SAP Solutions

CIO Guide

How to Use Hadoop with Your SAP® Software Landscape

February 2013

The Best-Run Businesses Run SAP™

Table of Contents

About the Authors

David Burdett and Rohit Tripathi, Product Architecture & Technology Strategy group at SAP.

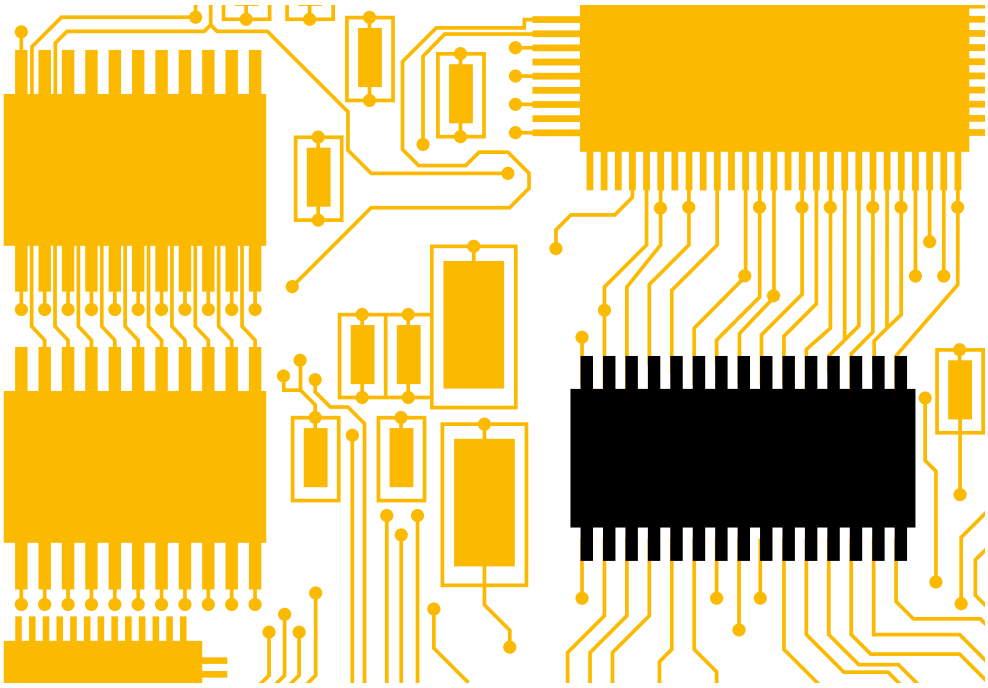
Executive Summary

The Apache Hadoop open-source software framework runs applications on large clusters of commodity hardware and is especially suited for data-intensive distributed applications. This paper explains how to leverage Hadoop in an environment where SAP® solutions are a major consideration. As open-source technology, Hadoop supports the analysis and processing of very large volumes of data from a great number of varied, structured and unstructured sources.

Among the technologies considered in this paper are the SAP HANA® platform, SAP Sybase® IQ database software, and Hadoop. In helping businesses find the right balance of data technologies that will solve its business problems, this paper discusses such questions as:

- When is Hadoop the best solution to a business problem?
- How should Hadoop be used alongside SAP solutions and technology?

It includes a definition of Big Data and an overview of Hadoop and draws a comparison between Hadoop and other database technologies. The paper describes key scenarios for which Hadoop could be the better solution to use. It provides a reference architecture for how to use Hadoop alongside SAP solutions, guidelines for implementing Hadoop, and an outline of future trends that have a bearing on Hadoop.



Introduction and Scope

Big Data is one of the top technology trends impacting information infrastructure in 2013 according to Gartner. "Significant innovation continues in the field of information management (IM) technologies and practices driven by the volume, velocity, and variety of information, and the huge amount of value – and potential liability – locked inside all this ungoverned and underused information."[1]

SAP HANA is already helping businesses to unlock this information by addressing one very important aspect of Big Data – fast access to and real-time analytics of very large data sets – that allows managers and executives to understand their business at "the speed of thought."

SAP has also announced SAP Real-Time Data Platform, which combines SAP HANA with SAP Sybase IQ and other SAP technologies as well as with non-SAP technologies, especially Hadoop, which is the focus of this paper. SAP Real-Time Data Platform can be used for both analytics and online transaction processing (OLTP). When used alone, each technology delivers business value. When used together, however, they can combine, analyze, and process all the data a business has, providing deeper insights into the business and opening up new business opportunities.

To achieve the best balance of data technologies to solve its business problems, a business must take into account many factors. Besides the cost of hardware and software, it must consider development tools, the operational costs associated with meeting its own service levels, and how it will fulfill its policies concerning security, high availability, secure backup, and recovery.

This raises questions that this paper aims to answer. They are:

- When is Hadoop really the "best" solution to a business problem?
- How should you use Hadoop alongside SAP solutions and technology?

There are some major differences between these technologies. At a high level, Hadoop uses commodity servers to handle data sizes in the petabyte and potentially the exabyte[2] range, which is much higher than the 100 TB range (or less) that SAP HANA and conventional relational database management systems (RDBMS) typically handle.[3] On the other hand, current versions of Hadoop are significantly slower than a conventional RDBMS, as well as much slower than SAP HANA, taking minutes or hours to provide analytic results. However, these versions can handle arbitrary data structures more easily and usually at much lower hardware storage costs per terabyte.

This means that Hadoop, unlike SAP HANA, will not enable you to understand your business at "the speed of thought." However, by allowing you to store and access more voluminous and detailed data at lower cost, it lets you drill deeper and in different ways into the data underlying your business.

The net result is that by putting SAP HANA and Hadoop together you have the potential to handle really big data really fast.

1. R. Casonato et al., *Top 10 Technology Trends Impacting Information Infrastructure*, 2013. Gartner Inc. http://www.gartner.com/newsroom/id/2359715.
2. 1 petabyte = 1,000 terabytes; 1 exabyte = 1,000 petabytes.
3. SAP HANA has been implemented using ~100 TB databases as of the time of writing this paper. With hardware advances, the practical implementation database size for SAP HANA is likely to increase significantly over time.

The key questions are **when is Hadoop really the "best" solution to a business problem** and **how should you use Hadoop alongside SAP solutions and technology**?

What This Paper Contains

Section	Summary
Big Data: A Definition	Get an overview of Hadoop. Read about the relationship and differences between conventional relational database management systems (RDBMSs), in-memory computing – especially the SAP HANA® platform – and Hadoop.
Key Scenarios	Find out how running Hadoop alongside SAP HANA and other SAP® solutions can help you solve new and different problems.
Reference Architecture	Learn how SAP technology can be used to support key scenarios and major use cases. Using SAP technology as a foundation, see how you can develop a plan to integrate Hadoop into your enterprise's SAP software landscape.
Example Use Cases	Review detailed examples based on the reference architecture that illustrate how Hadoop can be used with SAP solutions to solve business problems.
Hadoop Implementation Guidance	Become familiar with how to plan, use, and implement Hadoop in an SAP software landscape.
Future Trends	Review trends for Big Data technology, especially for Hadoop. Get insight into how you can position your enterprise to best leverage Hadoop and related technologies in light of how they are likely to evolve.

After reading this paper you should be able to understand Hadoop, its key features, and how it compares and relates to existing data technologies. You should be able to assess SAP solutions and SAP technology in order to determine the best time to leverage Hadoop. You should be able to start shaping your company's strategy for implementing Hadoop alongside SAP solutions and SAP technology.

Big Data: A Definition

Big Data has been defined in many ways. This paper uses the definition offered by the TechAmerica Foundation's Federal Big Data Commission: "Big Data is a term that describes large volumes of high velocity, complex, and variable data that require advanced techniques and technologies to enable the capture, storage, distribution, management, and analysis of the information." [4]

Big Data presents many challenges, often referred to as the "three Vs": velocity, volume, and variety. Not only is the volume of data large, it is arriving ever more rapidly – for example "machine data" generated on the factory floor or trading data generated by financial markets. It is also of many different types – from comments about products on Twitter or Facebook and logs of customer behavior on a company Web site to data describing rapidly developing world or weather events. All these types of data can have a significant effect on a business. Finding out quickly what the data means and understanding its importance provides a business with an ongoing advantage as well as the opportunity to realize competitive benefits.

There is a fourth challenge, "veracity." Being able to trust the data used to make decisions is not a new problem. At a minimum, a business must have some idea of its data's accuracy and source, which must be taken into account when decisions based on that data are being made. Big Data both extends and amplifies this problem by significantly increasing both the volume and variety of data, whose veracity requires assessment.

Yet understanding the meaning and importance of Big Data requires systems and solutions that can handle not just the velocity, volume, variety, and veracity of data but also cope with the changes in additional data sources yet to be identified. It must be possible to combine it with existing analytic and business data in order to derive a complete picture of what is going on. Finally, it must be delivered in a comprehensive and understandable way using quality business intelligence tools such as those provided by SAP BusinessObjects™ business intelligence (BI) solutions.

A CONVENTIONAL DISK-BASED RDBMS DOESN'T SOLVE ALL PROBLEMS

The three Vs present challenges for conventional disk-based relational databases. For a start, conventional databases were not designed to handle the database insert/update rates required to support the **velocity** at which Big Data arrives or the speed with which the Big Data must be analyzed. Conventional relational databases also require advance creation of database schemas that define what the data looks like. This makes it harder for them to handle **variety** in data. (The section "How Hadoop Handles the Three Vs" provides additional details.)

Some RDBMSs, including SAP Sybase IQ, have evolved to address these challenges. SAP Sybase IQ uses column-store technology, enabling it to compress data efficiently, and a parallel processing approach on multiple servers to handle multipetabyte data stores.

