





Now let's talk about the data compression in SAP HANA.

the columns stored tables allow the efficient completion of data.

This makes it less costly for SAP HANA database to keep data in main memory and it also speeds up searches

and calculations. In SAP HANA

there are two types of data compression, Dictionary compression, and advanced compression. Dictionary

compression is the default method of compression and is applied to all the column in table.

It involves the mapping a distinct column values to the consecutive numbers so that instead of actual value

being stored the typically must smaller consecutive number

is stored. In advance compression each columns can be further compressed using different compression

methods such as prefix encoding, run-length encoding, clustering encoding, and indirect encoding. SAP

HANA database uses a compression algorithm to determine which type of compression is most appropriate

for a column.

In this video, I will only explain to you about the dictionary compression type. Advanced data compressions

are out of the scope of this course.

Now let's take an example. here

this table shows sales information.

as we know, SAP HANA by default use dictionary compression to compress all columns in table, for

that purpose it creates dictionary which holds all the distinct values in a particular column.

For example here this dictionary holds all the distinct value in a column COUNTRY.

when we save this data into a database in place of saving all these actual values like UNITED STATE, FRANCE,

and NEW ZEALAND.

We will save its VALUE ID correspond to those values in the database.

So you can see here UNITED STATE get replaced by VALUE ID 1 since in dictionary UNITED STATE's VALUE

ID is 1.

similarly, FRANCE get saved as 2 because the VALUE ID of FRANCE in a dictionary is 2. similarly NEW ZEALAND get

saved as 3 and again UNITED STATE get saved as VALUE ID 1 and so on. you must be wondering

why we are not directly storing the actual values but we are storing the VALUE IDs correspond to that

actual values in database.

This is because, as you know for each character to store we require one BYTE.

So similarly if you want to save the complete column COUNTRY, it will take

71 BYTE.

But in place of storing the actual value

If you stored it in the form of an integer. let's say the VALUE ID type is an integer.

in that case, it will total cost to you 28 BYTE. so you can see the difference, this difference will be more if the

length of data in column COUNTRY is more.

SAP HANA is so intelligent that it directly works on compress column and not an actual column.

and this is how data compression works in SAP HANA.

Data compression in SAP HANA benefits us in 3 different ways

First, it reduced the amount of memory required.

Second, it speeds up the operation on columns because the columns can be loaded into CPU cache faster

and with fewer loading cycles. The third benefit is, it speeds up the operation on columns because many

comparisons become integer value comparisons since these are faster than string comparisons.

So this is how we get benefited by data compression in SAP HANA.