Before diving into the SAP BW on SAP HANA migration process, it’s important to understand the type of architecture that SAP HANA brings to the table. Here is an overview of the construction and interaction between SAP HANA’s software and hardware, as well as how the database works in the background. See what special considerations and options should be reviewed when using SAP HANA.
SAP BW on SAP HANA combines the speed benefits of SAP HANA with the comprehensive functions of SAP BW. In particular, reporting and loading processes are accelerated enormously. This chapter introduces you to the basic principles of this innovative product.

1 Introduction to SAP BW on SAP HANA

This chapter introduces you to SAP Business Warehouse (SAP BW) on SAP HANA. First, we’ll describe the differences between this system and other SAP HANA scenarios. We’ll then explain the differences between the side-by-side approach and the integrated approach. The topic of operational analytics and the latest challenges in the SAP BW environment are also discussed. Based on this, the chapter lists reasons for an SAP BW on SAP HANA migration and answers the question of why SAP HANA results in an acceleration in SAP BW.

1.1 Classifying SAP BW on SAP HANA Implementation Scenarios

SAP HANA is an SAP database technology that is designed for high performance. The SAP HANA’s special characteristic is its in-memory approach: all data is stored in the main memory. Processing large data volumes is thus much more efficient than it is with traditional databases that first have to load the data from the secondary memory (hard disk) with significantly longer access times. However, SAP HANA is not only a mere in-memory solution. Traditional databases already have appropriate caching procedures that can also be used to access sections of the dataset very efficiently. SAP HANA provides a new function: the usage of the in-memory technology in combination with additional innovative
software technologies. This includes column-based data handling, the usage of sophisticated compression and access procedures, and the partitioning of database tables. Section 1.4 discusses this in more detail.

In addition to these innovative software technologies, the usage of an appropriate hardware platform is one of the unique characteristics of SAP HANA. To synchronize the software and hardware components in an ideal way, you can operate SAP HANA exclusively on certified hardware platforms (SAP HANA appliance). This approach ensures that the hardware used has the required resources (main memory, cache size, number of processors, etc.) to process as many tasks as possible in parallel. Currently, only two operating systems (SUSE Linux Enterprise and Red Hat Enterprise Linux) can be used with SAP HANA, which allows for additional fine tuning among software, hardware, and operating system. So when you purchase an SAP HANA appliance, the hardware is delivered with optimized operating system parameters and pre-installed SAP HANA software.

Side-by-side approaches
When SAP HANA was announced in spring 2010 and implemented by selected customers in November of the same year, the first decision-makers already recognized that this technology could eliminate fundamental performance problems within the SAP system landscape. At that time, however, no one was ready to exchange a proven database solution of a very young SAP HANA technology. The risk seemed too high that the business could be affected by possibly immature software. Therefore, side-by-side approaches became popular: SAP HANA was used in parallel, or side-by-side, to already-existing systems. The concept of this approach is to continuously replicate the data from a rather slow database to the SAP HANA database and to use SAP HANA for a high performance data analysis. In general, one may distinguish between the data mart and the accelerator approach.

1.1.1 Side-by-Side and Integrated Approaches

Data mart approach
As Figure 1.1 (left side) illustrates with an SAP ERP system, the data mart approach continuously replicates the data from any database (often called AnyDB in literature) to the SAP HANA database. Specific analysis tools can then directly access this data using new interfaces, such as SAP BusinessObjects Analysis Edition for Office from the SAP BusinessObjects portfolio. With this approach, the data evaluation is extremely accelerated due to the powerful characteristics of SAP HANA. Chapter 7, Section 7.2.2 describes how you can directly access SAP HANA views using SAP Lumira and SAP BusinessObjects Design Studio and evaluate the data correspondingly.

Accelerator approach
On the right side of Figure 1.1, the accelerator approach is illustrated. With this approach, the data is also continuously replicated from a database to the SAP HANA database. Here, however, the focus is not the evaluation of the data using specific tools. Instead, certain transactions in the SAP system are adapted in such a way that they use, not the primary database for read accesses, but SAP HANA. On one hand, this leads to a considerable acceleration of the tasks within SAP GUI. On the other hand, this approach doesn’t make it necessary to replace an already-existing database. The disadvantage of this approach is that the data is kept in duplicate and must be updated continuously. Based on the accelerator approach, one of the first solutions marketed was SAP HANA Profitability Analysis (CO-PA Accelerator). This accelerator increases the speed of the profitability analysis within the SAP ERP system in controlling, for example, when using Transaction KE30 (Execute Profitability Report).
Today, SAP HANA is a mature product, and instead of side-by-side approaches, integrated approaches establish themselves in real life. Compared to the data mart and accelerator approaches, the main difference is that SAP HANA does not run in parallel to an existing database solution; instead, SAP HANA is integrated into the available architecture and replaces the old database. Consequently, data must be neither replicated nor retained redundantly. Based on the example of an existing SAP ERP system, Figure 1.2 illustrates how SAP HANA replaces the old database when the integrated approach is used. Oracle, DB2, MSSQL, or any other database is replaced with SAP HANA within the scope of a technical migration. Due to the performance characteristics of SAP HANA, this exchange already leads to a considerable acceleration in the corresponding applications.

Figure 1.2 Illustration of the Integrated Approach

The new programming paradigm (see Figure 1.3), mainly performance-intensive processes are moved to the SAP HANA level for acceleration. You can think of moving the programming logic to a lower level of the database. This is also referred to as code push-down. The application is then responsible for only the orchestration and solely triggers complex calculation operations. The actual calculation takes place directly in SAP HANA at a high speed. Eventually, the application simply consumes the results and forwards them to the presentation layer. The advantage here is that large data volumes do not have to be transferred from the database to the application layer (e.g., an SAP BW application server) first in order to perform calculations there, such as summations. Thanks to the in-memory technology, this or even more complex operations can be performed much more efficiently in the SAP HANA appliance. SAP can therefore successively adapt its own applications to SAP HANA for acceleration.

Figure 1.3 Traditional Against New Programming Paradigm

The code push-down principle, however, only works if you use SAP HANA as the primary database (see Section 1.4.5). If you use other databases, for example, to operate an SAP BW system, the application logic is not transferred to the database level. This also applies if the database that is used alternatively leverages an in-memory technology that is similar to SAP HANA. For compatibility reasons, however, it is not necessary to change to SAP HANA. All databases that have been used can still
Introduction to SAP BW on SAP HANA

SAP BW on SAP HANA is one of the integrative approaches that combines the functions of SAP BW with the speed benefits of SAP HANA. Thanks to the in-memory technology used in SAP HANA, all SAP BW data is directly stored in the main memory. In contrast to SAP Business Warehouse Accelerator (BWA; see Chapter 3, Section 3.1), reporting is thus accelerated for all SAP BW data. In combination with the other software innovations, such as column-based data handling, you no longer have to implement preaggregations for complex detailed analyses. The time effort for transformation and load processes in SAP BW is considerably reduced with SAP HANA because the code push-down principle is also applied for SAP BW. For example, the sometimes rather time-consuming process of activating data packages in DataStore Objects (DSO) for the generation of delta records was moved to SAP HANA. Chapter 5 describes further advantages and innovations of SAP BW on SAP HANA.

1.1.2 Operational Analytics

To evaluate operational enterprise data, SAP HANA provides new approaches. It allows for reporting and analyses even if you do not use an SAP BW system. For this purpose, the system creates individual data models (SAP HANA views) within SAP HANA that can be directly accessed by analysis tools. For example, if an SAP ERP system runs on SAP HANA, you can evaluate data directly in SAP Lumira or SAP Analysis Edition for Office using the corresponding SAP HANA views. In addition, if you use SAP BusinessObjects Design Studio, you can also create dashboards that are based directly on SAP HANA views. The advantage is that all data is available for evaluations in real time, and you no longer have to replicate the data to an SAP BW system in a time-consuming process. The disadvantage, however, is that the data from different SAP systems cannot be simply merged or harmonized within the scope of the analysis. For this purpose, an additional SAP BW system is necessary (see Chapter 9, Section 9.2). Nevertheless, it is beneficial to use operational analytics with SAP HANA in selected implementation scenarios.

One weakness first-introduced with operational analytics was that predefined data models and reports would go missing, forcing users to create them by hand. Today, default data models for the most important SAP HANA products are provided with SAP HANA Live. Analogous to SAP BW content, a virtual data model is now available that combines SAP HANA views in different layers. Depending on your license, you can also adapt or extend the SAP HANA views individually. Chapter 9, Section 9.1 provides more information on SAP HANA Live.

You might ask yourself why data models are necessary in SAP HANA at all. To answer this question, let’s take a deeper look. Unfortunately, the data that is supposed to be analyzed is rarely provided in only one table; e.g., master data and the corresponding texts are usually stored separately. If a report is supposed to list material texts in addition to specific material properties, then the various tables must be linked to each other. The relevant key figures are also often provided in specific tables. If you want the system to evaluate these key figures using different characteristics, the integration of further tables is required. The result is a complex structure of tables, also known as a star schema, in the SAP BW area. However, you can still link the various tables with each other using the corresponding SQL commands to allow for an evaluation. However, it is more efficient to use the already mentioned SAP HANA views because they are SAP HANA modules (e.g., OLAP Engine) optimized to perform calculations at a high speed to retrieve data considerably faster. Chapter 2, Section 2.3.2 provides more information on this.

SAP HANA views are modeled in SAP HANA Studio. This tool is based on Eclipse (see https://www.eclipse.org) and is the central development and administration tool. SAP HANA Studio supports users with a graphical modeling view for the creation of SAP HANA views. Like SAP BW objects, the created SAP HANA views must still be activated or deployed at the end of the process. Afterward, they are available in various analysis tools, such as SAP Lumira, SAP Analysis Edition for Office, or SAP BusinessObjects Design Studio. The development and modeling environment of SAP HANA Studio is detailed in Chapter 5, Section 5.2. Chapter 6 discusses the administration options of the SAP HANA database.
1.2 Current Challenges for SAP BW

After giving you an overview of SAP HANA, this section describes which current challenges in the SAP BW environment convince more and more enterprises to use SAP BW on SAP HANA. This section first analyzes how the general conditions have changed, and then it illustrates the disadvantages of relational databases from the technical perspective. Because of these disadvantages, in the past, data was often kept in various systems, each of which were optimized for a certain purpose. A distributed and redundant data concept, however, poses new business challenges to enterprises. These challenges are described at the end of this section.

