# Best-Practice Document
Sizing SAP HANA as a Database

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Introduction</td>
<td>4</td>
</tr>
<tr>
<td>2. Building blocks of SAP HANA sizing</td>
<td>5</td>
</tr>
<tr>
<td>2.1 Sizing relevant hardware components</td>
<td>5</td>
</tr>
<tr>
<td>2.2 Learning from source system: previous growth pattern / housekeeping behavior</td>
<td>8</td>
</tr>
<tr>
<td>2.3 When is an expert sizing needed</td>
<td>9</td>
</tr>
<tr>
<td>3. Migration sizing from a NetWeaver source system</td>
<td>10</td>
</tr>
<tr>
<td>3.1 General Procedure for migration sizing</td>
<td>10</td>
</tr>
<tr>
<td>3.2 Specifics for SAP Business Warehouse</td>
<td>11</td>
</tr>
<tr>
<td>3.3 Specifics for SAP Business Suite</td>
<td>13</td>
</tr>
<tr>
<td>3.4 Sizing reports and their interpretation, some tips and best practices</td>
<td>15</td>
</tr>
<tr>
<td>3.5 BW specifics</td>
<td>19</td>
</tr>
<tr>
<td>3.6 Suite specifics</td>
<td>21</td>
</tr>
<tr>
<td>3.7 BW and Suite: Application server sizing</td>
<td>22</td>
</tr>
<tr>
<td>4. Migration Sizing from a non-NetWeaver Source System</td>
<td>24</td>
</tr>
<tr>
<td>4.1 DB Memory sizing</td>
<td>24</td>
</tr>
<tr>
<td>4.2 DB CPU sizing</td>
<td>25</td>
</tr>
<tr>
<td>4.3 Application Server sizing</td>
<td>25</td>
</tr>
<tr>
<td>5. Greenfield sizing with no prior system</td>
<td>26</td>
</tr>
<tr>
<td>5.1 Approaches to sizing</td>
<td>26</td>
</tr>
<tr>
<td>5.2 Planning sizing verification</td>
<td>30</td>
</tr>
<tr>
<td>6. Sizing non-production systems</td>
<td>31</td>
</tr>
<tr>
<td>6.1 Sandbox system sizing</td>
<td>31</td>
</tr>
<tr>
<td>6.2 Development system sizing</td>
<td>31</td>
</tr>
<tr>
<td>6.3 Quality system sizing</td>
<td>31</td>
</tr>
<tr>
<td>6.4 Pre-production system sizing</td>
<td>32</td>
</tr>
<tr>
<td>6.5 Example for a sizing of non-production systems</td>
<td>32</td>
</tr>
<tr>
<td>7. Creating a deployment plan for all production and non-production systems</td>
<td>34</td>
</tr>
<tr>
<td>7.1 Example for a deployment plan</td>
<td>34</td>
</tr>
<tr>
<td>8. Sizing for System Growth</td>
<td>36</td>
</tr>
<tr>
<td>8.1 Sizing for growth</td>
<td>36</td>
</tr>
<tr>
<td>8.2 Sizing for consolidation</td>
<td>37</td>
</tr>
<tr>
<td>8.3 Scaling up and scaling out – general considerations</td>
<td>38</td>
</tr>
<tr>
<td>8.4 Keeping the table growth at a minimum</td>
<td>39</td>
</tr>
<tr>
<td>8.5 Table Partitioning</td>
<td>41</td>
</tr>
<tr>
<td>9. Expert sizing</td>
<td>43</td>
</tr>
<tr>
<td>9.1 Use Case: BW/DWH on HANA with Complex Queries on Raw Data</td>
<td>43</td>
</tr>
<tr>
<td>9.2 Use Case: Extra-large OLTP load</td>
<td>43</td>
</tr>
<tr>
<td>9.3 Use case: Carve out of functionality</td>
<td>44</td>
</tr>
<tr>
<td>10. Sizing verification in production system</td>
<td>45</td>
</tr>
<tr>
<td>10.1 Memory workload analysis</td>
<td>45</td>
</tr>
<tr>
<td>10.2 CPU</td>
<td>50</td>
</tr>
<tr>
<td>10.3 Data growth and disk consumption</td>
<td>51</td>
</tr>
<tr>
<td>11. Further Information</td>
<td>52</td>
</tr>
</tbody>
</table>
Change history

Version 2.0
- Changed focus to SAP HANA as a database only
- Extended chapter for customers new to SAP
- New chapter on migrations from non-NetWeaver based systems
- Considers Tailored Datacenter Integration (TDI)

Version 1.5:
- Disk space distribution formula changed.
- Changed CPU utilization graphs
- Lessons learned on memory allocation added
- Chapter on sizing verification and expert sizing/workload analysis added
- Appendix added

Authors, contributors
- Bernhard Brinkmöller, Peter Schmidt and Susanne Janssen, SAP Active Global Support, IT Planning
1 Introduction

This document describes different sizing procedures for SAP HANA as a database, either as part of a SAP NetWeaver platform or as a non-NetWeaver platform.

Sizing SAP HANA as a database can be quite easy. If you already have a NetWeaver-based system you can execute a sizing report which calculates a value for the amount of memory needed. Based on this value you can choose the adequate hardware from the catalog of certified hardware, which will be delivered as an appliance and per definition will have the required capacity for the different components (CPU, memory, disk, I/O). If you are new to SAP the process is very similar. There is a sizing tool called Quick Sizer which helps you to initially calculate sizing requirements for SAP HANA based on your inputs of transactional data.

<table>
<thead>
<tr>
<th>Starting Point</th>
<th>Sizing for BW on HANA</th>
<th>Sizing for Suite on HANA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Greenfield / new implementation</td>
<td>Quick Sizer questionnaire for BW on HANA</td>
<td>Quick Sizer questionnaires plus additional formula</td>
</tr>
<tr>
<td>Migration</td>
<td>Sizing report for BW on HANA</td>
<td>Sizing report for Suite on HANA</td>
</tr>
<tr>
<td>Documentation in SAP Notes</td>
<td>Sizing Report for BW on HANA</td>
<td>1872170  Suite on HANA memory sizing</td>
</tr>
<tr>
<td></td>
<td>1736975</td>
<td>1793345  Sizing for SAP Suite on HANA</td>
</tr>
</tbody>
</table>
2  Building blocks of SAP HANA sizing

Sizing of a SAP HANA database can involve more than just applying a formula or running a sizing report to determine the memory requirements. Depending on the complexity of the situation and planned system, different factors can be taken into account to amend a sizing result.

2.1  Sizing relevant hardware components

Different to a CPU-based sizing, where capacity is released when not required anymore, memory is allocated and released over time, but cannot be related 1:1 to user or background workload in the same way it is done for other hardware components, such as CPU or network bandwidth.

![Figure 2-1: Mindshift in sizing approach](image)

While the workload profile approach still applies to the application layer, for sizing SAP HANA as a database the approach is quite different.

2.1.1  Memory as the leading driver for sizing

The figure below shows the memory workload of a production SAP Business Suite powered by SAP HANA (SoH) system. The green line indicates how much memory SAP HANA can allocate, in this example around 4 GB. The blue line shows how much memory is actually used, and the red line shows how much memory is allocated. As you can see in the charts, every now and then the SAP HANA database releases allocated memory.

![Figure 2-2: Memory usage and allocation in a production system](image)

The main driver for memory sizing is the table data of the planned SAP HANA system. Most tables are located in the highly compressed column store of SAP HANA. For working memory of the database (various internal data structures, caches and working storage) and temporary calculations, for example, when queries are executed or new data are uploaded, we additionally require the same size. This factor is sometimes called “working memory” or “factor 2x”. Based on an analysis of the source DB, the memory requirement for the table data is calculated as shown in the below figure. The same amount of memory is assumed to be used for temporary working memory.
A SAP HANA database includes further memory areas, such as code, stack, caches, operating system and other system files. These areas are typically small compared to a typical database.

Note: SAP HANA sizing results always consider the total RAM required, unless otherwise specified.

2.1.2 CPU and parallelization

When you compare a CPU profile in a SAP HANA environment with a traditional database, you can see some differences. You will observe the CPU utilization hitting 100% more often, though typically for a very short period of time. The two figures below taken from the same system before and after migration demonstrate the different behavior.

The CPU utilization ("usr" + "sys") of the traditional database is about 50-60% in average peak. This DB was migrated to SAP HANA. The graph below shows the CPU utilization of the SAP HANA database under normal load three weeks after the migration.

The fact that the 100% are often hit does not necessarily signify a performance bottleneck in SAP HANA database which is designed to use all CPU resources available to speed up complex calculations.

In traditional databases sizing includes a safety buffer of an assumed target utilization of 65% per hour. We suggest keeping this buffer as it allows for different kind of assumptions.
2.1.4 Disk requirements
One of the preconditions to ensure that the SAP HANA database can always be restored to its most recent committed state, changes to data in the database are copied to disk. Like for other relational databases ‘redolog’ information is written to log files on disk as prerequisite for a successful DB commit. Besides this, per default, every 5 minutes savepoints are writing data from memory to the data files.

You must always ensure that there is enough space on disk to save data and logs. Otherwise, a disk-full event will occur and your database will stop working.

There are three main storage areas to consider:
- Data files: updated/filled by savepoints and delta merges
- Log File (redo log information): transactional data

The disk space sizing approach is straight-forward and derived from the anticipated memory requirements.

If you use SAP HANA as an appliance, disks are included as per the definition of the hardware. If you decide to deploy SAP HANA by re-using existing storage, the requirements are described in the document on SAP HANA Storage Requirements at http://www.sap.com/docs/DOC-4071. In the current version the storage requirements are about 2.5 to 3 times the required memory.

2.1.5 Network Requirements
Network sizing typically focuses on the bandwidth and is typically described in gigabits per second (gbps).

The graphic below shows the different networks in a SAP HANA installation (scale-out). They can be roughly subdivided into client zone for the application server, the end users and also data loading. In a scale-out implementation or when you replicate to another data center, data is sent over the “internal zone”. The storage zone includes backup network also. Last but not least there is an admin zone which includes virtualization, connection to HANA Studio or the boot network.

![Figure 2-6: Overview of networks in a SAP HANA landscape](image-url)

When you deploy SAP HANA as an appliance, “network” is provided for the internal and storage zones. Client zone and admin zone are within your responsibility.
Should you deploy SAP HANA using the SAP HANA tailored datacenter integration approach, the recommendation for internal zone and storage zone is minimum 10 GBit/s. For more information on network configuration in SAP HANA, including sizing recommendations, see the Web site http://www.saphana.com/docs/DOC-4380 and the SAP HANA Network Requirements whitepaper http://www.saphana.com/docs/DOC-4805

2.2 Learning from source system: previous growth pattern / housekeeping behavior

A look into the growth pattern of the source system is helpful to perform a kind of risk-based sizing approximation in the future. Not all systems grow in the same way, nor are all customers aware of how their system grew in the past or will grow in the future. The available sizing reports calculate HANA requirements based on the current database tables. Users may specify year-on-year growth projection, but the basis for the analysis is always as-is. We have seen customers with very different growth profiles, as the simplified image below shows.

1. Profile 1 is the one we innately assume, a steady profile indicating organic growth, maybe with regular housekeeping, maybe not. This is the one most people have in mind.
2. The saw-tooth profile of companies that perhaps perform housekeeping and data archiving at regular intervals.
3. The profile with a sharp increase is typical for phased go-lives or systems that completely lack housekeeping, data archiving or other data volume management activities.

The main task is not only to determine the HANA size at system start but also to project the growth in the time window set by the customer, typically for the next 2 – 3 years.

