Tools, tips, and strategies to optimize BEx query performance for SAP HANA

Pravin Gupta
TekLink International
In This Session

- If you are implementing a fresh SAP BW on SAP HANA system, or after you have finished upgrading to SAP BW on SAP HANA, the key design philosophy to keep in mind is LSA++

- LSA++ architecture results in reduction of layers, but along with that it is important to understand a few other query settings that can have a great impact on query performance in your SAP BW on SAP HANA system

- This session brings you practical tips to optimize the performance of your existing BEx queries and more on related topics
What We’ll Cover

- Introducing SAP BW 7.4 SP11 on SAP HANA
- Learning to evaluate query performance and tuning BEx queries to run better on SAP HANA
- Evaluating the new query design tool available in SAP BW 7.4 and getting best practices for designing SAP HANA optimized queries
- Understanding the SAP HANA analytic process and learning to leverage analytic manager to enable advanced analytic capabilities
- Wrap-up
Introduction to SAP BW 7.4 powered by SAP HANA SP11

- Why the combination of SAP BW 7.4 and SAP HANA is so fascinating
- Overview of new technical capability
- Converting new features into your advantages
What Is New with SAP BW 7.4

• Enhanced Data Modeling
  ◦ Common Eclipse-based modeling tools
  ◦ SAP BW/HANA Smart Data Access providing the logical EDW
  ◦ Easy integration of external data models with Open ODS Layer
  ◦ Further reduce data layers in SAP BW via Operational Data Provisioning
  ◦ New Composite Provider

• Push down further processing logic to SAP HANA
  ◦ SAP BW Analytic Manager
  ◦ SAP HANA Analysis Processes
  ◦ SAP BW Transformations
  ◦ PAK (Planning Application Kit) – Pushing down more planning semantics
**SAP BW 7.3 vs. SAP BW 7.4**

<table>
<thead>
<tr>
<th>Features</th>
<th>BW 7.3</th>
<th>BW 7.4 SP5</th>
</tr>
</thead>
<tbody>
<tr>
<td>BEx 3.x</td>
<td>Supported</td>
<td>Not Supported</td>
</tr>
<tr>
<td>Character Length</td>
<td>60 Characters</td>
<td>250 Characters</td>
</tr>
<tr>
<td>Long Text</td>
<td>60 Characters</td>
<td>1333 Characters</td>
</tr>
<tr>
<td>Open ODS layer</td>
<td>n/a</td>
<td>New feature</td>
</tr>
<tr>
<td>Eclipse-based modeling tool</td>
<td>n/a</td>
<td>New feature</td>
</tr>
<tr>
<td>Mobile Enablement</td>
<td>Limited</td>
<td>Enhanced</td>
</tr>
<tr>
<td>Converged planning solutions (i.e., BPC unified)</td>
<td>n/a</td>
<td>New feature</td>
</tr>
<tr>
<td>PSA layer</td>
<td>Required</td>
<td>Optional</td>
</tr>
</tbody>
</table>
Let’s look at the greatest drivers for enterprise synergy resulting from the combination:

- Speed (from SAP HANA)
  - Without having to design a classic star schema

- Simplification (from SAP HANA)
  - Eliminates the necessity for multi-staged persisted data processing
• **Available Content (from BW):** Over the years the business content for SAP BW has grown significantly – this can be used instantly by SAP HANA

• **Governance (from SAP BW):** SAP BW has a strong foundation in terms of data and structure relationships. This can be used as an advantage in the SAP HANA environment too.
LSA++ Open ODS Layer

- Integrate data into the EDW with more...
- Extensive and Flexible options ...

DataSources can be consumed virtually into an Open ODS View.

Combined with BW InfoObjects or Models.

Generate DataSource from the Open ODS View and ETL data into SAP BW.
Advanced DataStore Object (DSO)

- The next generation of DataStore Object is Advanced DSO
  - Combines InfoObject and field-based modeling
  - One type of InfoProvider with different settings to consolidate DSO and InfoCubes
  - The fast, no activation requires loading of the Write Optimized DSO
  - The 3-table approach in standard DSOs
  - The “every characteristic is key” approach of the InfoCube: Supports up to 120 key fields

Advanced DSO can be used for data acquisition layer, corporate memory, data propagation, or reporting layer

Source: SAP
The Power of Composite Providers

SAP HANA Tables/Views

SAP BW InfoProviders

Union and Join

MultiProvider InfoSet
Transient Provider
Virtual Provider

Composite Provider

Composite Provider is fully processed in SAP HANA
SAP HANA Views from Queries

- If a query definition can be expressed as a flat view the calculation can fully take place in SAP HANA and a corresponding SAP HANA view can be generated. To do so the query in Eclipse offers a flag to create an “External SAP HANA view.”
- This will generate an SAP HANA view based on the query definition into a separate SAP HANA schema which allows consuming the data, for example, via SQL or further modeling with native SAP HANA views.
What We’ll Cover

