Accessing SAP HANA with the semantic layer
Pierpaolo Vezzosi
Session 5001
Learning Points

Learn how to build a universe on top of SAP In-Memory Appliance (SAP HANA™). Understand the benefits of connecting to SAP HANA for your business intelligence (BI) needs. Discover dos and dont's when building a universe on HANA.

- Benefits of using SAP HANA for BI
- Best practices of universe definition on SAP HANA
- Step-by-step workflow for creating a universe on SAP HANA
Agenda

- What is SAP HANA
- Benefits of SAP HANA as a reporting data source
- SAP HANA basics for universe designers
- Building a universe on top of SAP HANA
- Best practices
- Q&A
SAP HANA: DEFINITION AND BENEFITS
SAP HANA stands for?

SAP High-Performance Analytic Appliance
What is SAP HANA?

- SAP HANA is a flexible, multi-purpose, in-memory appliance that combines SAP software components optimized on hardware provided and delivered by SAP leading hardware partners.

- In release 1.0, the appliance contains multiple integrated SAP software components including:
  - SAP In-Memory Computation Engine
  - Real-time replication services
  - Data modeling services
  - Data services
Benefits of SAP HANA as a reporting data source

SAP HANA improves a BI project experience with:

- In-Memory storage and computations
- Columnar data storage
- Optimized calculation engines and languages
- Reporting on real-time data
SAP HANA BASICS FOR UNIVERSE DESIGNERS
SAP HANA is a relational database

- Data in SAP HANA is stored in tables
  - Row storage tables
  - Column storage tables
- SAP HANA provides the usual relational concepts
  - Keys
  - Indexes
  - Etc.
- SAP HANA has a database management tool called HANA Studio
### SAP HANA tables: row store

<table>
<thead>
<tr>
<th>Name</th>
<th>City</th>
<th>Revenue</th>
<th>Cust_Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>John</td>
<td>Orlando</td>
<td>100</td>
<td>12</td>
</tr>
<tr>
<td>Tim</td>
<td>Las Vegas</td>
<td>0</td>
<td>12</td>
</tr>
<tr>
<td>Ted</td>
<td>Orlando</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>Dan</td>
<td>Toronto</td>
<td>50</td>
<td>14</td>
</tr>
<tr>
<td>Dan</td>
<td>Toronto</td>
<td>50</td>
<td>14</td>
</tr>
<tr>
<td>Dan</td>
<td>Toronto</td>
<td>50</td>
<td>14</td>
</tr>
</tbody>
</table>
### SAP HANA tables: row store

<table>
<thead>
<tr>
<th>Name</th>
<th>City</th>
<th>Revenue</th>
<th>Cust_Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>John</td>
<td>Orlando</td>
<td>100</td>
<td>12</td>
</tr>
<tr>
<td>Tim</td>
<td>Las Vegas</td>
<td>0</td>
<td>12</td>
</tr>
<tr>
<td>Ted</td>
<td>Orlando</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>Dan</td>
<td>Toronto</td>
<td>50</td>
<td>14</td>
</tr>
</tbody>
</table>
SAP HANA is more than a relational database

- On top of the columnar tables you can build **INFORMATION MODELS**
  *Aka Column Views, Aka HANA Cubes*

- Information models are business models which describe a part of your business

- Three types of information models:
  - Attribute views
  - Analytical views
  - Calculation views

- The models are developed with the same HANA Studio tool
Attribute views – the dimensions

- Attribute views are a representation of a dimension
  - Defined by joining all the tables of the dimension
  - It can contain hierarchical information
  - It can define calculated attributes
- Constraints as of SAP HANA 1.0 SP02
  - Only equi-joins
Attribute view: example in HANA Studio
Analytical views – the single fact cubes

- Analytical views are a dimensional model made of
  - Attributes (dimensions defined out of attribute views)
  - Measures (facts defined out of a fact table)
  - Calculated Attributes, calculated measures, restricted measures

- Analytical views can be consumed by
  - SQL, MDX, SQL Script

- Constraints as of SAP HANA 1.0 SP02
  - Single fact table
  - Only equi-joins
  - Any query on an Analytical view must contain a Group By or a Select Distinct if no measures are used
Analytical views: example in HANA Studio
Calculation views – the complex cubes

