Capgemini

SAP HANA

Lesson Name: Introduction to SAP HANA

Contents

Introduction to SAP HANA

Evolution of ABAP For SAP HANA

HANA Architecture

SAP In-Memory

SAP HANA Database Concepts

- Row Store
- Column Store

Database Compression

Code Pushdown

Data Provisioning

Lesson Objectives

After completing this lesson, participants will be able to -

- Know about SAP HANA
- Understand HANA Architecture
- Understand SAP In-Memory Concept
- Understand HANA Database Concepts
- Know about Database tables -
  - Row Store
  - Column Store
- Understand Data Provisioning

Introduction to SAP HANA

- SAP stands for **System, Application, Product** in Data Processing.

- HANA stands for **High-performance Analytic Appliance**.

- SAP HANA appliance is a hardware and software combination that integrates a number of SAP components

- SAP HANA is a flexible, data-source-agnostic appliance that enables customers to analyze large volumes of SAP-ERP data in real-time.

Introduction to SAP HANA

- SAP HANA Database is a hybrid in-memory database.
- SAP HANA combines row-based, column-based, and object-based database technology
- SAP HANA database consists of multiple servers
- SAP HANA database is developed in C++ and runs on SUSE Linux Enterprise Server

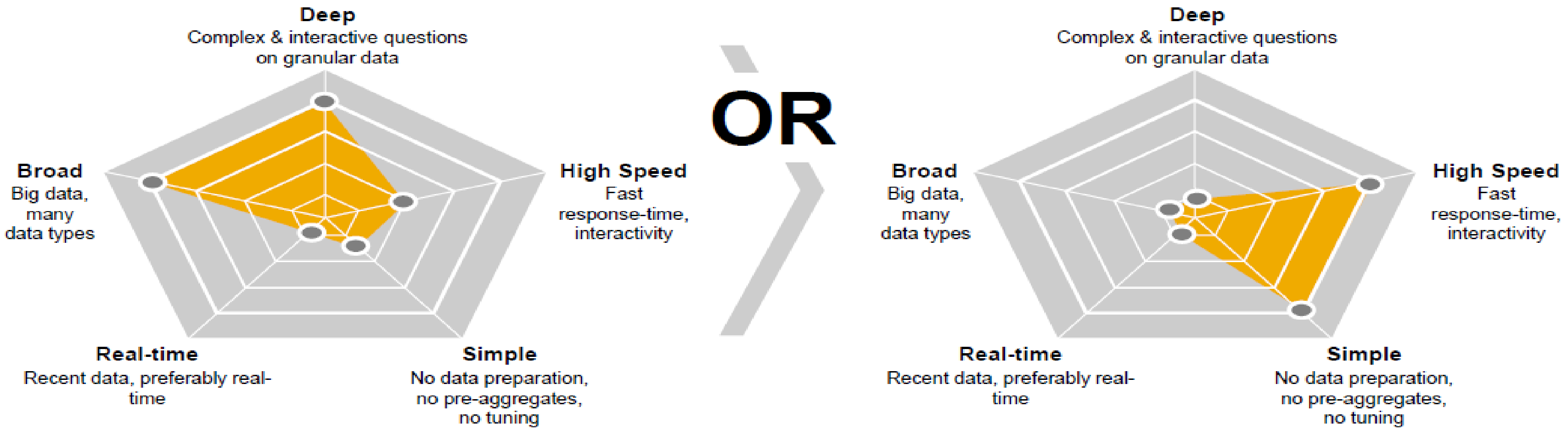
Introduction to SAP HANA

HANA is a Break-through Technology:

- The application developed so far will acknowledge the fact that this typically requires making several trade-off decisions.
- Trade Off is a situation that involves losing one quality or aspect of something in return for gaining another quality or aspect. If one thing increases, some other thing must decrease
- The graphics on the next slide show the five dimensions of requirement that are typically required for a business applications.
- With a traditional database, addressing these all 5 dimensions have been conflicting so far and challenged developers very badly.
- Development of an application which is both real-time and able to analyze a large amount of data at the same time is not possible.