1.2.1 Changing Environment

Since the introduction of SAP BW in 1997, the environment of this product has changed significantly. The following three essential aspects are particularly important in this context and are discussed in detail:

- Accelerated data growth
- Real-time data access
- Simple and fast operation

According to IDC studies, the worldwide data volume is doubling about every two years (see http://www.emc.com/leadership/programs/digital-universe.htm). This trend can also be applied to enterprise data. In this case, SAP assumes that the data volume is doubling every 18 months on average. As the central data warehouse solution, SAP BW in particular is affected. If an appropriate archiving solution is not used, the number of data records of the SAP BW InfoProviders will increase exponentially. This growth usually accelerates continuously because the status of reporting increases in general so that data from many new application areas and regions are transferred to the SAP BW system. This can lead to long response times when analyses are performed or long load times when data is further updated in SAP BW. Here, the often limited scalability of the existing IT infrastructure and database solution plays an important role.

However, it will be increasingly important to evaluate data in real time. Some years ago, it was sufficient to consolidate data once a week or once a month and to generate the reports during night processing. Today, the requirements have changed considerably. It is increasingly important to counteract undesirable developments in a targeted manner and at an early stage. In this context, access to real-time data is an important prerequisite. And, due to the introduction of smartphones, the latest key figures must now be available via mobile ad hoc accesses.

In other applications, SAP BW end users see every day that search processes in giant datasets—for example, Google search processes—take only a few moments. At the same time, various software providers, such as Apple, prove that modern graphical user interfaces can be designed with simple and intuitive views. SAP has already recognized the significance of these aspects some years ago. In 2011, Jim-Hagemann Snabe, one of the two chairmen of SAP SE until 2014, put it aptly when he said that SAP must become “Apple simple and Google fast.” SAP HANA contributes to this goal significantly.

1.2.2 Disadvantages of Relational Databases

One of the main tasks of an SAP BW system is the timely provision of data with the goal of being able to evaluate and analyze it efficiently. From the technical perspective, this requires efficient data handling with optimized read operations. In reality, relational databases are often used for this purpose. The data is stored in various, usually row-based tables. In a relational database, the data can be stored largely free of redundancies if certain normalizing rules are considered when modeling data (http://en.wikipedia.org/wiki/Database_normalization).

Especially in cases when an SAP BW system is used, the focus is not on redundancy-free data management, but on high speed read accesses. The data is stored internally in a denormalized star schema or extended schema in which one fact table (for key figures) and several dimension tables (for characteristics) form a star-like structure. This layout allows for
a high-performance and dimension-independent evaluation of key figures. Unfortunately, this also requires that the data be retained in various tables and linked to the corresponding primary and foreign keys. In this case, a specific logic that splits the data appropriately when it is transferred to the database and stores it in the tables provided for this purpose is necessary. In SAP BW, the load process (Extract, Transform, Load [ETL]) controls this, which requires additional time.

To access the respective data records efficiently, relational databases use database indexes. This index structure accelerates the search process and sorting by certain attributes in the database. But, the creation of these kinds of indexes is time-consuming. Furthermore, you must update the index structure after inserting a new data record. The SAP BW system also uses indexes internally for performance optimization. For example, an InfoCube may have an index in order to accelerate read accesses. When new data records are imported to the InfoCube, this index must also be updated; however, particularly in case of large data amounts, this may delay the load process. It is faster to delete the index before triggering the load process and then create it anew after completion of the load process. This is a common procedure for the creation of SAP BW process chains.

In SAP BW, the data analyses often use analytical operations, such as slicing (filter restriction using one dimension), dicing (filter restriction using several dimensions), or drill down (drilling aggregations down to a detailed level). These operations are usually directly executed in the SAP BW system. To perform the necessary calculations, the system must first retrieve the data from the database. From there, large data amounts are transferred between the database and the SAP BW application server, even if only a few data records are shown to the user at the end. It would be more efficient to already perform this kind of operation at the database level. This could considerably reduce the amount of data transferred and even accelerate the execution of the operations. In addition to analytical operations, the SAP BW system implements other performance-intensive process steps. These include, for example, the activation of DSO data packages and the execution of planning functions when the BW Integrated Planning (BW-IP) is used. If you use SAP BW on SAP HANA, analytical functions and numerous performance-intensive steps are directly executed in SAP HANA, which considerably accelerates these processes (see Section 1.4.5).

So that you can efficiently evaluate large data amounts with relational databases, materialized views are provided. Here, the comprehensive initial data is stored persistently and in a compressed way in an additional table. SAP BW works with this concept for the creation of aggregates for InfoCubes. The advantage is that the aggregation does not have to be performed at runtime so that the data can be evaluated more quickly. This makes sense particularly if several analysis scenarios are based on the same datasets. The disadvantage of this approach is that the dataset of the materialized views is outdated when the initial data is changed, so data inconsistencies may occur. In this case, you must update the materialized views, which again requires some time. Furthermore, performance problems can occur if the drilldown was changed during the analysis in such a way that the aggregation level changes. You can then no longer use the data of the aggregate and must access the InfoCube directly. If you use SAP BW on SAP HANA, you do not have to provide materialized views.

### 1.2.3 Distributed Data Retention

Due to the various requirements, various IT systems that partly leverage the same data are often used. Depending on the purpose, these systems are optimized for a certain application case. A parallel operation of operational transactional systems (OLTP, Online Transactional Processing) and decision-making systems for the execution of complex data analyses (OLAP, Online Analytical Processing) is a common practice. OLTP systems, such as SAP ERP, are generally used for traditional business applications and require high-performance write and update processes. The data amount that is processed in a transaction step is usually small, and the evaluation options are often limited. Here, the data is created mainly in normalized database tables. In contrast, OLAP systems, such as SAP BW, are designed to process various read queries, sometimes with very large data volumes. Write operations take place only within the scope of periodic data updates. To support this scenario, the data is usually stored
Introduction to SAP BW on SAP HANA

1.3 Reasons for Migrating an SAP BW System to SAP HANA

Let’s turn our attention to the reasons for an SAP BW on SAP HANA migration. On one hand, you have the restrictions of SAP BW, but on the other hand, you have the advantages of using SAP BW on SAP HANA.

1.3.1 SAP BW Restrictions

Based on your daily work with SAP BW, you may know some restrictions that the SAP BW system has, and have wished, at least once, for a technology to avoid them. The following are some key restrictions for SAP BW:

- Low performance and long waiting times in reporting
- Considerable time and effort for transformation and load processes
- Redundant data retention and inflexible data modeling
- High administration effort for the SAP BW system and the IT infrastructure

One of the major restrictions of SAP BW is the partially low performance during the execution of reports and analyses. Depending on the amount and complexity of the data, the processing of individual analysis steps in the SAP BW system may take several minutes and thus slow down the processes in the various business departments significantly. In one year, this can amount to numerous hours that employees have to wait. The possible side effects of this include a reduced quality of work.

Transformation and load processes also require a lot of time in SAP BW. Consequently, the corresponding processes run at night or on the weekend. This should ensure that the system resources are mainly available for analyses and are not affected by running background processes. The time frames that are available for the transformation and load processes become increasingly smaller: due to the international environment, it is no longer sufficient when a global SAP BW system is available for reporting only between 6 a.m. and 8 p.m. EST. Instead, it becomes more and more necessary to extend these time frames to ensure access from other countries and time zones. If the data from more regions and countries is centrally merged in SAP BW, the data volume also increases. Consequently, it becomes harder to solve this conflict with the existing IT infrastructure.

Section 1.2.3 already described redundant data retention and its consequences for SAP BW. This section addresses the various layers within the SAP BW system (see Chapter 5, Section 5.5). Often, the SAP BW
system transfers mainly unchanged data from a DSO in the data propagation layer to an InfoCube in the reporting layer to accelerate the reporting process. Redundant retention of identical data, however, leads to considerably long load times and requires additional memory capacities in the database. If you use SAP BW on SAP HANA, you do not necessarily have to use InfoCubes to accelerate reporting. Even worse, it can be a disadvantage to use InfoCubes in SAP BW without SAP HANA; for example, for data modeling or remodeling. If data has already been imported to an InfoCube, its structures, such as the dimensions, cannot be changed directly because an InfoCube in the database is mapped using numerous tables according to the star schema or extended star schema (see Section 1.2.2). The remodeling toolbox (Transaction RSMRT) enables you to adapt the InfoCubes without having to empty data content first. But, this is more complex because you previously have to define the appropriate remodeling rules and schedule a change run later on. For some adaptations, it may even be necessary to edit the InfoCube directly. In this case, the InfoCube data must be deleted first to implement the necessary modifications. Then, the data is re-imported to the InfoCube, and the index is created anew. Both approaches are relatively time consuming and prone to errors in real life. This looks different for SAP BW on SAP HANA. Here, direct remodeling is possible, for example, moving an InfoObject to other dimensions. You then do not have to use specific tools or empty the InfoCube first.

If you use SAP BW without SAP HANA, the administration effort for the SAP BW system, the primary database, and the often-used BWA is quite high. You require several tools for the administration of SAP BW, database, and BWA to, for example, create backups. The administrators must be familiar with all of these tools and trained correspondingly. Additionally, the capacity of the BWA is usually limited. If you want to replicate the data of a new InfoCube to the BWA (indexing) and the existing main memory is not sufficient, a manual intervention is required. Loading data in InfoCubes for mere performance reasons can result in significant costs and administration efforts. The duplicate data retention requires additional system resources, and the administrators might have to extend the file system more often—that is, configure hard disks or memory solutions and, if necessary, change the database administration. Furthermore, backups require a larger memory medium and take longer. If you use SAP BW on SAP HANA, you no longer require the BWA, and the administration effort is reduced considerably. This section discusses this in greater detail later on. Because aggregates, indexes, and database statistics are no longer necessary, the administration is easier (see Chapter 5, Section 5.4). For example, the execution of process chains no longer leads to errors with regard to InfoCube indexing. Because the mentioned process steps are omitted and the speed increases significantly with SAP BW on SAP HANA, administrators will be more flexible in the future with regard to the scheduling of load processes at night.

1.3.2 Advantages of SAP BW on SAP HANA

Now that we’ve discussed the essential restrictions in SAP BW, this section deals with the advantages resulting from the use of SAP BW on SAP HANA. Figure 1.4 illustrates the most central aspects.

