## Content

1. **Introduction** .......................................................... 3
   1.1 System Architecture .................................................. 5

2. **Factors That Influence Performance** ................................ 7

3. **Sizing Fundamentals and Terminology** ........................... 8

4. **Initial Sizing for SAP Vora** ........................................ 9
   4.1 Engines ..................................................................... 10
   4.2 Worker Nodes ........................................................... 12
1 Introduction

SAP Vora provides a set of in-memory query engines and a disk-based processing engine that are integrated in the Hadoop ecosystem and Spark execution framework. Able to scale to thousands of nodes, SAP Vora is designed for use in large distributed clusters and for handling big data.

Fast Query Execution

The SAP Vora relational in-memory engine holds data in memory and boosts the execution performance of Spark. Supporting just-in-time code compilation, it translates incoming SQL queries into machine-level code on the fly using a LLVM compiler, enabling them to be executed quickly and efficiently.

Data Analytics

SAP Vora makes available OLAP-style capabilities for data on Hadoop, in particular, a hierarchy implementation that allows you to define hierarchical data structures and perform complex computations on different levels of data. Extensions to Spark SQL also include enhancements to the data source API to enable Spark SQL queries or parts of the queries to be pushed down to the appropriate SAP Vora engines.

SAP HANA Integration

Data processing between the SAP HANA and Hadoop environments lets you combine data in SAP HANA with big data stored in Hadoop systems and process it in Spark or SAP HANA applications.

Graph Processing

A distributed in-memory graph engine allows you to execute commonly used graph operations on data stored in SAP Vora and is optimally designed for complex read-only analytical queries on very large graphs.

Time Series Analysis

The in-memory time series engine supports time series analysis algorithms that work directly on top of the compressed data, providing features such as standard aggregation, granularization, and advanced analysis.
Document Store

A distributed in-memory JSON document store supports rich query processing over JSON data.

Relational In-Memory Store

The relational in-memory store allows you to load relational data into memory for faster access and query processing than provided by disk storage.

Disk Storage

The disk engine provides relational column-based storage, allowing you to use relational capabilities without loading data into memory.

Business Functions

Business functions, such as currency conversion and unit of measure conversion, make it easier to use data in business settings.
1.1 System Architecture

The SAP Vora engines are services that you add to your existing Hadoop installation. SAP Vora instances (with the exception of the disk engine) hold data in memory and boost the performance of out-of-the-box Spark. To increase execution performance on the node level, you add an SAP Vora instance to each compute node so that it contains the following:

- A Spark worker and the necessary Hadoop components
- One or more SAP Vora engines

For more information, see the SAP Vora Installation and Administration Guide.
Related Information

SAP Vora Guides
Factors That Influence Performance

A number of factors have an impact on the performance of the system.

**Data Volume**

Consider the data volume that the users will process. This will have an impact on how many (worker) nodes you'll need to process the data efficiently. It will also determine the disk and memory size for each node. The SAP Vora relational engine is an in-memory query engine. It loads the entire table which is accessed into memory.

**Workload Characteristics**

You need to know the workload characteristics to effectively and accurately size your deployment. For SAP Vora this will mainly be the number and size of database objects (tables, time series collections, and so on) managed by SAP Vora, and the number and complexity of concurrent queries.

**Number and Size of Database Objects Managed by SAP Vora**

For example, during creation of a table you can define that it will be managed by SAP Vora. Most probably the decision will be based on the query type and the response time expected to execute the query. The number of tables and their size will determine how much memory you need to allocate.

**Number and Complexity of Concurrent Queries**

SAP Vora can process queries with a high level of parallelization - up to the maximum number of cores. This number and the expected response time of the defined queries will determine the CPU processing power needed to fulfill this KPI.
3 Sizing Fundamentals and Terminology

SAP provides general sizing information on the SAP Service Marketplace. For the purpose of this guide, we assume that you are familiar with sizing fundamentals. You can find more information at http://service.sap.com/sizing → Sizing → Sizing Decision Tree → General Sizing Information.

This section explains the most important sizing terms, as these terms are used extensively in this document.

Sizing

Sizing means determining the hardware requirements of an SAP application, such as the network bandwidth, physical memory, CPU processing power, and I/O capacity. The size of the hardware and database is influenced by both business aspects and technological aspects. This means that the number of users using the various application components and the data load they put on the server must be taken into account.

