

Introduction

Sunday, February 21, 2016 5:24 AM

HANA Database:

SAP HANA is an in-memory database:

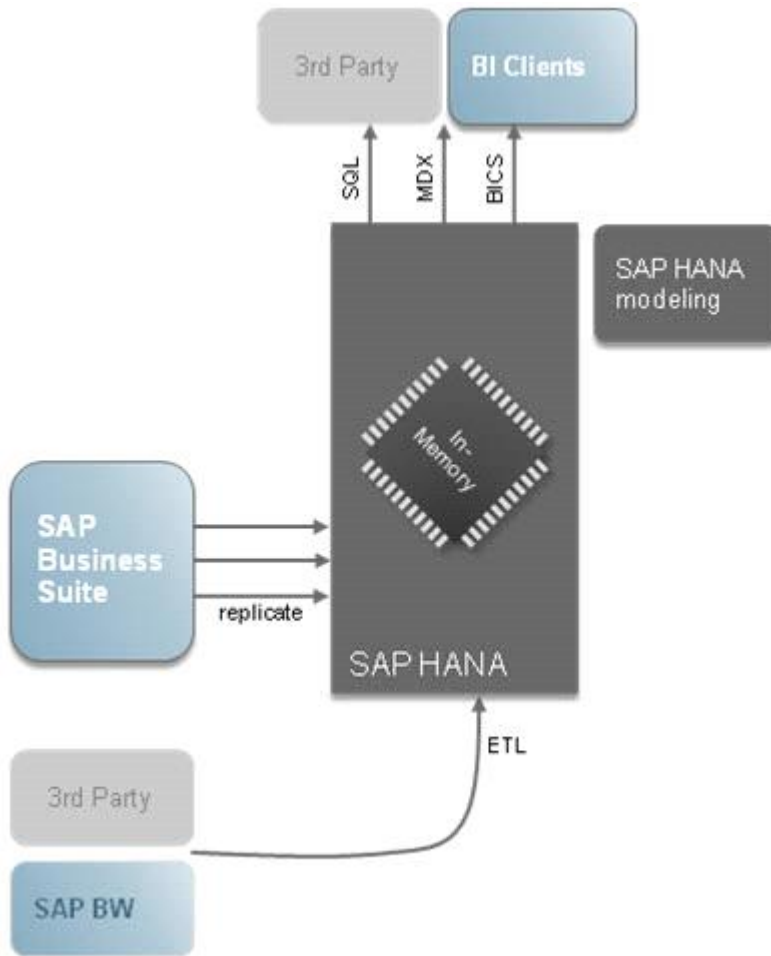
- - It is a combination of hardware and software made to process massive real time data using In-Memory computing.
- - It combines row-based, column-based database technology.
- - Data now resides in main-memory (RAM) and no longer on a hard disk.
- - It's best suited for performing real-time analytics, and developing and deploying real-time applications.

An in-memory database means all the data is stored in the memory (RAM). This is no time wasted in loading the data from hard-disk to RAM or while processing keeping some data in RAM and temporary some data on disk. Everything is in-memory all the time, which gives the CPUs quick access to data for processing.

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.

Complex calculations on data are not carried out in the application layer, but are moved to the database.

SAP HANA is equipped with multiengine query processing environment which supports relational as well as graphical and text data within same system. It provides features that support significant processing speed, handle huge data sizes and text mining capabilities.



With the help of technology like SLT replication, data can be moved to HANA in real time. It is also possible to copy data from SAP BW or other database into SAP HANA. In HANA, we can use modeling tool called HANA Studio to build the logic and structures and use tools e.g. SAP Business Objects, SAP Visual Intelligence to visualize or analyze data.

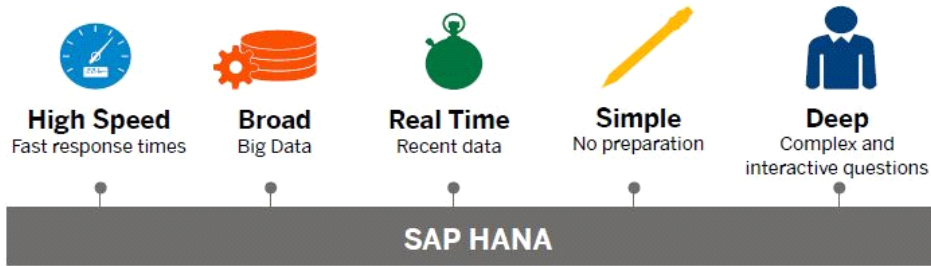
Top 10 Reasons Customers Choose SAP HANA

SAP HANA is one of the fastest growing products in SAP's history and is viewed by the industry as a break through solution for in-memory databases. SAP HANA claims that it accelerates analytics and applications on a single, in-memory platform as well as combining databases, data processing, and application platform capabilities.