Introduction to SAP HANA

Traditional System involved losing one quality or aspect. something in return for gaining another quality or aspect



Deep
Complex & interactive questions on granular data

Broad
Big data, many data types

High Speed
Fast response-time, interactivity

Real-time
Recent data, preferably real-time

Simple
No data preparation, no pre-aggregates, no tuning

OR

Deep
Complex & interactive questions on granular data

Broad
Big data, many data types

High Speed
Fast response-time, interactivity

Real-time
Recent data, preferably real-time

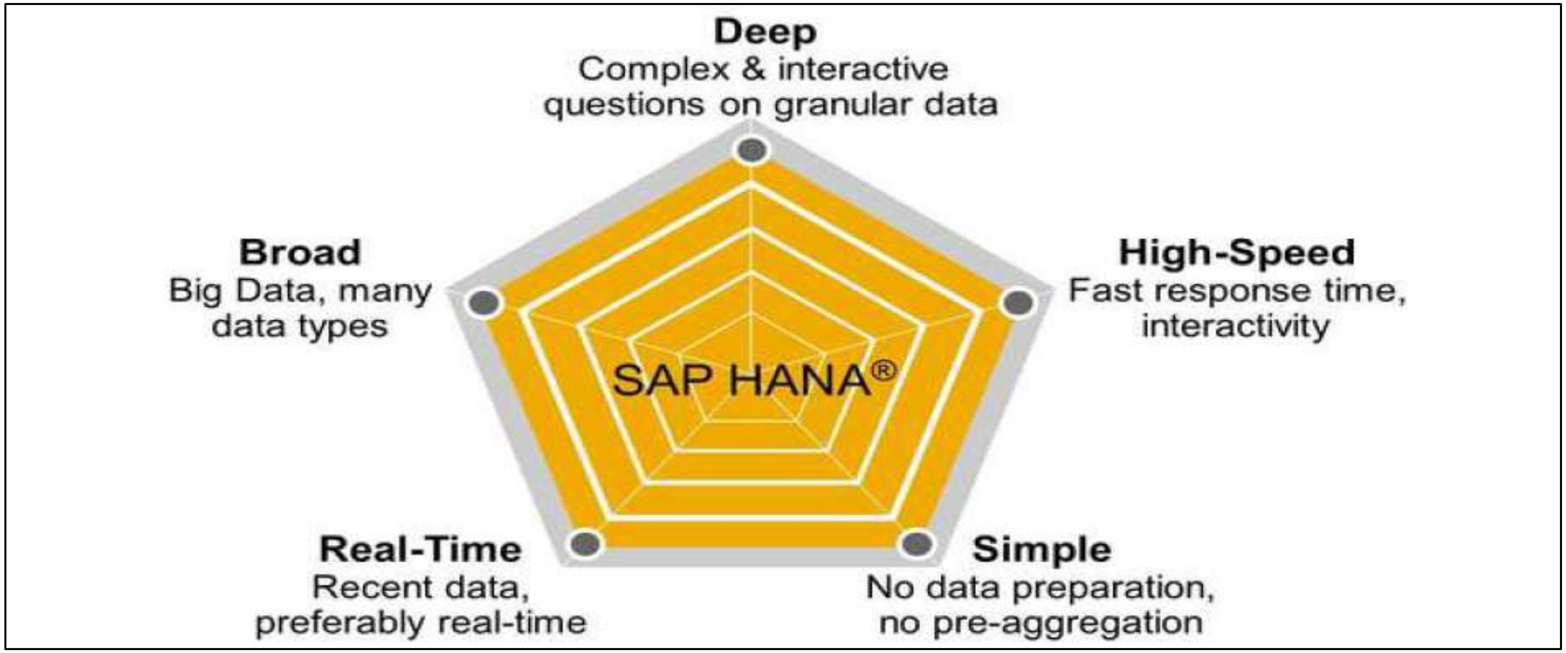
Simple
No data preparation, no pre-aggregates, no tuning

Before HANA, developing applications required several trade of decisions. Picture above, shows five typical dimensions while building a business applications.

As an example, we may have to decide between providing a high-speed application and developing an application which does not required special preparation of processing data e.g. calculating aggregations beforehand and sorting the result set. Also it was very difficult to develop an application which is both real-time and able to analyze a large amount of data.

Introduction to SAP HANA

HANA delivers all the five dimensions, so that you no longer have to make any trade-off decisions


Deep
Complex & interactive questions on granular data
Broad
Big Data, many data types
High-Speed
Fast response time, interactivity
SAP HANA®
Real-Time
Recent data, preferably real-time
Simple
No data preparation, no pre-aggregation

.