Figure 2-7: simplified growth patterns

We also found many source systems had no proper housekeeping or archiving strategy. This means that the sizing projection for SAP HANA includes a lot of waste – double waste, as for each byte of table data another byte is added for the working memory.

A question each customer needs to answer is whether or not they want to use the migration to SAP HANA as a chance to establish regular housekeeping or even archiving. We describe waste and sizing for growth in the chapters on interpreting the sizing reports and sizing for growth.
2.4 When is an expert sizing needed

There are no clear guidelines for situations which require an expert sizing. The list below contains examples for expert sizing approaches. Expert sizing is typically performed on real customer data.

- A customer wants to consolidate three functionally split ERP systems into one. A procedure how to approach this is described in the chapter on expert sizing.
- A customer made an initial sizing for the consolidation of four BW systems, deployed the first system which consumed more memory than anticipated and was afraid the hardware would not suffice. You’ll find more details on this example in the section on sizing for consolidation.
- A customer wants to carve out functionality from an existing system to SAP HANA, add new functionality and add new groups of users. You’ll find more on this in the section on expert sizing – how to deal with multiple influencing factors.
- A first-time customer migrates their high-volume legacy system to SAP HANA. In the case of high-volume migrations we recommend expert sizing to limit the “risks of the unknown”. Typically, we would build a day-in-the-life for the most important business processes, analyze the expected data volumes, measure resource consumption first in single user mode and then in a volume test. As with the vast majority of sizings, the expectation is that this sizing will be delivered in iterations in the course of weeks and months, depending on the phase of the customer project and the criticality of the business volumes.
3 Migration sizing from a NetWeaver source system

In this section we assume you want to migrate an existing SAP NetWeaver system from a traditional database to SAP HANA. Some basics for migrating to SAP HANA have to be considered:

- Not all tables are migrated and by housekeeping in the original system you can not only decrease the memory consumption of SAP HANA but also the migration time significantly. SAP offers a self-service with SAP Solution Manager, SAP HANA Quick Wins\(^3\), which we highly recommend. In addition, there is a documented list of basis tables that are good candidates for housekeeping.
- For example, in BW you can reduce the amount of data to be migrated. Good candidates for reduction of data are PSA and change log tables which will be migrated to the column store on the worker nodes in a scale-out system. Log tables are migrated to row store on the master node. Reduce the data volume of the row store as much as possible.
- You will still need some indexes, but less than in the source database (DB).

3.1 General Procedure for migration sizing

The general sizing procedure is as follows:

1. Perform an initial sizing estimation based on the source production system.
   - A rule-of-thumb type of assessment
   - A detailed approach based on a sizing report which analyzes the source system's data volume, table types, if running on Unicode, among others.
   
   **Recommendation:** Both approaches should be considered complementary; the first as a rough assessment and the second as more reliable and preferable, if the result of the rough assessment indicated physical appliance boundaries.
   
   **Recommendation:** It helps to have a good data volume management already in place; otherwise clean up the system prior to running the sizing report.

2. Consider additional factors such as planned growth or planned functional changes.
   - The sizing report for BW includes a field where customers can enter the planned year-on-year growth. Of course these are only rough estimates.
   - Here is an example why not to underestimate the growth impact: A customer ran the sizing report at the beginning of the year and go-live was planned for a year later. They purchased the hardware. As the project progressed, just before the go-live the customer re-ran the report to validate the requirements against the hardware. By that time the sizing had increased by more than 20 %, because they had forgotten that mid-year a new business unit had gone live and thus needed to add new hardware.

3. Assess the hardware configuration options (scale-up, scale-out)
   - Scale-up refers to deploying SAP HANA in one single, larger, physical host. It is also being referred to as single node. Typically, SoH systems are deployed on one host. Scale-out refers to a distributed system on multiple physical hosts, which is a typical deployment for SAP Business Warehouse powered by SAP HANA (BWoH) systems. For the current hardware configuration options see the Supported Hardware Platforms at [http://scn.sap.com/docs/DOC-52522](http://scn.sap.com/docs/DOC-52522).

As sizing for BWoH and SoH is slightly different, we describe the procedures separately.

---

\(^3\) To contain growth: [http://service.sap.com.sap/support/notes/706478](http://service.sap.com.sap/support/notes/706478) - Preventing basis tables from increasing considerably
3.2 Specifics for SAP Business Warehouse

Rule of thumb sizing for a very initial estimate

The rule of thumb sizing for an initial assessment is a straightforward, so-called “black-box approach”, because you do not analyze what is going on in the system. If your source system is in a good shape in terms of data volume management, apply the following formula:

- Current uncompressed DB Size⁴ * 0.25, where the factor 0.25 is composed of an assumed compression ratio of four, an assumed table footprint of 0.5 and a factor of 2x for the working memory (thus 0.5 * 2/4 = 0.25).
- Customer example: Source DB size: 15.7 TB * 0.25 = 3.9 TB.

This method provides only a high-level estimate of the migrated system size. In a first customer example, however, the estimate was pretty much in line with the result of the more detailed approach.

The sizing result from the above example was not so far off from the memory used by the BW system after migration. The sizing report was also run. It yielded a similar result, but additionally specified the planned row store and column store sizes, which is important for scale-out configurations (because the row store is located on the master node and the column store is distributed on the worker nodes).

Sizing report for a more detailed sizing

The sizing report analyzes the size and the data volume of each table in the database. For technical details such as minimum release or functional scope of the report, see the SAP Note on BW on HANA sizing report⁵.

The excerpt below shows the result of a simple execution that ran for an hour with only 4 parallel work processes⁶. The target size is estimated at 398 GB. The report performs a small “what-if analysis” in the upper section “Phys. Memory per node” by comparing how many nodes you need, assuming different server sizes.

The last line indicates how many nodes (hosts) are needed to meet the requirement. In this particular example, 1 node would be sufficient. Standby nodes are not part of the sizing result.

Note: Not all of the available memory per server can be used by the SAP HANA database! As the operating system also needs some memory, actually only about 490 GB resp. 980 GB can be allocated per server in servers of 512 GB and 1 TB memory, respectively. For example, if the BW system below were to grow by 30 % per year, the 512 GB server would be too small.

![Figure 3-1: Excerpt from a BW Sizing report.](image)

---

⁴ Important: The system must be running on Unicode; otherwise assume a minimum factor of 1.05 on the current disk size.

⁵ Note [https://service.sap.com/sap/support/notes/1736976 - Sizing Report for BW on HANA](https://service.sap.com/sap/support/notes/1736976)

⁶ The number of parallel work processes used by the report for its execution can be defined.
The two lines in the middle indicate the redo log space and data space required on disk, based on the calculated memory. This is only relevant if you want to deploy your system on enterprise storage as part of the SAP HANA tailored data center integration approach\(^7\).

The next section of the sizing report shows the requirements for the data tables and the planned usable memory for the migration in more detail. "Caches / services" in the list below are net requirements which are multiplied by the number of nodes. In this example 50 GB per node are recommended.

![MINIMUM SIZING RECOMMENDATION - CURRENT](image)

The sizing result also gives an estimation of the future growth based on growth rates you specified when configuring the report execution.

As stated before, this BW could be deployed in a single node with a node size of 512 GB or even 2048 GB. The question you can ask yourself now is how much safety buffer in addition to the planned growth you want to have. In case of a 512 GB host you have about 50 GB potential growth buffer, considering the 2x factor for temporary memory. How likely is it that the system would hit this limit? Assuming the buffer was not so generous as in this case, but, say 50 GB, it would make sense to take a closer look at the table details to check, if there is some saving potential by housekeeping – now and regularly in the future system.

As mentioned before, we recommend performing housekeeping on the production system. For further information please refer to the SAP note “BW Housekeeping Tasklist”\(^8\) and read the section on

\(^7\) The sizing for storage in a TDI environment is described in the document SAP HANA - Storage Requirements in Tailored Datadcenter Integration (TDI): [http://www.saphana.com/docs/DOC-4071](http://www.saphana.com/docs/DOC-4071)

\(^8\) Housekeeping task list: [SAP Note 1829728 - BW Housekeeping Task List](http://www.saphana.com/docs/DOC-1829728)

3.3 Specifics for SAP Business Suite

Rule of thumb sizing for a very initial estimate

The average compression rate of a SoH system during a SAP HANA migration is different to BW. One of the main reasons is the existence of pool and cluster tables, which have a lower compression than standard tables. The rule of thumb sizing for an initial estimate for SoH is straightforward and a so-called “black-box approach”, because you do not analyze what is going on in the system. If your source system is well in shape in terms of data volume management, apply the following formula:

- Current uncompressed DB Size\(^9\) * 0.6, where the factor 0.6 is composed of an assumed compression ratio of 4, a factor of 2x for the working memory and a safety factor of 20 % (thus \(2/4*1.2 = 0.6\)).
- Customer example: Source DB size: 2 TB * 0.6 = 1.2 TB.

Limitations: This method helps to get a rough idea on the system size, but does not consider the following for the migrated system:

- In what way the above compression ratio changes if the source DB is compressed – and most DBs are, nowadays.
- Different SAP Business Suite systems have slightly different compression behavior.
- Individual compression factors for different tables – different companies with the same table mix may require different HANA sizes.

Because of these limitations, we recommend using the sizing report for the SAP Business Suite. As an initial assessment the rule of thumb may be helpful, but it should never be used as the only reference.

Sizing report for a more accurate sizing

The sizing report for Suite on HANA analyzes the size and the data volume of the tables known to the ABAP dictionary\(^10\) and assesses their uncompressed size and projected compressed size in SAP HANA. A question that is often raised is whether the report considers DB compression or Unicode. The projections of the sizing reports are independent of both and thus valid also for non-unicode and deep-compressed source systems.

Below please find an excerpt of a test result. The report is regularly being updated and improved, so the screenshot below may be a bit outdated when you read this document. As with the BW sizing report, you can configure the calculation and output of the run, for example, you can configure a ranking of the top ten or top twenty tables that should be listed. We recommend taking the top 20, because it can help to know the relative size differences between the top tables.

---

\(^9\) Important: the system must be on Unicode, otherwise assume a factor of 1.2 on the current disk size.

\(^10\) The report is included in SAP Note https://service.sap.com/sap/support/notes/1872170 - Suite on HANA memory sizing. Make sure you always check for the latest version.
Wherever applicable, the report calculates the size for column store and row store tables which is the basis for the sizing. Based on heuristics, the total table size is doubled and is being referred to as “work space” in the table above (some refer to this memory buffer as “database management” or “temporary memory”). Added to this workspace is the “Hybrid LOB stored on disk”. Large Objects (LOBs) are often found in suite systems and are, for example, attachments, such as a long text in a workflow item. As of SAP HANA SP7, LOBs larger than 1000 bytes are not stored in memory by default, only on disk, and a reference marker to the LOB is kept in memory. By assumption only 20% of the LOB records are cached in memory. Neither LOB cache nor the space for “code, stack and other services” are doubled. The results are always displayed in GB.

**Note:** Please carefully read the FAQ document attached to the report. As the report is evolving to fit the latest HANA development and functionalities it is important to understand how to interpret the sizing results. We especially recommend reading the document to people who have experience with BW sizing as the approaches and assumptions are slightly different.

**Calculating server RAM requirements**

Using the result from above, you need to assess the total memory RAM for the system.

1. Anticipate the year-on-year growth, for example 10 % for the next three years. This leads to a factor of 1.33 (= 1.13).

   Using the above example: 1189.3 GB * 1.33 = 1582 GB

   Please note, this calculation is only a ballpark estimate. There are areas in SAP HANA that do not grow or whose growth is not much affected by the business growth. Examples for this are caches and services.

