IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 19, Issue 5, Ver. I (Sep.- Oct. 2017), PP 48-53
www.iosrjournals.org

An Overview of Projection, Partitioning and Segmentation of Big Data Using Hp Vertica

*N. Eswara Reddy[1]
[1](Senior Vertica DBA, i|Nautix Technology India Pvt Ltd. Chennai, India
Corresponding Author: N. Eswara Reddy
K V Kishore Kumar
Project Manager , DXC technologies, Chennai, India

Abstract: *This paper provides an overview of projection, partitioning and segmentation of big data using HP vertica. The main aim is to summarize the benefits and restrictions in utilizing HP vertica for big data analysis. We provide an example of projection, partitioning and segmentation that relies on big data analysis.*

Keywords: *Projection, big data, partition, segmentation, vertica*

Date of Submission: 31-08-2017 Date of acceptance: 13-09-2017

I. Introduction

Big data is a term that describes voluminous amount of data, whether it is structured, semi-structured or unstructured data which has a potential to be mined for information. A database that handles this huge volume of data has various technologies defined in order to play with the big data. Emerging technologies and devices create large amount of data out of various sources. To analyze the data many databases and tools have been introduced. HP Vertica is one among them which is an advanced SQL database that addresses big data analytics initiative. HP Vertica delivers speed without compromise, scale without limits, and has a broad range of consumption models. HP Vertica a vertical store database having various features covering four C's: Clustering, Column store, continuous performance and Compression. Comparing Vertica with traditional RBDMS explains it more clearly. As the name denotes the data is stored in vertical columns instead of horizontal rows. Unlike other RDBMS, in vertica a table is a logical entity, it does not have any data in itself. A Projection is a physical entity where data is stored. A super projection is created during the initial loading of the table. We can also create projections based on the query execution for fast data retrieval on few columns. These projections get auto refreshed when the data is being added or removed into the table. The data in the projections is distributed across nodes into small chunks of data called segments, the process is called segmentation. The segmentation can vary from projection to projection. It is advised to segment the columns from high to low cardinality columns. Once segmented across cluster partition takes place inside each node to group the data based on the partition column provided at the time of creation of table. Thus segmentation takes place across cluster and it can vary from projection to projection but partition takes place inside each node and remains same across all the projections. The number of default projection is defined by K-Safety which defines the rate of fault tolerance in Vertica. In other terms the number of duplication or redundancy of a database is described as K-Safety. Hence even if a node is down in the cluster the database will still continue to service the queries as long as it can pull the data from the surviving nodes. K safety value set during the database creation will determine the number of copies of data to be maintained. K-safety of 1 means, vertica database will maintain 2 copies of data and can with loss of 1 node. This can also be set at the time of database creation based on the organizational needs. Since vertica is a columnar store, a read operation only reads the specific column in the query hence reducing IO compared to RDBMS which read the entire row into the memory and reject the unwanted columns. It also has the ability to act on encoded data without decoding it which is another great feature. Traditionally, researchers have adopted the 3Vs criteria to define big data: volume (amount of data), velocity (speed of data in and out), and variety (i.e., range of data types and sources). Big data analytics has applications in health care, education, media, trade, banks, sports, science and research.

II. Projection, Partitioning And Segmentation Of Big Data

Projection is a physical entity where the real data get stored. When a query is executed against a table the database eventually select the best projection to answer the query and runs the data retrieval against that particular projection. Unlike Materialized view in RBDMS projects get refreshed automatically when the data is being inserted or removed from the table. The K-safety which decides the number of projection to be created on the cluster by default. Other than the super projection Vertica allows to create various query specific projection for faster retrieval of query.

In Vertica, partitioning and segmentation are separate concepts each having it own goals to achieving the localization of data. Segmentation refers to organizing and distributing data across the cluster nodes for fast data ingestion and query performance. Segmentation aims to distribute data evenly across multiple database nodes of the cluster so that all nodes participate in query execution. Segmentation is defined during projection creation. The segmentation distributes the data across the nodes in a cluster. Partitioning specifies how to organize data within individual nodes for distributed computing. This process of grouping the data happens inside each node. The partitioning method is defined during table creation A partition criterion is specified during the Table creation. And a Segmentation criterion is specified during the Projection creation. If no projections are specified for a table, a default super projection with the first 32 columns as segmentation clause will be created when the first batch of rows are inserted. Internally the data is first segmented across nodes and then it is partitioned inside each node i.e. partitioning is applied on top of segmented data. For example, let us assume that we have an ACME project which has 6 modules, such as BIM - Marketing Intelligence, FII - Financials Intelligence, HRI - Human Resources Intelligence, ISC - Supply Chain Intelligence, POA - Purchasing Intelligence, OPI - Operations Intelligence We assume that we have 3 executive team working in three different time zones. For the sake of simplicity each team is called developer, though many developers are working in each time. Also, for our convenience, we assume that each module is not depending on any other modules. There may be case that one developer will write the complete ACME project. As it’s only written by one developer, it takes more time and if that developer is unavailable because of any other priority, this project is at risk. Since we have three developers, who are available to contribute to the ACME project, we can allocate work equally among all the developers for much faster progress.