Initial Sizing

Initial sizing refers to the sizing approach that provides statements about platform-independent requirements of the hardware resources necessary for representative, standard delivery SAP applications. The initial sizing guidelines assume optimal system parameter settings, standard business scenarios, and so on.

Expert Sizing

This term refers to a sizing exercise where customer-specific data is analyzed and used to put more detail on the sizing result. The main objective is to determine the resource consumption of customized content and applications (not SAP standard delivery) by comprehensive measurements.

For more information, see http://service.sap.com/sizing → Sizing → Sizing Decision Tree → Expert Sizing.
4 Initial Sizing for SAP Vora

Initial sizing guidelines for SAP Vora.

Assumptions

- A sizing for your Hadoop cluster is already in place.
- You know the delta that SAP Vora puts on top.
  The delta is especially in the area of memory consumption as SAP Vora is mainly memory bound. For best performance, it is assumed that the SAP Vora engines are co-deployed with the Hadoop data nodes to provide the best data locality possible.
- You have adjusted your YARN configuration to make room for SAP Vora. SAP Vora is not integrated in YARN resource management.

CPU Sizing

In general, the CPU size per node can be taken from the Hadoop cluster sizing. For a more accurate sizing, you will need to know your workload characteristics. Based on this information a pilot can be set up to determine the CPU requirements.

Network Sizing

Network sizing can be taken from your Hadoop cluster sizing.
4.1 Engines

Vora Relational In-Memory Engine

You can estimate the memory requirements for the relational in-memory engine as follows:

- Parquet or ORC files:
  
  Memory usage = file size * 2 (for storing the data in memory) * 2 (for operations)
  
  Memory usage = 4 * file size

  
  - CSV files (compression of factor 2 can be expected):

  Memory usage = uncompressed file size / 2 (for storing the data in memory) * 2 (for operations)
  
  Memory usage = uncompressed file size

 Disk usage = <no additional disk space needed>

To load the Parquet or ORC files into the relational in-memory engine, main memory of the size of factor 3-4 of the files is needed on a temporary basis during loading.

Note that if memory is scarce, table columns will be evicted using an LRU algorithm.

Vora Disk Engine

You can use the following parameters to estimate the memory requirements for the disk engine:

- DataSize: Uncompressed data size on disk
- CompressionRatio: Compression ratio of the disk engine (assumed to be 2)

You can calculate the disk size requirements as follows (bear in mind that the disk space is outside of the Hadoop Distributed File System (HDFS)):

- Disk usage for storing data: DataSize / CompressionRatio
- Disk usage for operations: 0.2 * DataSize
  
  For safety reasons, we assume an additional storage of 20% (this means that full SELECT * queries will not be supported).

 Disk usage = DataSize / CompressionRatio + 0.2 * DataSize = 0.5 * DataSize + 0.2 * DataSize

 Disk usage = 0.7 * DataSize

 Memory usage <= 13 GB per node (depending on the Vora Disk Engine settings)
Vora Time Series Engine

You can estimate the memory consumption of the time series engine as follows:

\[
\text{Memory usage} = \frac{\text{size of the uncompressed data files}}{5} (\text{for storing the data in memory}) + 0.5 \times \text{size of the uncompressed data files} (\text{for operations}) \\
\text{Memory usage} = 0.7 \times \text{size of uncompressed data files} \\
\text{Disk usage} = \text{<no additional disk space needed>}
\]

Bear in mind that this is a very rough estimation. The size depends heavily on the data.

Vora Graph Engine

The memory usage of the stored JSG files (graphs) is roughly three times their disk size.

Currently, the recommendation is that the graph store should not use more than 1 TB of main memory for performance reasons. Data in memory will not be evicted if memory is scarce.

\[
\text{Memory usage} = 3 \times \text{size of JSG files} (\text{for storing the data in memory}) + 1 \times \text{size of JSG files} (\text{for operations}) \\
\text{Memory usage} = 4 \times \text{size of JSG files} \\
\text{Disk usage} = \text{<no additional disk space needed>}
\]

Vora Document Store

The memory usage of the JSON files is roughly the same as their uncompressed disk size. You require twice as much main memory for working on the results.

\[
\text{Memory usage} = 2 \times \text{uncompressed disk size of JSON files} \\
\text{Disk usage} = \text{<no additional disk space needed>}
\]

Additional Requirements

Logs

The following disk space requirements apply to all engines:

\[
\text{Additional disk usage for traces and logs: 5 GB per node}
\]
Table Eviction Disk Space
For table column eviction (in case of memory shortage), it is recommended to have available disk space of the size of the main memory.