2. Check the sizing results against the allocation limit of the different possible configurations. The allocation limit is determined by the memory required by the operating system.

   Implementing Suite on HANA on one single node is the preferred option. Scale-out options are only available with restrictions.

<table>
<thead>
<tr>
<th>Node size</th>
<th>Typical Memory allocation</th>
</tr>
</thead>
<tbody>
<tr>
<td>4+ TB</td>
<td>3900+ GB</td>
</tr>
<tr>
<td>2 TB</td>
<td>1920 GB</td>
</tr>
<tr>
<td>1 TB</td>
<td>940 GB</td>
</tr>
<tr>
<td>512 GB</td>
<td>430 GB</td>
</tr>
</tbody>
</table>

**Figure 3-4: Usable memory per node in TB**

If the projected size comes close to meet a border between two configurations, for example of 512 GB and 1 TB (with approximately 430 GB and 9490 GB memory which can be allocated), you should consider the following aspects before deciding on a specific configuration.

---

11 The sizing report also allows for what-if analyses, in case you want to cache more LOBs in memory.
- How well is data volume management done in the system? Is there a lot of “waste” to be found and housekeeping to be done? If you are in doubt, that this can be achieved, it probably makes more sense to choose the bigger machine for more space.
- What was the growth pattern of the system in previous years, steady, exponential? Will this pattern change in future?
- How realistic are the growth assumptions? Are consolidations planned, for example?
- What are the performance impacts, if I move certain tables from row store to column store? This might reduce the memory footprint but negatively impacts performance.

Depending on your decision, for example to clean up the system and run the sizing report again, a smaller or a larger server recommendation could be the result. SAP also offers to license less memory in larger boxes. This makes a later scale-up expansion easier than a later scale-out.

3.4 Sizing reports and their interpretation – tips and best practices

It is an advantage as well as an disadvantage of the sizing reports that that they provide a number, that is, the projected RAM requirements for SAP HANA. Some people take these numbers for granted, ask one or several hardware vendors for a configuration and then they are done. By doing this they are not always doing themselves a favor, as both reports offer sufficient food for thought that may alter an initially conceived target configuration – not always towards the higher end, but sometimes also to a smaller configuration.

In BW, assess different configuration options

In the example below the report allows two different configuration options, with six 0.5 TB hosts or three 1 TB hosts – even a third option might be possible, that is, a single node on a 2 TB host, provided a number of conditions apply. One decision factor whether to go for smaller or larger nodes is the individual sizing result for the master node. The row store – mainly residing on the master node – grows much less than the column store. With an initial result of 265 GB for the master node, there is sufficient headroom even for non-anticipated load. If this master node were to be deployed in a 1 TB node, more than 700 GB “free” memory can be considered as waste. In theory, you could manually shift additional column store tables to the master node to make use of these 700 GB as a contingency.
Figure 3-5: Example result from BWoH (excerpt, version 1.8)

We considered it important to mention this theoretic possibility, but would not have recommended it to this particular customer. If the customer could reduce the overall sizing result by a minimum of 400 GB and if the system is stable and does not grow very much, then even a 2 TB single node implementation could be considered. There are numerous “ifs” in that consideration, but we see a lot of “table waste” in the systems. One big advantage of migrating to SAP HANA is the motivation to establish regular housekeeping and archiving.

Check for the size of the row store and its ratio to column store

This check applies to BWoH as well as to SoH. For scale-out, the size of the master node, thus to a great extent the row store size, determines the configuration. For this check we can also use the example above. Total column store is about 2 TB and row store is calculated at 175 GB. This is a very typical example for BW systems. In another BW example, the following result was calculated:
The row store had 623 GB and total column store result was 16,400 GB, row store thus constituted 5% of the total RAM size. The larger the system, the smaller the ratio for the row store. By the way, in this example, the calculation of the master node sizing requirements of 936 GB comes very close to a hardware limit. You may remember that not all RAM of a host may be allocated by SAP HANA. In 1 TB nodes it is approximately 950 GB. Even though row store does not grow much, the conclusions of this result should be drawn carefully. In this case it would make sense to check if space can be reduced in the row store tables or if 2 TB boxes are the safer choice.

In SoH systems, the row store tends to be similarly small: In the following example 42 GB for row store in HANA and 585 GB for column store.

Only once did we see a difference which led to immediate optimizations by the customer:
The row store was calculated as ~180 GB and the column store at 170 GB. The smaller the system, the higher the proportion of the row store. In this case however, the largest system table by far was DBTABLOG, a basis table logging changes. As SoH is usually deployed as a single node implementation, this analysis is not as vital as for a BW scale-out, but nevertheless can be quite helpful.

**SoH: Tables among the top ten**

When you run a HANA sizing report you may see tables on top that are not on the top of the list in the traditional DB. There are multiple reasons for this. One of them is that some tables are now de-clustered, for example BSEG and KONV from financials. These tables can grow very quickly. Also, at least in previous report versions, there were tables among the top ten which include attachments or large objects (LOBs) such as SOFFCONT1 and DBTABLOG. Here is a result from an older sizing report version where a number of interesting tables are listed:

Figure 3-9: Example result from SoH sizing report (excerpt, version 33)

The first table is from SAP Office and can include email text or attachments. As of HANA SPS 7, not all of the attachments are loaded into memory, therefore, in a recent HANA implementation the size of 630 GB which we see above, will be reduced significantly. Then there are KONV and BSEG as mentioned and a table from General Ledger. These are the tables which we assume will grow quickly. You can see that KONV already has 13 billion entries, which is a lot.

**Focus: which 5 – 10 tables dominate the sizing result**

The example above can serve for this rule-of-thumb: the ratio between the largest and the 10th largest table is 9:1. The total result for column store in this system was roughly 3000 GB. The top ten tables account for about 2000 GB of them. The situation is not always as clear cut as in the system above,

---

12 Other de-clustered tables we often see among the top 10 are: CDPOS and EDID4.
but still it is helpful to take a look at these ratios and consider what benefits different kinds of “space saving” activities might bring.

Addressing some frequently asked questions

**Question: What is the performance impact of running the sizing reports?**

The impact is usually noticeable, but not very high. You can influence the impact by specifying the number of work processes used on the entry screen of the report. A higher degree of parallel work increases the performance footprint at any one time, fewer work processes increase the duration of the report run. As a rule of thumb for planning: a suite report with 10 work processes took about 20 minutes on a 800 GB compressed DB.

**Question: Should I use low, medium or high precision?**

The precision defines how many samples are taken to calculate the sizing result. We found that even low precision tends to give a good result already; the default is set to medium. However, there are exceptions. These exceptions depend on whether you use BW or Business Suite or the source DB or the table mix. In general, we consider low and medium precision to be sufficient for BW.

For the suite we recommend setting the precision level to high, if your source DB is MaxDB and if the first (medium-precision) execution of the sizing report showed that you main sizing contributors are basis tables. If you run the report again with high precision, most of the times, the sizing result will probably be smaller, but again, this depends on the table mix, and there is no exhaustive list.

**Question: Do the reports consider non-unicode implementations or source DB compression?**

As both reports work on the ABAP layer, their results are valid independent of whether or not the source system is compressed and/or on non-unicode.

**Question: What NetWeaver releases and HANA revisions do the reports consider?**

Different target releases have some impact, which is less for the business suite and higher for BW. The reason is that over time, more and more tables are shifted from row store to column store. In a scale-out implementation this will reduce the size of the master node. In general, we do not expect the differences to be so big.

For further questions and answers, see the documents attached to the sizing reports for BW and business suite.

### 3.5 BW specifics

This section will provide some high-level guidelines for BW sizing and hardware configuration. They represent lessons learned from numerous customer interactions and joint discussions with hardware vendors. These may seem obvious at first glance, but there is more than first meets the eye.

1. Understand the sizing result for the master node – housekeeping
2. Understand the impact of the growth on the master node – should be negligible
3. In a scale-out, consider the total number of hosts

**Housekeeping on the master node**

We already mentioned that the master node is composed of row store data and some column store data. The top tables of the master node are typically admin tables, as you can see in the example below.
If a system is stable, that is, not many more structures will be created, these admin tables should be stable as well. We assume that basis tables, such as the printing table TST03, are regularly house-kept. The system below looks well-kept, for example. Of course, it is also much smaller than the one above. What you can observe is that the relative difference between the top contributors and the subsequent tables is not so very large.

Figure 3-11: Example result from SoH sizing report (excerpt, version 1.8)

Because the master node determines the hardware configuration options both in size and in the number of the nodes, housekeeping is an exercise worth performing.

**Decision criteria**

Therefore, one indicator of the status of the master node is in a good shape the relative size difference between the top tables of the row store.

Another criterion asks what type and size the tables of the column store are on the master node. In what way will business growth affect them?

A last question is how mature the system is. If it is mature, few new structures will be created, thus the (main) admin tables will not increase. We assume that a mature system with good housekeeping shows less than 5% growth on the master node per year.

**In scale-out, what node size and how many nodes to choose?**

If the system is subject to business growth, for example 30% per year over three years, then you can assume that these 30% will almost exclusively affect the column store data and thus the slave nodes. If we take the example from the report below, after three years we would assume the following sizes:

- **Master node:** ~ 1400 to 1500 GB RAM
- **Slave nodes:** ~15,000 GB RAM (6958 GB * 1.3³)

Conclusion, the configuration of 2 TB nodes will remain adequate and the number of active slave nodes will increase from 4 to 8 or maybe nine to be on the safe side.
If the system configuration allows choosing between different node sizes, as in our first BW example (see report result) you could plan for 6 nodes with 512 GB or 3 with 1 TB, the first decision criterion should be the space on the master node, the second on the number of nodes.

**Decision criteria for node sizes and numbers**

- If the row store is calculated at 200 GB maximum, you can take 512 GB nodes
- If the row store is calculated at more than 500 GB, take 2 TB nodes
- For good efficiency and operational effort, try to avoid too many nodes. So if you can choose between ten 1 TB nodes or five 2 TB nodes, five 2 TB nodes would generally be the better choice. The main reason is the operational effort and the robustness (less risk of memory shortage on a single node)\(^\text{13}\).  

These recommendations are based on the experience of monitoring small and large production HANA systems and different proof of concept studies. As always, exceptions may apply.

New versions of the BW HANA sizing report address these considerations by specifying “minimum configurations” and “recommended configurations” in terms of the proper number of nodes.

### 3.6 Suite specifics

The preferred deployment option for SoH is single node. So the focus of the report interpretation is more on column store than on row store tables, especially if the system is in-between server sizes. For example, at the moment, vendors offer different models for 1 TB nodes and for 2 TB nodes\(^\text{14}\). Also for SoH, we expect less growth on the row store than on the column store, but as many customers do not perform regular housekeeping, we suggest a planned growth of 20% on the row store and 80% on the column store. To exemplify, we do the following calculation:

\(^\text{13}\) A counter argument may be in instances of well-partitioned data, and considerations of the larger number of cores per dollar in a distributed system.

\(^\text{14}\) Certified Appliance Hardware for SAP HANA: [http://scn.sap.com/docs/DOC-52522](http://scn.sap.com/docs/DOC-52522)
Best-Practice Document
Sizing SAP HANA as a Database

- Assumed yearly business growth: 10%
- Target year for sizing: 3 years from now
- Thus: growth factor is 1.33 (1.1³)
- Original row store sizing result: 128 GB
- Original column store sizing result: 1102 GB
- Projection of row store size in 3 years: 136 GB (128 GB * (1.33 - 1.33 * 0.2))
- Projection of column store size in 3 years: 1388 GB (1102 GB * (1.33 - 1.33 * 0.8))
- Conclusion, the initial configuration of 2 TB nodes can remain.