Certainly, the most important advantage of SAP BW on SAP HANA is the high speed with which analyses and reports can be performed. This results from the in-memory technology, the numerous software-related innovations, and the optimized hardware (see Section 1.4). For example, in several projects, SAP BW reports were reported as being more than 30 times faster. The execution duration of a report could also be reduced from one minute to less than two seconds, which allows for a considerably more frequent and interactive usage of the reports in SAP BW. This enables you to optimize existing business processes or design completely new process flows. SAP sometimes refers to Yodobashi, a Japanese electronics goods retailer. At this enterprise, the introduction of SAP HANA reduced the calculation of incentive payments for customers from three days to two seconds (see http://www.news-sap.com/hana-teched-2011-plattner-in-memory). Yodobashi can now inform its customers anytime about the current value of their credit memos collected in the incentive program, as well as about the new status of the bonus credits after a purchase. SAP BW on SAP HANA therefore not only significantly accelerates reporting, but also allows for the implementation of completely new business processes that would have otherwise been impossible to carry out.
In addition to an increase in performance, SAP HANA also makes data retention processes more efficient. Due to the column-based data storage, the memory size of the database is reduced because specific and powerful compression methods are used. Column-based data handling is particularly efficient for SAP BW data because aggregates are often formed on the basis of numerous rows, but only a few columns (see Section 1.4.2). Faster access to data and other performance characteristics of SAP HANA make it possible that SAP BW data no longer has to be provided in denormalized tables. A high reporting performance for SAP BW on SAP HANA is always guaranteed, irrespective of whether the data is provided in a flat DSO table or in an InfoCube. Correspondingly, if you use SAP BW on SAP HANA, you can usually omit InfoCubes (see Chapter 5, Section 5.1) and redundant data retention in SAP BW, such as the storage of data in a DSO and in an InfoCube, in order to accelerate reporting. If you have created data flows that enrich or modify data on its way to the InfoCube, these InfoCubes are still required after a migration to SAP BW on SAP HANA. However, you can convert them into SAP HANA-optimized InfoCubes (see Chapter 5, Section 5.3), which accelerates the reporting performance considerably and allows for a direct remodeling of InfoCubes.

In the future, SAP HANA will be the basis for further SAP products, and existing products will be increasingly optimized for SAP HANA. It is thus a future-proof solution with a high level of protection on investment. It may be worthwhile to become familiar with this state-of-the-art technology today. Furthermore, there are already products that are exclusively available for SAP HANA, such as Operational Process Intelligence (see http://help.sap.com/hana-opint). If it becomes necessary to use these kinds of products in the future, it pays off if you are already acquainted with the technology. SAP BW on SAP HANA is the ideal initial scenario because it was one of the first SAP HANA solutions on the market. The product now is rather mature, so it can be used in production without special risks.

As already mentioned, you no longer need the BWA for SAP BW on SAP HANA, so the maintenance and monitoring effort for this component is omitted. Administrators must still be introduced to the SAP HANA-specific tools once (see Chapter 6), but a duplicate administration of the two separate memory technologies is no longer necessary. Because the data retention is more efficient and materialized views are no longer required in SAP BW on SAP HANA, the administration effort is further reduced, for example, for the generation of database statistics, the deletion or creation of indexes, or the maintenance of aggregates. Until now, it was nearly impossible in SAP BW to efficiently perform complex analyses with a large set of detailed data, so the data was often aggregated in the SAP BW data flow, for example, to aggregate data from a daily basis to a monthly basis. This made it possible to execute the respective evaluations in a reasonable time frame because it reduced the required data volume. If you use SAP BW on SAP HANA, you no longer have to aggregate data. Thanks to the efficient data retention, you can create reports in SAP BW on SAP HANA directly on the basis of detailed data (document level). This results in a higher level of detail in reporting and still allows for high-performance evaluations of non-aggregated mass data.
Finally, SAP BW on SAP HANA also significantly accelerates the load processes when data is further updated in SAP BW. The code push-down (see Section 1.4.5) moves performance-intensive process steps directly to SAP HANA for efficient processing. In SAP BW, for example, this is the activation of DSO data packages. Because the denormalized star schema or extended star schema is not used and the data is stored directly in the main memory, the load processes in SAP BW are accelerated. In real life, it has been proven that an SAP BW on SAP HANA migration can also reduce the execution times of the process chains considerably. In one case, for example, the initial execution of a rather comprehensive process chain could be reduced from more than 10 hours to about six hours. For this purpose, the system was migrated to SAP BW on SAP HANA, and the InfoCubes were converted to SAP HANA-optimized InfoCubes (see Chapter 5, Section 5.3). Additional optimizations, such as the use of DSO instead of InfoCubes for reporting, allow for further significant accelerations of the load processes.

**Summary**

This section discussed the numerous reasons for an SAP BW on SAP HANA migration, made prominent by the current restrictions in SAP BW and the various advantages associated with the use of SAP BW on SAP HANA. In addition to a considerable increase in performance for analysis and reporting, topics such as protection of investment, efficient data retention, reduced administration effort, and acceleration of the load processes play an important role. The advantages of SAP BW on SAP HANA are generally based on the software and hardware innovations of SAP HANA, which are discussed in the following section.

### 1.4 Basic Technical Principles

Databases, the relational database model, and SQL as the query language date from the 1970s and are based on IBM’s R database system. All databases that are largely used today (DB2, Oracle, Microsoft SQL Server, and so on) have the same basic principles. Since the early days of standard software and SAP, the amount of data to be stored in enterprises and the computing power of the processors have increased significantly.

The performance of hard disks has fallen far behind, and even today’s SSD hard disks cannot catch up with this performance lead. The system bottleneck and the runtime of database queries consequently depend on the transmission speed between the hard disks and the main memory.

Databases aim to provide data promptly to allow for making business decisions on the basis of this data. The hard disk speed has already not met these latest requirements for a long time. To compensate, only some data has been copied in the form of caches to the main memory, so the access speed is increased considerably for a small amount of data. SAP HANA goes one step further and leverages the in-memory technology. This means that the entire database is provided in the main memory. However, the innovations of SAP HANA should not only be limited to the in-memory aspect. Innovations such as column-based data retention, the insert-only procedure, the partitioning of tables, and the push-down principle of the SAP systems also contribute to the performance of SAP HANA.

The result is a significant speed benefit compared with other databases, which makes previously impossible scenarios feasible for the first time. The technology aims to adapt the runtimes of data analyses to the speed of today’s internet search engines in order to also change the usage pattern for standard software.

#### 1.4.1 In-Memory Technology

The concept of in-memory databases is not new. For example, with TM1 (today: Cognos TM1), IBM’s portfolio has provided a database that performs processes in the main memory since 1984. At that time, the idea of replacing slow hard disks with memory as the storage medium for data of a database seemed appealing.

In TM1, the amount of data to be processed was limited by the high main memory requirements. The past hardware did not allow for mass data processing in the main memory, so TM1 could not establish itself against other databases. However, SAP took up the idea again and started to develop in-memory databases with TREX in the 1990s, when the price for main memory was already considerably lower. First, TREX served as a search and indexing service, and later on, as the basis for BW...
Accelerator (BWA), which was published as an enhancement of the SAP NetWeaver system in 2005. It does not replace the main database of the SAP BW system but quickly provides selected InfoCubes from the main memory for it.

Studies conducted under the lead of Hasso Plattner at the Hasso Plattner Institute of the University of Potsdam (HPI) aimed to implement analytical and transactional operations (OLTP and OLAP) in one system on the basis of the same in-memory–based database. Furthermore, the response time was supposed to be reduced considerably to enable completely new application scenarios. This vision was first implemented with the SanssouciDB database and in the SAP HANA product at the Hasso Plattner Institute. Standard business applications, such as SAP CRM or even SAP ERP, can now be operated on a completely main memory–based database. It is provided as an SAP HANA platform that has various interfaces to SAP or external systems in addition to the database. The platform also contains a tool that enables you to directly access the database and its administration: SAP HANA Studio. Furthermore, you can develop native applications on the database, and it provides function libraries for analytical processes, such as forecasts.

To ensure an appropriate hardware performance, SAP HANA is provided only in a package with certified hardware (SAP HANA appliance). Chapter 2, Section 2.1 discusses the hardware in more detail.

The ratio between the amount of main memory and the number of processor cores is of particular importance because every main memory area is processed by an assigned processor. A single SAP HANA server (single node) can currently reach a size of up to 4TB main memory and 80 processor cores. By combining several systems, you can create even larger databases (multi node). If these large servers are combined to a multi-node system (scale-out approach), huge SAP HANA systems are feasible. In various experiments, system sizes of up to 100TB main memory were checked for performance, so SAP HANA can also be used for extremely large systems.

This size of main memory leads to new hardware problems, which must be solved in collaboration with hardware providers and SAP. Chapter 2, Section 2.1.5 discusses this in more detail.

### 1.4.2 Column-Based Data Retention and Compression

Despite the price decline over the decades, main memory is still an expensive resource. The data to be stored must thus be compressed in an in-memory database as efficiently as possible. For this purpose, SAP HANA uses a combination of various technologies.

A characteristic feature of SAP HANA is the column-based storage of the most tables. Because data in a column has the same data type and includes similar data (for example, gender, nationality, date), the data compression rate is considerably higher than in the usual row-based databases.

A common technology is *dictionary encoding*. Here, every value that occurs in a column is entered in a dictionary and assigned to an ID, respectively. If the same value occurs several times in the column, memory space is saved by storing only its ID, and not the raw data (see Figure 1.5).

You can use more of the potential of the column-based storage by clustering successive IDs with the same value (*cluster encoding*). For this purpose, you divide the table with IDs into several clusters of the same size. If the ID within such a cluster always has the same value, it is stored only once. Depending on the size of the selected cluster and the structure of

![Dictionary Encoding](image)
the data, you can merge numerous entries of the column. This par-
ticularly applies to sorted columns. Figure 1.6 illustrates this using an ex-
ample with cluster size 3.

A similar compression technology does not use clusters but additionally
stores the number of successive same values: run length encoding.

The compression methods mentioned are only examples of numerous
possible measures that enable the most efficient compression. The meth-
ods used depend, among other things, on the data type of the column.

The database’s compression rates that can be achieved depends largely
on its structure and content. SAP says that they have already achieved
compression rates of more than 20, so for 100GB of user data, less than
5GB of main memory would be required. Our empirical values show
that a ratio of 5:1 to 7:1 is common.

Database tables of SAP systems may contain up to 150 columns. If you
want the system to calculate the total of one column—for example, all
sales of the last year—the system first has to search every single row in
a row-based database for the corresponding column. In column-based
databases, the relevant figures are stored successively, anyway, and can
thus be read and aggregated considerably faster. In analytical systems,
these queries are primary queries, which means that they usually benefit
significantly from column-based systems.

Regardless, SAP HANA can also work with row-based storage of data,
which is often used by system tables, such as in statistics.
Because the delta merge transfers several changes to the main storage at the same time, the performance bottleneck has less impact than if each change were transferred individually. This compensates for the disadvantages of column-based databases.