• Introducing SAP BW 7.4 SP11 on SAP HANA
• Learning to evaluate query performance and tuning BEx queries to run better on SAP HANA
• Evaluating the new query design tool available in SAP BW 7.4 and getting best practices for designing SAP HANA optimized queries
• Understanding the SAP HANA analytic process and learning to leverage analytic manager to enable advanced analytic capabilities
• Wrap-up
Understanding Query Runtime

- **Client + Network**: SAP HANA cannot speed up rendering cells on the front end
- **SAP BW OLAP Engine = SAP HANA Calculation Engine + SAP BW Analytic Manager**
- **DB time** includes executing the SQL statement and also the OLAP calculations that are pushed down to SAP HANA

Source: SAP
So Where Are the Improvements for the Query?

- Scenarios with very large master data tables (>100 million rows)
- Query execution plan: SAP HANA has a rule-based optimizer. The rules are continuously improved. E.g., inversion of the JOIN direction for certain accesses and table sizes.
- Materialization of intermediate results – Overcomes memory problems on smaller BWA blades
How to Check for “Push Down” to SAP HANA

- Check for the OLAP functions that can be processed by SAP HANA
- Understand how the push down works … through BW Statistics
- Is an SAP HANA Calculation Scenario generated and used?
- Is an Exception Aggregation a candidate for push down?
### Query Features and Their Handling

<table>
<thead>
<tr>
<th>Analytic Manager operation</th>
<th>Release*</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aggregation</td>
<td>BW 7.0 + BWA 7.00</td>
</tr>
<tr>
<td>Cell-based calculations (FEMS)</td>
<td>BW 7.0 + BWA 7.00</td>
</tr>
<tr>
<td>Hierarchy-processing (part 1)</td>
<td>BW 7.0 + BWA 7.00</td>
</tr>
<tr>
<td>MultiProvider-UNION</td>
<td>BW 7.3 + BWA 7.20</td>
</tr>
<tr>
<td>Exception aggregation COUNT</td>
<td>BW 7.3 + BWA 7.20</td>
</tr>
<tr>
<td>Exception aggregation for keyfigures w/o currency/unit conversion</td>
<td>BW 7.3 + BWA 7.20</td>
</tr>
<tr>
<td>Exception aggregation for keyfigures w/ currency conversion</td>
<td>BW 7.3 + BWA 7.20</td>
</tr>
<tr>
<td>Exception aggregation for keyfigures w/ unit conversion without reference InfoObject</td>
<td>BW 7.3 + BWA 7.20</td>
</tr>
<tr>
<td>Time-dependent currency conversion</td>
<td>BW 7.3 (SP10) on HANA</td>
</tr>
<tr>
<td>Avoid intermediate result set materialization (Layer “Q” virtualization)</td>
<td>BW 7.4 (SP05) on HANA</td>
</tr>
<tr>
<td>“Current member” calculation</td>
<td>BW 7.4 (SP05)</td>
</tr>
<tr>
<td>CompositeProvider (e.g., JOIN between fact data)</td>
<td>BW 7.4 (SP05) on HANA</td>
</tr>
<tr>
<td>Stock coverage keyfigures</td>
<td>BW 7.4 (SP05)</td>
</tr>
<tr>
<td>Unit conversion with Reference InfoObjects (see SAP Note 2001947 to switch off)</td>
<td>BW 7.4 (SP08) on HANA</td>
</tr>
<tr>
<td>Parts of Handling of Inventory keyfigures (see SAP Note 2001947 to switch on)</td>
<td>BW 7.4 (SP08) on HANA</td>
</tr>
<tr>
<td>Leverage HANA hierarchy processing</td>
<td>planned with BW 7.5(SP00) on HANA</td>
</tr>
<tr>
<td>List-based calculations (like conditions)</td>
<td>under discussion</td>
</tr>
<tr>
<td>Time-Series calculations</td>
<td>under discussion</td>
</tr>
<tr>
<td>Temporal Hierarchy JOIN</td>
<td>planned</td>
</tr>
<tr>
<td>Internal Business Volume elimination</td>
<td>under discussion</td>
</tr>
<tr>
<td>Formula exception aggregation for non-dimensional keyfigures</td>
<td>planned</td>
</tr>
<tr>
<td>Formula exception aggregation</td>
<td>planned</td>
</tr>
</tbody>
</table>

---

**Note:**

Keep checking SAP Note 2063449 | BWA functionality is fully covered by SAP HANA, not vice versa
Checking BW Statistics

- Run the query with RSRT settings as shown
- Review the steps and timing through the various layers, including the SAP HANA Calculation Engine layer
- Change the Query Execution Modes and repeat to review differences in execution, steps, and performance
Query Execution Mode in Transaction RSRT

- SAP BW can access data in SAP HANA in two different ways:
  - Standard SQL
  - Using SAP HANA API against the Calculation Engine. This supports the OLAP/Calculation features.