For Intel Ivybridge CPU based systems, it makes sense to pay particular attention to the following physical considerations. If the sizing result is:

- Close to 1 TB memory. Most vendors offer different models for up to 1 TB configurations and 2+ TB configurations: 4-socket configurations (up to 1 TB memory) or 8-socket configurations (2+ TB memory).
- Close to and more than 2 TB memory. Not all vendors offer configurations with more than 4 TB, and currently only a few, relatively expensive, configurations exist with 6 TB of memory.
- More than double-digit TB. While some servers are available for 12 TB and beyond, if the report yields such a result, we recommend expert sizing. See the respective sections on scale-out assessment and expert sizing.

If you are unsure how much housekeeping can bring, there is a service offering on determining easy wins in SAP HANA. For details see the chapter on keeping database tables at a minimum size.

3.7 BW and Suite: Application server sizing

If the SAP releases of the source application and the target application are identical, and also the functional remains the same, migrating to SAP HANA does not greatly affect the capacity requirements of the application layer.

The main driver for the application servers is typically the CPU consumption.

If a customer, in preparation for the migration to SAP HANA, upgrades to a HANA-ready NetWeaver version, additional CPU capacity is required on the application servers. When upgrading over several releases it makes sense not simply to add capacity as per the upgrade sizing recommendations in the respective SAP Notes15, but to perform a workload analysis beforehand. By doing this, you can also perform some simple overall health checks. This workload analysis is part of proactive performance and capacity management.

If there is a need to upgrade over several releases, the projections become more variable, because not all transactions are affected to the same degree. As a rough rule of thumb you could take the values in the table below. The bigger and more complex the system, the more we recommend expert sizing, especially on the most important business processes and volume drivers.

<table>
<thead>
<tr>
<th>Source release</th>
<th>Target release</th>
<th>Lower upgrade factor</th>
<th>Upper upgrade factor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Web AS 6.20</td>
<td>NetWeaver 7.30</td>
<td>1.2</td>
<td>1.4</td>
</tr>
<tr>
<td>Web AS 6.20</td>
<td>NetWeaver 7.40</td>
<td>1.2</td>
<td>1.5</td>
</tr>
<tr>
<td>NetWeaver 7.00</td>
<td>NetWeaver 7.30</td>
<td>1.1</td>
<td>1.3</td>
</tr>
<tr>
<td>NetWeaver 7.00</td>
<td>NetWeaver 7.40</td>
<td>1.1</td>
<td>1.4</td>
</tr>
</tbody>
</table>

Figure 3-13: Rules of thumb for upgrade sizing (see SAP notes “Resource consumption”)

Independent of the above factors, if your system falls into either or all of the following categories, sizing verification or load testing on custom data is definitely recommended, independent of the migration to SAP HANA:

- Very large (both in terms of the number of application instances and in the total capacity)
- Tightly integrated into mission-critical business processes covering multiple components

Some customers keep their current application servers and others that take the opportunity to purchase new hardware. The only trend is to virtualize the application servers. If a software virtualization layer is added, you can calculate an additional 10% capacity requirement on top of the currently used capacity.

Note that some response time impact can occur when virtualizing the systems. The reason for this is partially the additional layer (for example, hypervisor), but most often non-optimally configured network interfaces and network cards. In case you observe increased response times after a migration without functional or release changes, check the communication path between application instance and database.
4 Migration Sizing from a non-NetWeaver Source System

When you want to migrate data from a non-NetWeaver based system into a SAP HANA system, there are two use cases, SAP HANA will be used as non-NetWeaver standalone application or the source data will migrate into a SoH or BWoH.

4.1 DB Memory sizing

Initial sizing approach

SAP Note for SAP HANA in-memory database sizing\(^\text{16}\) includes a sizing formula for RAM as well as a script to assess the source DB size (exceptions apply). The basic formula is as follows:

- **SAP HANA total RAM = Source data footprint \( \times \frac{2}{7} \times c \)**
  - Source data footprint refers to the total size of the source non-SAP DB’s tables
  - \(2\) is the factor for the temporary memory consumption
  - \(7\) is the assumed HANA compression ratio
  - As many RDBMS compress table data, \(C\) refers to the compression ratio in the source DB.

  **Example:**
  - A source non-SAP database has a relevant size of 2000 GB compressed.
  - The source DB compression ratio is 2
  - Total RAM = \(~800\) GB (2000 GB \(\times\) 2 / 10 \(\times\) 2)

  Of course, this standard SAP HANA compression ratio of seven is an assumption. It may be higher, which is why we recommend verifying the initial sizing assumption by measuring the real compression ratio of the most important tables.

  The compression ratio may also be different in case a non-SAP DB is migrated into a SAP NetWeaver based system. In our experience most customers will have a factor of 4-7, depending on the system and types of tables involved. In this case, a target sizing in form of a range may make sense. Ranges are helpful to determine whether or not the target HANA system may run into the risk of crossing hardware boundaries.

  **Example:**
  - A source non-SAP database has a relevant size of 2000 GB compressed.
  - The source DB compression ratio is 2
  - Result A: Total RAM = \(~800\) GB (2000 GB \(\times\) 2 / 10 \(\times\) 2)
  - Result B: Total RAM = \(~1200\) GB (2000 GB \(\times\) 2 / 7 \(\times\) 2)
  - Result C: Total RAM = \(~2000\) GB (2000 GB \(\times\) 2 / 4 \(\times\) 2)

  The initial sizing range for assessing the needed hardware configuration therefore is 1200 - 2000 GB until further validation by uploading sample source tables into HANA.

Sizing for migrations of non-SAP content to SAP HANA always requires a sizing verification.

Sizing verification

If you want to perform a proactive sizing verification we recommend a test migration of the top 5 to 10 most important DB tables. By doing the verification on a test system or sandbox system you have sufficient time to react to changes in the assumptions.

A reactive sizing verification could happen after the migration using SQL scripts in SAP HANA. This is described in more detail in the chapter on sizing verification.

\(^\text{16}\) SAP Note 1872170 Suite on HANA memory sizing report
4.2 DB CPU sizing

The CPU requirements for migrating to SAP HANA standalone are very difficult to anticipate, as we have no real reference against which to compare. Therefore the sizing in the above mentioned note has the following formula: 300 SAPS per active user / 0.65 for a CPU utilization buffer. An active user is one that consumes CPU power at a given point in time. Very often in sizing, customers overestimate the (overlapping) activity patterns of their end users. Some end users also may perform more or less intensive calculations on DB level.

Consider this formula as an initial estimation that needs verification. The more users there will be on the system, the less likely will this formula be accurate. The decision of whether you invest time into further CPU analysis depends upon the risk of reaching CPU limits. SAP HANA servers with two sockets, for example, deliver round about 60,000 SAPS.

In case you want to verify the CPU requirements a test with the top 5 to 10 “SAP HANA transactions” can be helpful, either within a single user test or a load test.

4.3 Application Server sizing

This only applies if you migrate to SAP HANA on SAP NetWeaver, when you require application servers. If you use SAP HANA as standalone, the end users could access the DB directly. In the first case we recommend implementing the application instances in a virtual environment and using the same capacity as for the non-SAP system initially. Most software and hardware virtualization solutions allow increasing and decreasing capacity on the fly. Another option is that if you already have SAP systems in your environment, you can use the application server sizes of the production systems and an approximation.

If you are completely new to SAP, see also the next section on sizing without prior SAP system experience.
5 Greenfield sizing with no prior system

In this section we describe the sizing of a SAP HANA system for customers without prior knowledge of SAP systems. The sizing exercise is targeted at determining the capacity requirements for the production system’s (PRD) DB server and the application servers, if applicable. We also describe how to size the main non-production systems development (DEV), quality assurance (QAS), pre-production (PRE) and sandbox (SBX). Most often, our SAP HANA customers have a 3-system landscape with DEV, QAS and PRD. Many customers have an SBX system or others a PRE system.

The main procedure of sizing a complete SAP landscape is as follows:

1. Size the production system’s DB. If applicable, size the production system’s application layer and middleware layer
   To give you a better idea on how to do this we describe the sizing approaches in this chapter and give you some examples from the sizing tools.

2. Size the QAS and DEV systems and other non-production systems.
   As these systems will be deployed first, you can also start with this exercise. We describe the procedure in the next chapter.

3. Create a deployment plan of the technical system landscape considering high availability, disaster recovery and other boundary conditions.
   The chapter on the deployment plan contains some examples you might find helpful for your own documentation.

In this chapter we focus on the first item, sizing approaches the production system, with a focus on HANA DB sizing.

5.1 Approaches to sizing

The general approach to capacity management is described in ITIL processes. It all starts with understanding the main business transactions and applications. In this section we describe two basic approaches which both have their validity depending on the actual customer situation. Small and non-critical systems, and situations in which the customer has a certain flexibility in the hardware deployment allow for a pragmatic approach. Systems which need to serve complex business processes with tight response time or throughput requirements and those where there is a lot of custom development need more care in the sizing approach.

A tool that has proven to be very effective in an initial is the creation of a day-in-the-life scenario, ideally when the business blueprint is already finished and there is a good knowledge of the key business processes. Please note that this document only contains a rough overview of this approach.

For more details see the best practice document on proactive performance and capacity management.

5.1.1 Create a day-in-the-life from a purely business view

This exercise helps at understanding the potential performance-critical processes as well as prioritizing any subsequent expert analysis.

You choose the assumed busiest period of the year and document what main processes happen at what point in time in parallel. Typically, 5-10 such processes will dominate the overall capacity requirement. It is important to also look across systems, so that synchronous or asynchronous communication steps are considered. It is also good to create a 24-hour profile to understand if

17 https://websmp207.sap-aq.de/~sapidb/0110003587000001231382011E
different processes with maybe different characteristics (online usage in parallel to massive batch processing, for example) overlap.

<table>
<thead>
<tr>
<th>No.</th>
<th>What</th>
<th>Where</th>
<th>When</th>
<th>Occurrence</th>
<th>Volume</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>ECC_01</td>
<td>BW Extraction</td>
<td>ECC</td>
<td>03:00-05:00</td>
<td>Daily peak</td>
<td>~30 mio records</td>
<td>To BW</td>
</tr>
<tr>
<td>ECC_02</td>
<td>Online users</td>
<td>ECC</td>
<td>14:00-16:00</td>
<td>Daily peak</td>
<td>1000 avg., 500 in peak</td>
<td></td>
</tr>
<tr>
<td>ECC_03</td>
<td>Planning runs</td>
<td>ECC</td>
<td>10:00-18:00</td>
<td>Daily avg.</td>
<td>Triggered by users. Small peak overlapping with peak users</td>
<td>To SCM</td>
</tr>
<tr>
<td>ECC_04</td>
<td>Period end closing</td>
<td>ECC</td>
<td>00:00-23:00</td>
<td>Periodical</td>
<td>20% on top of average load, base noise</td>
<td>To tax system</td>
</tr>
<tr>
<td>ECC_05</td>
<td>Warehouse processes</td>
<td>ECC</td>
<td>17:00-21:00</td>
<td>Daily</td>
<td>avg.: 20,000 transport orders per day, in peak 45,000</td>
<td></td>
</tr>
<tr>
<td>ECC_06</td>
<td>Pricing calculations</td>
<td>ECC</td>
<td>21:00-23:00</td>
<td>Daily</td>
<td>3,000 products</td>
<td>To CRM</td>
</tr>
</tbody>
</table>

Figure 5-1: Example for day-in-the life of key processes to determine the anticipated peak time of the day

From the exercise above you can make the following deductions that affect performance and capacity management. Very likely:

- For end user performance experience the most critical time will be between 16:00 and 18:00, when there is an expected concurrence of high end user activity (ECC_02), planning runs (ECC_03), warehouse processes (ECC_05) and occasional period end closing (ECC_04).
- Application server CPU and memory sizing will be dominated by these processes (include these processes in a performance test once they are functionally stable).
- It needs to be investigated if these processes could be memory intensive on SAP HANA.
- Some of the processes may be more intensive on the application layer, others on the DB layer.

