**HANA - High performance Analytic Appliance.**

* It enables customers to analyze large volumes of SAP ERP real time data.
* This appliance is a combination of HW and SW that integrates a no of SAP components.
* These components are:
  + A in memory HANA database
  + Session management
  + Disk storage
  + Persistence Layer
  + Relational engines
  + replication server
* HANA database consists of multiple servers:
  1. **Index Server (most imp server)**
  2. Name Server
  3. Statistics Server
  4. Preprocessor Server
  5. XS Server/Built in Application server/XS Advance (from SPS 11 onwards)
* SAP HANA Data base is a hybrid IN MEMORY DATABASE that combines Row based, Column Based and Object Based database technology.
* HANA is sold with dedicated HW- containing SSD to store records according to ACID properties, data is saved regularly to HDD to allow reboot in case of power failure.
* Getting a license of HANA means getting the HANA modelling tool and a loading or replicating tool.
* In HANA the calculations are executed at the database layer which is in the memory level as against the ABAP method where calculations used to happen at the application layer:
  + **For e.g. in ABAP, we write below code:**

Loop at itab into wa.

Case wa-tempt.

When ‘L’.

Res= ‘Low’.

When ‘H’.

Res= ‘High’.

When ‘M’.

Res= ‘Medium’.

Endcase.

If pdate-ddate <=30.

Interest = 3% on invest.

If pdate-ddate between 35 and 45.

Interest = 5% on invest.

Else

Interest = 8% on invest.

EndIf.

EndLoop.

* **For same code in HANA ABAP:**

Select CID, CNAME,profit, indate, pdate,

CASE res

WHEN ‘L’ then ‘LOW’

WHEN ‘H’ then ‘HIGH’

WHEN ‘M’ then ‘MEDIUM’

END AS res,

Days\_between( indate, pdate) AS Days,

CASE WHEN Days\_between(indate, pdate) BETWEEN 1 and 30

THEN profit \*3/100

WHEN Days\_between(indate, pdate) BETWEEN 30 and 60

THEN profit \*5/100

END interest

FROM ZCTABLE INTO ITAB.

* + - * So here the raw data from the table as well as the resultant of all the calculations like case statements (3 statements in the above code is pushed into the itab. All the calculations are done at the database layer itself.

Evolution of SAP HANA (slide 18):

1. ABAP can **access** the SAP HANA Db at the same time retaining ABAP as the primary DB and HANA would be the secondary DB. This method is used whenever you are satisfied with majority of the benefits of SAP ABAP and just need **barely 10-20% of** **optimization** from HANA. So, the code will be written in ABAP and it will access HANA for just that 10-20% of the code. **Also known as HANA as secondary DB or HANA as accelerator** or **Side by Side Scenarios with ABAP** as u r executing HANA and ABAP side by side. **For this method we need to know HANA SQL which is needed to access the HANA DB.**

ABAP Code

For 20% of code

HANA views or procedures

for 80% of the code

ABAP/Oracle DB

HANA DB

1. ABAP can **run** on HANA DB but there is no optimization on ABAP. There is a 3rd tool which is optimized that is BW (business warehouse). So, HANA with SPS05(service Pack 05) and Netweaver 7.31 gives ability to run ABAP on HANA DB but with no improvement on ABAP. The BW is optimized bcz of HANA not ABAP. **You need to know HANA SQL.**

In both above methods its like U have a 4G SIM but your phone supports only till 2G so the benefits of 4G cannot be met. Or its like U live in an underdeveloped region with no shops, schools, hospitals, malls for everything u need to go to the city. So, the 3rd phase is the most beneficial one.

1. Optimized ABAP running on HANA DB. Here U have a phone which can support 4G Sim or ur region is developed. You use **Open SQL**, **so no need to know HANA SQL.**

The 5 dimensions delivered by SAP HANA:

1. Deep (Complex and interactive questions on granular data)
2. High Speed (Fast response Time and interactive)
3. Simple (No data preparation or pre aggregation)
4. Real Time (Recent and real time data)
5. Broad (Big Data and many data types)

It combines both OLTP (Online Transaction Processing) and OLAP (Online Analytical Processing).