- By default all queries use Mode “3.” A Calculation Scenario is created in SAP HANA at runtime combining the Column Views of the InfoProviders. Then, a single statement is executed via the HANA-API for this cluster. The other InfoProviders are accessed independently, but in parallel.
Query Execution Mode “3”

- Pushes the UNION operation to SAP HANA
- Decreases the amount of data that has to be transferred from the database to the application server
- SAP HANA can also reuse the result of certain operations that have to be performed for each single InfoProvider in the cluster (e.g., a JOIN between the SID-table and a master data [X-/Y-] table)
Query Execution Mode “0”

- Mode “0” means no operations in SAP HANA
- Each single InfoProvider is accessed via the generated SQL statement
- May be used in debugging situations when you want to force the usage of the JOIN engine
- **SAP Note 1858333 – Query Execution Mode “2/3” reads more data than “0” has more details on unique situations when Mode “0” may be the fastest**

Source: SAP
• Mode “2” means no operations in SAP HANA
• Each single InfoProvider is accessed via the generated SQL statement
• May be used in debugging situations when you want to force the usage of the JOIN Engine
Query Execution Mode “6”

- Mode “6” is specifically to deal with exception aggregation in SAP HANA
- Check the query runtime statistics, especially the “records transferred” value, to give an indication of the level of performance gain by the push down

Source: SAP
Read Mode

- The read mode determines how the OLAP processor gets data during navigation. Three alternatives are supported:
  - A = When accessing the database, the system uses the most suitable aggregate table and, if possible, aggregates in the database itself
  - X = You can improve the performance of queries with large presentation hierarchies by creating aggregates in a middle hierarchy level that is greater than or equal to the start level
  - H = The “Read when navigating” setting, in contrast to “Read when navigating/expanding the hierarchy,” only has a better performance for queries with presentation hierarchies
Query Cache Mode

• The cache mode defines whether and how the query results and navigational states calculated by the OLAP processor should be saved as highly compressed data in a cache
• Can be defined at InfoProvider and query level
• Caching is a means of improving query performance.
   But you need to balance:
   • Benefits: Caching improves the response time for queries
   • Costs: Expenditure involved in organizing and retaining data in the cache
Different Cache Modes: Local and Database

- Cache Mode I: Cache is inactive
  - New data is read during every single navigation step
- Cache Mode 0: Local cache
  - The database cache is deactivated. In this case, the system works exclusively with the local cache.
- Cache Mode D: Cache in database
  - The cache data is persistent in database tables. In this mode, no data is displaced, and the memory size is unlimited. The database instance for the table carries the load. More space is required, but this method also saves time.
- Cache Mode P: Every provider as set in the InfoProvider definition
  - This mode is only available for queries on MultiProviders and CompositeProviders. It makes it possible to cache the data from the contained InfoProviders in accordance with the cache made set in the definition of each individual InfoProvider.
Cache Usage Mode

- The cache usage mode is an expert extension to the cache mode. It defines in which situation cache entries can be used for answering a query, and in which cases data needs to be read anew.

- “S” Always display data from query start (Stable Data)

- “C” Always display Current Data

- “ ” Data from query start if possible, otherwise current data (default behavior)
The data integrity profile offers a simple way to control the data integrity of a query without the need to understand all the expert settings of data integrity, cache mode, read mode, InfoProvider grouping, and cache usage mode.

Use: By using the data integrity profile, the system automatically uses correct settings for other individual technical query settings. This prevents contradictory settings of the expert settings, which might not produce the required query behavior.
Use Selection of Structure Elements

- The “Use Selection of Structure Elements” option affects system performance. This function should therefore normally be activated.
  - Example: Query with the restricted key figures (Sales 2013) and (Sales 2014): A drilldown by year in the rows would show the years 2013 and 2014 in the rows. If the query is filtered by the key figure (Sales 2013), then only the row for 2013 would be shown if the flag is switched on. If the flag is switched off, still the two rows for 2013 and 2014 would be shown but the cell (Sales 2014) for the year 2014 would be empty.
Process Key Figures with High Precision