The next steps are to analyze the planned data volumes in more detail.

5.1.2 Attach anticipated volumes to the steps (throughput requirements)

Volumes in this context can be the anticipated number of concurrently active users or the number of specific business documents they create or modify. Or it can be the number of automatically uploaded business documents. We need the data volumes for the peak day for application server CPU sizing and, in addition, the volumes per year for SAP HANA memory sizing. Again, the top 20 document types should be sufficient to get an approximation of both the CPU capacity requirements and the HANA RAM. For a better approximation of space it is good to specify the retention time of the documents, how long do they need to reside in the DB, for 3 months, 3 years or 30 years? Retention times are different for different business documents and, sometimes, in different countries.

<table>
<thead>
<tr>
<th>No.</th>
<th>What</th>
<th>Where</th>
<th>Yearly volumes</th>
<th>Retention</th>
</tr>
</thead>
<tbody>
<tr>
<td>ECC_04</td>
<td>Period end closing</td>
<td>ECC</td>
<td>20 million financial documents</td>
<td>6-12 months</td>
</tr>
<tr>
<td>ECC_05</td>
<td>Warehouse processes</td>
<td>ECC</td>
<td>50 million logistics related documents (transport orders, goods issues, shipments, stock orders, ...)</td>
<td>2 months to 3 years</td>
</tr>
</tbody>
</table>

Figure 5-2: Example for HANA-memory sizing relevant yearly business volumes

In the example above we determined the business documents that will likely dominate the capacity requirements for SAP HANA memory sizing.

5.1.3 Determine response time requirements (optional)

The next exercise is important for all customers who have very specific performance requirements.

With response time, two dimensions are meant: the first dimension is the end user response times. For example, it could be: “After saving an order on our Web Site using a mobile device, the end user should receive the e-mail with the order confirmation from our Web Site within less than 1 minute.” Another dimension of response time refers to processing times or background processes or batch chains that need to finish at a very specific time of the night, for example.

In our example, very likely the planning run process (ECC_03) will have specific response time requirements for the end users.
5.1.4 Size the main processes – apply sizing guidelines

In this step, you assign “price tags” to the day-in-the-life scenarios. For small and simple systems you can do so using SAP standard sizing tools such as Quick Sizer, where you enter the business volumes in the pre-determined business document lines and obtain an initial result. However, please note that the Quick Sizer assumes very streamlined processes so that the sizing result will often seem quite low. You need to include add capacity buffers for custom development, interfaces and so on.

For large and complex systems using the standard sizing tools is not enough, here an expert sizing required for at least the performance-critical processes and those with an anticipated high data volume.

Even if you do not use the results of the standard guidelines, these tools can be quite helpful in that they provide a structured approach to the typical key business processes in an application.

5.1.5 Using Quick Sizer to determine the SAP HANA DB size and application server CPU capacity requirements

5.1.5.1 SAP HANA DB sizing in Quick Sizer

There are two main flavors for sizing SAP HANA as a database, BW on HANA and Business Suite on HANA.

Sizing BW on HANA

If you do not migrate from an existing SAP BW system but build BWoH from scratch, you can use the Quick Sizer which provides a sizing questionnaire for BW on HANA. The graphic below partially shows the questionnaire where you can specify the size of one or more InfoCubes and Data Store Objects (DSO). In addition, you can specify active users and the amount of data uploaded into the BW system for CPU sizing.

Figure 5-3: Quick Sizer Questionnaire for BW on HANA

Figure 5-4: Quick Sizer result for above entries
The sizing procedure is analogous to sizing a BW system for a traditional database, only the results are already converted into a SAP HANA result.

In the example above, the most important result is the 1.2 TB RAM. Currently the largest physical server for single node BW on HANA is 2 TB\(^{18}\). This means you need to think about deployment options. For more information on deployment options in SAP HANA, see the document “Technical Deployment Options for SAP Systems with SAP HANA”\(^{19}\).

**Sizing Suite on HANA**

If you consider implementing your business suite or business warehouse directly on SAP HANA, the sizing tool of choice is the Quick Sizer. As of September 2014 SAP offers a SAP HANA-flavored Quick Sizer which has the same look and feel than the Quick Sizer for traditional databases. In the Quick Sizer SAP HANA version you can perform a standard throughput-based sizing with realistic retention times of the business data. User-based sizing does not make sense for SAP HANA memory sizing, as the user-based sizing algorithms for Quick Sizer are very simple and do not consider growth, for example.

The Quick Sizer converts your entries in the questionnaires into RAM, CPU\(^{20}\) and disk sizing requirements. For example, the result of Quick Sizer project “SoH sizing example” (customer number 188213) calculates about 1.7 TB RAM as you can see in the screenshot below:

![Joint Quick Sizer result for SoH and BWoH](image)

**Understanding the boundary conditions**

- You can see warning. Results above 1 TB are marked as XXL result where you should get in touch with SAP or the hardware vendor for more analysis.
- Quick Sizer does not consider custom tables and custom parameterization. Tables which we often see in production systems (see also the section on interpreting the sizing reports) are anticipated only by an offset.
- The “DB disk” space is not relevant. In a SAP HANA appliance, disk space is included, anyway. For TDI implementations, there is a storage sizing algorithm included.
- Quick Sizer often provides small CPU sizing requirements on the database because it does not consider custom development, system integration (unless you do a workaround), for example.
- For large complex systems an additional expert sizing is recommended.

---

\(^{18}\) For certified hardware configurations for SAP HANA confer with [http://scn.sap.com/docs/DOC-52522](http://scn.sap.com/docs/DOC-52522)


\(^{20}\) SAPS is a unit to describe the processing power of any given hard- and software configuration in the SAP world. The values quoted here are derived using measurements with traditional DBs and not SAP HANA.

5.1.5.2  Application Server sizing

For the application servers we recommend a tiered approach. We recommend deploying the application instances in a virtual environment in which you can dynamically deploy hardware as it is needed.

*Using Quick Sizer*

This is the easiest approach if you already filled in Quick Sizer. Then you can just take the CPU requirements in SAPS for the application. As of today, 2000 SAPS equate to roughly CPU core in a modern Intel-based processor, give or take 10-15%. If you used the throughput sizing approach as recommended above, you’ll get an average sizing result for the application server, which is the minimum.

*Pragmatic approach*

If you have a similar application as the new one already running in your environment, you can take the overall capacity of this one as a reference and adapt to the real requirements either after proactive sizing verification or in production – if you have phased go-lives, for example by business units.

5.2  Planning sizing verification

As mentioned above, Quick Sizer calculations are based on SAP standard and do not consider custom-specific implementations, or interface load or additional development. Again, if you have a small system with few modifications, almost no custom development or no special response time requirements, you can do the verification in production. Otherwise we recommend proactive performance and capacity management.

This you can do as soon as the implementation has reached a functional stability. This means, that you can perform tests on business processes where the main functionality is as close to production state as possible.

You can do this based on your day-in-the-life exercise described above. Take the main business processes and measure:

- In single user tests: End-to-end (E2E) response times for main business processes. Are they as expected?
- Also in single user tests: measure the CPU resource consumption
- In volume tests: “age” the system with business documents to check the memory requirements by SAP HANA. If the business process is implemented across multiple systems, in volume tests you should treat each system individually first before you do an end-to-end test with all main systems.

Example

The *sizing verification* in a production system is described in a separate chapter.
6 Sizing non-production systems

In this section we describe the sizing for non-production systems such as sandbox (SBX), development (DEV), quality assurance (QAS) and pre-production (PRE) systems. The sizing describes the overall capacity requirements. Please note that the strict CPU-memory ratio required for the production systems does not apply to the non-production systems.

6.1 Sandbox system sizing

The sandbox system is like a training field where SAP business content is created for evaluation purposes. Usually, these systems are small and temporary. Sometimes they are based on a SAP model company, sometimes customers migrate a part of their production DB into the sandbox.

Guidelines
- **RAM**
  - If you use a SAP model company, the DB size should not exceed 128 GB.
  - On application server: 8 GB RAM per CPU core as a minimum. Hardware vendors typically deploy 12 or 16 GB RAM per core as a default.
- **CPU**
  - Database CPU has the rough guideline: 4-8 active users per CPU core
  - Application CPU has the rough guideline: 8 active users per CPU core or 4000 SAPS for only a few users.

6.2 Development system sizing

Most development systems have an artificial database created through testing, so their size, even for large systems, does not usually exceed 256 to 512 GB HANA size. There are exceptions, for example, when customers want to test on a large amount of data also in the development system. In this case, the data volume has to be considered as well. Typically, however, DEV systems tend to be small. For the recommendations below we assume up to 50 developers.

Guidelines
- **RAM**
  - 256 GB RAM
  - On application server: 8 GB RAM per CPU core as a minimum. Hardware vendors typically deploy 12 or 16 GB RAM per core as a default.
- **CPU**
  - Database CPU has the rough guideline: 4-8 active users per CPU core
  - Application CPU has the rough guideline: 4 active developers per CPU core

6.3 Quality Assurance system sizing

Quality systems have very varied usages, such as interface testing, unit testing, or regression testing, for example. Some customers populate the QA system with a fraction of the (scrambled) production data volume, others use a full DB copy, which requires a production-like sizing of the DB. In our guideline we refer to typical use cases with up to 80 users in total and assume a fraction of the production size.

Guidelines
- **RAM**
  - 512 GB RAM as a typical size, but can be higher if production data are migrated into QAS
  - On application server: 8 GB RAM per CPU core as a minimum. Hardware vendors typically deploy 12 or 16 GB RAM per core as a default.
Best-Practice Document
Sizing SAP HANA as a Database

- CPU
  - Database CPU has the rough guideline: 8-16 active users per CPU core, unless the system is used for load testing. In this phase we recommend 4 users per CPU core.
  - Application CPU has the rough guideline: 12 users per CPU core, unless the system is used for load testing, then we assume 4-8 users per CPU core.

6.4 Pre-production system sizing
In pre-production systems the focus is usually on the quality of the data in the DB. There are very few users and very few tests done. In the vast majority of cases, the pre-production system will be a copy of the production system. There are, however, customers who keep a subset of the production data, for example using SAP Test Data Management System (TDMS).

Guidelines
- RAM
  - Size equal to the production system RAM
  - On application server: 8 GB RAM per CPU core as a minimum. Hardware vendors typically deploy 12 or 16 GB RAM per core as a default.
- CPU
  - Database CPU: You can configure less CPU than in the production system
  - Application CPU has the rough guideline: 2 to 6 CPU cores or an equivalent of 4000-12,000 SAPS in one application instance

6.5 Example for a sizing of non-production systems
This is an example for the sizing of non-production systems for a mid-size company and a Suite on HANA system. The production system has been sized with 4 TB of RAM.

Typically, we do not size the CPU for the non-production systems in SAP HANA.