### 1.4.4 Partitioning

Partitioning a database means having the system automatically subdivide tables and their content into several small tables. In general, you distinguish between horizontal and vertical partitioning, that is, a division according to column or row (see Figure 1.8). The users of the database still view a large, contiguous table. SAP HANA leverages this function for column-based tables only. All partitioning methods that are available in SAP HANA are horizontal partitionings.

**Advantages**

SAP HANA uses partitioning to avoid various restrictions. For example, only two billion entries can be stored for each table. With partitioning, the entries are distributed across the number of partitions, and a table of more than two billion entries exists only for the database users. The performance of the table access also increases because various operations are performed in parallel without impacting each other. In addition, the delta merges explained in Section 1.4.3 are accelerated because only parts of the table, and not the entire table, must be reorganized.

You obtain the most benefits, however, if you use an SAP HANA cluster in which partitioned tables are distributed across several physical servers. This way, the workload is distributed efficiently across the servers involved.

SAP HANA can work with various partitioning types. They differ by the way in which data records are transferred to particular partitions.

To distribute the data as equally as possible, you should use the hash partitioning method. Here, the system calculates a hash based on one or several columns of the table’s primary key. This hash is then used to determine to which partition the corresponding data record is moved, as follows:

- **Round-robin partitioning**
  Round-robin partitioning distributes data records across all partitions according to their sequence.

- **Range partitioning**
  With range partitioning, the data records are distributed based on the column of the primary key. This method enables you to define partitions for individual values or value ranges.

- **Multi-level partitioning**
  You can also combine several partitioning algorithms to allow for an individual distribution of the data. This method is called multi-level partitioning.

The SAP system usually defines for itself how it partitions its tables, but you can also partition the tables manually. Hash partitioning is ideal for tables whose content you don’t know, while range partitioning should be used for a distribution on the basis of periods.

### 1.4.5 Push-Down Principle

The defined goal of SAP HANA is to allow as many calculations as possible to be directly performed in the database, and not by the application system. SAP wants to utilize the fact that the data in SAP HANA is...
already provided in the main memory for processing and that sufficient processor resources are available. To perform complex calculations at the database level, more and more parts of the application system logic are moved to the database. This is referred to as the push-down principle (see Figure 1.3). During the execution, the system checks at specific places whether an SAP HANA database is operated in the minimum required version. If this is the case, a modified program logic is executed that transfers calculations using specific SQL queries or database procedures to the database.

Push-down in SAP BW

Because SAP BW has supported SAP HANA as the primary database for several years, the push-down principle is used often. For example, a new Data Transfer Process (DTP) execution module, SAP HANA Execution, is available since SAP BW 7.4 SPS 05. Figure 1.9 illustrates the difference between a DTP based on SAP BW on SAP HANA (on the right-hand side) and a DTP in an SAP BW system on another database (on the left-hand side).

Figure 1.9 Push-Down Using a Sample Transformation

In the SAP HANA execution DTP mode, all calculations of the corresponding transformation are performed within SAP HANA, as well as the transfer from the source tables to the target tables. This concept is particularly useful for transformations because both the source table and the target table are located in the SAP HANA database.

In the database, Application Function Libraries (AFL) are used for this purpose. In these libraries, SAP combines selected application functions. These are procedures that are written in C++ and can directly access the SAP HANA resources. Their close connection to the programming code ensures that the performance is higher than that of database procedures that were defined in SQLScript or R.

Currently, there are two Application Function Libraries: Business Function Library (BFL), which encapsulates business functions, and Predictive Analysis Library (PAL) for forecasts. Their functions are used in SAP products that are based on SAP HANA but can also be utilized in custom procedures or programs. Their usage, however, is not automatically covered by every SAP HANA license.

Since SAP BW 7.4 SPS 05, SAP BW on SAP HANA provides the PAL functions in the new SAP HANA Analysis Processes, for example. They allow for the processing of data using the default functions available in the PAL. You can then store and further use the results in a database table or DSO InfoProvider, for example. This enables SAP BW users to create processes themselves that benefit from the performance of the push-down principle.

1.5 Summary

The usage of SAP HANA has changed over the years: first, the focus was on the implementation of side-by-side approaches to accelerate selected evaluations (data mart approach) or applications (accelerator approach). Today, SAP HANA is implemented almost exclusively on the basis of integrated approaches. The existing database, e.g., in an SAP BW system, is replaced by the SAP HANA database. The SAP BW system then benefits from the unique performance characteristics of the SAP HANA database, which leads to a significant acceleration of reporting and load processes.
Introduction to SAP BW on SAP HANA

Because the usage of SAP HANA as a database for SAP BW systems today achieves excellent results, SAP now wants to focus on the development and usage of SAP HANA for further products, such as SAP Business Suite. SAP Business Suite, powered by SAP HANA also transfers and correspondingly accelerates performance-intensive processes to the database using a code push-down. However, it is not yet possible to operate SAP BW and SAP Business Suite on a shared SAP HANA database. SAP still sticks to the vision of Hasso Plattner to develop SAP HANA to a database for the simultaneous operation of OLAP and OLTP applications, but it will surely take some years until this goal is achieved. In the meantime, data must still be transferred between OLTP and OLAP systems. This will change only if both systems can access the same tables. SAP has already taken the first steps into this direction. For example, Operational Data Processing (ODP; see Chapter 5, Section 5.1.1) already enables the data from the source systems to no longer have to be stored in the Persistent Staging Area (PSA). Instead, the required data can be called directly from the source system for further processing in SAP BW. If you also use Open ODS views, redundant data retention in SAP BW can be completely omitted (see Chapter 5, Section 5.1.2).

SAP BW on SAP HANA's success is based on the existing restrictions in SAP BW and the numerous advantages of SAP BW on SAP HANA to overcome these restrictions. The challenges in SAP BW include the criticized performance and long waiting times in reporting. SAP BW on SAP HANA is convincing due to a high performance, which is based on the in-memory approach and column-based data handling. The fact that numerous customers have already experience in the usage of the in-memory technology due to the integration with BWA also contributes to the acceptance of SAP BW on SAP HANA. Overall, SAP BW on SAP HANA has reached a level of maturity that justifies an explicit recommendation of its usage. You can benefit from the additional functions that are available exclusively in the SAP BW on SAP HANA scenario and reduce the complexity of your SAP BW system significantly. Today, SAP BW users have the following complaints:

- It takes too long until new scenarios are implemented in SAP BW.
- The night processing is too short to load the data.
- The execution time of some reports is too long.
SAP HANA is delivered as an appliance and is therefore a complete product consisting of server hardware and an operating system. This chapter discusses these individual parts in detail. You learn about their significance, which selection options you have, and what you need to consider.

2 SAP HANA Architecture

Chapter 1, Section 1.4 already described which innovative principles set SAP HANA apart from other databases. Let’s now discuss the architecture, that is, the interplay of the various components of an SAP HANA system. The term architecture refers to hardware and software in general, and SAP HANA processes in particular.

Section 2.1 discusses the criteria for a certified SAP HANA system. We will explain the terms scale-up and scale-out for upgrading an existing SAP HANA system. Deploying an in-memory database imposes special requirements on data security and main memory management. For this reason, these topics are outlined in separate sections. As an alternative, you will also learn about SAP HANA’s cloud options.

Section 2.2 deals with the question of how to enable parallel usage of an SAP HANA server for various purposes. Operating SAP HANA on virtual machines is discussed separately. Finally, we’ll discuss some specifics for selecting the operating system.

Within the scope of architecture, you receive a detailed description of SAP HANA’s functionality. Chapter 6 and other parts of this book will refer to SAP HANA processes and engines. Their functionality is described in Section 2.3. The index server process is discussed separately because it is the core of SAP HANA and is thus responsible for a multitude of tasks.
While reading, bear in mind that SAP HANA is a fast-developing product: while this book was being written, another operating system was released for productive use in SAP HANA, some details of the main memory management were changed, and live operation of virtual machines was permitted. To keep you up-to-date, reference is made to the corresponding SAP notes whenever possible. If a topic is of particular interest to you, we recommend that you obtain the latest information from these notes.

### 2.1 Hardware

Like desktop PCs, today’s servers follow a flexible hardware concept. Their hardware and software supports the highest possible combination variety of the various hardware components. This becomes clear when you consider the effort that hardware providers, operating system manufacturers, and customers must make to develop, maintain, and install drivers.

In the IT world, however, there are some famous counter-examples, for instance, Apple devices; Sony game consoles, Microsoft, or Nintendo; or mini-computers like Raspberry Pi, which are very popular among tinkerers. For these products, the collaboration between hardware and software is as close as possible. The less programming and driver codes must be adapted in the various components, the lower the effort will be for development and execution. Consequently, software projects emerge that feature stunning performance. An application development that strongly focuses on hardware therefore results in better performance.

SAP HANA follows a similar concept. For this purpose, guidelines were created for the hardware and software environment on the servers so that the database operation always takes place in a highly homogeneous environment. This is to ensure high performance and low error-proneness for each appliance. At the moment, only a certified selection of servers and operating systems is supported for productive usage in SAP HANA. The complete product consisting of hardware, operating system, SAP HANA, and all related tools is referred to as an appliance. Customers can therefore choose among predefined packages when purchasing SAP HANA for production purposes. Depending on the scenario, several servers are deployed for scale-out, test and development systems, or high-availability scenarios.

The following discusses the various options of which hardware you can deploy and how to use it.

#### 2.1.1 Certification

Certified hardware is required particularly for the productive usage of SAP HANA. SAP has already certified several servers from most SAP hardware partners for usage with SAP HANA. The certification process checks the hardware components for their performance; this involves the speed of processors, the main memory, and their ratio. For an SAP BW on SAP HANA system, for example, the necessary ratio is 16GB per processor core when using an 8-core processor. Additionally, the speed of the network interface and hard drives is taken into account. At the moment, support is provided for servers with Intel processors only. Servers that have passed this process successfully are listed in the Product Availability Matrix (PAM). They are subdivided into servers that are run with SUSE Linux Enterprise Server for SAP Applications or with Red Hat Enterprise Linux. Some of them also support virtualization via VMware vSphere for live operation.

The tailored data center integration is an alternative. Here, you can individually compile the hardware package, which is usually delivered in an appliance, in cooperation with your hardware partner. You still require a certified server and a certified storage solution. Savings are possible if you already own parts thereof. You may then deploy this hardware for productive usage of SAP HANA. Different from the appliance approach, you must then install and configure the operating system and SAP HANA yourself. The employee responsible requires a certificate for attending a training for SAP HANA installation administrators. This can be a cost-efficient alternative for customers who already have appropriate high-performance hardware. For this purpose, you should first execute the test tool in SAP Note 1943937 (http://service.sap.com/sap/support/notes/1943937). This tool checks the components of the server and
provides an initial assessment of its suitability. Here, the network connection and the speed of the storage solution are tested in particular.