- Defining a certain calculation accuracy decides which number format will be used internally for calculating key figures
  - The level of calculation accuracy can be set in the InfoObject maintenance transaction for each key figure, and in the query monitor for all key figures of a query

- There are two levels of precision:
  - Decimal floating point numbers with 16 places (#short decfloat#)
  - Decimal floating point numbers with 34 places (#long decfloat#)

- Short decfloats are used by default in the Analytic Engine for numerical data to minimize the memory requirements for the main memory and the OLAP cache
No Parallel Processing

- Parallel processing can be deactivated for an individual query. This can be beneficial because the query then uses fewer system resources if you use non-parallel processing.
- In the case of queries with very fast response times, the effort required for parallel processing can be greater than the achievable time gain.
  - In this case, it may also make sense to turn off parallel processing.
    - The query contains cells with a constant selection.
Operations in SAP HANA

- If you are using the SAP HANA database, the system always checks whether optimized operations in the SAP HANA DB are possible for the InfoProvider that the selected query is based on. You can choose one of the following options:
  1. No optimized operations in SAP HANA: If there are virtual chars/key figures, the system sets the No Optimized Operations in SAP HANA
  2. Individual access for each InfoProvider: A separate access is made for each participating InfoProvider
Materialize Intermediate Query Result

• If the query property “Operations in SAP HANA” is set to “Exception Aggregation” or higher, a major part of query processing takes place in SAP HANA.

• Normally it is faster to recalculate the intermediate results each time, because in this case subtotal level specific optimizations can be used.
  • In some exceptional cases it is faster to materialize the intermediate result sets.

• These cases involve a combination of the following:
  • A large number of characteristics in drill, each with subtotals.
  • A huge reduction of data by standard aggregation.
  • Nearly no reduction of data based on filters.
What We’ll Cover

• Introducing SAP BW 7.4 SP11 on SAP HANA
• Learning to evaluate query performance and tuning BEx queries to run better on SAP HANA
• Evaluating the new query design tool available in SAP BW 7.4 and getting best practices for designing SAP HANA optimized queries
• Understanding the SAP HANA analytic process and learning to leverage analytic manager to enable advanced analytic capabilities
• Wrap-up
Eclipse-Based Query with BW Modeling Tool (BW-MT)

- Create simple queries in Eclipse
  - BW-MT 1.6 required
  - A subset of query functionality can be created directly in Eclipse
- Choose between various functions
  - Create structures in row and columns area of the query
  - Conditions and exceptions
  - Currency and unit conversion
  - CANNOT create reusable restricted and calculated key figures and variables
Generate SAP HANA Views Based on BW Queries SP09+

- If a query definition can be expressed as a flat view, the calculation can fully take place in SAP HANA and corresponding HANA views can be generated.
- Generate an SAP HANA model for a simple BW query including security.
- Generated SAP HANA model is also able to read the data from NearLine Storage.
Create a Query Using Eclipse-Based Query Designer

- Invoke the context menu on an SAP BW project and select New → BW Query
Create a Query Using Eclipse-Based Query Designer (cont.)

- New BW Query dialog box comes up, where we can give the name and description of the query

Note: When creating the query itself, it’s possible to have the query in the Project Explorer “Favorites,” for the SAP BW project, by selecting the “Add to Favorites” option
Select the InfoProvider for Query

Select the desired InfoProvider and select “Finish” in the New BW Query dialog box

InfoProvider – Which has the characteristics, key figures, and reusable components

Five tabs – General, Filter, Sheet Definition, Conditions, and Exceptions
Define the Query: Filters and Sheet Definition

- Navigate to the Sheet Definition and Filter tabs and drag and drop the desired characteristics and key figures from the InfoProvider. You can also use Variable for Dynamic filters.
Preview of the Query Using the “Show Preview” Button
For Further Tuning You Can Set Query Properties

• For example, you would like to suppress the ZERO Values

Before Zero Suppression

After Zero Suppression
What We’ll Cover

• Introducing SAP BW 7.4 SP11 on SAP HANA
• Learning to evaluate query performance and tuning BEx queries to run better on SAP HANA
• Evaluating the new query design tool available in SAP BW 7.4 and getting best practices for designing SAP HANA optimized queries
  • Understanding the SAP HANA analytic process and learning to leverage analytic manager to enable advanced analytic capabilities
• Wrap-up
Analysis Process Designer