Database appliances have been built to address these problems, including the SAP HANA database and SAP HANA software, which have demonstrated[5] a greater than 20x compression rate where a 100 TB five-year sales and distribution data set was reduced to 3.78 TB and analytic queries on the entire data set ran in under four seconds.

Another approach is to use nonrelational data stores, such as Hadoop, that use commodity servers to handle multiple petabytes and potentially exabytes[6] of a wide variety of data at high arrival rates with acceptable processing times.

WHAT'S HADOOP?

Since Hadoop is one of the potential technologies for handling Big Data, what is it and how does it work?

At a high level, four main features are central to understanding Hadoop:

- A cluster of commodity servers
- The Hadoop MapReduce programming model
- The Hadoop software architecture
- The Hadoop ecosystem

4. TechAmerica Foundation, *Demystifying Big Data: A Practical Guide to Transforming the Business of Government*. www.techamerica.org/Docs/fileManager.cfm?f=techamerica-bigdatareport-final.pdf.
5. SAP white paper, *SAP HANA Performance: Efficient Speed and Scale-Out for Real-Time BI*. http://www.saphana.com/servlet/JiveServlet/previewBody/1647-102-3-2504/HANA%20Performance%20Whitepaper.pdf. Compression rates and performance depend heavily on the characteristics of the actual data. Individual results may vary.
6. 1 petabyte = 1000 terabytes; 1 exabyte = 1000 petabytes

Cluster of Commodity Servers

Hadoop runs on hundreds or thousands of commodity (low-cost) servers. Most of the servers are, in Hadoop terminology, "DataNodes," each of which contains just a part of the data. By breaking down a processing job into hundreds or thousands of smaller jobs running in parallel on individual machines, Hadoop can scale to handle petabytes or more of data by just adding more DataNode servers. Hadoop also provides automatic replication of data between DataNodes, which means if one DataNode fails, both the Hadoop jobs running at the time and any lost data can be recovered.

Current versions of Hadoop have one "NameNode"[7] that manages the data stored on the DataNodes and data replication (see Figure 1). The NameNode also initiates and manages the analytic and processing jobs and restarts them if any fail, helping guarantee that a Hadoop job will complete.

The Hadoop MapReduce Programming Model

The MapReduce programming model is what Hadoop follows to execute analytic and processing jobs in parallel (see Figure 2). The model divides execution into two parts:

- The **map phase** splits the problem to be solved into multiple smaller jobs, each of which produces intermediate partial results.
- The **reduce phase** combines data from the intermediate results of the map phase to produce the final result. The reduce phase may also consist of multiple jobs processing in parallel.

Hadoop hides a lot of this complexity, for example, by managing the distribution of jobs and restarts.

Figure 1: Cluster of Commodity Servers

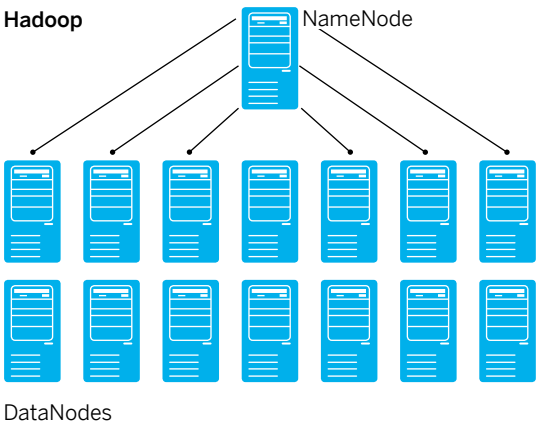
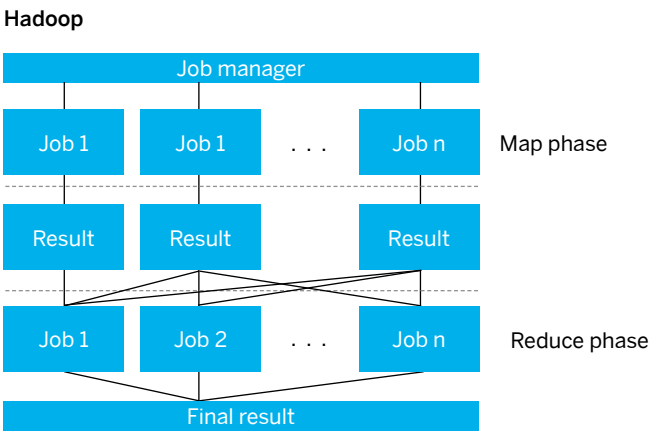


Figure 2: The Hadoop MapReduce Programming Model

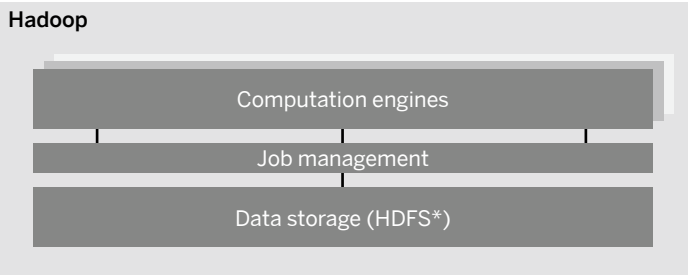


7. Hadoop 2.0 will have multiple NameNodes to provide greater resilience from NameNode failure.

Hadoop Software Architecture

A key feature of the Hadoop software architecture is the way it separates how data is stored from the way it is processed. Hadoop has a single level of data storage called the Hadoop Distributed File System (HDFS). It stores data using native operating system (for example, Linux) files. This means Hadoop can support any type of data and data can be dumped in HDFS without directly using Hadoop software. This architecture allows multiple computation engines to run on top of Hadoop and leverage both HDFS and the MapReduce programming model (see Figure 3).

Figure 3: Hadoop Software Architecture



*Hadoop Distributed File System

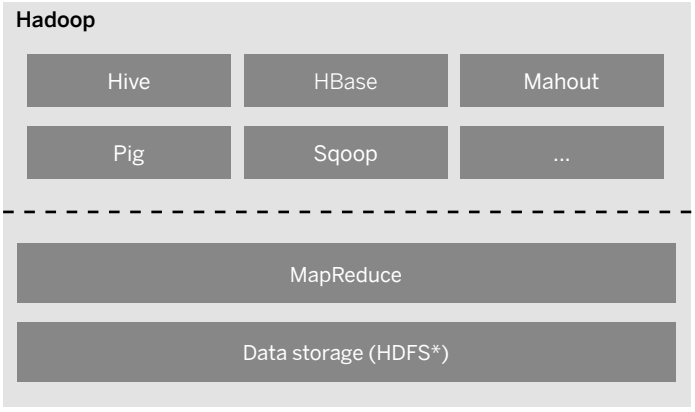
Hadoop Ecosystem

One of the core parts of the Hadoop software architecture is "Hadoop common," a set of core components that provide services that the computation engines can use. These core components serve as a foundation for an entire Hadoop ecosystem that, through its initiatives, provides additional features on top of Hadoop (see Figure 4). Just some of the many features these initiatives have produced include:

- HBase provides APIs that allow Hadoop to store and retrieve multimegabyte documents rapidly using a key, thus serving as a real-time "key value" store.
- Hive makes it possible to use Hadoop as a read-only relational database, although it's not fully Structured Query Language (SQL) compliant.
- Mahout provides a library of statistical and analytical software that runs on Hadoop and can be used for data mining and analysis.
- Pig generates MapReduce programs written in Pig Latin, a procedural language that can be used for data analysis.
- Sqoop transfers bulk data between Hadoop and structured data stores such as relational databases.

Hive and HBase are examples of NoSQL data technologies. The term was used to refer to data technologies that do not use SQL as the query language but the term is now accepted as indicating data technologies that use SQL as well as other query languages. It emphasizes that there are other ways of storing, processing, and retrieving data beyond that which conventional RDBMSs offer. A lot of new solutions are being built in this area, many of which are built on Hadoop (see the "Future Trends" section).

Figure 4: Hadoop Ecosystem



*Hadoop Distributed File System

How Hadoop Handles the Three Vs

The section "A Conventional Disk-Based RDBMS Doesn't Solve All Problems" outlined the challenges the three Vs – velocity, volume, and variety – present for disk-based relational databases. Let's look at these challenges in more detail and find out how Hadoop handles them.

Hadoop handles **data velocity** by avoiding, or at least postponing, the overheads associated with inserting data into a conventional RDBMS. These overheads can be caused by:

- **Validating data** – Before data can be inserted into an RDBMS, it must be validated to ensure that it is compliant with the database schema. This requires lookups in other tables to support referential integrity.
- **Creating indexes** – Typically, database indexes are created at the time records are inserted to support database joins and faster searching. This may involve multiple disk operations to complete.
- **Ensuring consistency** – A major reason for using an RDBMS is the way it maintains data consistency. An example is how an RDBMS ensures that an order is either inserted into a database in its entirety or not at all. The RDBMS prevents simultaneous updates to the same record and keeps logs of database changes to roll back updates when an update fails midstream and recover data if it gets lost or damaged. This approach supports the Atomicity, Consistency, Isolation, and Durability (ACID) of the data and is commonly called ACID compliance.

It can take a significant amount of time and resources for these processes and checks to complete, which may limit data insertion rates unless the RDBMS, including the way it inserts records, is designed to avoid these delays. Hadoop is different in that it stores the data in its raw form and replicates the data on multiple DataNodes to make sure it does not get lost. Any processes for validation, index creation, or consistency checks that may be needed are performed later when they will not impair the rate of data insertion.

Hadoop handles **data volume** by splitting data, as well as its processing, between multiple DataNodes (see the "Cluster of Commodity Servers" section). As data volumes or the processing workload on each DataNode increase, the data in the node can be split and more nodes added.