SAP HANA is a next-generation business platform which brings together

- Business transactions
- Advanced analytics
- Social media
- Mobile experience

- Collaborative business
- Design connections



You may be thinking, “So what?” or “How does this help my business?” or “How can SAP HANA help my company make more money?”

In this article, we look at what we consider to be the top 10 reasons why customers should choose SAP HANA.

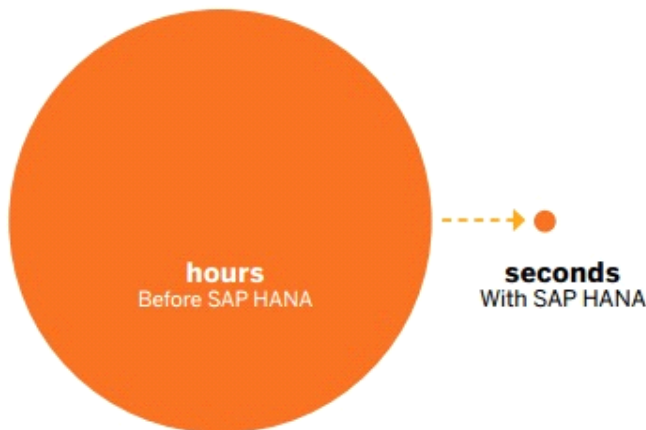
1. Speed:

“The speed SAP HANA enables is sudden and significant, and has the potential to transform entire business models.”

SAP HANA manages massive data volume at high speeds.

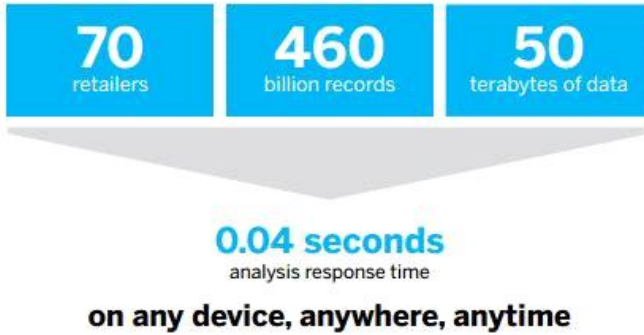
It delivers the “real” real-time enterprise through the most advanced in-memory technology.

SAP HANA provides a foundation on which to build a new generation of applications, enabling customers to analyze large quantities of data from virtually any source, in real time.



A live analysis by a consumer products company reveals how SAP HANA analyzes current point-of-sale data in real time—empowering this organization to review segmentation, merchandising,

inventory management, and forecasting information at the speed of thought.



2. Real Time:

SAP HANA delivers the “real” real-time enterprise through the most advanced in-memory technology

Pull up-to-the-minute data from multiple sources. Evaluate options to balance financial, operational, and strategic goals based on today’s business

3. Any Data:

SAP HANA helps you to gain insights from structured and unstructured data.



SAP HANA integrates structured and unstructured data from internal and external sources, and can work on detailed data without aggregations.

4. Any Source:

SAP HANA provides multiple ways to load your data from existing data sources into SAP HANA.

SAP HANA can be integrated into a wide range of enterprise environments, allowing it to handle data from Oracle databases, Microsoft SQL Server, and IBM DB2.

5. Insight - Unlock new insights with predictive, complex analysis:

Before SAP HANA, analytics meant:

- Preconfigured dashboards based on fixed business requirements.
- Long wait times to produce custom reports.
- Reactive views and an inability to define future expectations.

With SAP HANA, you can:



Ask any question

Quickly and easily create ad-hoc views without needing to know the data or query type - allowing you to formulate your actions based on deep insights



Get fast answers

Receive quick reactions to newly articulated queries so you can innovate new processes and business models to outpace the competition.