### 2.1.2 Cloud

SAP HANA can be operated in the cloud for both testing and production purposes. Cloud offerings are usually deployed if the emphasis is on a cost-efficient scaling and particularly low IT overhead. A wide variety of offers has emerged in recent years.

**Definition**

The cloud is one of the latest technology trends. Because this vogue term is often used incorrectly, we want to first explain what a cloud offering actually is. In 2011, the National Institute of Standards and Technology (NIST) provided a definition (see [http://csrc.nist.gov/publications/nist-pubs/800-145/SP800-145.pdf](http://csrc.nist.gov/publications/nist-pubs/800-145/SP800-145.pdf)) that establishes the following characteristics for cloud services:

- The consumer can provision hardware resources automatically without human interaction.
- The service is available in the network (usually via the Internet).
- The provider’s virtual and physical resources are assigned to the customers.
- Resources can be adapted and scaled flexibly to customer requirements as needed.
- The load and usage is measured to provide transparency for customers and providers.

**Service and channel**

In summary, a cloud is an IT service that is provided to multiple customers via a network. Here, you must distinguish which service is offered via which channel. A distinction is usually made among the following three types:

- **Infrastructure as a Service (IaaS)** provides the customers with their share of hardware that is managed by the provider.
- **Platform as a Service (PaaS)** does not permit direct access to the hardware, but upload and operation of self-developed software. Interaction is made using tools provided by the provider.
- **Software as a Service (SaaS)** only permits the usage of an application that is provided and operated by the provider on separate servers and made available via a network, usually the Internet. All SAP offers in the SAP HANA area are assigned to this method.

A distinction is made by the customer range of a cloud service. If it is used by one organization exclusively, it is referred to as a **private cloud**. In this cloud, customers are, for example, departments or individual employees of an enterprise. The provider can be an external company or the enterprise-internal IT department. If, however, the service is available to all organizations, this is referred to as a **public cloud**. If the provider is an external company, it also assumes liability for failures, maintenance, and operating costs. These are typical reasons for using a cloud solution. The definition of the private cloud applies to several products that already existed before this term was created, starting with simple network drives.

Besides the new SAP HANA-based offers, SAP also provides the cloud-based SAP Business ByDesign or solutions of enterprises like SuccessFactors and Ariba that were acquired by SAP. Enterprises like Amazon offer cloud hosting of “regular” SAP applications within the scope of Amazon Web Services (AWS).

There’s a wide variety of cloud offerings for SAP HANA:

- **SAP HANA One/development edition**
  
  Several cloud offers exist with which you can try and test SAP HANA quickly and without hardware expenditure. SAP HANA One is a HANA instance with 60GB that can be used via Amazon Web Services, for example. Currently (April 2015), the price amounts to about $3.50 per hour. You have full control over a virtual machine with a preinstalled SAP HANA database. If you develop your own applications on this virtual machine, you may use it in live operation.

  This doesn’t apply to other providers of the SAP HANA cloud development edition. They vary in their hardware dimensioning but are often more cost-efficient. These variants are not suited for usage in the SAP BW on SAP HANA environment, but they can give you a first impression of SAP HANA at a reasonable price.
Trial offers
Some offers are available for free for a limited period of time. At the moment, trial programs are available for the SAP HANA development edition, SAP BW on SAP HANA, and SAP ERP on SAP HANA. These offers can usually be deployed via the Amazon cloud, which involves some minor costs, depending on the usage duration. But you should still use these inexpensive offers to receive an initial impression of the corresponding solution.

SAP HANA Infrastructure, DB Services, and App Services
The SAP HANA Infrastructure Services offer is also provided by Amazon Web Services and includes only the appropriate hardware for operating SAP HANA in the cloud. The license is not included in the price, so you must purchase the SAP HANA license yourself. In exchange, the SAP HANA Cloud Infrastructure systems offer up to 1 TB main memory and support scale-out. If you don’t have a license of your own, you can deploy SAP HANA DB Services because they already include a license. SAP HANA App Services additionally provide advanced tools and options for application developers.

SAP HANA Cloud Platform
SAP HANA Cloud Platform cannot be used for operating SAP HANA systems. It is intended for developing native SAP HANA applications only. For this purpose, each SAP partner or customer can create a developer account and try the platform for free. For financial reasons, SAP operates the free trial version on servers that several developers access at the same time. As a result, their rights are restricted considerably. In particular, user management, administration, and monitoring cannot be made. Data administration, development of own applications on SAP HANA, and modeling of SAP HANA views is possible without any problems. Thus, the trial version of SAP HANA Cloud Platform is a good and free option to deal with these topics.

SAP HANA Enterprise Cloud (HEC)
SAP HANA Enterprise Cloud is intended for productive operation of SAP applications on SAP HANA. It is also the default solution for the cloud operation of SAP BW on SAP HANA. The servers used for this purpose are provided by SAP directly or by selected partners around the world. In this offer, the software is fully managed and maintained for you. A support team is available 24/7, and the data center equipment ensures availability. Consequently, SAP HANA Enterprise Cloud is the first choice for enterprises that want to outsource their IT activities completely.

2.1.3 Scale-Up/Scale-Out
Every database server reaches its storage capacity limits after a while. If you don’t want to reorganize your data or utilize a nearline storage solution, then you must upgrade your hardware resources (more information is available in Chapter 8). In the case of SAP HANA, this entails the purchase of additional main memory and further processors, as well as higher requirement of hard disk memory for backups, logs, and data images. Two upgrade methods exist: scale-up and scale-out.

Scale-up means to upgrade the already existing server(s) without increasing their number. For this purpose, the existing servers are upgraded; many certified SAP HANA servers are built on various blades for this reason. A blade includes all typical hardware components and thus presents a work unit. Another board is added for the upgrade to prevent major intervention of the hardware structure. Scale-up is possible only up to the limits that SAP certified for this server. These limits currently amount to 2TB or 4TB main memory and 80 processor cores.

A scale-out scenario is implemented for larger upgrade projects. Here, the existing system is supplemented with additional servers. SAP HANA now runs on several servers in parallel. The server cluster that emerges here includes a master server that assumes coordination and operates some SAP HANA processes alone from then on (for example, XS Engine and statistics server). This master server and the remaining servers distribute the load among themselves. This is achieved, for example, by table partitioning; the partitions are then distributed to the individual servers. Calculations and database accesses are made on different cluster servers, depending on the required data. For this reason, you should distribute the data to the servers using partitioning after you’ve implemented a scale-out scenario (see Chapter 1, Section 1.4.4). More detailed information on this procedure is available in SAP Note 1908075.
Optimal Number of SAP HANA Servers

The basic principle applies that the performance of an individual server outweighs the performance of several connected servers with the same overall hardware because no communication is required between the connected servers via the network.

When you operate SAP systems, however, it often occurs that several processes compete for the same resources. Deploying an SAP HANA database that is distributed to several servers can therefore result in acceleration. SAP Note 1702409 provides information on this topic, with a focus on SAP BW on SAP HANA.

You should also consider operating one or more servers that allow for continued operation after failure in case of emergency. Servers of the development or test systems are often used for this purpose.

Collective data access
When you operate several servers, they must all have access to data. Logs and data are stored, for example, on a shared network drive or exchanged between servers. These scenarios must be implemented by the hardware partner.

If a server fails, you must take measures to resume operation again. The next section examines this.

2.1.4 High Availability/Data Availability

Power outage results in complete loss of data in the main memory. Because SAP HANA stores this data in the main memory, you must prevent this loss in case of emergency. An initial step is to continuously create save points, logs, and backups, which is discussed in Chapter 6, Section 6.3.2. Additionally, SAP HANA supports various techniques:

- **Mirroring**

  Mirroring is a parallel operation for identically structured servers whose datasets are synchronized (mirrored) continuously. If the primarily used server fails, the mirror server recognizes this and continues operation smoothly. This is the safest but also the most expensive method. Normally, the mirror server doesn’t serve any purpose but incurs considerable costs for hardware procurement and operation. You should also note that the mirror server and the original server are set up in different locations. Only then can you prevent the two servers from being inoperable for the same reason (flood, power outage, fire, and so on).

- **Cold standby**

  In an SAP HANA cluster, one of the servers can be used as a cold standby server. This is an inactive server that automatically steps in if another server fails. To continue operation, this server must first reconstruct the data of the failed server’s main memory. In contrast to a mirror server, this cannot be done continuously because you don’t know in advance which server will fail.

- **Auto restart service**

  A software error or incorrect configuration of SAP HANA can result in an unplanned cancelation of an SAP HANA process. In this case, SAP HANA will restart the process immediately and restore the original state.

2.1.5 Main Memory Management

The entire main memory is divided into areas that are each managed by a specific processor.

The shared usage of the main memory therefore requires communication between the processors. If you use the Non-Uniform Memory Access (NUMA) architecture, one board can comprise up to four processor sockets that are interlinked via interfaces. All processors are linked with one another for optimal performance (see Figure 2.1).

![Figure 2.1 NUMA Architecture](image-url)
For boards with more than four processor sockets, the current design does not permit an interface between all processors. If a processor core wants to access data that is assigned to another processor core, the data must be transported in several steps. On average, access to the main memory features lower performance in this case. So far, such systems often use eight processors in the form of two NUMA architectures that are linked with one another.

Because a lot of time is lost for this type of communication and with an increasing number of processors and sockets, it becomes more and more important in larger systems to reference the calculations for a specific main memory area to the assigned processor.

Typically, memory management is carried out by the operating system. However, the operating system doesn't have any information on the structure and relevance or the dependencies of database tables and their interim results. SAP HANA is supposed to distribute calculations to as many processor cores as possible to keep the processing time as short as possible. For this reason, SAP HANA assumes main memory management completely. To do so, SAP HANA's index server process creates a separate main memory pool when the database is started. During operation, it reserves main memory for program code, database tables, temporary calculations, and other internal processes, such as database statistics. If the reserved memory is no longer required, it remains in the main memory pool and is not returned to the operating system again, as is the case in other applications. If the memory is to be occupied again, SAP HANA can decide which areas are suited here. For this reason, the operating system will report a higher main memory usage that is caused by SAP HANA processes than actually exists.

Figure 2.2 shows a sample structure of the memory pool. The memory for tables is subdivided into column store for column-based tables and row store for row-based tables. Only SAP HANA knows the amount of free memory and its composition; the operating system is informed about only the total amount of main memory that is occupied by the index server.
If the runtime environment requires main memory from SAP HANA, a request is sent to the operating system and the respective SAP HANA process receives virtual memory. This memory has not been assigned to a concrete location in the main memory, but it authorizes SAP HANA to occupy main memory in the further course of the program. If this memory is then physically filled with data, it is referred to as resident memory. It is assigned to a main memory area and can then be used by the process. Accordingly, there’s a difference between virtual memory and resident memory that results from the reserved main memory that has not yet been utilized.