<table>
<thead>
<tr>
<th>Type</th>
<th>Planned Usage</th>
<th>DB RAM and CPU</th>
<th>Application server CPU and RAM</th>
<th>Comment</th>
</tr>
</thead>
</table>
| SBX  | Functional testing using a SAP model company | - 64 – 128 GB  
- Approximately 4-8 users per CPU core | - 2 CPU cores or an equivalent of 4000 SAPS.  
- Total 16 GB RAM | Assuming an SAP model company. |
| DEV  | Development | - 128 – 256 GB  
- Approximately 4-8 users per CPU core | - Rule of thumb: 4-6 active developers per CPU core. 20 active developers means 5 CPU cores or an equivalent of 10,000 SAPS  
- 8 GB RAM per core | |
| QAS  | Unit testing, Interface testing | - 256 – 512+ GB  
- Approximately 4-8 users per CPU core | - Rule of thumb: 8-12 users per CPU core. 20 active developers means 5 CPU cores or an equivalent of 10,000 SAPS  
- 8 GB RAM per core | If QAS is also used for load testing, you will temporarily need more hardware. This depends on the planned test volume. |
| PRE  | | - As in PRD (4 TB in our example)  
- CPU as in PRD | - 1 application instance is typically sufficient. Allocate 10,000 SAPS or an equivalent of 4 CPU cores  
- 8 GB RAM per core | |

Figure 6-1: recommended sizing for non-production systems
For more information on the hardware configuration for non-production systems see also the blog on cost-optimized SAP HANA infrastructure for non-production systems\(^{21}\).

### 6.6 Co-Deployment of Failover Systems and other Non-Production Systems

When you size a landscape, additional factors need to be considered, for example high availability and disaster recovery measures (HA/DR). The [document on SAP HANA deployment options](http://www.saphana.com/community/blogs/blog/2014/08/12/cost-optimized-sap-hana-infrastructure-requirements-for-non-production-usage) is a good source of information.

If you decide to assign multiple use cases to hardware, for example by co-deploying standby hardware with temporary or permanent non-production systems you need to consider this impact on the hardware capacity and the system configuration.

When we talk about co-deployment of non-production and failover systems, we refer to SAP HANA system replication, not host auto failover, which constitutes a special case.

**Impact on capacity**

The first rule is a general one on sizing and it says that in principle all individual requirements add up to a target sizing result. For example, a production system was sized at 850 GB, and a QAS system was sized at 240 GB. If you plan the physical host to hold both the failover system and the memory-loaded QAS system in parallel, the target sizing means 1090 GB or a 2 TB server.

The second rule is that when you use system replication, not all data needs to be held in memory all the time. This means for the above case, you could configure memory for the QAS at 240 GB and assign the rest of the memory to the production failover system.

In case of failover SAP HANA uploads all data required, row store in parallel to column store data. As these kinds of failovers are manual you can switch of the QAS system and thus “free” memory on the host so that 1 TB overall RAM should be sufficient in our above example.

If different systems share a physical host, make sure you have sufficient storage space, which you again need to consider separately and add. If the 1 TB host includes both systems, you require 850 GB + 240 GB for data volume\(^{22}\).

---


\(^{22}\) See the storage requirement document for more information: [http://www.saphana.com/docs/DOC-4071](http://www.saphana.com/docs/DOC-4071)
Creating a deployment plan for all production and non-production systems

Especially when you plan for a system landscape (with multiple applications and their non-production system) it makes sense to document the sizing results and map them to the different hardware layers. It is also a valuable tool to document any assumptions made.

Ideally, a deployment plan contains the following data:

- Assumptions (sizing, SAPS valuation), boundary conditions
- The hardware deployed, especially when you are dealing with hardware of different processing power.
- When you use virtualization technology, you can document the initial guaranteed or minimum capacity or the potential maximum capacity for the virtual machine – to make sure the physical boxes are initially well populated.
- Which system are you dealing with (ERP, BW, PI, etc.), is it PRD, QA or DEV, what is the SID?
- We suggest one line for each layer (DB, Central Services, Application)
- In the columns you can specify CPU, memory and optionally disk and IOPS. Note: in virtualized environments do not build a simple sum overall KPIs, by e.g. totaling the number of SAPS. This result will be too high.
- Optionally, you can add the respective deployment for the secondary site. You can either include it in the same table or have a second one.
- Important: for each KPI include the guiding principles or assumptions for the calculations

7.1 Example of a deployment plan

In the table below you can see a simple initial deployment plan for two systems in a 4-system landscape for DEV, QAS, PRE and PRD. It includes SAP HANA RAM size and disk as well as CPU for HANA DB and the application servers. It basically includes the sizing requirements for each system, each system type and each technical layer.

![Table Example of deployment plan](image)

In a next step, you would map the sizing requirements to the planned hardware, assess if systems can share one physical box. The hardware plan would also consider requirements for high availability and
disaster recovery and can be typically found in the technical design documents of the implementation project.

The final deployment plan should also include the number of standby nodes planned in case of scale-out implementations.
8  Sizing for System Growth

In this chapter we share experience how to deal with data growth over time. The most interesting aspect here is what happens if the system outgrows the hardware boundaries, both in a single node or in a scale out deployment.

8.1  Sizing for growth

In this section we describe how to size for two use cases: “organic” year-on-year business growth and additional data volume created by adding another business unit or another system.

Note: These approaches are only a rough approximation for planning hardware expenditures. They do require verification through measurements such as by sample migrations and checking the resource consumption in production.

As a rule of thumb, we can say that as long as the projections are in the range of a maximum of 20-30% the risk of wrong sizing is not too high, unless the system sizes reach hardware limitations. With higher percentages projections should only be made based on actual measurements.

8.1.1  Sizing formula for year-on-year growth

If you have a mature system, you can take indicators such as the Early Watch Alert to obtain the organic growth over the last 12 months. The percent can serve as a basis for SAP HANA and the sizing projection is very straight forward.

The example below shows the growth over the last twelve months of a SAP Business Suite system, it is about 11%.

![Excerpt from EWA report](image)

Figure 8-1: Excerpt from EWA report

The sizing result for this system based on the sizing report was 6 TB. Assuming a target sizing for the next three years the planned sizing will be: 8.2 TB HANA RAM = 6 TB * 1.11^3.

The respective sizing formula is: HANA RAM = HANA_SIZE * year-on-year to the power of the number of years into the future.

While this formula: data * (1 + growth%)
planned_years is very straight forward, it may be inaccurate because it assumes that all tables grow the same way. While we recommend projecting the growth in this way, because it will be sufficient most of the times and good enough for small systems, we also suggest performing an expert sizing if at least one of the following situations applies:

- The database size is dominated by a few very fast-growing tables.
- The “simple” approach is over multiple years and includes an average yearly growth assumption of more than 30%.
- The sizing result is close to a hardware boundary, e.g. when the result means that you need to move from a single host implementation to scale out.
In a single node the memory cards need to be exchanged with more powerful (and expensive) ones.

The hardware needs to be changed for another model (currently, 1 TB on 4 sockets and 2 TB on 8 sockets are on different hardware).

Expert sizing in this context means analyzing in more detail in what way the business growth would impact the volume drivers for their impact, for example, the largest and fastest growing tables.

The sizing reports do allow growth projections, you can specify in advance how much year-on-year growth you assume in % for a specific number of years. The calculation algorithm is quite similar to the one above. However, also for the reports the same rules apply as for the rule of thumb: exceptions apply.

### 8.1.2 Sizing for additional business units or load

For this scenario there are also two typical use cases: the additional business units are “known” in terms of their relative weight or a new load factor needs to be approximated.

The first use case is straight forward. In many business suite systems the volume drivers can be found in financials or sales tables. Other business document types are also possible. If you can build a relationship between the number of business documents in the receiving system and the relative number of business documents of the incoming load.

Building on the previous example: In the system we showed in the previous graphic, a further analysis shows that the volume driver is sales and the relative table is VBAK. VBAK therefore represents 8.2 TB of HANA RAM in three years. If the sending system has 15% of the entries in VBAK, you can add the relative amount of memory to 8.2 TB. The sizing projection is then 9.4 TB RAM. It must be understood that numerous boundary conditions apply: a likeness in the business process design, a similar age of the data, similar maintenance and so on.

If, for example, the sending system is not based on SAP NetWeaver, the approximation is more difficult. You could use the method described in the chapter on non-NetWeaver based systems. If your business department gives you some high level estimates in terms of relative business acumen (by saying, for example: “the new business generates about 20% of the current system”), you can use a factor of 1.2 to calculate the projected capacity. Of course this is a rather simplified approach. Nevertheless, you can use it as an early indicator and verify it by measurements.

### 8.2 Sizing for consolidation

In case you want to consolidate multiple systems into one, the results of the sizing reports do not necessarily add up. You can do that, of course, add all the results of all the sizing reports. However, by consolidation, also some duplicates are removed.

This is best explained in an example. A customer was unsure if their production system BW on HANA system would be able to include the load of three additional BW systems. In the production system (Receiving BW) about 3.5 TB had been deployed on 7 nodes with 512 GB each, and the workload of the production system was consuming about 1 TB. They ran sizing reports for all BW systems, including the production one, and added the results up, like in the table below.

<table>
<thead>
<tr>
<th></th>
<th>Master node</th>
<th>Slave nodes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RS size by report</td>
<td>CS size by report</td>
</tr>
<tr>
<td>Receiving BW</td>
<td>150</td>
<td>200</td>
</tr>
<tr>
<td>BW1</td>
<td>174</td>
<td>480</td>
</tr>
<tr>
<td>BW2</td>
<td>216</td>
<td>300</td>
</tr>
<tr>
<td>BW3</td>
<td>116</td>
<td>160</td>
</tr>
<tr>
<td>Total</td>
<td>656</td>
<td>1,140</td>
</tr>
</tbody>
</table>
Figure 8-2: Example from a consolidation sizing

As the row store size is the most important driving factor for the hardware decision, the result of 650+ GB for the row store plus 1140 GB for the column store on the master node showed that 512 GB nodes would not be sufficient any more, but rather 2 TB nodes.

However, when we analyzed the sizing reports more closely, for example the column store tables on the master node, we saw duplicates and tables that can be easily deleted, also in the production system.

Figure 8-3: Excerpt from a sizing report result for one of the to-be migrated systems

In fact, especially when you consolidate BW systems, the row store will not usually increase, only when you define new structures. As a rough assumption, we assumed that on the master node, each new system will only add 10% more memory in maximum. The customer additionally specified that only 70% of the slave node size will be migrated. So the result looked quite different. We assumed 10% per system for master node.

Figure 8-4: Second example from a consolidation sizing

The result shows a tight projection. In theory, SAP HANA can allocate about 490 GB on a 512 GB node (table data plus temporary data). In scale-out, the master node is the limiting factor. Also, the master node is not usually affected as much by growth as the slave nodes. However, we made a fairly crude assumption. For the slave nodes, definitely 1 or 2 nodes need to be added – provided they really migrate 70% of the current load.

8.3 Scaling up and scaling out – general considerations

Suite on HANA systems are usually deployed as a single node or a single physical host. Many BW systems are deployed across multiple hardware nodes, in a scale-out configuration. When projecting the future size of the SAP HANA system it is important to understand how hardware and software can scale.

8.3.1 Scaling the hardware

In this section we describe three typical questions we receive from customers when we discuss the future HANA size. For details on HANA deployment options, see also the best practice document on deployment options (login required).

8.3.1.1 Scale-up: Impacts of adding memory cards

In theory you can deploy a node with 8 sockets and gradually add capacity starting with 2 TB, then change to 4, 6, 8 or 12 TB by adding or changing the memory cards. Of course, this is subject to certified hardware, more memory may impact network and storage configuration, how this is done varies strongly between the different hardware vendors. In general we can say that large memory cards for single nodes can become quite expensive.
A number of customers started with a smaller configuration for SAP HANA and added capacity as required. This is especially advisable if the future growth pattern relies on many assumptions. Other customers started with a larger hardware configuration but had much less memory activated.