Plan and predict

Enable state-of-the-art, interactive analyses such as simulations and pattern recognition to create measurable, targeted actions.

6. Innovation - The ultimate platform for business innovation:

SAP HANA is an early innovator for in-memory computing. Its configurability, easy integration, and revolutionary capabilities make it flexible enough for virtually anything your business requires.

Some examples of this include:

Energy Management

Utility companies use SAP HANA to process and analyze vast amounts of data generated by smart meter technology, improving customers' energy efficiency, and driving sustainability initiatives.

Real-time Transit Routing

SAP HANA is helping research firms calculate optimal driving routes using real-time GPS data transmitted from thousands of taxis.

Software Piracy Detection and Prevention

Tech companies use SAP HANA to analyze large volumes of complex data to gain business insights into software piracy, develop preventive strategies, and recover revenue.

7. Simplicity - Fewer layers, simpler landscape, lower cost:

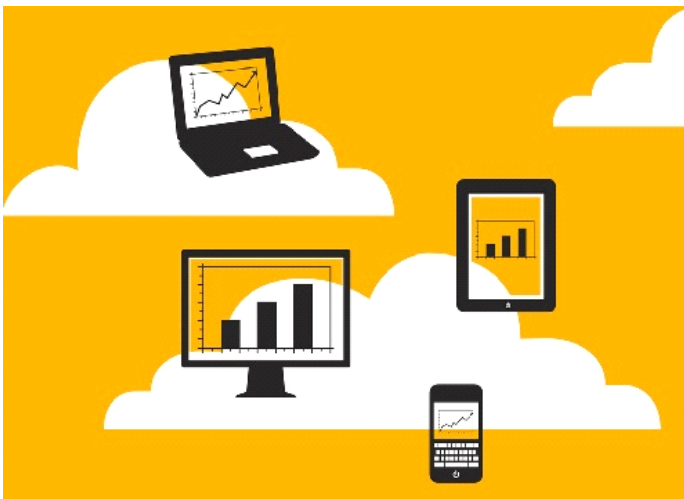


Reduce or eliminate the data aggregation, indexing, mapping and exchange-transfer-load (ETL) needed in complex data warehouses and marts.

Incorporate prepackaged business logic, in-memory calculations and optimization for multicore 64-bit processors.

Spend less on real-time computing

8. Cloud:



Step up to one of the world's most advanced clouds.

SAP HANA powers SAP's next-generation enterprise cloud.

Fast:

A single-location stack removes latency – enabling real-time collaboration, processing, and planning.

Scalable:

A highly robust cloud service allows quick deployment of current and next generation applications, scaled to your business needs.

Secure:

We secure your data through the entire cloud solution with independently audited standards of data security and governance.

9. Cost:

SAP HANA reduces your total IT cost so you can increase spending on innovation.

10. Choice:

SAP HANA provides you choice at every layer to work with your preferred partners.

- **Run on the hardware of your choice.**
- **Work with the software you prefer.**

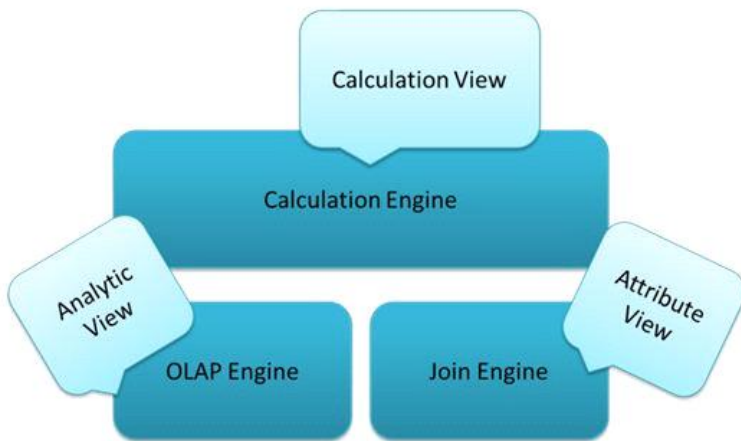
Collaboration with a number of partners means that SAP can complete the software stacks of our diverse customer base in configurations that make sense for their business.

Plus, a variety of different options means that you won't be locked in by a single provider.



SAP HANA Modeling View Processing:

SAP HANA has mainly 3 types of engines that are used based on the views required by the model.