Hadoop handles **data variety** by storing data as Linux operating system files without checking or processing them first. This means that absolutely any type of data can be stored, and it removes the need to understand and define the structure of the data beforehand.

Hadoop and Data Veracity

Current versions of Hadoop do not provide help in addressing **data veracity**. Consequently, if the data stored in Hadoop is to serve as a trusted basis for decision making, the data quality must be assessed, understood, and improved. This may include:

- **Validating data** – This involves looking for inconsistencies in the data and correcting them. It could be as simple as correcting spelling or other mistakes in the data.
- **Enriching data** – You can add additional metadata, for example, by mapping product names provided as text to their respective product identifiers.
- **Assessing data quality** – You can analyze data in Hadoop for the number and types of errors it contains.

More detail on assessing and improving data veracity is provided in the "SAP Data Services" section, which describes how that software can be used for this purpose.

CONTRASTS BETWEEN RDBMS, IN-MEMORY (SAP HANA), AND HADOOP

Choosing the data technology to use in an OLTP or analytical solution requires understanding the differences between the choices. The table highlights the main differences between a conventional RDBMS; an in-memory database, specifically SAP HANA; and Hadoop. You will note the table is not product specific and is therefore somewhat of a generalization.

Technology is undergoing rapid innovation and the details shown in the table below are sure to change (see "Future Trends" section). However, it is worth noting that the following key differentiating characteristics of each database type will likely continue to hold true in the future:

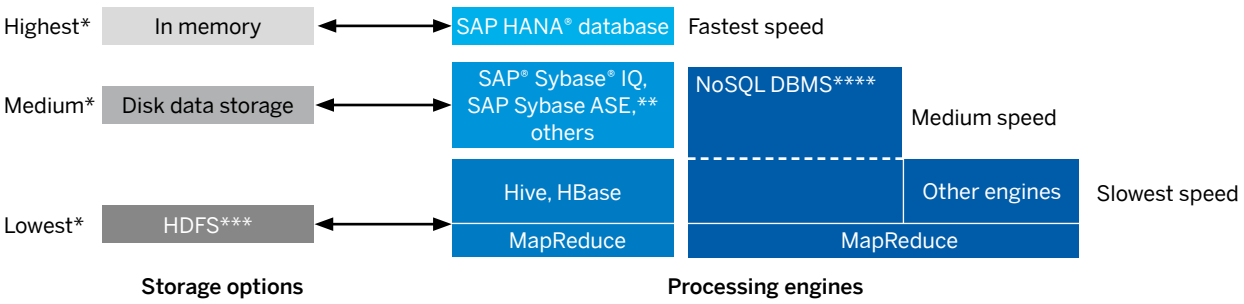
- **RDBMS** will continue to be an acceptable solution for many problems, especially for straightforward OLTP, where time criticality is of no business importance.
- **In-memory computing** embodied in such products as SAP HANA is best suited when speed is important, for example, for real-time data updates and analytics, but the volume of data is not excessively large and the cost is justifiable given the business need.
- **Hadoop** is better suited when the data volume is very large, the type of data is difficult for other database technologies to store (for example, unstructured text), and slow data analysis and processing are acceptable or costs must be minimized.

RDBMS, In-Memory (SAP HANA) Databases, and Hadoop

Relational Database Management System	In-Memory (SAP HANA®) Database	Hadoop
Structured data stored on disk	Mainly structured data in memory	Any data or file structure on disk
Slow data access (~10 ms)	**Very fast access (~<1 ms)**	**Very slow data access (seconds to hours)**
Predefined schema	Predefined schema	**No schema or postdefined schema**
1 server (~8 to 32 cores[8])	1 or many servers (100s of cores)	Distributed servers
Scale-up architecture	Scale-up/scale-out architecture	Scale-out architecture
Server-level failover	Server-level failover	**Query and server-level failover**
Existing server technology $	Database appliance $	**Commodity (low-cost) servers**
Excellent OLTP[9]	**Excellent OLTP**	**No OLTP**
Acceptable OLAP[10]	**Excellent OLAP**	**Slow OLAP**
High data consistency – based on ACID[11] principles	High data consistency – based on ACID principles	Eventual data consistency (BASE)[12]
Enterprise-ready administration tools	**Enterprise-ready administration tools**	Few enterprise-ready administration tools
Evolution rather than innovation	**Rapid innovation**	**Rapid innovation**
No IT skills shortage	**IT skills shortage**	**IT skills shortage**
License fees required	License fees required	No fees – open source

8. Total number of CPU cores available to the database software
9. Online transaction processing
10. Online analytical processing
11. Atomicity, Consistency, Isolation, and Durability
12. Basic Availability, Soft state, Eventual consistency

Figure 5: Map of Database Storage Options to Processing Engines



*Cost per terabyte **SAP Sybase Adaptive Server® Enterprise ***Hadoop Distributed File System ****Database Management System

CHOOSING THE "BEST" DATA TECHNOLOGY

When deciding on the best balance of technologies to manage your business challenges, you will have to make some trade-offs. For example, the basic Hadoop software is open-source software, has no license fees, and can run on low-cost commodity servers. However, the total cost of running a Hadoop cluster can be significant when you consider the hundreds or potentially thousands of servers that will need to be managed. In achieving the best balance, you must consider that the relative performance and costs of the different components are also changing. For example, the cost of memory is steadily decreasing – it is also getting faster. As a result, the cost of hardware required to store a terabyte of data in memory will probably also decrease, which might make SAP HANA a better technology to use for a specific situation. Moreover, if your application requires real-time analysis, then an in-memory computing technology, specifically SAP HANA, is likely to be the only one that will meet the need.

Although this paper focuses on Hadoop, Hadoop may not always be the best solution. Other technologies should be considered as well, including SAP Real-Time Data Platform, which combines SAP HANA, other SAP technologies like SAP Sybase IQ, and Hadoop. The section "Hadoop Implementation Guidance" can help you decide where and how to use Hadoop. That said, we would now like to focus on the kinds of problems Hadoop can help you solve. The next section provides examples.

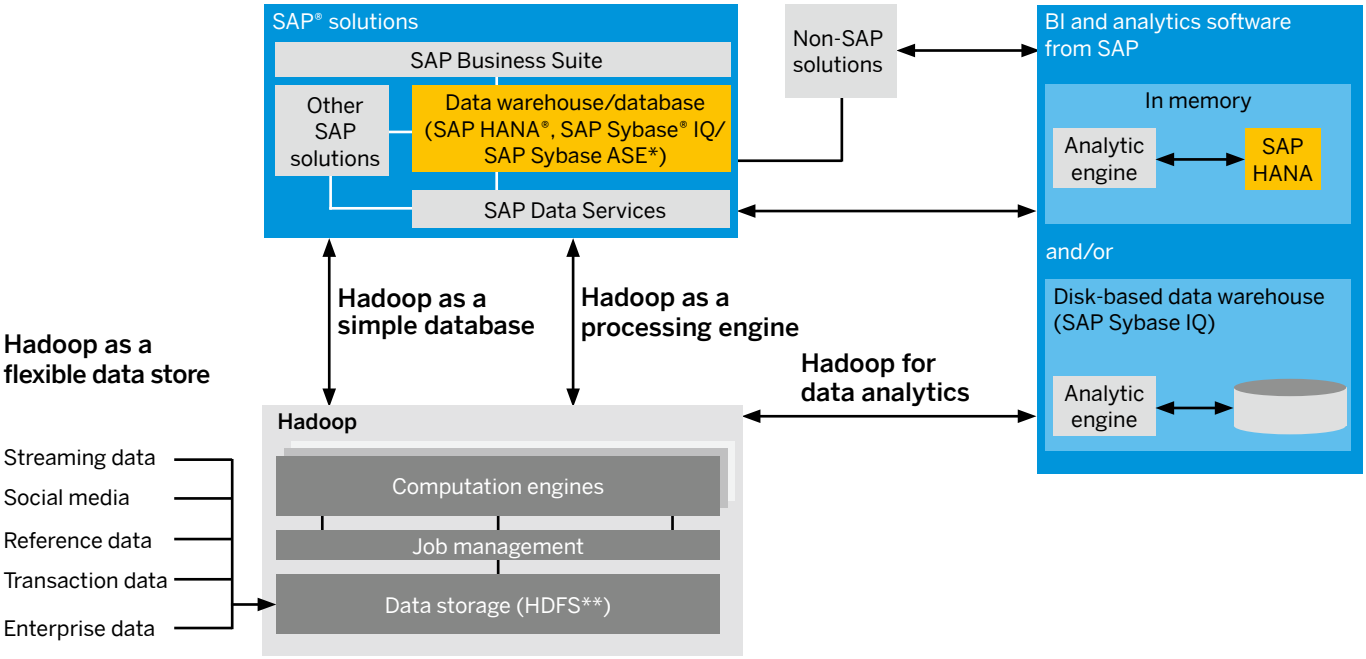
Key Scenarios

This section describes how and where Hadoop could be used to solve Big Data problems, either on its own or combined with software like SAP HANA, SAP Data Services software, or SAP Sybase IQ software. The goal of the section is to provide ideas that will help businesses identify further similar uses of Hadoop for themselves. The scenarios are grouped into the following areas:

- **Flexible data store** – Using Hadoop as a flexible store of data captured from multiple sources, including SAP and non-SAP software, enterprise software, and externally sourced data
- **Simple database** – Using Hadoop as a simple database for storing and retrieving data in very large data sets
- **Processing engine** – Using the computation engine in Hadoop to execute business logic or some other process
- **Data analytics** – Mining data held in Hadoop for business intelligence and analytics

The relationships between these scenarios are illustrated in Figure 6.