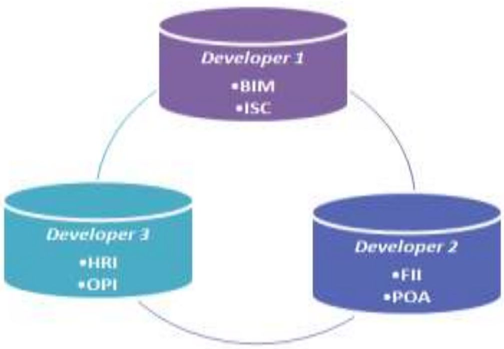

Figure 1: Module distribution among the developers

Each developer can focus on two modules and this reduces the time it takes to complete the ACME project. Being members of executive team, one of the developers was asked by CEO to focus on another high priority project and developer 1 is now unavailable to work on delivering this ACME project. Hence, completion of this ACME project is under risk. The other two developers of the ACME project realized this risk and they accepted some additional work load to minimize this risk. Let’s review how they are planning to minimize risk and deliver the ACME project on time. Every Developer understands the current state of modules written by another developer and also version copy of task done by other developer.

Let’s review how the diagram will look like in this scenario:-

If developer 1 is unavailable for some time, developer 2 can take the work of developer 1 and developer 2. However, there will be an increase in the time and resources for developer 2 to complete. When a specific module is available with two developers, delivery of the content will be fast compared to delivering content only from one developer.

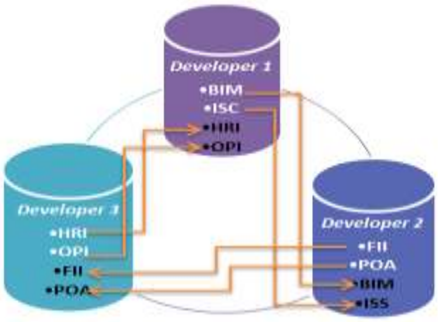

Figure 2: Segmentation of modules across developers

Correlating the entities with the HP vertica:-

- ACME project which we are publishing is called Projection, which stores the data in the form of logical table.
- Developer is called as Node in the vertica cluster
- Distributing modules across all developers is called segmentation
- BIM and ISC module which belongs to developer but available also with developer 2 is called Buddy Projection Scenario when the developer1 is unavailable

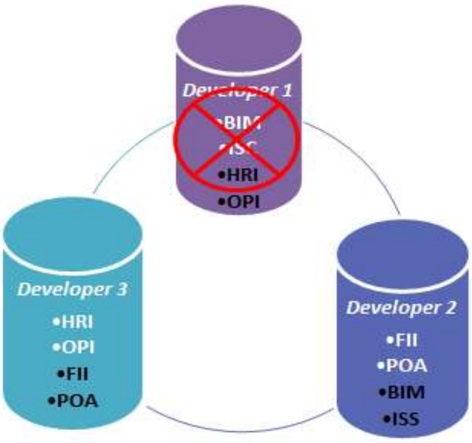

Figure 3: Case when a single developer is unavailable

Case when developer 1 is unavailable developer 2 can finish modules BIM, ISC, HRI, OPI (with less efficiency as developer 2 is overloaded with 4 modules – FII, POA, BIM, ISC) whereas developer 3 can finish HRI ,OPI and also share the work of FII and POA modules of developer 2.

In vertica terms, Node 2 can provide the data when node1 is down, since the data is available from the buddy projections. Node 3 will continue to deliver data in normal way.