Join Engine: Used for Attribute Views

OLAP Engine: Used for Analytic Views (without calculated columns)

Calculation Engine: Used for Analytic views with calculated attributes, Calculation views

HANA

Sunday, February 21, 2016 5:26 AM

- 1- Catalog: It is a folder inside hana studio where we will store all type of data.
- 2- Content: It is a folder where we are going to perform all modelling related activities.
- 3- Provisioning : It is a folder used for data fetching from another systems.
- 4- Security : It is a folder where we are going to create different users, authentications, privledges.

Catalog:

Sunday, February 21, 2016 5:58 AM

- 1- Inside the catalog first step we have to do is create a schema.
- 2- Schema: It is type of sub folder where we are going to store our data.
- 3- Right click on catalog and open sql console.
- 4- Write sql syntax for creating schema "create schema 'schema name' ".
- 5- Inside the schema we are going to maintain our all tables.
- 6- For Importing the tables go to Quick view and choose Import option.
- 7- In that go to sap content folder.
- 8- Choose sap content from local file
- 9- Then choose your source file , choose your target schema.
- 10- Give table name same as your source file.
- 11- Than choose one common coloumn and choose that as primary key.
- 12- Finish.

Content Folder

Saturday, February 27, 2016 5:30 AM

- 1- We have to create our own package by right click on content folder.
- 2- Package- Is the type of sub folder where you are going to do all your modelling related stuff.
- 3- Types of Data : A- Master Data (Dimensions in Hana) - Fixed Data where we can't perform any calculation.
- 4- B- Transaction Data (Measures in Hana)- Is type of volatile data where on top of that you can perform mathematical operations.

Attribute View

Saturday, February 27, 2016 6:51 AM

- 1- Attribute view : Is a concept given by sap hana meant for maintaining master data .
- 2- Attribute view will consist of two parts : a) Data foundation Layer : Is used to hold all your tables which we want to use in our views.
b) Semantic Layer : Here we will find all selected object properties.
- 3- Whenever we will try to create any view in Hana studio an user is going to be created by system automatically inside users folder under security with the name of 'Sys_Repo'.
- 4- To this _Sys_Repo user we need to give Grant select Privileges.
- 5- For this just go to your system select it, right click and open sql console.
- 6- Now give sql statement for grant select : (grant select on schema "schema name" to _sys_repo with grant options).
- 7- After executing attribute view go to attribute view and select data preview.
- 8- There you can see all data and you can play around with data for reporting purpose.

Attribute View:

- Attribute views are dimensions just like BW characteristics or master data.
- Attribute views are used to join to a dimension or attribute view.
- In most cases used to model master data like entities (like Product, Employee, Business Partner)
- Highly re-used and shared in Analytic- and Calculation Views

Example : An attribute view "FLIGHT" can show together Airline Code, Airline Connection Number and flight Date into one Object. This attribute view can be used in analytic and calculation views where entities are relevant.

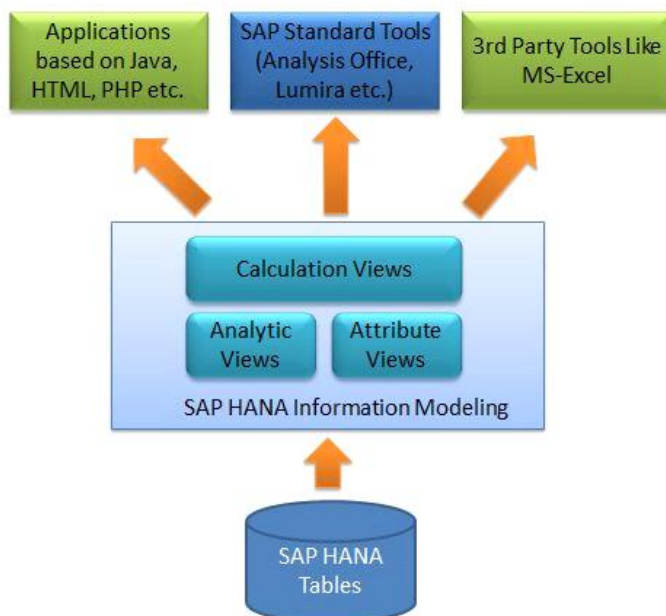
Generally attribute views represent master data. But, however technically there is no restriction and it's possible to make attribute views on transaction data.