In SAP HANA, used memory refers to the memory that is actually reserved by program code, tables, temporary calculations, and database management (but may not be occupied yet). The difference in the memory reserved by the operating system (virtual memory) is the free memory of the main memory pool. If main memory was reserved and is no longer required, it still remains in the resident memory and is thus part of the pool. The used memory can therefore be greater or less than the resident memory.

The used memory is the decisive variable for main memory usage. It may not exceed the predefined main memory limit.

### 2.2 Software

The dimensioning of hardware, as well as the database innovations of SAP HANA, live up to their potential particularly when you deploy the software that was developed on this basis. Existing products like SAP BW have been adapted for this purpose. Additionally, an entire product portfolio of native SAP HANA applications arises. Let’s now discuss the question of how to operate them jointly.

The original operating concept of SAP HANA included a dedicated (that is, sole) operation of the database on the server. It was not released for productive systems to use several applications on an SAP HANA appliance or several databases on a server. In the meantime, the regulations are considerably more flexible and develop further. For this reason, we can provide only a snapshot that illustrates the additional options that open up and where you can obtain information on the relevant topics.

#### 2.2.1 SAP HANA and Other Applications

If you deploy an SAP HANA database in your enterprise, you can utilize it for different purposes at the same time. SAP refers to this principle as Multiple Components One Database (MCOD). However, this principle does not apply to every conceivable combination, which is why SAP publishes a continuously updated whitelist that lists all permitted combinations. You can find details in SAP Note 1661202. Bear in mind, however, that the parallel use of several applications on SAP HANA must be taken into account when you plan the resource requirement. This information is supplemented by SAP Note 1666670, which deals with the question of whether various SAP BW systems may use the same SAP HANA database. At present, this is possible only for test and development systems; for production systems, it is expressly forbidden. Figure 2.4 illustrates these scenarios at the top right and the bottom left.

Another option is the parallel operation of several SAP HANA databases on the same server. Each installation of SAP software is usually identified with a three-digit SID. For this reason, this scenario is referred to as multi-SID. SAP Note 1681092 currently states that multi-SID scenarios
are permitted only for test and development systems. Figure 2.4 presents this scenario at the top left. The implementation via virtual machines, which is discussed in Section 2.2.2 forms an exception here. Figure 2.4 shows this scenario in action at the bottom right.

Some customers use external products like backup or monitoring software, as well as anti-virus products. They are also subject to restrictions, which are discussed in more detail in SAP Note 1730928.

Figure 2.4 provides a summary of the different scenarios. Note that these conditions may change in the course of time. If you want to examine a specific method in more detail, you should consult the SAP notes that have already been mentioned.

2.2.2 SAP HANA on Virtual Machines

A virtualization software divides the available hardware for several systems and programs. As a result, several programs or systems can operate completely independently of one another on the hardware. Virtualization is thus an alternative to the normal operation of several applications or to multi-SID installations in SAP HANA.

There are two completely different approaches for virtualization that we want to discuss for this purpose: hardware virtualization and operating system virtualization.

Hardware Virtualization

In the case of hardware virtualization, a component that is referred to as hypervisor or virtual machine monitor (VMM) controls and allocates resources. A hypervisor may already be preinstalled, depending on the server. Imagine a hypervisor as some kind of upstream operating system: only the hypervisor can access the hardware and enable system monitoring and administration. After it has been installed, you can use it to define virtual machines that contain their own (guest) operating systems. This ensures completely independent operation of virtual machines.

In the meantime, the CPU must differentiate among the processes of the various virtual machines. Instead of using the hypervisor for this purpose, AMD and Intel developed special command sets (Pacifica and Vanderpool, respectively) that can be addressed by the guest systems without detours.

This results in an environment with different operating systems that are completely independent. Hardware is accessed directly and with high performance. The best known hypervisors include XEN (Citrix), KVM (Red Hat), and vSphere (VMWare).

SAP Note 1788665 deals with this topic and is updated on a continued basis. At present, only the usage of VMware vSphere is supported for operating SAP HANA on virtual machines. For this purpose, you must still use a certified SAP HANA server or a server that is verified by SAP HANA tailored data center integration. In this case, however, live operation is supported. If you want to deploy this scenario, you should also refer to SAP Note 1995460. It discusses all restrictions, important documents, and tips that are required for setting up virtual machines. Right now, it must still be decided whether several SAP HANA databases are to be operated on a virtual machine in parallel.
Operating System Virtualization

Operating system virtualization follows a different approach. In this case, an operating system that was installed normally provides several of its guest instances that are executed in separate environments (containers).

Live operation of SAP HANA in such a container is not supported at present (April, 2015). This scenario can be used for test and development systems, however.

With this method, only one operating system is run that usually simulates an image of the actual hardware. It can be addressed more directly, and the overall overhead can be only 1%. The effort of setting up a container for virtualization is lower than with hardware virtualization.

Containers use their own applications, users, services, and directories, but they can also be inherited by the operating system. Internally, the superordinate operating system groups the container’s executed processes. These kernel control groups can be monitored and managed with a high level of detail. It is possible, for example, to restrict to concrete main memory areas or CPU cores.

2.2.3 Operating System

The operating system is one part of the SAP HANA appliance. Only SUSE Linux Enterprise Server for SAP Applications has been permitted so far for optimal operation. In the meantime, the selection was extended with Red Hat Enterprise Linux Server for some certified servers. Some special regulations, which are discussed first, apply to the two operating systems.

SUSE Linux Enterprise Server for SAP Applications

In addition to its server operating system, the enterprise SUSE Linux also publishes a version adapted for SAP whose packages are compatible with SAP applications. This version also comprises a proprietary high-availability extension, an installation wizard for SAP applications, and support tailored for SAP customers.

Some recommendations exist for operating SAP HANA on this operating system. They are summarized in SAP Note 1944799. It includes the selection of software packages and instructions for maintaining and supporting the operating system.

Red Hat Enterprise Linux Server

Red Hat Enterprise Linux Server is developed by Red Hat and is the leading product for Linux operating systems on servers in the enterprise segment. Red Hat’s products are heavily represented on the U.S. market. Because it was released for SAP HANA after the SUSE operating system, the number of SAP HANA appliances that run with Red Hat is still low.

To use SAP HANA on Red Hat, you must first make some adaptations to ensure optimal operation. All necessary steps are summarized in SAP Note 2013638. These steps are not required for SUSE Linux Enterprise Server for SAP Applications because it has already been adapted for operating SAP applications.

General Notes

New versions of the installed software packages are required for both operating systems if SAP HANA is to be deployed as of version 80. You can find a list of the relevant changes in SAP Note 2001528.

Each configuration change to the operating system should first be checked for its impact on the SAP HANA operation. General instructions for configuration changes are available in SAP Note 1730999. A blacklist with prohibited configuration changes is maintained separately in SAP Note 1731000. We recommend that you take these SAP notes seriously because incorrect configuration or non-compatible software packages can result in unpredictable behavior in SAP HANA. Frequently, the causes are no longer known in this circumstance, and the responsible persons search for the error only in the database.
2.3 Processes

The tasks of SAP HANA are implemented through several processes. You can monitor both the resource utilization and possible error sources of these processes via the operating system or SAP HANA Studio (see Chapter 6, Section 6.3).

Therefore, this section discusses the tasks of these individual processes. You will also learn about the internal functioning of the particularly important index server process:

- **Index server**
  The index server is the core of SAP HANA. It contains and processes all data of the database and therefore occupies a considerable amount of the main memory. Calculations are made by different engines depending on their type. An exception is the analysis of text data that is run in the preprocessor process. Section 2.3.1 discusses the functioning of the index server. SAP HANA engines are described in a separate section (see Section 2.3.2).

- **Statistics server**
  The statistics server records a huge number of performance and hardware data during an SAP HANA operation. Over time, this allows you to identify weak points or performance peaks (see Chapter 6, Section 6.3). The statistics server’s task can be assumed by the name server as of HANA Revision 70 (see Chapter 6, Section 6.4.3).

  In an SAP HANA cluster, the statistics server runs on a host only.

- **Name server**
  The name server is required for operating an SAP HANA cluster. It knows all SAP HANA servers and the data they manage.

- **Preprocessor server**
  SAP HANA is capable of evaluating texts with regard to language and content. It also includes the analysis of positive or negative sentiments in texts (sentiment analysis). This can be used, for example, for interpreting large amounts of texts from social media. The preprocessor runs this analysis.

- **XS engine**
  The Extended Application Services (XS) engine manages and operates native SAP HANA applications that can be developed in JavaScript (XSJS). SAP utilizes this option, too. Components for the administration and monitoring of SAP HANA are implemented in XS to an increasing extent. Even entire SAP applications—for example, Operational Process Intelligence—are developed and operated in XS. The operation of SAPUI5 applications and the creation of ODATA interfaces for application development also take place in XS.

  In an SAP HANA cluster, the XS engine runs on a host only.

2.3.1 Index Server

The index server is the central SAP HANA process. All pure database activities are run on this server. Accordingly, its structure is complex and documented for the public only to some extent. The following, therefore, deals with the tasks of the index server and provides a rough overview of how they are carried out.

**Database Accesses**

Applications, SAP systems, and technical and human users constantly connect to SAP HANA to run operations. The index server ensures the management of sessions. This particularly concerns the authentication during the logon process and the rights management of existing database connections. Chapter 6, Section 6.2 discusses rights management in more detail. Authentication can also be assumed by external systems (Kerberos, SAML-base identity providers). In this case, the index server ensures the communication with these systems. You can also set up an encrypted communication via HTTPS.

**Administration of Database Objects**

All tables, SAP HANA views, database views, procedures, development objects, schemas, and other database components are managed by the index server. This means, among other things, that it retains all metadata. Metadata is, for example, the structure, authorizations, names, and descriptions, as well as code to be executed.
Managing Stored Data

In SAP HANA, you can store data based on a column or row. The technical implementation is performed by the two index server components, row store and column store. The storage of data for restore on the hard disk is also organized by the index server. Chapter 1, Section 1.4.3 describes the functioning of the column store.

Executing SQL Commands

Engines

Read and write operations in SAP HANA are usually transferred via SQL. When an SQL statement is received, it is first analyzed and then forwarded to the corresponding component of the index server. Procedures, for example, are processed by a separate component, and special SQL commands of the planning engine are forwarded to the corresponding planning component. These components are referred to as engines. They carry out the actual processing of statements. A separate engine, the calculation engine, is also used for calculations in SAP HANA views (see Chapter 5, Section 5.2).