Figure 6: Key Scenarios for Hadoop



*SAP Sybase Adaptive Server® Enterprise **Hadoop Distributed File System

Each of these high-level scenarios can be used to solve a number of more specific problems. These are described in the following sections.

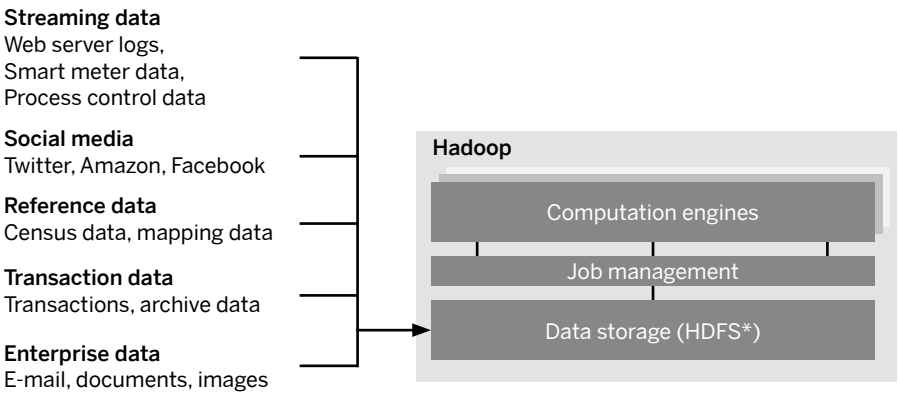
Hadoop can be used as a flexible data store, a simple database, for processing massive amounts of data as well as for data analytics.

HADOOP AS A FLEXIBLE DATA STORE

Data can help drive better decisions because it provides a deeper understanding of what is going on. Hadoop can be used cost-effectively to capture any data produced by a business including lower level, granular data such as raw-data feeds from low-level instrumentation or line-item-level transactional data. By keeping the lowest level of data, rather than aggregate data, there are no theoretical limits on the types of analyses that can be carried out, as none of the data has been discarded.

Figure 7 provides an overview of the types of data Hadoop can capture and store.

Figure 7: Scenarios for Hadoop As Flexible Data Store for Any Kind of Data



*Hadoop Distributed File System

The table provides a few sample use cases for using Hadoop as a flexible data store. Note that these are only a few examples of what is possible.

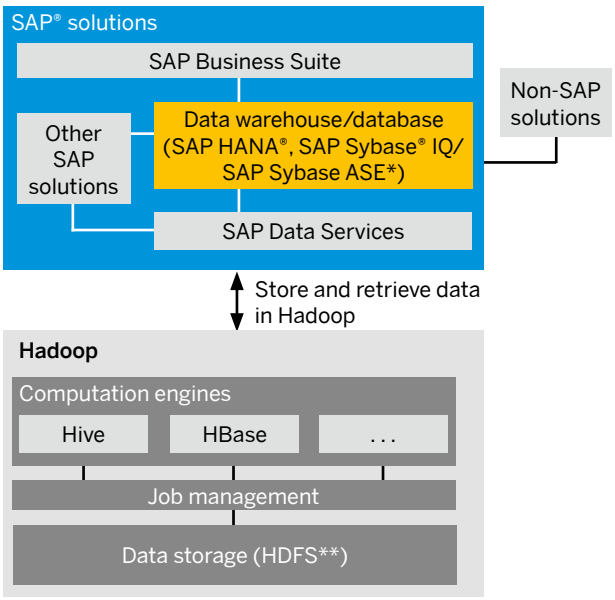
Scenario	Description	Sample Use Cases	Comment
Data stream capture	Real-time capture of high-volume, rapidly arriving data streams	Smart meters, factory floor machines, real-time Web logs, sensors in vehicles	Analyze low-level data to gain insight.
Social media	Real-time capture of data from social media sites, especially of unstructured text	Comments on products on Twitter, Facebook, and Amazon	Combine social media data with other data, for example, customer relationship. management data or product data, in real time to gain insight.
Reference data	Copy of existing large reference data sets	Census surveys, geographic information systems, large industry-specific data sets, weather measurement and tracking systems	Store reference data alongside other data in one place to make it easier to combine for analytic purposes.
Audit and risk management	Capture of business events from multiple systems for later risk analysis and audit	Transactions from SAP and non-SAP software and external systems	Correlate and analyze data from disparate systems to help identify risks.
Low-level transaction data	Long-term persistence of transactional data from historical online transaction processing (OLTP)	Call center and other process transactions	Capture low-level transaction data to enable almost any type of analytics, especially to improve business processes.
E-mail histories	Capture of logs of e-mail correspondence an enterprise sends and receives	Fulfillment of legal requirements for e-mail persistence and for use in analytics	Combine data from e-mail with other data to support, for example, risk management.
Document storage	Capture of business documents generated and received by a business	Healthcare, insurance, and other industries that generate or use large volumes of documents that must be kept for extended periods	Store unlimited number of documents in Hadoop, for example, using HBase.
Data archive	Capture of archive logs that would otherwise be sent to off-line storage	Computer system logs or other archive data	Lowers cost when compared with conventional solutions

HADOOP AS A SIMPLE DATABASE

Hadoop can be used as a simple database, when the main focus is on simple data storage and retrieval for use by other systems. To do this, computation engines such as Hive or HBase must run on top of Hadoop. (See section "Hadoop Ecosystem" for more details on Hive and HBase.)

Figure 8 provides an overview of the sample scenarios in which Hadoop can be used as a simple database.

Figure 8: Scenarios for Hadoop As a Simple Database



*SAP Sybase Adaptive Server® Enterprise
**Hadoop Distributed File System

The table provides a few sample use cases – only a few examples of what is possible.

Scenario	Description	Sample Use Cases	Comment
Extract, transform, and load (ETL) from other systems to Hadoop.	Pushing data stored in Hadoop to another software system, such as the SAP HANA® platform or other data warehouse software	Combine analytic data in SAP HANA with data from Hadoop; aggregate data in Hadoop to create online analytical processing (OLAP) fact tables for upload to SAP HANA.	SAP® Data Services software provides ETL support for data transfer from Hadoop to SAP HANA; using Hadoop frees SAP Data Services for other tasks.
Get near-real-time access from other systems to Hadoop.	Structured data stored in Hadoop treated as if it were in a relational database using Hive's SQL-like interface	Carry out direct queries on smart-meter data or other low-level data stored in Hadoop.	Hive is much slower than a relational database management system. SAP Data Services leverages Hive for ETL. SAP Sybase® IQ database software provides direct access to data in Hadoop.
Provide real-time database for very large documents and very high data volumes.	Rapid store and retrieval of "blobs" of data in Hadoop using HBase	Use as a key to store and retrieve any large document, for example, a PDF, image, or video.	This functionality is used by Facebook and other social media Web sites for storing and retrieving data.

HADOOP AS A PROCESSING ENGINE

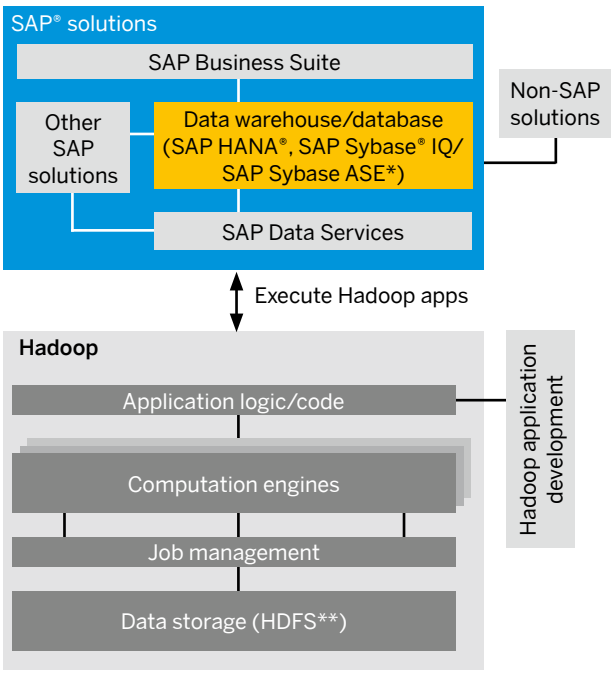
Although Hadoop can be used as a simple database when the focus is on data retrieval, MapReduce programs can be written and deployed that execute process logic on Hadoop data for many purposes, such as Pig for data analysis and Mahout for data mining or risk analysis (see section "Hadoop Ecosystem").

Figure 9 shows how this works.

Typically, developers create MapReduce application code in their language of choice, which is then deployed and executed on Hadoop. The result may be a stand-alone software system that runs on Hadoop or one that is part of or a step in a larger software system or application.

There is often a need for replication of data, especially master data, between SAP (and non-SAP) software and Hadoop, so that it can be used by the MapReduce code. (This can also be useful when using Hadoop as a simple database.)