The goal is to enable the development of applications that combine Online Transaction Processing (OLTP) and Online Analytical Processing (OLAP) usage patterns.

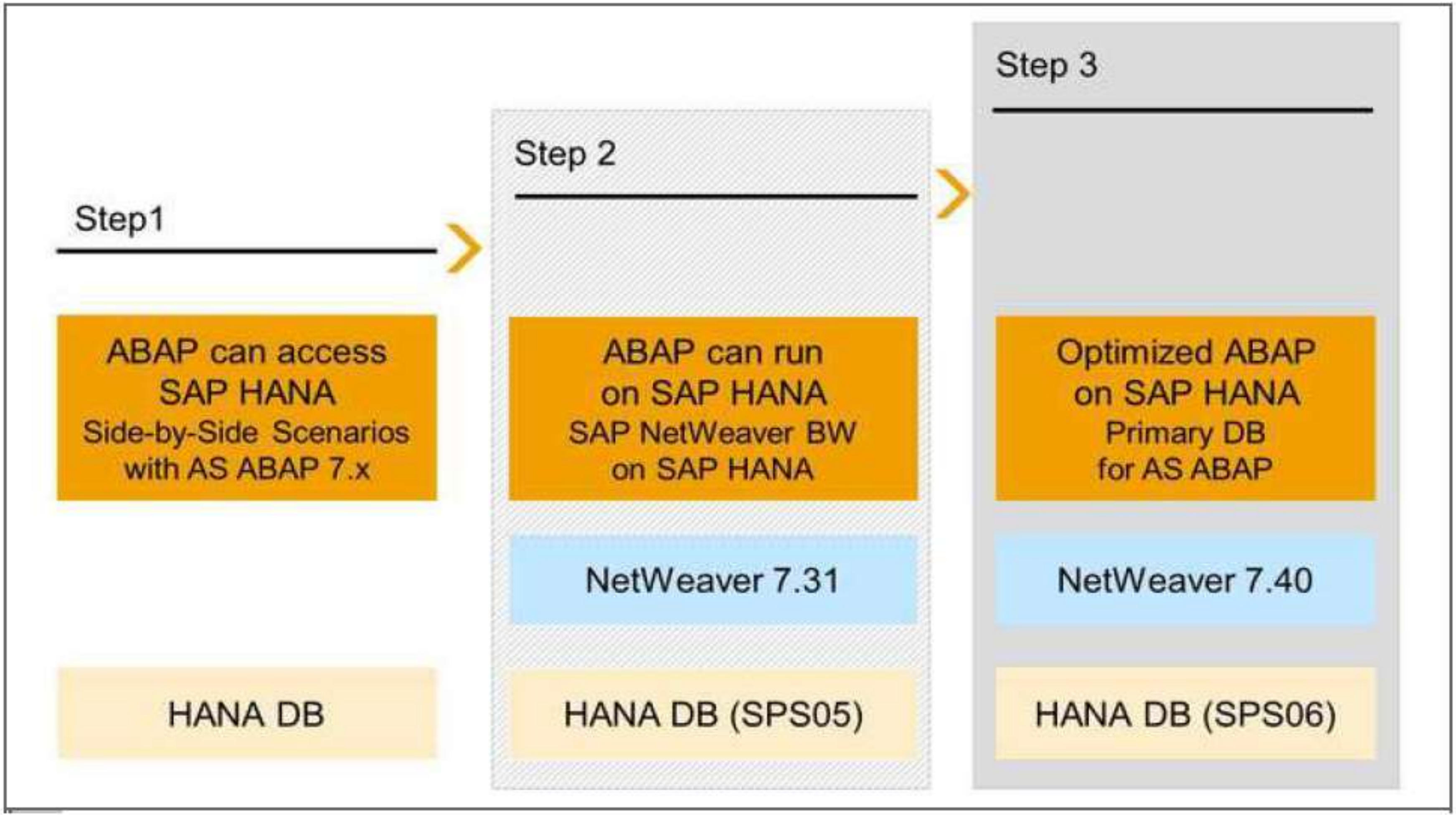
Evolution of ABAP for SAP HANA

With support package stack (SPS) 05 for SAP NetWeaver 7.4, SAP has brought ABAP and SAP HANA together with features that enable developers to leverage the best of both worlds.

ABAP lays the foundation for countless SAP applications, with a broad range of features and functionality for creating powerful business solutions.