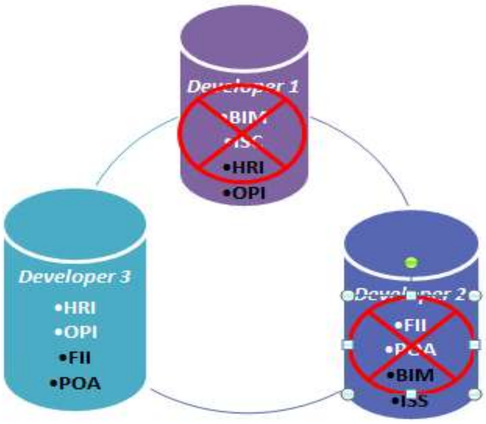

Figure 4: Case when two developers are not available

Scenario when two Developers are not available - Developer 3 dosen’t have details of modules BIM and ISC. At this time however we will not be able to complete the project. Hence even though we maintained the redundacy of the data across cluster when a single segmenation is not available, the cluster will not be able to answer the request for that segmant and it fails this causes the database to go down completely. That is if two Nodes go down, vertica cluster cannot function fully hence it shuts down all the nodes.

If our BIM consists of information about investment services, investment management and wealth management, it is suggested to partition the module into three divisions and keeping the information about each division into a specific partition. With this approach, readers who are focused only on wealth management can directly read the content from that partition. If developer wants to remove complete content of a specific division, they can just remove those divisions without impacting other divisions.

In HP vertica terms, each division is a partition. Partitions are specific to a node and projections are spread across all the nodes.

To overcome the loss of database that is caused by above scenario, vertica provides a solution for this. That can be achieved by increasing the k-safety to two which in turn requires minimum of 5 nodes and there will be three projections that will be created. One super projection and two buddy projection which means three copies of the data will be available in across the cluster. In this case data will be segmented and distributed across all these five nodes.

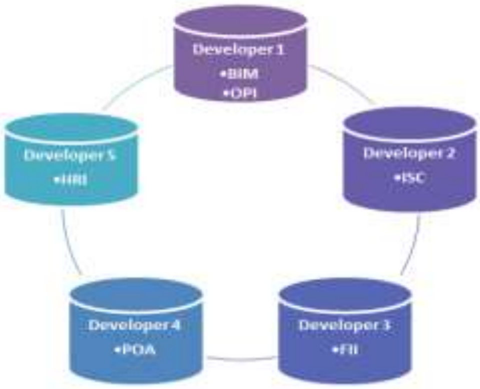

Figure 5: Module distribution among five developers

When we see through the developer module use case, there will be five developers available for these six modules. So in this case each developer will be assigned with its own module as primary (white) and will take up the code of modules from its former and successive developers (black).

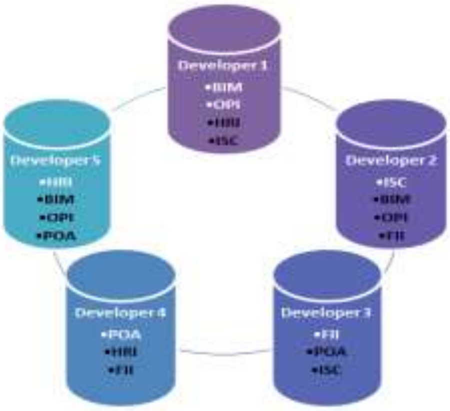

Figure 6: Segmentation of modules across developers

Case when one developer is unavailable to proceed with the module the task will be shared by two other developers. Consider if developer 2 is not available due to some other high priority task then the module ISC which is the primary module of developer 2 is available with developer 1 and developer 3. So developer 1 will perform its own modules BIM, OPI and will also contribute to ISC which is the primary module for developer 2. Developer 3 will perform its own module FII and will also contribute to ISC. Hence in this case the work is distributed among two other developers when a developer is not available.

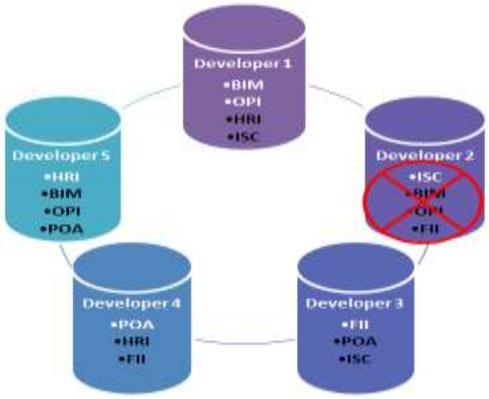

Figure 7: When a single developer is unavailable

Let's see the other case when two developers are unavailable. Consider developer 4 is also unavailable due to some unavoidable reasons. In that case module POA of developer 4 is available with developer 5 and developer 3. Thus the work of the developer 4 will be shared by other two developers.