Terms/Operations of SAP HANA DB:

1. **Table Partitioning:** Data Organization scheme in which Table Data is divided across multiple storage objects known as data partitions. It may involve splitting column store tables into horizontal sub tables or partitions.
   1. A nonpartitioned SAP HANA table can contain only 2 billion rows.
2. **Avoidance of aggregate Table:** Data should be stored at highest level of granularity. The aggregation speed of HANA provides computing from the source data really fast with same performance and quality as aggregated DB systems.
3. **Insert Only on Delta :** Whenever there is a new entry into the table, it is appended to the table with the same PK thus avoiding overwriting the table. HANA adds a transaction key to the entry and while reading only the data with highest transaction ID is read.
4. **Deletion:** It just marks the entry as deleted but does not right away delete data from the table.
5. **Same with Insertion:** Insertion does not happen to the table right away. It gets inserted to an uncompressed delta table and gets joined to the table when it is selected/read.

Architecture of HANA database consists of:

* 1. **Index Server (most imp server):** Contains the actual data stores and the engines for processing the data. It processes incoming SQL or MDX(Query language) statements.
  2. **Persistence Layer:** Responsible for durability and atomicity of the transactions. Ensures DB can be restored to recent commit and transactions are fully executed or undone.
  3. **Name Server:** Owns info about topology of HANA system-meaning in a distributed system, this server will know which components are running and where all the data is located on a server.
  4. **Statistics Server:** Has the info about status, performance and resource consumption from other servers in the system. Also provides info about history of measurement of data for further analysis.
  5. **Preprocessor Server:** Used by Index server to analyze the text data and extract the information on which the text search capabilities are based.
  6. **XS Server/**Built in Application server/XS Advance (from SPS 11 onwards) : Optional Component. Users can connect to HANA DB using XS Server through HTTP.
* HANA is an in-memory DB. Data that is in real format now resides in RAM and not in Hard disk. It has row based and column-based technology (slide 34 diag). You can do the calculations at the database layer and not in the application layer as the dB is now in the memory.
  + - HANA – In Memory DB = Normal DB or Normal ABAP DB.
* HANA Db has its own scripting language named **SQL Script.** Also supports R language.
* SAP HANA **Modeler** Views can only be created on the top of **Column** based tables.
* In a conventional database, data is stored in Row based structure i.e. horizontally. SAP HANA stores data in both row and Column based structure. This provides Performance optimization, flexibility and data compression in HANA database. By default every table created in SAP HANA is columnar.
* Storing Data in Columnar based table has following benefits:
  1. Data Compression
  2. Faster read and write access to tables as compared to conventional Row based storage
  3. Flexibility & parallel processing (HANA supports multi core processing just like what the i3/i5/i7 processors support – carrying out multiple processing on 3/5/7 processors at the same time) – For this we need columnar based DB. Each column of the DB is processed individually by each core. If required each column can be further divided into multiple cores to ensure that all the cores are being utilized effectively thereby guaranteeing faster processing.

***Example: U have 10 counters (10 cores) would u just have 1 queue for just 1 counter and leave all the other 9 counters w/o any queue? NO u wd divide the queue among all the 10 counters***.

* 1. Perform Aggregations and Calculations at higher speed
  2. It costs less to store data in column based in main memory
* Data in Column tables can be compressed using 2 ways:
  1. Dictionary Compression
  2. Advanced Compression
* Explain slide 45
* To store the contents of a column, the SAP HANA database creates a minimum of two data structures:
  1. a dictionary vector
  2. an attribute vector
* Explain slide 47
* **Code Pushdown/Code to Data Paradigm**: you can push down data intense computations and calculations to the HANA DB layer instead bringing all the data to the ABAP layer and the processing the data to do computations. slide 49.
* Data provisioning involves importing data from multiple SAP and non-SAP system into SAP HANA.