SAP HANA provides the opportunity to innovate with new and sophisticated technologies.

Evolution of ABAP for SAP HANA



Step1

ABAP can access SAP HANA
Side-by-Side Scenarios
with AS ABAP 7.x

HANA DB

Step 2

ABAP can run on SAP HANA
SAP NetWeaver BW
on SAP HANA

NetWeaver 7.31

HANA DB (SPS05)

Step 3

Optimized ABAP on SAP HANA
Primary DB
for AS ABAP

NetWeaver 7.40

HANA DB (SPS06)

SAP In-Memory

SAP HANA is an in-memory database:

- An in-memory database means all the data is stored in the memory (RAM).
- Data now resides in main-memory (RAM) also.
- It’s best suited for performing real-time analytics, and developing and deploying real-time applications.
- The speed advantages offered by this RAM storage system are further accelerated by the use of multi-core CPUs, and multiple CPUs per board, and multiple boards per server appliance.

Code Pushdown:

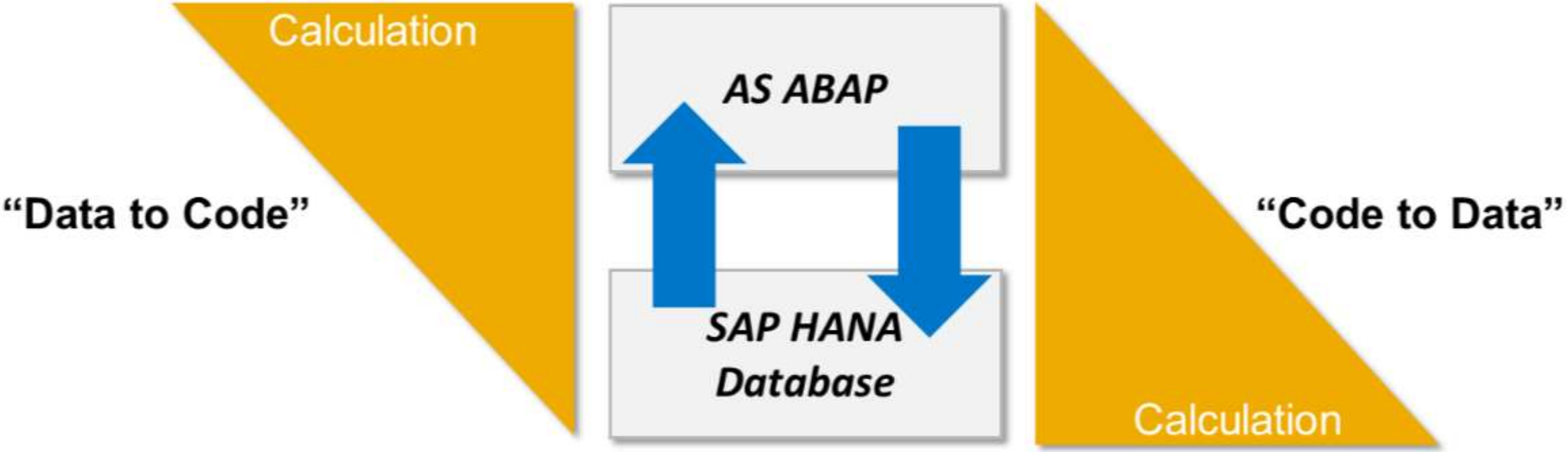
In HANA, you can push down data intense computations and calculations to the HANA DB layer instead of bringing all the data to the ABAP layer and then processing the data to do computations.

This is what is termed as Code-to-Data paradigm in the context of developing ABAP applications optimized for HANA.

It is basically a programming style in ABAP where you code to ‘push down’ data intensive computations to the HANA DB layer.

It is one of the key differences for developing applications in ABAP for HANA.

Code Pushdown:



Review Question

- SAP HANA Studio provides an environment for Administration, Modeling and Data Provisioning. (True/False)

- SAP HANA is a _____________ database.

- Real time replication supported by _____________.

- For small and extensive write operations,_________ store is advisable.

- SLT uses ______ technology to transfer data.

Review Question

- The SAP HANA database is developed in _______ and runs on _______ Server.

- SAP HANA database consists of _______ Server, ______ Server, _______ Server, ___________ Server and ________ Engine.

- SAP HANA database cannot store more than ________rows for a non-partitioned table.