Modelling

Sunday, February 28, 2016 4:29 AM

SAP HANA Modeling at a glance:

- SAP HANA Information Modeling which also known as SAP HANA Data Modeling is the heart of HANA application development.
You can create modeling views on top of database tables and implement business logic to create a meaningful report.
- These modeling views can be consumed via Java or HTML based applications or SAP HANA native applications. You can also use SAP tools like SAP Lumira or Analysis Office to directly connect to HANA and report modeling views. It is also possible to use 3rd party tools like MS-Excel to connect to HANA and create your report.



- Modeling SAP HANA Information Views are important for successfully exploiting the power of SAP HANA. These views are classified as
 - **Attribute Views**
 - **Analytic Views**
 - **Calculation Views**
- At run-time these views make implicit use of optimized SAP HANA In-Memory calculation engines and thus enable for best performance.
- HANA Data Modeling is only possible for Column Tables i.e. Information Modeler only works with column storage tables.

Calculation Views

Analytic
Views

Attribute
Views

Analytical View

Sunday, February 28, 2016 4:35 AM

Analytic View:

- Analytic views are star schemas or fact tables surrounded by dimensions, calculations or restricted measures.
- In the language on SAP BW analytical views can be roughly compared with Info Cubes or Info Sets.
- Analytic views are typically defined on at least one fact table that contains transactional data along with number of tables or attribute views.
- Analytic views leverage the computing power of SAP HANA to calculate aggregate data, e. g., the number of bikes sold per country, or the maximum power consumed per month.
- It is specifically designed to execute star schema queries

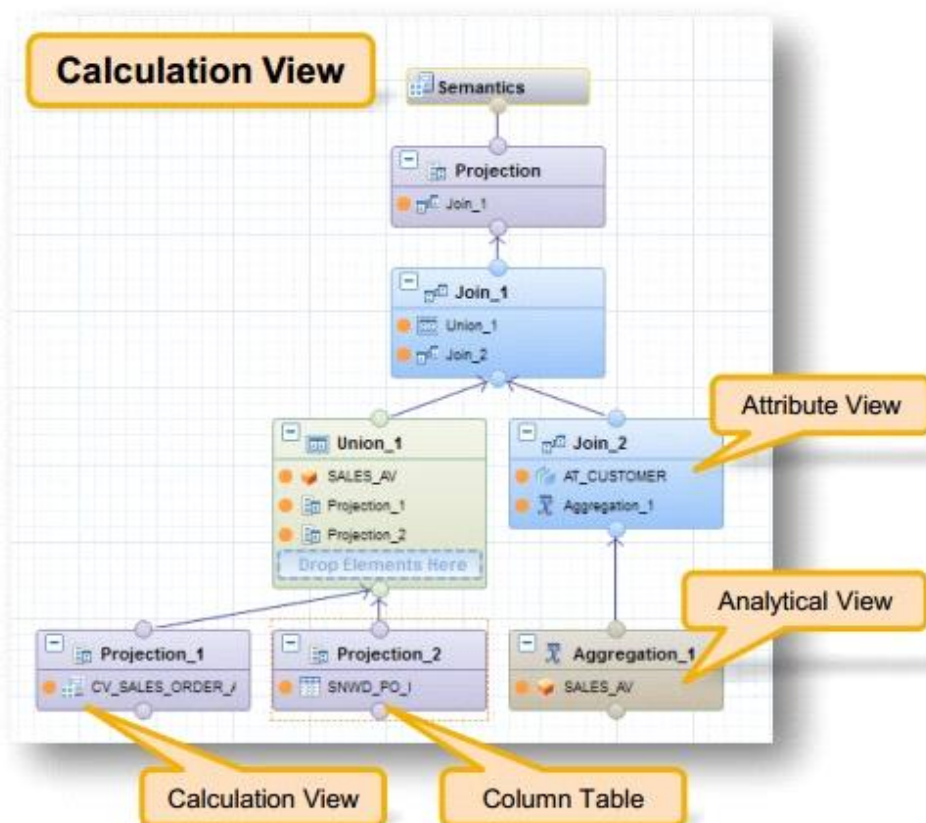
Calculation View

Sunday, February 28, 2016 4:36 AM

Calculation View:

- Calculation views are composite views used on top of analytical and attribute views.
- It can perform complex calculations not possible with other views.
- It can be defined as either graphical views or scripted views depending on how they are created. Graphical views can be modeled using the graphical modeling features of the SAP HANA Modeler. Scripted views are created as sequences of SQL statements.
- Calculation views can be referred as combination of tables, attributes views and analytical views to deliver a complex business requirement. They offer to combine different analytical views into one source of data for reporting.