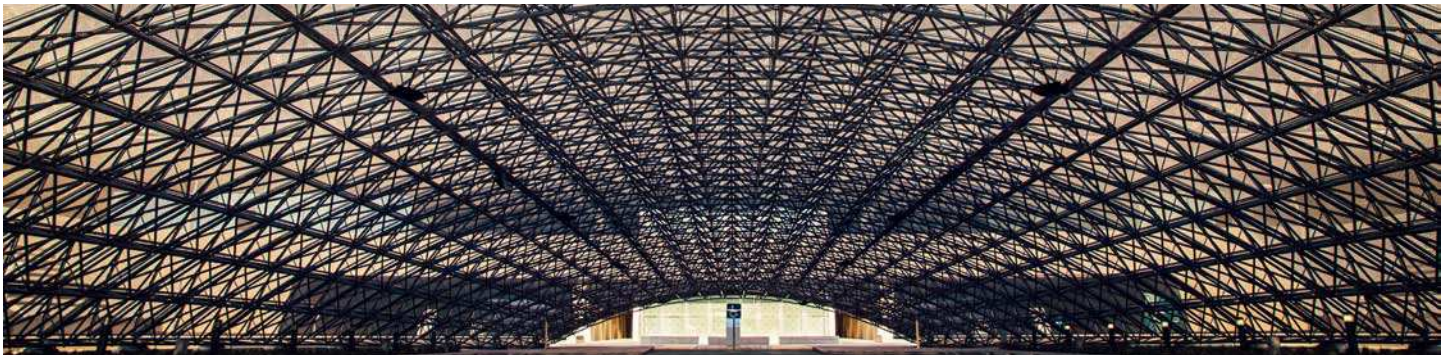
Figure 9: Hadoop As a Processing Engine



*SAP Sybase Adaptive Server® Enterprise
**Hadoop Distributed File System

The table provides a few of the potentially many scenarios that can leverage Hadoop as a processing engine.

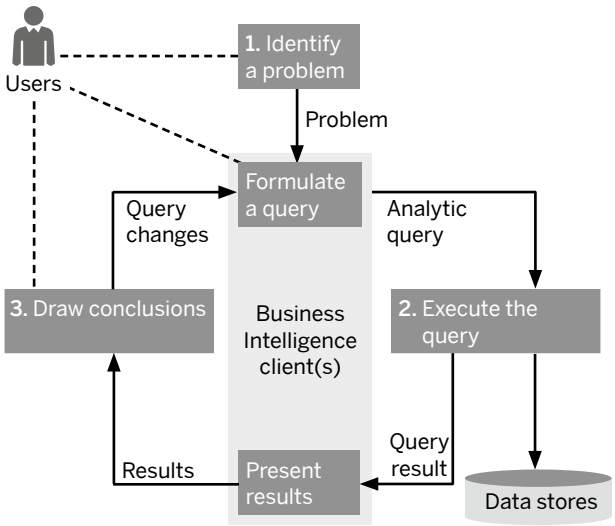
Scenario	Description	Sample Use Cases	Comment
Data enrichment	Analyze data in Hadoop for problems; enhance data with additional information.	Fix problems and inconsistencies in raw data in Hadoop; add demographic or other data to, for example, customer Web logs.	Perform this to fix problems and enhance data prior to using Hadoop as a simple database or processing engine.
Data pattern analysis	Analyze data in Hadoop to look for patterns (examples follow).	See different types of data pattern analysis below.	See examples below.
Data mining	Look for patterns, data clusters, and correlations in data in Hadoop.	Correlate customer behavior across diverse systems. Analyze machine data to predict cause of failures.	Use of computing engines such as Mahout (see "Hadoop Ecosystem") running on top of Hadoop is required.
Risk analysis	Look for known patterns in data in Hadoop that suggest risky behavior.	Manage risk in banking products, for example, credit cards; identify rogue traders.	Data mining can be used to help identify risk patterns.
Identifying data differences	Identify differences between large but similar sets of data.	Identify differences in DNA samples.	Hadoop, using MapReduce, can be much faster than conventional approaches.



HADOOP FOR DATA ANALYTICS

Hadoop does not fundamentally change the type of data analytics that can be carried out. However the inclusion of Hadoop does change the way analytics solutions work. This could have an impact on the scenarios for which a business chooses to use Hadoop. This section explains how including Hadoop as part of business intelligence and data analytics solutions can change the way those solutions need to work. Let's start by setting a base line for the basic data analytics process.

Figure 10: The Data Analytics Process



In data analytics, the following approach is typically followed (see Figure 10):

1. A user identifies a problem that must be solved – this can be for any analytic purpose or for any of the scenarios identified earlier in this paper. The user (perhaps with the help of a business intelligence client) formulates an analytic query that will help provide the answer or, at least, provide insight into the answer.
2. The query is executed and the data analyzed, which creates a query result that can be presented to the user.
3. The user uses the results to draw conclusions and may, in turn, identify query changes to be made to the original query. The cycle repeats.

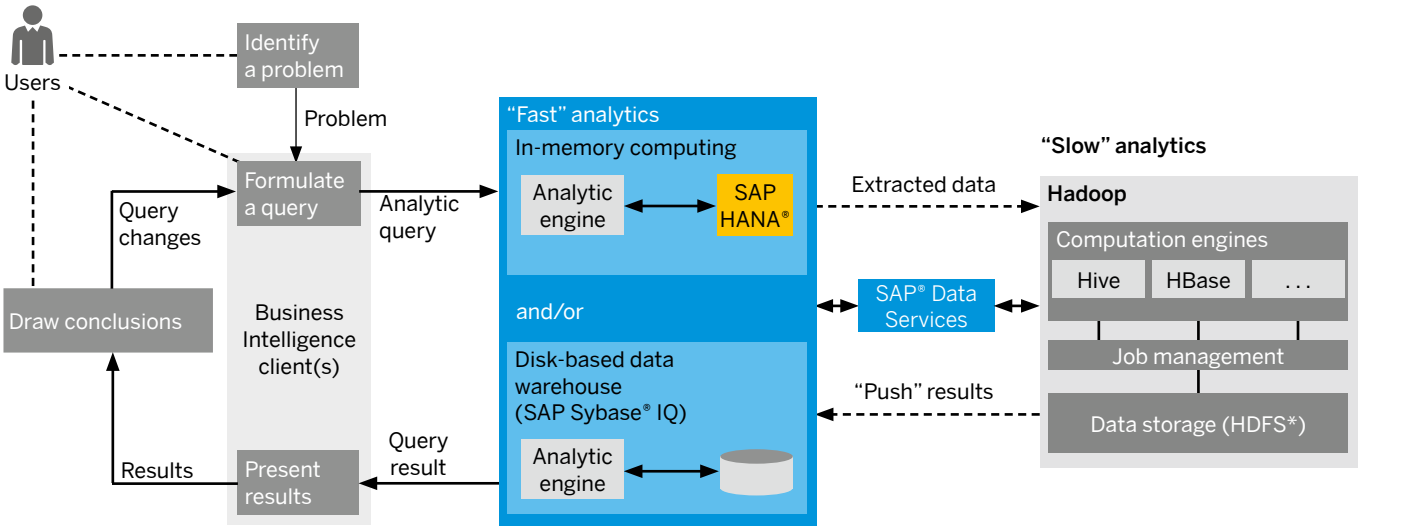
Hadoop does not really change this process, but does affect it in two ways.

Because the data volumes stored by Hadoop can be so vast, its raw data cannot be readily copied into an existing disk-based or in-memory data warehouse. It is too much data to economically store permanently, and moving data from Hadoop just in time for analysis would take too long. This means that some of the data analysis must be carried out in Hadoop itself. Also, queries executed in current versions of Hadoop usually take much longer to run, anywhere from minutes to hours. Unless managed, this will significantly slow down the "formulate query –> analyze data –> present results –> draw conclusions" cycle to the point where it can no longer be done in real time. It's also likely that many data analyses will require comparison and consolidation of results on data in multiple places, for example, with some data in Hadoop and some in a data warehouse or in an in-memory database such as SAP HANA. When these differences are put together, using Hadoop results in two fundamentally different approaches: two-phase analytics and federated queries.

Two-Phase Analytics

The need for two-phase analytics arises because of the long execution times that can occur when running queries on Hadoop. To address this, Hadoop is used to work continually in the background to analyze the data in Hadoop and then, periodically, push the results to a faster analytics solution, such as SAP HANA, so that it can be made available to the user. This is illustrated in Figure 11.

Figure 11: The Data Analytics Process – Two-Phase Analytics



*Hadoop Distributed File System

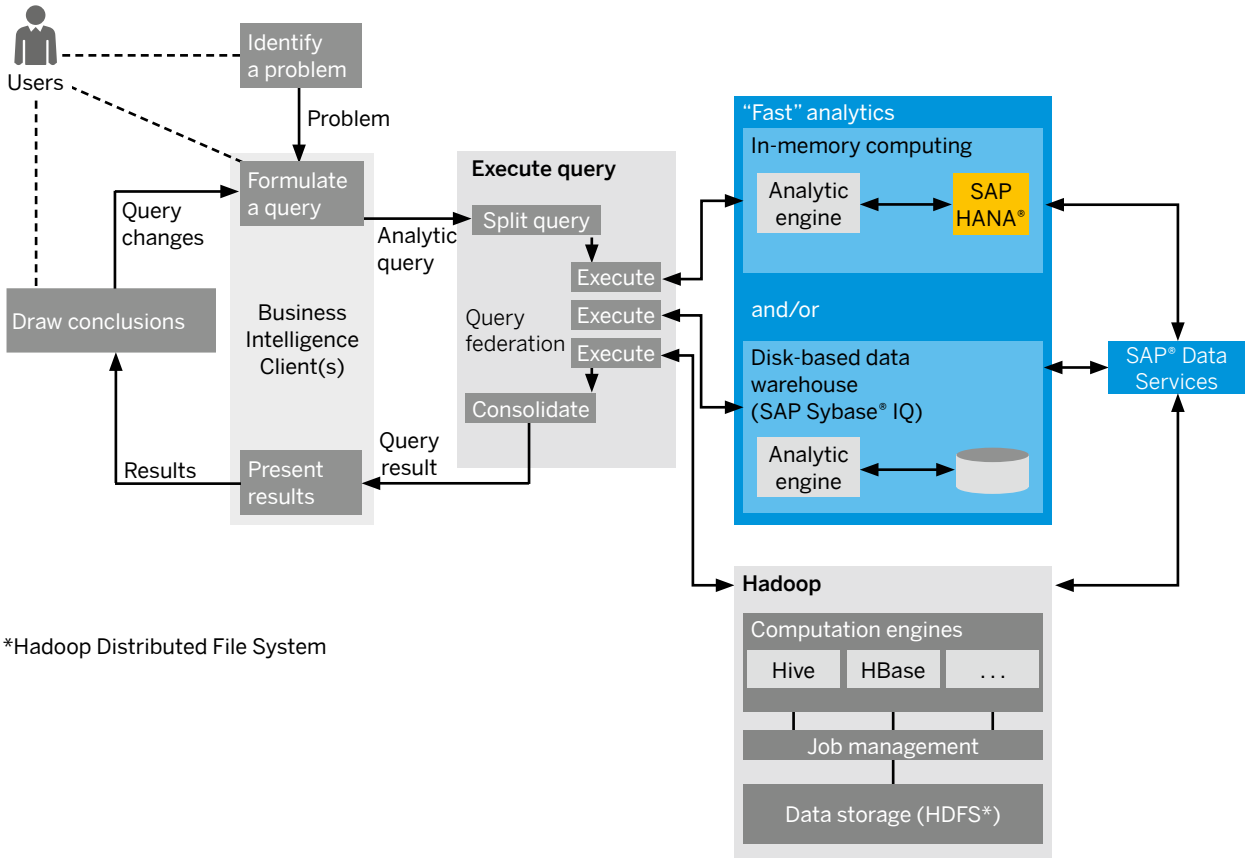
The analytics running on Hadoop could be, for example, data mining, risk analysis, or OLAP fact table creation. Using SAP Data Services, the results of the analytics could then be pushed to a fast analytics store, such as SAP HANA or SAP Sybase IQ, where they can be combined with data already there and made available to the user as a query result. This is essentially an extract, transform, and load (ETL) process.