- Calculation views are a dimensional model
  - Putting together multiple fact tables or multiple Analytical views
  - Defined either graphically or using SQL script
  - SQL Script is a language which can contain:
    - SQL functions
    - CE functions – work in progress
    - L/R scripts – work in progress

- Calculation views can be consumed by
  - SQL, MDX, SQL Script
Calculation view: example (SQL Script) in HANA Studio

```sql
/* Begin Procedure Script */
BEGIN
    var_out =
        SELECT
            STATE,
            CITY,
            SHOP_NAME,
            ARTICLE_LABEL,
            sum(MARGIN) as MARGIN,
            sum(AMOUNT_SOLD) as AMOUNT,
            sum(QUANTITY_SOLD) as QUANTITY
        FROM
            "II050827"."OUTLET_LOOKUP" INNER JOIN "II050827"."SHOP_FACTS" ON
                INNER JOIN "ARTICLE_COLOR_LOOKUP" ON ("ARTICLE_COLOR_LOOKUP"."AR

GROUP BY
    STATE,
    CITY,
    SHOP_NAME,
    ARTICLE_LABEL;

END /* End Procedure Script */
```
Calculation view: example (graphical output)
Summary

The following consumption objects are available from SAP HANA:

- Database catalog objects:
  - Row tables
  - Column tables

- Information models:
  - Attribute views
  - Analytical views
  - Calculation views

* plus other objects which are not relevant at this time for universe designers such as stored procedures
BUILDING A UNIVERSE ON SAP HANA
## What maps to what

<table>
<thead>
<tr>
<th>In SAP HANA</th>
<th>In a universe</th>
</tr>
</thead>
<tbody>
<tr>
<td>Relational objects</td>
<td>Table</td>
</tr>
<tr>
<td>Row table</td>
<td></td>
</tr>
<tr>
<td>Column table</td>
<td></td>
</tr>
<tr>
<td>Information models</td>
<td>Table</td>
</tr>
<tr>
<td>Attribute view</td>
<td></td>
</tr>
<tr>
<td>Analytical view</td>
<td></td>
</tr>
<tr>
<td>Calculation view</td>
<td></td>
</tr>
</tbody>
</table>

As simple as that!
Step by step procedure

- In the information design tool
  - Create a new project or reuse an existing project
  - Create a new connectivity to SAP HANA
    - JDBC or ODBC
      - Use JDBC if you want to access information models
  - Create a new data foundation
    - Drag and drop your tables
    - Join tables to build your schema
      - Caveats and tips on using information models as tables in the next section!
  - Create a new business layer
  - Publish your universe

As simple as that!
Attribute views appearance

In HANA Studio

In IDT
Analytical views appearance

In HANA Studio

In IDT
Calculation views appearance

In HANA Studio

```sql
/********** Begin Procedure Script **********/
BEGIN
  var_out =
  SELECT
    STATE,
    CITY,
    SHOP_NAME ,
    ARTICLE_LABEL,
    sum(MARGIN) as MARGIN,
    sum(AMOUNT_SOLD) as AMOUNT,
    sum(QUANTITY_SOLD) as QUANTITY
FROM
  "II050827"."QUIET_LOOKUP" INNER JOIN "II050827"."SHOP_FACTS" ON
INNER JOIN "ARTICLE_COLOR_LOOKUP" ON ("ARTICLE_COLOR_LOOKUP"."AS
GROUP BY
  STATE,
  CITY,
  SHOP_NAME ,
  ARTICLE_LABEL;

END /********** End Procedure Script **********/
```

In IDT

<table>
<thead>
<tr>
<th>Output Parameter</th>
<th>var_out</th>
</tr>
</thead>
<tbody>
<tr>
<td>STATE</td>
<td></td>
</tr>
<tr>
<td>CITY</td>
<td></td>
</tr>
<tr>
<td>SHOP_NAME</td>
<td></td>
</tr>
<tr>
<td>ARTICLE_LABEL</td>
<td></td>
</tr>
<tr>
<td>MARGIN</td>
<td></td>
</tr>
<tr>
<td>AMOUNT</td>
<td></td>
</tr>
<tr>
<td>QUANTITY</td>
<td></td>
</tr>
</tbody>
</table>

Real Experience. Real Advantage.
BEST PRACTICES, TIPS, DONTS
Two questions to ask yourself

- Should I build my universe on information models or on tables?

- How to make sure my universe on HANA has a good performance?
Should I use HANA information models in my universe?