Calculation views are **composite views** and can be used to **combine other views**. It can consume **other Analytical, Attribute, other Calculation Views & tables**. It can perform complex calculations not possible with other views.

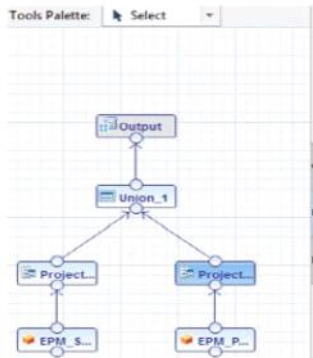


Types of Calculation Views:

Calculation views can be created using a Graphical editor or the SQL Script editor.

- **Graphical calculation views** are created using the graphical editor.
- **Scripted calculation views** are created using SQL Editor. Complex calculations which are not possible thru graphical approach, can be created using SQLScript.

GRAPHICAL



SQL Script

```
Details of Script_View
/***** Begin Procedure Script *****/
BEGIN
  po = CE_OLAP_VIEW('EPM PROCUREMENT', ['NET_AMOUNT', 'GROSS
    'INVOICING_STATUS', 'CONFIRM_STATUS', 'C
  so = CE_OLAP_VIEW('EPM SALES', ['NET_AMOUNT',
    'DELIVERY_STATUS', 'COUNTRY']);
  var_out = CE_VERTICAL_UNION(:po, ['NET_AMOUNT' AS 'Purchase
    'COUNTRY' AS 'Purchase
    :so, ['NET_AMOUNT' AS 'Sales Amount', 'COUNTRY' AS
    'Sales Cc
END /***** End Procedure Script *****/
```

Capabilities of Calculation Views:

Graphical Calculation Views

- Can consume other Analytical, Attribute, other Calculation Views & tables
- Built-in Union, Join, Projection & Aggregation nodes
- Provides additional features like Distinct, Count, Calculation, dynamic joins
- No SQL or SQL Script knowledge required

Row Data Storage and Column Data Storage

Overview of Row Data Storage and Column Data Storage

Relational databases typically use row-based data storage. However Column-based storage is more suitable for many business applications. SAP HANA supports both row-based and column-based storage, and is particularly optimized for column-based storage.

As shown in the figure below, a database table is conceptually a two-dimensional structure composed of cells arranged in rows and columns.

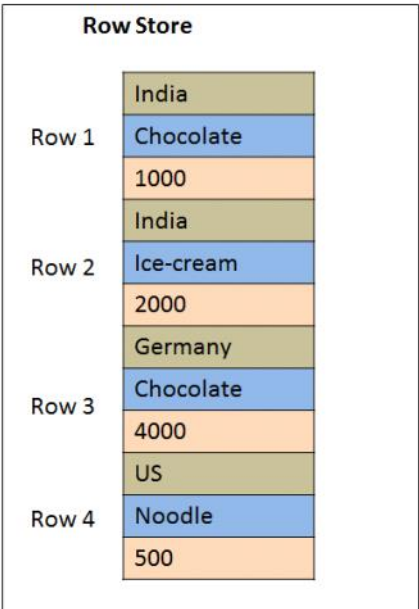
Because computer memory is structured linearly, there are two options for the sequences of cell values stored in contiguous memory locations:

Row Storage - It stores table records in a sequence of rows.

Column Storage - It stores table records in a sequence of columns i.e. the entries of a column is stored in contiguous memory locations.

Table

	Country	Product	Sales
Row 1	India	Chocolate	1000
Row 2	India	Ice-cream	2000
Row 3	Germany	Chocolate	4000
Row 4	US	Noodle	500



Traditional databases store data simply in rows. The HANA in-memory database stores data in both rows and columns. It is this combination of both storage approaches that produces the speed, flexibility and performance of the HANA database.

Advantages of column-based tables:

Faster Data Access:

Only affected columns have to be read during the selection process of a query. Any of the columns can serve as an index.