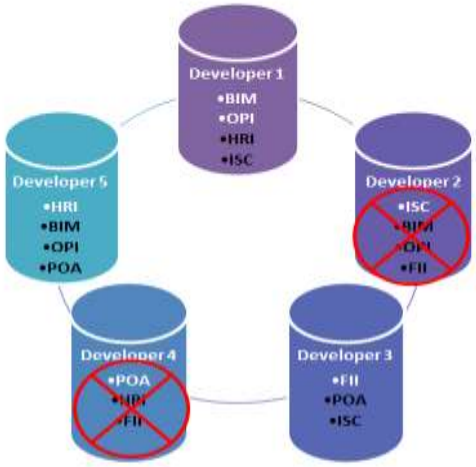

Figure 8: When two developers are unavailable

In case of vertica, even when two nodes are not available the database will still continue to run and answer the queries till the segments are available in other nodes. Hence 6 nodes with safety two creates one super projection and two buddy projection. Thus three replication of data will be created and distributed across nodes. Database will continue to run till less than 50% of the nodes are unavailable.

III. Benefits and Restrictions

The main goal of partitioning is to organize the data in a particular sort order that helps in improved performance. Partition helps us set the data retention policy which will make the data to be removed more certainly based on the partition key. Each partition on a projection is assigned with partition key which helps us to identify the partition. Using Partition keys, partitions can be moved, swapped and copied between tables. In case of archiving the data, moving the data between source and target tables is the only thing to-do. Dropping a partition is more efficient than using delete on the table in the case when the entire data from one partition is to be deleted. During drop partition the appropriate ROS (READ_OPTIMISED_STORE) containers of a partition are eliminated hence each partition stores data in a separate ROS container.

Segmentation of data across the cluster helps in load balancing the data thus the workload of a query is shared across the nodes and all nodes will participate in query execution by retrieving the data from the segments available on respective nodes. Segmentation also helps during the recovery of the node by adding or removing the segments in the recovery node from other nodes for the transactions that are committed during the time when node was down.

Vertica allows us to create projection for a table based on the frequent query execution. Various types of projection can be created on a table other than super and buddy projection. Database design also helps in determining the projection requirement for a table based on the frequently used query. When a query execution is triggered against the database the query optimizer determines the best projection for query execution for fast and efficient data retrieval.

Suppose let us consider that a query that is asked by the senior management every day on the start of the day to generate the analytics. By creating a projection for such query it enables to retrieve the data much faster. The created projection has to segment based on the cardinality values of the column from high to low.

These are some of the restrictions in Vertica:

Partitions are created on the top of the tables hence the partition remains the same for all the projection that will be created based on that table. Creating many number of partition impacts the performance. For an optimal performance the number of projection need to be between 20 and 30 however vertica permits 1024 partitions per projection. A partition cannot be created on NULL value columns. It should be defined as NOT NULL. Segmentation has to be defined on the table columns with low cardinality values to high cardinality values which have to be determined and mentioned explicitly. Segmentation clause cannot be altered. In other words the projection definition cannot be altered. It needs to be dropped and recreated. In case of super projections segmentations are generated automatically. That is if a table contains primary key defined, the projection is segmented by the primary key else projection is segmented by the first 32 columns of the table. The query performance improves by creating appropriate projections. However, we need to increase the amount of disk space as well to store those data. The amount of time required to load data will also increase.

IV. Conclusion

Data is growing every day and big data store is one of the widely used databases to manage and perform analytics. Vertica is one such database that provides high-availability, and petabyte scalability on commodity enterprise servers. It is used by companies such as Facebook, Apple etc.

References

[1]. Gartner. Gartner says solving 'big data' challenge involves more than just managing volumes of data. USA: Gartner. [Online] Available from: http://www.gartner.com/newsroom/id/1731916

[2]. Hewlett Packard Enterprise on 2/8/2017 at 8:08 AM released https://my.vertica.com/docs/7.2.x/ for 7.2 HP vertica enterprise/community edition release notes

[3]. http://vertica.tips/resources/development/

[4]. https://my.vertica.com/docs/7.2.x/HTML/index.htm#Authoring/AdministratorsGuide/Partitions/PartitioningAndSegmentingData

IOSR Journal of Computer Engineering (IOSR-JCE) is UGC approved Journal with Sl. No. 5019, Journal no. 49102.

N. Eswara Reddy. "An Overview of Projection, Partitioning and Segmentation of Big Data Using Hp Vertica." IOSR Journal of Computer Engineering (IOSR-JCE), vol. 19, no. 5, 2017, pp. 48–53.