr>
<td>Customer Panel</td>
<td></td>
</tr>
<tr>
<td>▪ 1:15 PM - 2:15 PM</td>
<td></td>
</tr>
<tr>
<td>▪ Moscone South – 252</td>
<td></td>
</tr>
</tbody>
</table>

*Visit us at the DEMOgrounds! Moscone South, Booths SL-011 and SL-012*
Hardware and Software

Engineered to Work Together