Vora Services
The memory and disk requirements of the other SAP Vora services, such as the Catalog, Distributed Log, Landscape Server, Thriftserver, Vora Tools, and Vora Manager, are negligible.

4.2 Worker Nodes
In general, the memory size defined by the Hadoop cluster sizing can be used. If a range is defined, take the high value and assign memory exclusively to SAP Vora.

Example
Your Hadoop cluster sizing recommends 128-256 GB of RAM.
Use the 256 GB and assign 50-80% to be used by SAP Vora. The percentage will depend on what kind of workload you expect: balanced workload, CPU/memory-bound or I/O-bound workload.

Estimating Memory Requirements
SAP Vora memory requirements heavily depend on the workload characteristics. Best performance will be achieved if 100% of the tables managed by SAP Vora can be loaded into memory.

You can roughly estimate your memory requirements based on how many tables will be managed by SAP Vora and their size, since the components of SAP Vora (except the disk engine) load the entire tables into memory.

<table>
<thead>
<tr>
<th>Parameters for estimating memory requirements</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MemoryRequirements</td>
<td>Memory usage of all engines in total in GB</td>
</tr>
<tr>
<td>NodeSize</td>
<td>Worker node memory size in GB</td>
</tr>
<tr>
<td>RatioVora</td>
<td>Ratio of memory size per node reserved for SAP Vora (usually between 0.5 and 0.9, where 0.5 = 50% and 0.9 = 90%)</td>
</tr>
<tr>
<td>WorkerNodes</td>
<td>Number of worker nodes</td>
</tr>
</tbody>
</table>

You might want to add some space to cope with data growth (for example, data growth of 10% per month).
The main variables to accommodate your memory requirements will be NodeSize and WorkerNodes, with NodeSize set to something between 128 and 512 GB.

Example
The memory requirements for all engines is in total 5000 GB. Each worker node is equipped with 512 GB of RAM, of which 50% is reserved for SAP Vora:
WorkerNodes \geq \frac{\text{MemoryRequirements}}{\text{(NodeSize} \times \text{RatioVora})} = \frac{5000 \, \text{GB}}{(512 \, \text{GB} \times 0.5)} = 20

You roughly need 20 worker nodes to keep 100% of your tables in memory.
Important Disclaimers and Legal Information

Coding Samples

Any software coding and/or code lines / strings ("Code") included in this documentation are only examples and are not intended to be used in a productive system environment. The Code is only intended to better explain and visualize the syntax and phrasing rules of certain coding. SAP does not warrant the correctness and completeness of the Code given herein, and SAP shall not be liable for errors or damages caused by the usage of the Code, unless damages were caused by SAP intentionally or by SAP's gross negligence.

Accessibility

The information contained in the SAP documentation represents SAP's current view of accessibility criteria as of the date of publication; it is in no way intended to be a binding guideline on how to ensure accessibility of software products. SAP in particular disclaims any liability in relation to this document. This disclaimer, however, does not apply in cases of willful misconduct or gross negligence of SAP. Furthermore, this document does not result in any direct or indirect contractual obligations of SAP.

Gender-Neutral Language

As far as possible, SAP documentation is gender neutral. Depending on the context, the reader is addressed directly with "you", or a gender-neutral noun (such as "sales person" or "working days") is used. If when referring to members of both sexes, however, the third-person singular cannot be avoided or a gender-neutral noun does not exist, SAP reserves the right to use the masculine form of the noun and pronoun. This is to ensure that the documentation remains comprehensible.

Internet Hyperlinks

The SAP documentation may contain hyperlinks to the Internet. These hyperlinks are intended to serve as a hint about where to find related information. SAP does not warrant the availability and correctness of this related information or the ability of this information to serve a particular purpose. SAP shall not be liable for any damages caused by the use of related information unless damages have been caused by SAP's gross negligence or willful misconduct. All links are categorized for transparency (see: http://help.sap.com/disclaimer).