Commands are executed by the SQL engine in several steps. The following describes the SQL and calculation engines in more detail.

2.3.2 SAP HANA Engines

In SAP HANA, the actual execution of data operations takes place in engines. Each calculation type is mapped by a separate engine. The following sections discuss some of these engines providing greater insight into the functionality of SAP HANA.

Calculation Engine

The calculation engine is used, for example, by SQL scripts or calculation views. Chapter 5, Section 5.2 outlines the modeling of calculation views. In calculation views, you can even carry out functions of the calculation engine directly. This can result in faster evaluations if you have sufficient knowledge of the SAP HANA architecture and the engine performance.

Application function libraries are part of the calculation engine. They contain functions written in C++ that the calculation engine can access directly. However, they cannot be developed manually. Popular examples include the predictive analysis library, which comprises functions for developing forecasts, and the business function library, which offers functions for financial calculations.

The functions of the application function libraries can be deployed via special SQL commands or functions that are predefined by SAP. By means of the predictive analysis library, for example, you can set up your own processes via SAP HANA Analysis Process in an SAP BW on SAP HANA system and then schedule their execution in a process chain (see Figure 2.5). This allows you to utilize the benefits of the calculation engine from the SAP BW system without detailed technical knowledge.

Figure 2.5 Definition of an SAP HANA Analysis Process
performance. This includes the handling of temporary results, merging of data from various tables, or the search within tables.

Additionally, this ensures that parallel reading and writing to the same tables does not lead to deviation, resulting in inconsistencies. The optimized command can then be executed, and the row store and column store of the corresponding tables can be accessed.

### SQL Engine for SAP HANA Calculation Views

When you model calculation views, they are executed by the SQL engine instead of the calculation engine. This can result in a considerably faster execution because the SQL engine optimizes command chains more strongly. You must observe some restrictions, however, when you choose the SQL engine. For example, you may not use any analytic views, attribute views, or calculation views with SQL script as the data source. All restrictions that must be observed are available in the SAP HANA modeling guide at help.sap.com/hana_platform.

#### Join Engine

The only task of the join engine is to merge data from several tables. It is used by attribute views. The calculation engine also provides a method for joins of tables. To understand this necessity, you must be familiar with the resource consumption of the engines. If several engines process the same command, data exchange is necessary between them. Particularly for SAP HANA views, the performance varies greatly if you deploy only one instead of several SAP HANA engines.

#### OLAP Engine

The OLAP engine is used by analytic views and InfoCubes from the SAP BW system. It is aligned with the architecture of the star schema (see Chapter 1, Section 1.2.2) and is equipped with its own optimizer. A particular benefit of this engine is the fast execution of analytical calculations through parallel implementation of aggregation.
# Contents

<table>
<thead>
<tr>
<th>Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction</td>
</tr>
<tr>
<td>1 Introduction to SAP BW on SAP HANA</td>
</tr>
<tr>
<td>1.1 Classifying SAP BW on SAP HANA Implementation Scenarios</td>
</tr>
<tr>
<td>1.1.1 Side-by-Side and Integrated Approaches</td>
</tr>
<tr>
<td>1.1.2 Operational Analytics</td>
</tr>
<tr>
<td>1.2 Current Challenges for SAP BW</td>
</tr>
<tr>
<td>1.2.1 Changing Environment</td>
</tr>
<tr>
<td>1.2.2 Disadvantages of Relational Databases</td>
</tr>
<tr>
<td>1.2.3 Distributed Data Retention</td>
</tr>
<tr>
<td>1.3 Reasons for Migrating an SAP BW System to SAP HANA</td>
</tr>
<tr>
<td>1.3.1 SAP BW Restrictions</td>
</tr>
<tr>
<td>1.3.2 Advantages of SAP BW on SAP HANA</td>
</tr>
<tr>
<td>1.4 Basic Technical Principles</td>
</tr>
<tr>
<td>1.4.1 In-Memory Technology</td>
</tr>
<tr>
<td>1.4.2 Column-Based Data Retention and Compression</td>
</tr>
<tr>
<td>1.4.3 Insert-Only Procedure</td>
</tr>
<tr>
<td>1.4.4 Partitioning</td>
</tr>
<tr>
<td>1.4.5 Push-Down Principle</td>
</tr>
<tr>
<td>1.5 Summary</td>
</tr>
<tr>
<td>2 SAP HANA Architecture</td>
</tr>
<tr>
<td>2.1 Hardware</td>
</tr>
<tr>
<td>2.1.1 Certification</td>
</tr>
<tr>
<td>2.1.2 Cloud</td>
</tr>
<tr>
<td>2.1.3 Scale-Up/Scale-Out</td>
</tr>
<tr>
<td>2.1.4 High Availability/Data Availability</td>
</tr>
<tr>
<td>2.1.5 Main Memory Management</td>
</tr>
<tr>
<td>2.2 Software</td>
</tr>
<tr>
<td>2.2.1 SAP HANA and Other Applications</td>
</tr>
<tr>
<td>2.2.2 SAP HANA on Virtual Machines</td>
</tr>
<tr>
<td>2.2.3 Operating System</td>
</tr>
</tbody>
</table>
Contents

2.3 Processes ................................................................. 64
  2.3.1 Index Server ...................................................... 65
  2.3.2 SAP HANA Engines ............................................. 66

3 Migration and Implementation of SAP BW on SAP HANA .................................................. 69
  3.1 Migration and Implementation Scenarios ......................... 69
    3.1.1 New Installation ............................................. 72
    3.1.2 Manual Migration ........................................... 80
    3.1.3 Migration Options: Database Migration Option (DMO) and Post Copy Automation (PCA) ... 85
    3.1.4 Rapid Deployment Solutions ......................... 96
  3.2 Technical Requirements for Migration ...................... 99
  3.3 Preparation Steps ................................................ 104
    3.3.1 Creating a Homogenous System Copy ............. 105
    3.3.2 Cleanup Activities (SAP BW Housekeeping) .... 108
    3.3.3 Identifying and Checking Sizing Requirements ........................................... 121
    3.3.4 Preliminary System Validation .................... 132
    3.3.5 Additional Tools ........................................ 137
  3.4 SAP BW on SAP HANA Migration ......................... 142
    3.4.1 Export Preparations .................................. 145
    3.4.2 Export Phase .......................................... 149
    3.4.3 Import Preparations .................................. 158
    3.4.4 Import Phase .......................................... 160
  3.5 Post-Migration Steps ........................................... 166
    3.5.1 General Post-Processing .............................. 167
    3.5.2 SAP BW-Specific Post-Processing ................. 169
    3.5.3 Identifying and Remedyin...
# Index

## A

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABABVARCHARMODE</td>
<td>205</td>
</tr>
<tr>
<td>ABAP report</td>
<td>447</td>
</tr>
<tr>
<td>ABAP Routine Analyzer</td>
<td>137, 231</td>
</tr>
<tr>
<td>ABAP development</td>
<td>210</td>
</tr>
<tr>
<td>performance tips</td>
<td>229</td>
</tr>
<tr>
<td>SAP HANA procedure</td>
<td>230</td>
</tr>
<tr>
<td>Accelerator approach</td>
<td>19</td>
</tr>
<tr>
<td>Access level group</td>
<td>373</td>
</tr>
<tr>
<td>Activate R</td>
<td>355</td>
</tr>
<tr>
<td>Add SAP HANA database server</td>
<td>322</td>
</tr>
<tr>
<td>Administration effort</td>
<td>30, 33</td>
</tr>
<tr>
<td>Administration perspective</td>
<td>322</td>
</tr>
<tr>
<td>Administration Console</td>
<td>321</td>
</tr>
<tr>
<td>Administration Editor</td>
<td>335</td>
</tr>
<tr>
<td>AFL/L50478 Application Function Library</td>
<td></td>
</tr>
<tr>
<td>Aggregation</td>
<td>33, 118</td>
</tr>
<tr>
<td>Alert</td>
<td>345</td>
</tr>
<tr>
<td>Allocation limit</td>
<td>333</td>
</tr>
<tr>
<td>Alternatives to SAP BW on SAP HANA</td>
<td>193</td>
</tr>
<tr>
<td>Amazon Web Services</td>
<td>51</td>
</tr>
<tr>
<td>Analytic privilege</td>
<td>243</td>
</tr>
<tr>
<td>view</td>
<td>242</td>
</tr>
<tr>
<td>Analytical index</td>
<td>310</td>
</tr>
<tr>
<td>Application Function Library</td>
<td>43, 67</td>
</tr>
<tr>
<td>Architected data mart</td>
<td>264</td>
</tr>
<tr>
<td>Archiving</td>
<td>186, 411, 417</td>
</tr>
<tr>
<td>BW object</td>
<td>120</td>
</tr>
<tr>
<td>with nearline storage</td>
<td>412</td>
</tr>
<tr>
<td>Attribute view</td>
<td>242</td>
</tr>
<tr>
<td>Authentication</td>
<td>65</td>
</tr>
<tr>
<td>Authorization Management</td>
<td>326</td>
</tr>
<tr>
<td>Authorization missing</td>
<td>329</td>
</tr>
<tr>
<td>Auto restart service</td>
<td>55</td>
</tr>
</tbody>
</table>