Some facts

- Information models are a business representation of your data
- Information models are business models, they don’t influence the performance of queries
- Analytical views have SQL constraints (i.e. always need a GROUP BY or a SELECT DISTINCT to retrieve data)
- Calculation views might improve the performance of data calculation and data retrieval when using SQL Script functions making use of HANA specific low level calls
Should I use HANA information models in my universe? Best practices

- **You can, but you do not need to use** information models in universes: universes on tables and SQL views have an equal performance (*exception with Calculation views using low level calls in the SQL Script)*

- If you use information models, they should not be joined to anything else in the Data Foundation (joining different constructs reduces the performance of the query as multiple query engines are used)

- Use information models in universes if you want to make sure the query syntax follows the constraints which guarantee an execution of the SQL defined in HANA Studio
Should I use HANA information models in my universe?

Bottom line:

- Choosing to build universes on information models rather than on tables is mainly a **modeling decision, not a performance decision.**
## Should I use HANA information models in my universe?

**Bottom line:**

<table>
<thead>
<tr>
<th>Universes on top of a single Analytical View</th>
<th>Universe on top of a single Calculation view</th>
<th>Universe on top of tables</th>
</tr>
</thead>
<tbody>
<tr>
<td>Most modeling is done in HANA Studio</td>
<td>Most modeling is done in HANA studio</td>
<td>Modeling is done in the information design tool</td>
</tr>
<tr>
<td>I want to reuse an existing Analytical view</td>
<td>I want to reuse an existing Calculation view</td>
<td>I want to build my universe from scratch</td>
</tr>
<tr>
<td>The model definition is more guided and constrained in order to guarantee performance</td>
<td>The model definition is flexible with a programming language approach.</td>
<td>The model definition is less guided and constrained in order to provide flexibility</td>
</tr>
<tr>
<td>Queries are guaranteed to be optimized and performing</td>
<td>Queries might trigger low level function calls to the HANA system for fast processing</td>
<td>Queries are performing if the SQL is well formed (same performance than information models on the same SQL)</td>
</tr>
</tbody>
</table>
How to write a performing universe on SAP HANA

- Always prefer column store tables to row store tables for retrieving data
- Do not mix column store tables with row store tables in a query
- Avoid calculations in joins or filters or GROUP BY
  - E.g. T1."user_ID"+T1."user_code"= T2."user_global_id"
  - E.g. group by T1."user_ID"+T1."user_code"
  - E.g. where a*b=10
- Prefer defining table columns which already contain the calculated values
How to write a performing universe on SAP HANA

- Avoid implicit type casting e.g.:
  - (slow) where date_string < SYSDATE
  - (fast) where date_string < TO_CHAR(SYSDATE, 'YYYYMMDD')

- Avoid, whenever possible, non-equijoins

- Avoid using Exists and OR in the same filter

- Use Not Exists instead of Not In predicates

- Avoid Union, Intersect, Except predicates
  - Usually those predicates are found in derived tables
  - Prefer Coalesce rather than Union
  - Except and Intersect can sometimes be substituted by better filters
How to write a performing universe on SAP HANA?

- Fine tune the connection options for best performance
  - Adapt the connection “Array fetch size” to the network and the expected quantity of data to be retrieved (higher values provide a faster data transfer, but increases the server memory consumption)
- Add compulsory filters in the universe to decrease the quantity of retrieved data
How to write a performing query on SAP HANA

REMEMBER: HANA’s value is to manipulate, aggregate, filter large amount of data, not to transfer a large amount of data to client tools

- Always retrieve the least possible quantity of data
- Retrieve aggregates, not details when possible
- Retrieve only the columns you need
  - If you don’t ask for a column, HANA doesn’t fetch it at all
Summary

- Information models provide a more constrained query experience but guarantee performance.
- Universes provide a more versatile query experience but have to be defined carefully to take into account all possible optimizations.
KEY LEARNINGS AND Q&A
Key Learnings

In this session we discussed about

- What is SAP HANA and what are its benefits for BI projects
- What are the main query objects available in SAP HANA
- How to build a universe on top of SAP HANA
- How to fine tune a universe on top of SAP HANA
Q&A

- My contact information

Pierpaolo Vezzosi

pierpaolo.vezzosi@sap.com
Thank you for participating.

Please remember to complete and return your evaluation form following this session.

For ongoing education on this area of focus, visit the Year-Round Community page at www.asug.com/ycr