<table>
<thead>
<tr>
<th></th>
<th>SAP BW 3.x</th>
<th>SAP BW 7.01</th>
<th>SAP BW on SAP HANA</th>
</tr>
</thead>
<tbody>
<tr>
<td>5 DataSources</td>
<td>5 DataSources</td>
<td>6 DataSources</td>
<td></td>
</tr>
<tr>
<td>14 Transformations</td>
<td>16 Transformations</td>
<td>16 Transformations</td>
<td></td>
</tr>
<tr>
<td>8 Data Targets</td>
<td>9 Data Targets</td>
<td>10 Data Targets</td>
<td></td>
</tr>
</tbody>
</table>
What Is an SAP HANA Analysis Process (HAP)

- Using SAP HANA analysis process, functions of the SAP HANA database and functions in the SAP BW systems can be combined
- Always made up of exactly one data source, a function for data analysis, and data target
- SAP recommends to use SAP HANA analysis process instead of Analysis Process Designer (APD) to take advantage of improved performance
- It is possible to execute SAP HANA functions from different libraries (e.g., PAL, AFL, R) directly on SAP BW InfoProvider data such as clustering, association algorithms, regression analysis, weighted score, exponential smoothing, etc.

Source: SAP
Steps to Create an SAP HANA Analysis Process

- In the Modeling Workbench (RSA1), choose SAP HANA Analysis Process under Modeling
Steps to Create an SAP HANA Analysis Process (cont.)

- On the Overview tab, select a data source
- Select the Function or Script or Procedure for the data analysis
- Select the data target
Steps to Create an SAP HANA Analysis Process (cont.)

- Data Source Tab: Make assignments between the fields of the data source and the entry table of a function or procedure, and create filters and calculations

Open order Qty is mapped to the Value
Steps to Create an SAP HANA Analysis Process (cont.)

• Data Analysis Tab: You start by choosing a PAL (Predictive Analysis Library) function.

• On this tab you can change the parameters of the function selected. This tab is only visible if you choose Function or Script in the Overview tab.

You can find details on PAL functions here:
Steps to Create an SAP HANA Analysis Process (cont.)

- Procedure Tab: You start by choosing a PAL (Predictive Analysis Library) function.
- On this tab, you can change the parameters of the function selected. This tab is only visible if you choose Procedure in the Overview tab.
- You can enhance the default logic of the generated procedure.
Steps to Create an SAP HANA Analysis Process (cont.)

- **Data Target Tab**: You can edit loading behavior settings here.
- **Depending on the selections in the first page**, **Data Analysis and Analytic Index tabs** can be selected.
- **Analytic Indexes**:
  - Cannot be transported
  - No backup and restore
  - Best used for ad hoc scenarios only
  - Is used to generate a Transient Provider
What We’ll Cover

- Introducing SAP BW 7.4 SP11 on SAP HANA
- Learning to evaluate query performance and tuning BEx queries to run better on SAP HANA
- Evaluating the new query design tool available in SAP BW 7.4 and getting best practices for designing SAP HANA optimized queries
- Understanding the SAP HANA analytic process and learning to leverage analytic manager to enable advanced analytic capabilities

- Wrap-up
Where to Find More Information

• You can find articles and presentations on SAP HANA and SAP BW 7.4
  • BW 74 Powered by SAP HANA: Use it to your advantage
  • Simple Models in HANA
  • Extending the Reach of LSA++

• Operations in SAP HANA/BWA on the SAP Help Portal
  • https://help.sap.com/saphelp_nw73/helpdata/en/4b/bda228a8b43c22e1000000a42189b/frameset.htm

• Marc Hartz, “How to check the SAP BW Query push down to SAP HANA?” (SAP Community Network, June 2015).
  • http://scn.sap.com/community/bw-hana/blog/2015/06/18/how-to-check-the-sap-bw-query-push-down-to-sap-hana
7 Key Points to Take Home

- SAP BW on SAP HANA does not mean that your queries are always rocket speed
- Take the time to understand the query execution plan, modes, and stats. This brings you insight into the steps at different layers.
- Keep check on new information from SAP as there is a lot of development going on with respect to query processing and feature push down
- Query performance improvements are just one of many improvements and enhancements that SAP HANA can bring to you
- SAP BW 7.4 coupled with SAP HANA brings an exciting range of new modeling capabilities and other features
- The new query design tool shows you a preview of the data in design mode
- HAP makes it possible to execute SAP HANA functions from different libraries (e.g., PAL, AFL, R) directly on SAP BW InfoProvider data
Please remember to complete your session evaluation

How to contact me:
Pravin Gupta
Director, Business Analytics
TekLink International
Pravin.Gupta@teklink.com
Disclaimer

SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP SE (or an SAP affiliate company) in Germany and other countries. All other product and service names mentioned are the trademarks of their respective companies. Wellesley Information Services is neither owned nor controlled by SAP SE.