## B

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Backup</td>
<td>125, 167, 357</td>
</tr>
<tr>
<td>create</td>
<td>357</td>
</tr>
<tr>
<td>restore</td>
<td>358</td>
</tr>
<tr>
<td>Basepath</td>
<td>353</td>
</tr>
<tr>
<td>BW Analyzer</td>
<td>399</td>
</tr>
<tr>
<td>BW query</td>
<td>70, 212, 308, 375, 377</td>
</tr>
<tr>
<td>optimization</td>
<td>206</td>
</tr>
<tr>
<td>BI Launch Pad</td>
<td>363, 364</td>
</tr>
<tr>
<td>alerts</td>
<td>365, 367</td>
</tr>
<tr>
<td>applications</td>
<td>366</td>
</tr>
<tr>
<td>content linking</td>
<td>369</td>
</tr>
<tr>
<td>display document</td>
<td>366</td>
</tr>
<tr>
<td>documents</td>
<td>364, 366</td>
</tr>
<tr>
<td>initial screen</td>
<td>364</td>
</tr>
<tr>
<td>messages</td>
<td>365</td>
</tr>
<tr>
<td>BI lock server</td>
<td>169</td>
</tr>
<tr>
<td>BI platform to SAP BusinessObjects</td>
<td></td>
</tr>
<tr>
<td>Business Intelligence</td>
<td></td>
</tr>
<tr>
<td>BI server</td>
<td>371</td>
</tr>
<tr>
<td>BI workspace</td>
<td>368</td>
</tr>
<tr>
<td>content linking</td>
<td>370</td>
</tr>
<tr>
<td>create</td>
<td>368</td>
</tr>
<tr>
<td>module</td>
<td>369</td>
</tr>
<tr>
<td>BICS interface</td>
<td>373</td>
</tr>
<tr>
<td>Big data</td>
<td>360</td>
</tr>
<tr>
<td>Blade</td>
<td>53</td>
</tr>
<tr>
<td>Business transformation layer</td>
<td>264</td>
</tr>
<tr>
<td>Business Warehouse Accelerator</td>
<td>70</td>
</tr>
<tr>
<td>BW Application Server Java</td>
<td>177</td>
</tr>
<tr>
<td>BW Integrated Planning</td>
<td>313</td>
</tr>
<tr>
<td>BW Migration Cockpit</td>
<td>141</td>
</tr>
<tr>
<td>BW modeling tools</td>
<td>231</td>
</tr>
<tr>
<td>installation</td>
<td>232</td>
</tr>
<tr>
<td>update</td>
<td>233</td>
</tr>
<tr>
<td>BW Transformation Finder</td>
<td>139</td>
</tr>
<tr>
<td>in SAP HANA</td>
<td>140</td>
</tr>
<tr>
<td>BW virtual data mart layer</td>
<td>291</td>
</tr>
<tr>
<td>reporting</td>
<td>291</td>
</tr>
<tr>
<td>BWA/L50478 Business Warehouse Accelerator</td>
<td></td>
</tr>
</tbody>
</table>
CompositeProvider, 212, 218, 232, 238, 291, 306

area of use, 274
configure, 239
create, 238
InfoProvider, 219
join, 274
SAP HANA view, 219
tips, 220
transport, 219
use, 218

Compression, 37, 122
Concept of non-active data, 294
Consistency check, 171
Consistent EDW core, 269
advantages, 270
Container, 62
Corporate memory, 264
Correct inconsistency in table, 171

Dashboard, 326
Dashboard Builder → BI workspace

Data
acquisition layer, 263
administration, 66
analysis, 361
archiving, 411, 417
availability, 54
BW managed, 278, 279
classification by access frequency, 300
consolidation, 434
data mart approach, 18
dispatcher process, 182, 282, 285
flow template, 266
growth, 24

HANA Cookbook, 180

Hard disk, 339

hardware certificate, 49
scale-out, 53
scale-up, 53
sizing, 121
Open ODS view (Cont.)
create, 216, 236
Smart Data Access, 218
Open Operational Data Store layer
→ Open ODS layer
Open planning requests, 318
Operating system, 62
virtualization, 62
Operational analytics, 22
Operational Data Provisioning, 220
DataSource, 223
Open ODS view, 224
prerequisites, 225
source system, 222
Operational data store, 264
Operational Delta Queue, 220

P
PaaS → Platform as a Service
Pacific, 61
Package, 243
PAK → Planning Application Kit
PAM → Product Availability Matrix
Parallel export and import, 165
Partitioning, 40, 53
type, 41
Password change, 175
Performance, 31, 206, 342
historical data, 342
Persistent Staging Area, 118
cleanup, 118
Perspective, 319
show, 321
Physical memory, 337
Planning Application Kit, 70, 177, 313
activate, 177, 317
disaggregation, 315
performance, 314
Planning scenario, 436
Platform as a Service, 50
POC → Proof of Concept
Post-migration report, 168
Preprocessor server, 64
Private
cloud, 51
Private (Cont.)
view, 431
Privilege
adapt, 175
Procedure, 243
Process chain, 258
build, 260
evaluate, 138
performance, 261
step, 145
structure, 259
Product Availability Matrix, 49
Production system, 201
migration, 201
Proof of Concept, 188
advantages, 188
analysis, 193
BW scenario, 190
cost, 190
costs, 189
execution time of reports, 191
hardware, 189
Knowledge transfer, 189
PSA → Persistent Staging Area
Public cloud, 51

Q
Quality and harmonization layer, 263
Quality assurance system, 200
migration, 200

R
R3 program, 182
Range partitioning, 41
Read-optimized table, 39
Real-time
analysis, 360
data access, 25
Red Hat Enterprise Linux Server, 63
Redo log, 354
Relational database, 25
reload_tables, 355
Remodeling Toolbox, 30
Reports
RSDU_TABLE_CONSISTENCY, 197
SMIGR_CREATE_DDL, 147
Reporting, 231, 262, 291, 359, 427
EDW propagation layer, 273
layer, 264
purpose, 359
third-party data, 277
trend, 360
Resolve DDIC inconsistency, 173
Resource utilization, 343
Restricted column, 248
Reuse view, 431
Rights management, 65
Role, 326
create, 331
Round-robins partitioning, 41
Run length encoding, 38
Runtime, 158
S
SaaS → Software as a Service
SAP Business Warehouse
Planning and Consolidation, 374
SAP Business Warehouse
administration effort, 30
benefit, 433
data consolidation, 434
data extraction, 284
data preparation, 435
demo scenario, 310
EDW service, 279, 289
integration service, 278, 279
modeling tools, 218
nearline storage, 414
operational data services, 278, 282
process chain, 258
real-time data replication, 283
use standard function, 230
SAP Business Warehouse Accelerator
→ SAP BWA
SAP BusinessObjects Analysis
edition for Microsoft Office, 399
edition for OLAP, 402
SAP BusinessObjects Business
Intelligence, 361, 363
SAP BusinessObjects Dashboards, 395
SAP BusinessObjects Design Studio, 391
component, 394
data sources, 392
SAP BusinessObjects Explorer, 404
SAP BusinessObjects Web
Intelligence, 388
Rich Client, 391
SAP HANA view, 390
SAP BW on SAP HANA
advantages, 22, 31
architecture, 434
nearline storage, 411
reporting, 427
vs. SAP HANA Live, 433, 436
SAP BWA, 28
SAP Crystal Reports, 383
Crystal Reports 2013, 383
Crystal Reports for Enterprise, 383
SAP customer message, 166
SAP Data Services, 263
SAP GUI, 375
SAP HANA
architecture, 47
basic principles, 17
check version, 181
checklist tool, 132
cloud, 50
CPU load, 334
data model, 23
development, 36
hard disk capacity, 125
hardware sizing, 123
hardware virtualization, 61
live operation, 52
main memory, 123, 333
management, 56
models, 280
operating system virtualization, 62
procedure, 230
released BW components, 77
requirements, 36
server, 49
virtualization, 60
SAP HANA App Services, 52
SAP HANA Cloud Platform, 52
SAP HANA DB Services, 52
SAP HANA Development perspective, 320
SAP HANA Enterprise Cloud, 52
SAP HANA Infrastructure Services, 52
SAP HANA Live, 23, 429
SAP HANA Live Browser, 429
SAP HANA Modeler perspective, 320
SAP HANA One, 51
SAP HANA PlanViz, 320, 344
SAP HANA Studio, 23, 182, 231, 319
BW modeling perspective, 234
configuration file, 351
determine version, 182
install, 321
modeling perspective, 241
monitoring, 333
Project Explorer, 234, 235
user administration, 328
SAP HANA view, 304, 428
access to, 228
activate, 249
consume in BW, 303
data source, 246
generate, 227
graphical element, 246
model, 240, 243, 246
requirement, 243
semantic information, 248
structure, 245
SAP IQ, 411
SAP LT Replication Server, 283, 288
SAP Lumira, 406, 409
SAP Note, 53, 54, 182, 183, 196, 197
implement, 183
overview, 442
search, 184
SAP OS/DB migration check, 201
SAP Predictive Analytics, 362, 409
SAP Profitability and Cost Management, 374
SAP QuickStere, 126
sap.hana.xs.admin.roles:: SQLCCAdministrator, 332
Save point, 340
Scale-out, 53, 124
approach, 36
Scale-up, 53, 124
Security aspects, 194
SELECT *, 230
Server, 49
components, 363
number, 54
Set up system users, 160
Shared memory, 338
Single point of truth, 265
Single sign-on (SSO), 375, 377
Sizing
report, 126
sample calculation, 130
SLT -> SAP LT Replication Server
Smart Data Access, 214
open ODS view, 215, 218
SAP HANA view, 215
Software as a Service, 51
Software Provisioning Manager, 183
SQL
ingine, 67
plan cache, 344
statement, overview, 452
Standard users, 327
Star schema, 23, 25
Start export run, 157
Statistical data, 113
Statistics server, 64
disable, 356
Stop job processing, 146
Stored procedure, 436
SUSE Linux Enterprise Server for SAP Applications, 62
SWAP memory, 333
SWPM, 144
settings, 160
System copy, 187
homogeneous, 105
System landscape, 186
migration, 195
multilevel, 195
System validation, 132
options, 135
Systems view, 322
Tailored data center integration, 49
Test transport, 200
Trace, 349
table access, 350
type, 349
Trailing space, 205
Training, 185
Transaction
overview, 448
RSMigRHANADB, 196
RSMRT, 30
RSRT, 192, 206
SARA, 186
Transformation
code push-down, 225
TransientProvider, 281, 310
TREX, 35
Trigger read ratio, 341
Trigger write ratio, 341
Troubleshooting, 166, 345
Update Table DBDIFF, 147
Used memory, 337
User
create, 327
V
Vanderpool, 61
Version
DBSL, 182
SAP HANA client, 182
SAP HANA database, 181
SAP HANA Studio, 182
SAP kernel, 182
Virtual data mart layer, 273
Virtual data model, 431
Virtual Machine Monitor, 61
Virtual memory, 337
Virtualization, 60
layer, 264
VirtualProvider, 281, 311
create, 311
VMM -> Virtual Machine Monitor
VMware vSphere, 61
Web-based analysis, 404
WHERE condition, 230
Work process, 182
Workbench, 319
X
Xcelsius /L50478 SAP BusinessObjects Dashboards
XS
debugger, 356
engine, 64
jobs, 356
Matthias Merz is head of the Center of Excellence for SAP HANA at Camelot ITLab GmbH, advising companies in the area of SAP BW on SAP HANA. He has a degree in business informatics and has been working with SAP for over ten years.

Torben Hügens Dr. Torben Hügens heads the Center of Excellence for SAP BusinessObjects BI at Camelot ITLab GmbH. Previously, he was an employee of SAP Germany AG & Co. KG and worked as a consultant in the area of SAP BusinessObjects BI and SAP BW.

Steve Blum Steve Blum is a junior consultant at Camelot ITLab GmbH and works in the Center of Excellence for SAP HANA. He specializes in XS development, SAP HANA administration, and nearline storage.