### 8.3.1.2 Scale-out: Impacts of adding a node

For OLAP activities, e.g. in BW, it is important to have a sufficient number of CPUs to be able to get fast and highly parallel running index scans. Therefore the requirement for 1 TB nodes is to have 4 CPU sockets and for 2 TB to have 8 sockets of Ivy Bridge CPUs. This means scale-up is only a very limited option in BW. Larger BW systems are running in scale-out configurations. Table distribution and partitioning rules are standardized in BW and are automatically applied during a landscape reorganization.

Some hardware vendors even provide the feature to add new nodes in the scale-out case without downtime. After the node is physically in the server frame, this node can then be added to the HANA DB without stopping business operations in SAP HANA. Then single tables or table partitions can also be moved manually via SAP HANA Studio to the new node. Tables can also be redistributed and partitioned by using the landscape reorg tool from the SAP HANA Studio. In principle this step can be done without business downtime, however based on our experience it is better to run it during a business downtime window to avoid lock situations and performance degradation. Based on our experience the runtime for the landscape reorganization e.g. for a 20 TB SAP HANA DB is about 10 to 15 hours. For smaller databases this time is much shorter.

### 8.3.1.3 Customer experience

A customer’s BW system is in production for more than one year. They started with nine nodes with 512 GB each. 8 of these were active nodes, one was a standby node. Since then the (planned) growth required adding two nodes.

They decided to take two nodes from a HANA QA system. First, they used SAP HANA Studio to isolate these nodes online in the QA system. In this step SAP HANA automatically moves all tables to the other remaining nodes. No data is lost. After reallocating all tables on the other nodes, the isolated nodes were removed from the landscape and no longer visible in the QA HANA DB. This first step took around 3 hours.

The next step was to remove the host on OS level and add it to the productive HANA DB. Then the node was added to the HANA DB of the production system and from now on tables can be placed on the new node. The customer afterwards moved some large tables/partitions manually to this node. All these steps were executed during production uptime without affecting the business operations.

### 8.3.2 Software-related scalability

One way to prepare for growth in SAP HANA is a lean database. Over the last year or two we analyzed many production systems and while the recommendations in the next sections seem very obvious we still find too many systems in which adhering to the recommendations hardware could be saved.

### 8.4 Keeping the table growth at a minimum

Even though SAP HANA compresses data much better than traditional databases, it still makes sense to perform a proactive data volume management. In this section we describe the two most effective mechanisms.
8.4.1 Housekeeping: SAP HANA Quick Wins

The best solution is to get rid of data as soon as possible. The table below shows how much data could be reduced per document type. The graphic is an example from a service called “DVM Best Practice Self-Service”. It is available as of SAP Solution Manager 7.0. The self-service generates a best practice document that describes how to handle your largest data objects using the methodologies of data avoidance, summarization, archiving, and deletion.

![Table showing data reduction](Image)

Figure 8-5: Excerpt from DVM best practice document


8.4.2 Data on disk

By default, SAP HANA does not upload all data in memory. For example, Large Objects (LOBs or Binary Large Objects, BLOBs) are often found in business suite systems and are, for example attachments, such as GIF or PDF. As of SAP HANA SPS07, LOBs larger than ~1 KB are not loaded in memory, but remain on disk. Only a reference to the LOB is kept in memory. The reference itself consumes some memory.

Another example where SAP HANA manages data both in memory and on disk is by “unloading” table partitions to free memory. Unload in this context does not mean that SAP HANA physically moves table columns or partitions to disk – they are already there, in the data volume on disk. Instead, SAP HANA frees memory of least recently used columns and partitions. When the data are needed, they are uploaded into memory.

From a sizing projection perspective, we

- Consider LOBs in the sizing report by assuming that 80% of the LOBs reside on disk (assumption can be changed).
- Do not consider unloading. It is primarily an operational behavior and should not be quantified.

8.4.3 Assessing external storage

If you cannot avoid data growth in the database, you can integrate Sybase IQ with the HANA DB as external storage for Nearline Storage (NLS) in BW or with SAP HANA dynamic tiering for any kind of

[23](http://www.sdn.sap.com/irj/scn/go/portal/prtroot/docs/library/uuid/00ac7012-5a21-3210-45a7-ec1fa58fa0db?QuickLink=index&overrideLayout=true&59575491387829)
HANA application. This concept is mainly built for cold data. Cold data is data which is only accessed very rarely (e.g. historical data).

Another option to avoid physical data replication into HANA is accessing the data via a so-called HANA virtual table, which acts like a view on data in another data source such as an external database. In this case the data physically stays in the source and is fetched when needed. This functionality in HANA is called Smart Data Access (SDA). It is built for reading data from external sources, which e.g. frequently changes in the source and the read performance is not that critical.

Figure 8-6: SAP HANA and external data

8.5 Table Partitioning

When column store tables grow above a certain limit SAP HANA starts partitioning them. The main reason for this is to speed up delta merging whose duration is dependent upon the size of the original table. Additionally there is also a hard limit of two billion records for a single partition of column store tables/objects.

When tables are partitioned there may be some impact on performance and scalability. Performance is improved especially when a partition is spread across multiple nodes in BWoH. There is no impact on sizing, but the question of handling growth often comes up in customer meetings. We separate our lessons learned in table partitioning for growth between BW on HANA and Suite on HANA.

8.5.1 Table Partitioning in BW on HANA

Table partitioning is a standard feature in BWoH. During the migration or when a landscape reorganization in the HANA Studio is performed, tables are partitioned automatically depending on the number of rows in the table and the number of nodes of the HANA DB. The more HANA nodes are installed, the more partitions are created for very large tables. However, as a general principle, the system never creates more partitions than the number of available nodes.

As a basic rule, level 1 partitioning (hash) starts in BW for tables with more than 40 million records:

<table>
<thead>
<tr>
<th>Number records</th>
<th>Number of Partitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt; 40 Mio</td>
<td>1</td>
</tr>
<tr>
<td>40 - 120 Mio</td>
<td>3</td>
</tr>
<tr>
<td>120 - 240 Mio</td>
<td>6</td>
</tr>
<tr>
<td>&gt; 240 Mio</td>
<td>12</td>
</tr>
</tbody>
</table>

Figure 8-7: SAP HANA partitioning in BWoH

Table partitioning does not affect sizing, we only mention it here because several questions came up in customer discussions about planned system growth.
The above mentioned level 1 partitioning is used for all type BW objects and is of the type hash partition.

For some objects (Infocubes and DSO) there is an application-defined level 2 range partitioning defined in BW, which is automatically triggered from BW.

In a nutshell: table partitioning in BW is standardized and automatically applied during landscape reorganization or during the migration. Usually there is no need for manual partitioning of tables.

Further information can be found in:

- SAP Note (1908073 - BW on SAP HANA: Table distribution and table partitioning)

The rules for SoH are slightly different.

### 8.5.2 Table Partitioning in Suite on HANA

While BWoH benefits mostly from an increase in parallelization and load distribution, table partitioning in Suite on HANA can improve the scalability by exploiting a feature called partition pruning. Put in simple terms, it can be described as “do not scan partitions where there can be no matching values”. This statement analysis reduces the overall load on the system and improves the response time for very large tables. Very large in the context of SoH refers to tables with more than 500 million to 1 billion entries.

**Use hash partitioning on the most selective part of the primary key**

SAP recommends using hash partitioning using the part of the primary key that can best guarantee that no partition will exceed the size limit while allowing that most queries find all relevant data in one single partition.

In most business suite systems the largest tables are those containing line item information of transactional data (for example, BSEG, VBAP or VBRP) For these types of tables we recommend using hash partitioning on the document number only, so that information concerning a single document can be found in one partition.

This is a straightforward algorithm, but it cannot be fully automated as in BWoH because it requires some knowledge of the semantics of the fields of the affected table. For more information on table partitioning see also the dedicated document on table partitioning.
9 Expert sizing

The best way to describe expert sizing is using examples.

9.1 Use Case: BW/DWH on HANA with Complex Queries on Raw Data

For BW or HANA native data warehouse solutions consultants or customers very often only apply memory-based sizing. If you require an 8 TB HANA Database, you get e.g. a scale-out solution with 8 * 1 TB nodes, each with 1 TB main memory and likely 60 CPU cores.

The assumption for this sizing is that the customer has a data warehouse which typically consists of many different tables with different sizes serving different application areas (e.g. FI, Logistics, HR, etc.). Some of these tables might be small, others might be larger (e.g. more than 1 billion records). Many different queries are executed accessing different content areas.

In this case the standard sizing usually fits very well.

However the picture changes if the customer builds up data warehouse scenarios, with only a few, but very large tables with very granular data on line item level. An example is a retail customer, who built up a native HANA DWH and stores detailed Point of Sales (POS) data in only one table. This table contains billions of records and all queries on this DWH are accessing this table. As the data is more or less raw data from the POS server, many calculation views have been built on top of this table to calculate reporting relevant KPIs during runtime of the query.

Each query is scanning large data volumes and HANA is calculating additional KPIs. HANA is built and optimized for this type of reporting, but of course many CPUs will be utilized in parallel for each single query.

In this case the standard approach of a memory-based sizing is not appropriate, because very likely you will get a CPU bottleneck if many users are executing queries in parallel.

Here it is required to perform an expert sizing to determine the required CPU setup based on a volume test in combination with a proper partitioning and distribution of the tables on different nodes.

9.2 Use Case: Extra-large OLTP load

The question what makes a business suite system extra-large in terms of Online Transaction Processing (OLTP) load cannot be described in simple terms. As a very basic rule of thumb we could possibly say that systems which consume more than 100,000 SAPS on the DB layer may fall into this category.

The goal of an expert sizing for systems with a very high OLTP load is to assess a potential higher impact on memory consumption caused by indexes. When you implement HANA you have much fewer indexes than in a traditional row-based database. Some indexes, however, remain by default. When dealing with very large OLTP systems indexes may be helpful to improve performance, but will consume memory, especially concatenated indexes.

OLTP load basically consists of a very high number of select statements which utilize highly selective WHERE clauses to return a small result set each from the DB. By SAP default, most of these queries utilize the primary key index or other inverted column indexes to access the DB tables. This is included in the sizing reports.

While secondary indexes are missing that might improve the runtime of less frequently used queries, the scan speed of SAP HANA compensates for this small disadvantage.

If you have selects in custom applications that are executed each time a document is processed and that do not use the primary key, we recommend assessing the benefit of additional inverted column
indexes or concatenated indexes. Particularly the concatenated indexes on the largest tables of a system can increase memory sizing. Therefore, when a few very large tables dominate the overall system load, check the custom access to these tables. If the access pattern suggests that additional indexes are required, you need to add the memory accordingly.

**Inverted vs. concatenated index**

Adding an inverted column index will increase memory consumption of the table around 1-5%. So unless several such indexes need to be added, the effect on overall sizing is negligible.

Depending on how many fields you choose for a concatenated index the memory consumption is typically around 10% for the table.

In most cases the analysis will prove that concatenated indexes that would be needed on a row based DB to support the access will not be necessary to achieve response time goals and can be either dropped completely or can be replaced by an inverted column index on the most selective column.

9.3 **Use case: Carve out of functionality**

We performed an expert sizing for a customer who wanted to carve out their HR implementation from their current ERP system. In addition they wanted to add a bunch of new functionality and increase the HR user base by about 60%. In the following description we focus only on the HANA DB sizing aspect.