The trigger for transferring the data can vary depending on the requirement. For example, if an OLAP fact table is being created, then the table (or updates to the table) could be sent every day, every hour, or every few minutes. On the other hand, if Hadoop is being used to identify something specific, for example, a "risky situation" as part of a risk analysis process, then the identification of such a situation could be used as a trigger for the transfer of data.

Federated Queries

Although two-phase analytics can provide faster responses, the types of analyses available are limited to the data and queries or jobs that were previously defined to run on Hadoop. Sometimes the user will want to run queries or analyses that are perhaps more "exotic" and that have never been asked before. These will require that a new job is run on Hadoop. The approach to solving this type of problem is to use federated queries as illustrated by Figure 12.

Figure 12: The Data Analytics Process – Federated Queries



*Hadoop Distributed File System

In a federated query, the original query is split into separate parts that can run on any combination of Hadoop, SAP HANA, SAP Sybase IQ, and other solutions. These are then executed separately.

Often, a query running on one data store, for example, Hadoop, will have to perform a database "join" with data in another, for example, SAP HANA. There are two choices: retrieve data from SAP HANA while the Hadoop query is running or copy the data in SAP HANA to Hadoop before the analysis query on Hadoop starts.

There may also be dependencies between the parts of the query since, for example, a job running on Hadoop may need to complete first so that the results it produces are available to complete a query in, for example, SAP HANA. Because part of a federated query can take a long time to run, it's not reasonable to expect a user to wait for the response to arrive. The federated query must have a way of executing queries asynchronously, where a query is accepted and control is passed back to the user so they can continue with other activities. Later, when the query is complete, the user is notified that the query has finished. The user can examine the result at that time.

The table describes a number of different ways federated queries can be implemented.

Scenario	Description	Sample Use Cases	Comment
Client-side federation	Business intelligence (BI) client executes queries separately and consolidates data from Hadoop and other data warehouses, for example, the SAP HANA® database.	Any analytic use case where the data being analyzed is held in Hadoop and other data stores	Only really practical when the volume of data returned by a Hadoop query is small, no database joins across data stores are required, and the query executes quickly
Stand-alone analytics	BI client analyzes data in Hadoop directly.	Any analytic use case where all the data being analyzed is contained in Hadoop	A subset of client-side federation; more value is often realized by analyzing data from multiple locations rather than just Hadoop
Query federation	The data warehouse server executes queries on multiple data sources, with results consolidated and combined afterwards.	Any analytic use case where the data being analyzed is held in Hadoop and other data stores	Better approach if data volumes returned by Hadoop are large; provides more flexibility and better performance by persisting or caching results
Data federation	Data in Hadoop is treated as an external table by another database management system.	Any analytic use case where the data being analyzed is held in Hadoop and other data stores	A good approach if the "external table" is not too large

Data Analytics and the SAP Real-Time Data Platform
Performing data analytics on data in multiple different data stores is not straightforward; many different parts need to be brought together and integrated. SAP Real-Time Data Platform is designed to facilitate much of this by including support for both query federation and data federation.

KEY SCENARIOS – FINAL WORDS

The scenarios and related use cases listed above are just examples of how a business can use Hadoop alongside SAP software, as well as non-SAP software. As mentioned earlier, the list and examples are by no means exhaustive. That said, there are common approaches to architecting and implementing Hadoop and the technology that surrounds it that will provide benefit primarily through more rapid development and lower marginal implementation costs. These are outlined in the next section, which provides a reference architecture for implementing Hadoop.

Unlike SAP HANA, Hadoop won't help you understand your business at "the speed of thought." But it lets you store and access more voluminous, detailed data at lower cost so you can **drill deeper and in different ways** into your business data.

Reference Architecture

This section provides a reference architecture showing how Hadoop can be used in an SAP software environment. It also suggests the SAP technologies that can be used with Hadoop. Figure 13 shows the architecture at a high level and in four main parts: data storage components, data sources, BI tools and analytics solutions, and applications. Note that the architecture is based on technology available as of early 2013 and is subject to change.

DATA STORAGE COMPONENTS

The data storage components – shown in the center of the reference architecture diagram – are used as follows:

- **Data warehouse/database** – To persist data in Hadoop as well as in databases and data warehouses, such as SAP HANA, SAP Sybase IQ, and the SAP NetWeaver® Business Warehouse (SAP NetWeaver BW) application
- **Data exchange** – To move data between databases and data warehouses using, for example, SAP Data Services
- **Data governance** – To manage and help improve the quality and veracity of data stored in Hadoop
- **Hadoop landscape and operational management** – To deploy and manage data in Hadoop

SAP Data Services

SAP Data Services software provides a versatile set of data integration tools that can:

- Access any type of data – structured, semi-structured, or unstructured
- Load it into any target – Hadoop, a data warehouse, or an in-memory database
- Navigate data sources that exist on premise or in the cloud
- Work in batch and real time
- Handle both small and large volumes of data

SAP Data Services features a design tool for modeling data and workflows. Using these tools, ETL developers can create and sequence the steps required to create ETL functions. They can extract, load, parse, integrate, cleanse, and match data in Hadoop. MapReduce code to do the work is generated automatically.

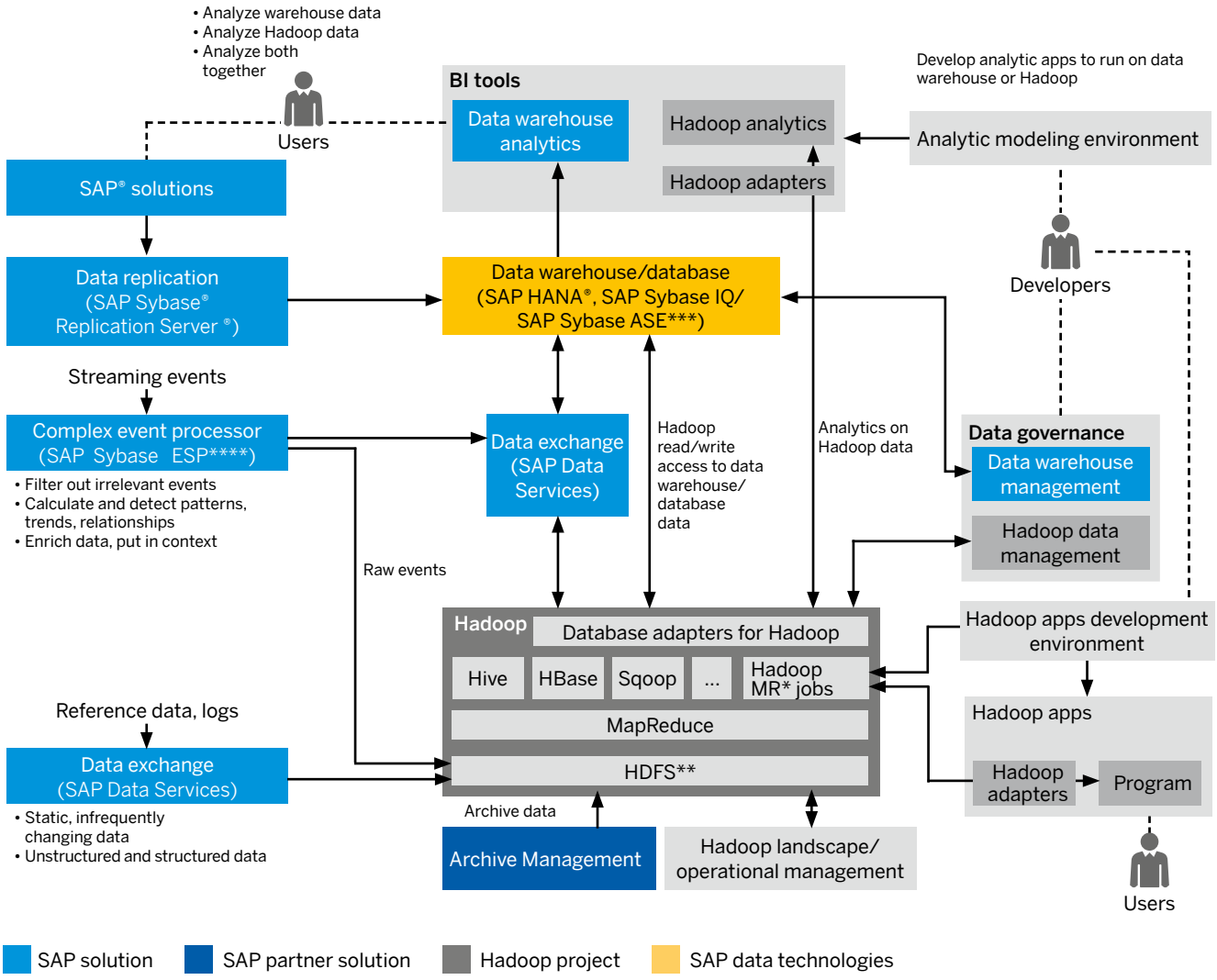
SAP Data Services integrates with Hadoop in three main ways:

- Hive tables – SAP Data Services can generate and execute HiveQL statements to query, retrieve, and load data into Hive tables.
- Hadoop Distributed File System (HDFS) – SAP Data Services can work on Linux operating system files natively or using Pig scripts.
- Text data processing transformations – Jobs containing an HDFS source and a text transform algorithm are pushed down by SAP Data Services into Hadoop via Pig scripts to carry out text analysis directly on files in HDFS as MapReduce jobs. This data could include analysis of Web logs, surveys, content fields, social media data, and geographic information system data.

Both Hive and Pig use MapReduce jobs to provide highly parallel processing.

SAP has technologies for integrating Hadoop into your SAP Software Landscape.

Figure 13: Reference Architecture



* MapReduce
** Hadoop Distributed File System
*** SAP Sybase Adaptive Server ® Enterprise
**** SAP Sybase Event Stream Processor

Data Governance

Understanding the veracity of data as well as improving its quality is important when making decisions. This applies as much to data stored in Hadoop as to data stored using more traditional database technologies. Given the volume of data often stored in Hadoop, the value of making quality improvements to the data has to be weighed against the costs of making them – see section "Hadoop Implementation Guidance." SAP Data Services can be used to improve and verify the quality of data in Hadoop. It has features to:

- Enhance data with additional data about industries, localities, and data domains, including customers, products, and materials
- Improve data quality using data cleansing, fuzzy matching, data merges, and survivorship

SAP Information Steward software provides business analysts, data stewards, and IT users with a single environment to discover, assess, define, monitor, and improve the quality of their enterprise data assets.

SAP Sybase IQ

SAP Sybase IQ supports federated queries (see section "Federated Queries") to access Hadoop. This support includes:

- Data federation, where the structure of files in HDFS are defined in SAP Sybase IQ and then treated as an external table within SAP Sybase IQ
- Query federation, where queries are executed separately on Hadoop using Hive and the result is then combined with other queries executed natively in SAP Sybase IQ

During data federation, the data is copied from Hadoop to SAP Sybase IQ and then analyzed there. For larger volumes, this takes longer than query federation, which returns the results of each separately executed query.

Landscape and Operational Management

The number of servers in a Hadoop landscape can run into the hundreds or even thousands. Like any other "server farm," these servers need to be managed from both a deployment and operational perspective. The current (early 2013) open-source distributions of Hadoop software do not provide many tools for this purpose. However, the proprietary distributions of Hadoop do. You should consider using these proprietary solutions.

DATA SOURCE COMPONENTS

Data source components are shown on the left and along the bottom of the reference architecture diagram. The data sources that can feed Hadoop are many. They include **SAP solutions, streaming events, reference data, and archive data**. Each of these is described below.

Data from SAP solutions, such as SAP Business Suite software, is best sent indirectly using data replication. This data should first be pushed to SAP HANA, SAP Sybase IQ, or a data warehouse using SAP Sybase Replication Server® software. The data can then be pushed to Hadoop using ETL processes and SAP Data Services.

Streaming events are voluminous and arrive rapidly. Processing them often requires a complex event processor, such as SAP Sybase Event Stream Processor (SAP Sybase ESP), to capture them, perform real-time analysis on them, and send the results to a database. The database, for example, SAP HANA, then uses the results for analytics and reporting.

Complex event processors usually keep the "raw events" for only a short period of time, because the amount of data they can store is limited. Hadoop helps by taking the output from SAP Sybase ESP and persisting some, if not all, raw events so that they can be used or analyzed later. This requires an ESP adapter, custom developed for the data source, to load the data into Hadoop. If real-time analysis of raw events is not necessary, the raw events can be sent directly to Hadoop for persistence using, for example, SAP Data Services.

Reference data is relatively static structured and unstructured data that is stored in Hadoop HDFS. The data can be loaded directly into Hadoop, but if worthwhile, SAP Data Services can be used to enhance the data and improve or assess its quality before the data is stored.

Archive data is data that would previously have been stored offline. In this reference architecture, it is stored in Hadoop HDFS using an archive management solution. This data can include e-mail correspondence, document content, and transaction logs.

BI TOOLS AND ANALYTICS SOLUTIONS

The business intelligence and data analytics components shown in the reference architecture diagram at the top and top right consist of two main areas: an **analytic modeling environment** and the **BI tools** used at run time:

- The analytic modeling environment provides the opportunity to build data models that can be used by the BI tools to answer queries. A semantic layer has tools for developing models (called "universes") on data in SAP HANA, SAP Sybase IQ, and non-SAP data sources, including Hive on Hadoop.
- Universes can be used directly by nontechnical end users working with BI tools, such as SAP Crystal Reports® software, SAP BusinessObjects Web Intelligence® software, SAP BusinessObjects Dashboards software, and SAP BusinessObjects Explorer® software.

APPLICATIONS

Using Hadoop for building applications is shown on the right of the reference architecture diagram. At a high level, building a Hadoop application requires developing and deploying MapReduce jobs, then executing them on Hadoop. Frequently, the generated result is pushed to another database, for example, SAP HANA.

MapReduce jobs are written in Java, although other technologies can be used, including Hive, Pig, and Mahout (see "Hadoop Ecosystem" section). Application logic can be automatically generated using technology such as SAP Data Services, which can generate Pig scripts and use Hive and Sqoop to extract data from Hadoop (see "SAP Data Services" section).

REFERENCE ARCHITECTURE – FINAL WORDS

The reference architecture describes how to integrate Hadoop with SAP solutions, where Hadoop and each of the SAP technologies are treated as stand-alone. SAP Real-Time Data Platform, with SAP HANA at its core, can combine Hadoop with SAP Sybase IQ and other SAP technologies. It thereby provides a single platform for OLTP and analytics, with common administration, operational management, and lifecycle management support for structured, unstructured, and semistructured data. This single platform makes it much easier to perform analytics on multiple data stores.

The **SAP Real-Time Data Platform**, with SAP HANA at its core combines Hadoop with SAP Sybase IQ and other SAP technologies to provide a single platform for OLTP and analytics, with common administration, operational management, and lifecycle management support for structured, unstructured, and semistructured data.

Sample Use Cases

This section looks in more detail at how Hadoop and SAP HANA can be used together to solve real problems. It describes four use cases covering:

- Preventative maintenance of hardware
- Retail recommendations in real time
- Problem identification and management of a telecom operator network
- Petabyte-scale data warehouse migration

USE CASE: PREVENTATIVE MAINTENANCE OF HARDWARE

In this use case, a high-tech company that manufactures servers wants to become better at preventative maintenance. To do this, it wants earlier identification of problems that have arisen so that prompt action can be taken to prevent the same problem from occurring at other customer sites.

To achieve this, the company wants to combine data from multiple sources:

- **Call center data**, which contains details of the problems recorded by call center staff when responding to calls from customers who are having problems with their servers
- **Questionnaires** completed by the customer that provide feedback on the company's products' features and use
- **Customer relationship management (CRM) data** about the customer to capture the history of purchases and interactions with the customer
- **Hardware monitoring (system) logs**, which were generated by the company's products in the field and transmitted to the company for analysis and which include real-time status information as well as details of errors and problems that have occurred
- **Bill of material data** about products to identify individual components of a product as well as uses of a component in other products
- **Production and manufacturing data** for the products to be used to track any problems to specific components or production runs

The call center data can be voluminous and includes text provided by both the customer and the call center staff, alongside structured information such as customer and model numbers. Similarly the questionnaires include text explanations and comments as well as ratings on product features. The company knows that there is valuable information about problems with the company's servers that can be extracted from this data and used to provide better preventative maintenance. However, analyzing the data using current, mainly manual methods takes too long.