Better Compression:

Columnar data storage allows highly efficient compression because the majority of the columns contain only few distinct values (compared to number of rows).

Better parallel Processing:

In a column store, data is already vertically partitioned. This means that operations on different columns can easily be processed in parallel. If multiple columns need to be searched or aggregated, each of these operations can be assigned to a different processor core

Advantages and disadvantages of row-based tables:

Row based tables have advantages in the following circumstances:

- The application needs to only process a single record at one time (many selects and/or updates of single records).
- The application typically needs to access a complete record (or row).
- Neither aggregations nor fast searching are required.
- The table has a small number of rows (e. g. configuration tables, system tables).

Row based tables have dis-advantages in the following circumstances:

- In case of analytic applications where aggregation are used and fast search and processing is required. In row based tables all data in a row has to be read even though the requirement may be to access data from a few columns.

Which type of tables should be preferred - Row-based or Column-based?

In case of analytic applications where aggregations are used and fast search and processing is required row-based storage are not good. In row based tables all data in a row has to be read even though the requirement may be to access data from a few columns. Hence these queries on huge amounts of data take a lot of time.

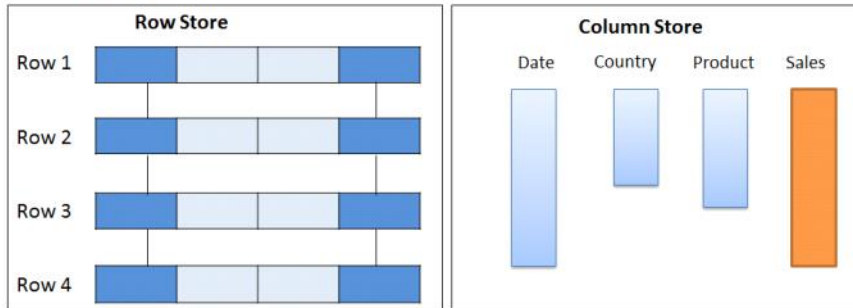
In columnar tables, this information is stored physically next to each other, significantly increasing the speed of certain data queries.

The following example shows the different usage of column and row storage, and positions them relative to row and column queries. Column storage is most useful for OLAP queries (queries using any SQL aggregate functions) because these queries get just a few attributes from every data entry. But for traditional OLTP queries (queries not using any SQL aggregate functions), it is more advantageous to store all attributes side-by-side in row tables. HANA combines the benefits of both row- and column-storage tables.

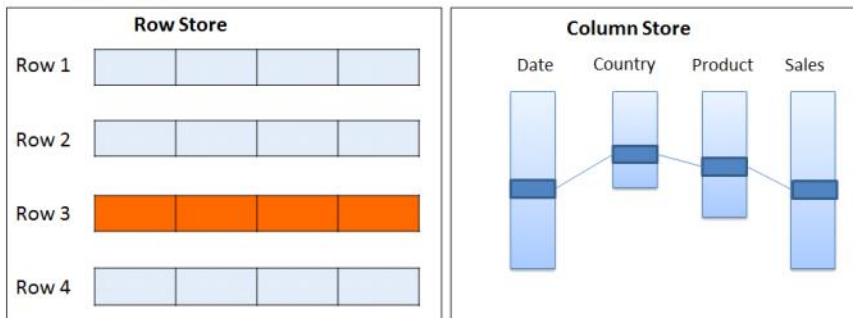
Table - SALES

	Date	Country	Product	Sales
Row 1	2013-01-01	India	Chocolate	1000
Row 2	2013-01-10	India	Ice-cream	2000
Row 3	2013-02-20	Germany	Chocolate	4000
Row 4	2013-03-01	US	Noodle	500

Column Operation: SELECT SUM(SALES) FROM SALES WHERE DATE > 2012-01-01



Row Operation: SELECT * FROM SALES WHERE COUNTRY = 'INDIA'



Conclusion:

To enable fast on-the-fly aggregations, ad-hoc reporting, and to benefit from compression mechanisms it is recommended that transaction data is stored in a column-based table.

The SAP HANA data-base allows joining row-based tables with column-based tables. However, it is more efficient to join tables that are located in the same row or column store. For example, master data that is frequently joined with transaction data should also be stored in column-based tables.