9.3.1 **The structured approach to a sizing with multiple dimensions**

1. Determine the table data footprint for the carve-out relevant tables – you can use the SoH sizing report for that.

2. Project the future growth percentage of the current user base-- you can do this based on an analysis of the leading DB tables, using, for example transaction TAANA, which shows you in which month or year a record was created.

3. Determine, if all table content will be migrated or only a part of it, e.g. only the last three years. If it is only the last three years, you can relate the total HR-relevant size to the entries of the leading tables in the last three years.

4. Assuming the new user base will perform the same functionality as the old one, perform a relative projection for the target year, e.g. three years from now.

5. Use standard sizing tools and guidelines for the new functionality of the end user base and perform the black box sizing approach using the formula described in the section on [Suite on HANA sizing](#).
10 Sizing verification in production system

Once a system is in production, it is subject to changes, either based on the business growth, or mergers or functional changes and others. Every once in a while it makes sense to check the workload to re-size the system or to perform an expert sizing.

In comparison to other databases, sizing SAP HANA is easier and more complicated at the same time. It is easier because it is an appliance and with appliances, you need to understand the boundaries and in what way the system behavior reaches these boundaries. It is more complicated at the same time because it is driven by memory usage, and memory usage behavior does not compare with CPU utilization where capacity is released when not needed or disk space which grows and gets reorganized every once in a while to avoid fragmentation.

Black box and white box sizing

If you want to perform an easy sizing, you just check if certain memory boundaries are reached or not. If they are, you scale-up or scale-out, and that's your sizing. The black box approach can make sense in a lot of cases, such as in small and uncritical systems. It bears the risk of sizing waste but is rather straight-forward.

When you want to understand how your system behaves over time, which memory areas drive the workload

10.1 Memory workload analysis for sizing verification

10.1.1 Understanding the relevant memory areas

At the beginning of this document we showed a simplified graph with the memory consumption for table data and dynamic/working memory. Now we would like to take a closer look into the memory areas, which of them are relevant and which are not.

Figure 10-1: Graphic to show the different memory areas

The graphic above shows quite nicely the principle of memory usage. For the sizing purpose “used memory” is the most important one, for determining the performance of a system the allocation lines are important. Note that the lines “HANA code & stack” and “used by your data” are actually included in the “used memory”.

---

The table below gives an overview of the different memory areas and how relevant they are. It includes two additional memory areas which are not depicted in the figure above. The memory areas are shown from top to bottom.

<table>
<thead>
<tr>
<th>Memory Area</th>
<th>What Is It</th>
<th>Relevant for Sizing?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Physical memory</td>
<td>The total memory per node, such as 512 GB, 1 TB, 6 TB, or more.</td>
<td>No</td>
</tr>
<tr>
<td>Allocation limit</td>
<td>Each HANA implementation requires reserved space for the operating system (around 3+ %). This means a server with 512 GB has an allocation limit of around 440, or a 1 TB server has an allocation limit of ~940 GB.</td>
<td>Indirectly</td>
</tr>
<tr>
<td></td>
<td>The allocation limit determines how much memory can actually be used in a node. If memory requirements exceed the allocation limit, tables are unloaded and performance deterioration can be expected.</td>
<td></td>
</tr>
<tr>
<td>Pre-allocated memory</td>
<td>SAP HANA pre-allocates and manages its own memory pool, used for storing in-memory tables, for thread stacks, and for temporary results and other workspace data structures. When more memory is required for table growth or temporary computations, it is obtained from the pool. When the pool cannot satisfy the request, the SAP HANA memory manager will increase the pool size by reserving more memory from the operating system, up to the predefined Allocation Limit.</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>But: If the pre-allocated memory hits the allocation limit, performance degradations can occur. Note: this is sometimes referred to as pool.</td>
<td></td>
</tr>
<tr>
<td>Used memory</td>
<td>Consists of three memory areas visible on this screen, HANA code and stack, table data and dynamic or working memory which is allocated and released when necessary. In addition, used memory is made up of shared and heap memory parts. As of today, row store is in shared memory and column store in heap memory.</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>If you analyze used memory over time and compare it with table data, you can get a heuristic ratio of dynamic to table memory that will help you determine the new sizing. Used memory should never hit the allocation limit, otherwise SAP HANA runs out of memory.</td>
<td></td>
</tr>
<tr>
<td>Used by your data (or table data)</td>
<td>This is table data for row store and column store that have been loaded into SAP HANA. Note that tables are only loaded when needed. When about 80%-90% of the global allocation limit is reached, tables are removed from memory (&quot;unloading&quot;).</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>For sizing verification and to assess the growth rate of the system. Note: It is part of used memory.</td>
<td></td>
</tr>
<tr>
<td>HANA code and stack</td>
<td></td>
<td>Negligible</td>
</tr>
<tr>
<td></td>
<td>Typically this accounts for 2-5 GB</td>
<td></td>
</tr>
</tbody>
</table>

25 In the sizing reports an additional 50 GB are reserved for caches and services per node, but they are not part of the allocation limit.
Not loaded tables and unloaded tables | Not all tables are loaded into memory at system start, but only when required. This means that used memory or table memory may not reflect the actual data growth in the system.

Unloaded tables are not included in the graphic above. When about 80%-90% of the global allocation limit is reached, SAP HANA temporarily removes partitions from memory ("unloading") according to a least recently used algorithm.

Yes (production sizings) | The not-loaded tables have potential impact on the growth of the system. Therefore, to determine the overall memory requirements, the not loaded tables have to be factored in as well.

If no tables are unloaded, you can take used memory as a basis for sizing. If there are unloaded ones, you need to add their size to table data.

In HANA Studio, you may come across the term resident memory. This is the physical memory as seen by the operating system and typically smaller than the used memory. It does not have a sizing impact.

Used memory is the most important KPI. The graphic below visualizes the memory areas that make up “used memory”. Used memory includes shared used, heap used, code and stack and also includes cached data.

### 10.1.2 Examples for memory consumption in production systems

**Used memory**

Used memory includes used shared memory, used heap memory, code and stack. The example below shows the used memory consumption together with the allocation limit in a SoH system. The system is well-filled and still has some room to grow.
You can also see in SAP HANA Studio how much memory is used for column store tables and row store tables.

10.1.3 Analyzing the performance of SAP HANA as a system health check

If you want to do a sizing, it makes sense to check the system behavior in general to see if load can be added to the system easily or if optimizations should be performed.

Here you consider inter-relationships between different KPIs. These KPIs are allocation limit, allocated memory and used memory. Whenever used or allocated memory reaches the allocation limit, the probability of performance problems increases strongly. Performance problems can be caused by unloading of tables and partitions, for example. Out-of-memory situations also affect the end user experience.

The next three graphics from three different systems show an underused, a well-used and a system with performance issues. The KPIs shown are: used memory, allocated heap memory and allocation limit. In this system, at the beginning of the month, some period closing program ran, creating a massive amount of data. Still, there is enough memory available.
tables or partitions have been unloaded to see if there are performance issues. In this case there were very few, less than one per day.

Figure 10-5: Almost fully loaded system

The last graphic is from the master node of a scale-out system and shows overload. At first glance, the graph does not look very dramatic, but you can see that the allocated memory (red) constantly reaches the allocation limit, and even the used memory reaches the allocation limit. The analysis of the number of unloads revealed several unloads per day.

For scale out workload analysis we recommend to separate master node from the secondary nodes. Unless performance problems occur, the secondary nodes can be summed up.

Figure 10-6: Almost fully loaded system

By the way, unloads can also happen in a system without capacity limitations, for example the one below. There were a couple of unloads for several days even if the system was not overloaded. The unloads are displayed on a secondary axis with the number of unloaded tables and partitions. When unloading, SAP HANA follows the least recently used principle.
The above KPIs cannot really be used for performance analysis. For this root cause analysis tools are available in the Solution Manager as well as the SAP internal Ad hoc memory analysis tool.26

One important parting remark: When used or allocated memory hit the allocation limit, performance problems can occur, that is, the likelihood increases. It does not mean they will occur.

10.2 CPU

For CPU monitoring you can use HANA Studio load section. The basis for the CPU utilization as shown below is a statistics table with CPU snapshot data. Each hour has multiple snapshots, and of these, the maximum is taken. If you cannot get CPU utilization data from the Operating System, Solution Manager or HANA Studio, this is the next best approximation and looks as follows. Each bar is a snapshot of one hour. While the average load is somewhere between 10% and 20%, high utilization can occur at times. Because of the way the data is taken by HANA Studio, you cannot compare these kinds of CPU graphs with SAPOSCOL or other tools you may be used to.

---

10.3 Data growth and disk consumption

For snapshot you can look at SAP HANA Studio overview page in the section on disk usage and data volume size. For a historic view, there are also SQL scripts available which analyze how much data was backed up.

The graphic below shows the backup size in GB and backup duration in hours for the last three months. Note that backup is not compressed per default and depends on the sub system. Since then it grew by about 10% from about 2900 to now 3200 GB. This is not a typical growth rate for SAP HANA.

Figure 10-9: Backup size in GB and backup duration in hours
11 Further Information

If you read carefully through the document you will have noticed that SAP HANA is continuously being further developed. Therefore it is important to stay up to date. Below you will find a list of SAP Notes and documentation we found helpful when compiling this document. Whenever you plan to follow any of the recommendations in this paper, validate them against possible new information.

Sources of published sizing documentation:

- [http://service.sap.com/sizing](http://service.sap.com/sizing)
- Access to Quick Sizer
- Access to sizing guidelines, for example HANA accelerators
- [http://service.sap.com/notes](http://service.sap.com/notes)
- SAP Note 1736976 - Sizing Report for BW on HANA
- SAP Note 1855041 - Sizing Recommendation for Master Node in BW-on-HANA
- SAP Note 1829728 - BW Housekeeping Task List
- SAP Note 1872170 - Suite on HANA memory sizing
- SAP Note 1793345 - Sizing for Suite on HANA.
- SAP Note 1698281 - Assess the memory consumption of a SAP HANA System
- SAP Note 1661202 – Support for multiple applications on SAP HANA
- SAP Note 1666670 – Multiple SAP HANA DBs on one appliance
- SAP Note 1786665 - SAP HANA running on VMware vSphere VMs
- [http://service.sap.com/goinglivecheck](http://service.sap.com/goinglivecheck)
- Goinglivecheck for SAP HANA scenarios
- SAP Solution Manager
  - Guided Self Service: Sizing for SAP HANA: HANA sizing rules given in SAP notes and also SAP Quick Sizer. It calculates necessary memory and disk sizes of the planned HANA appliance

Other useful documentation:

- Explanation of the most important Statistic Tables: [https://wiki.wdf.sap.corp/wiki/display/ngdb/Metric+Tables](https://wiki.wdf.sap.corp/wiki/display/ngdb/Metric+Tables)
- How does the Delta Merge work? [https://community.wdf.sap.corp/docs/DOC-206668](https://community.wdf.sap.corp/docs/DOC-206668)

To integrate SAP HANA into the data center may have an impact on your overall infrastructure strategy. For this we created the SAP HANA infrastructure roadmap, which is designed to help you identify and prioritize the actions to design a durable hardware infrastructure for SAP HANA, irrespective of whether is it a new implementation or a migration from a traditional database. As of Solution Manager content package ST-ICO 150_700 SP 36, you will find this roadmap in the SAP Solution Manager, in the Project Administration section, see also the SAP Library - Solution Documentation Assistant Work Center.

There is another roadmap that will help you in the migration process from a traditional database to SAP HANA, which is available as of Solution Manager’s ST-ICO 150_700 SP 33.