Use Case Implementation

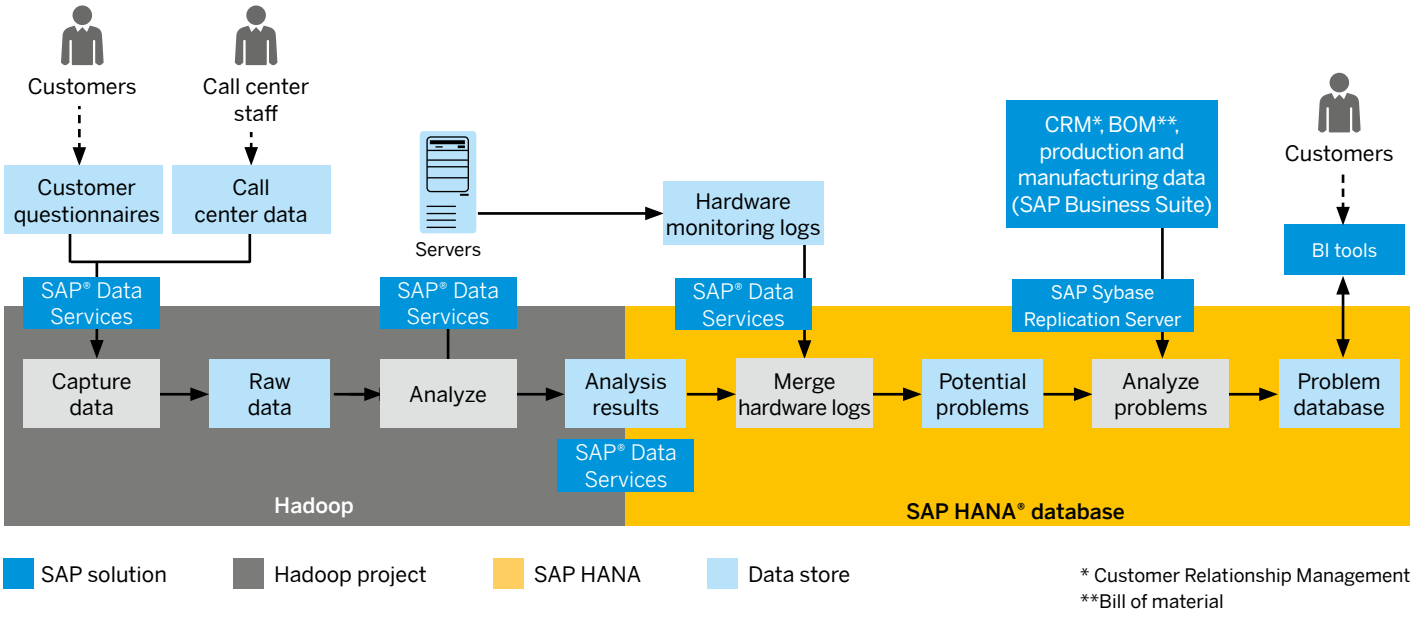
The company implements a solution in two parts.

1. Use Hadoop to analyze call center data and questionnaires and identify patterns that indicate that a particular product, component, or customer is having an unusual number of problems. This uses the text analysis API in SAP Data Services software to analyze the text.
2. Use SAP HANA to:
   a. Take the results of the analysis of call center and questionnaire data from Hadoop
   b. Merge it with the hardware monitoring logs for those customers
   c. Match it to individual products installed at customer sites
   d. Look for patterns in the merged data that indicate an unusual number of problems occurring
   e. Combine the results with CRM, bill of material, and production and manufacturing data for reporting purposes

Figure 14 illustrates the implementation and indicates where SAP software components are used.

Note how SAP Data Services and SAP Sybase Replication Server are used to integrate the various parts of the solution.

Figure 14: Implementation of Preventative Maintenance of Hardware



USE CASE: RETAIL RECOMMENDATIONS IN REAL TIME

In this use case, a retailer wants to make purchasing recommendations in real time to customers who are members of its loyalty program as they are browsing its e-commerce Web site. The retailer has both "brick and mortar" stores as well as a major Web presence.

To make real-time recommendations, the retailer wants to combine information from multiple sources:

- **Social media data** – After getting customer permission, the information in the customer's social media accounts, such as Facebook and Twitter, would be analyzed.
- **Point-of-sale data** – This is the data captured when the customer makes purchases either in-store or on the company's e-commerce site.
- **Historical Web log information** – This is a record of the customer's past browsing behavior on the company's Web site.
- **Inventory and stock Information** – This data reflects what products are in stock at which locations and which are on promotion.
- **CRM data** – This data is a record of all the interactions the customer has had with the company through the company's support site.
- **Real-time Web activity** – This data records the user's browsing activity on the company's e-commerce site in real time. It is during this activity that immediate product recommendations need to be made.

The basic idea of the solution is to analyze the first three data sources to understand the customer's likes, dislikes, and previous buying behavior. That information will then be combined with the inventory and stock information, the CRM data, and information on what the customer is doing in real time on the e-commerce Web site. Based on all this, immediate recommendations will be made for products the customer may be interested in purchasing as well as local stores where those products are in stock.

Use Case Implementation

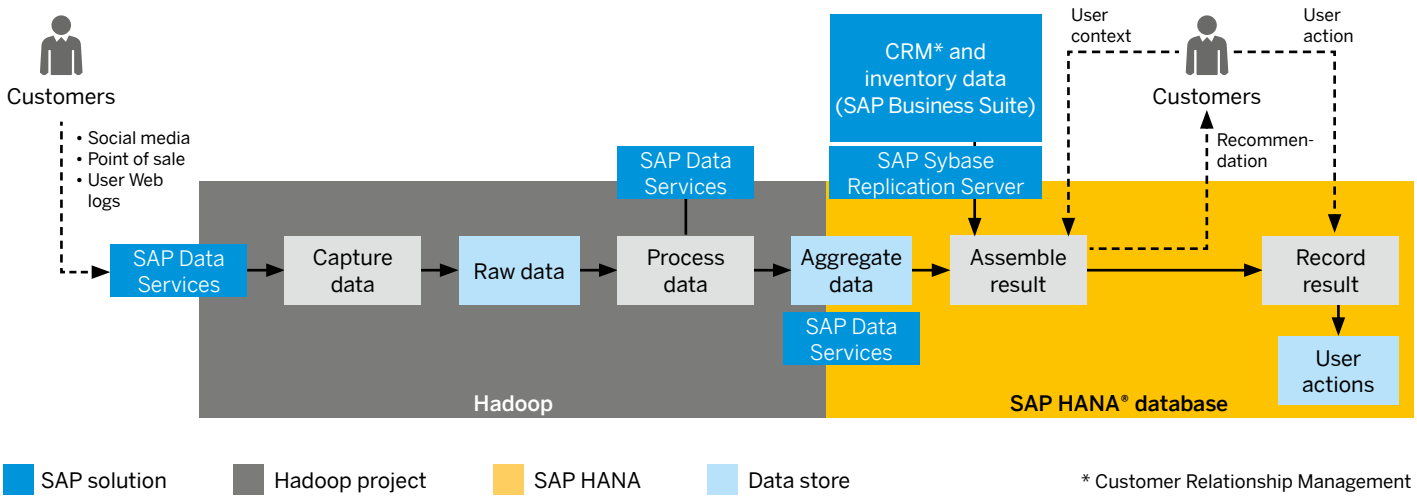
The company implements the solution in two parts.

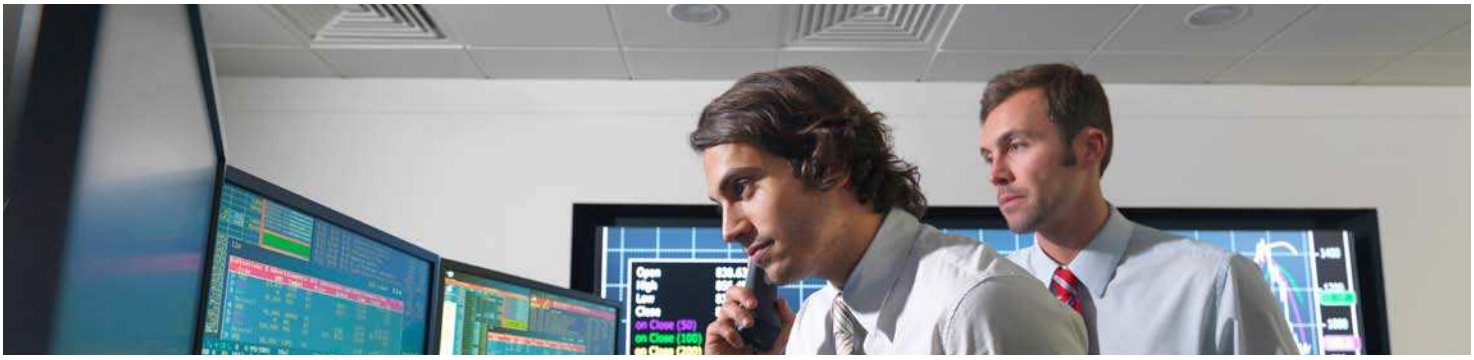
1. Use Hadoop to capture and analyze social media data, point-of-sale data, and historical Web log information to aggregate data about the customer's demographics, buying preferences, and past purchasing behavior. This uses the text analysis API in SAP Data Services for analyzing social media data.
2. Use SAP HANA to combine the results from Hadoop with the customer's real-time Web activity and inventory and stock information to make immediate recommendations on what the customer might like to buy.

Figure 15 illustrates the implementation and indicates where SAP components are used.

Again note the use of SAP Data Services and SAP Sybase Replication Server for data integration.

Figure 15: Implementation of Real-Time Retail Recommendations





USE CASE: PROBLEM IDENTIFICATION OF TELECOM OPERATOR NETWORK

In this use case, a telecom operator monitors its network to capture issues as they arise, for example, network outages or changed network loads that need to be managed. Predicting outages or workload changes in advance offers advantages in terms of lower operational costs and better service levels. However, new network issues and workload changes regularly arise that were not predicted, so improving the quality of predictions and the speed of reaction to changes in bandwidth needs is an ongoing task.

To address this problem the telecom operator needs to:

- **Analyze historical network data** to:
  - Improve the quality of predictions – for example, by searching for as yet undiscovered patterns in the network data that are found to correlate to problems that were not identified in advance
  - Answer ad hoc queries and produce new reports that can be used in, for example, network development planning
- **Analyze sensor and other data** on the network in real time to identify patterns that are known indicators or predictors of problems
- **Report problems to staff** as they are identified, so personnel can take corrective action with minimal latency
- **Provide near real-time service and other reports** to management, its customers, and to company carriers

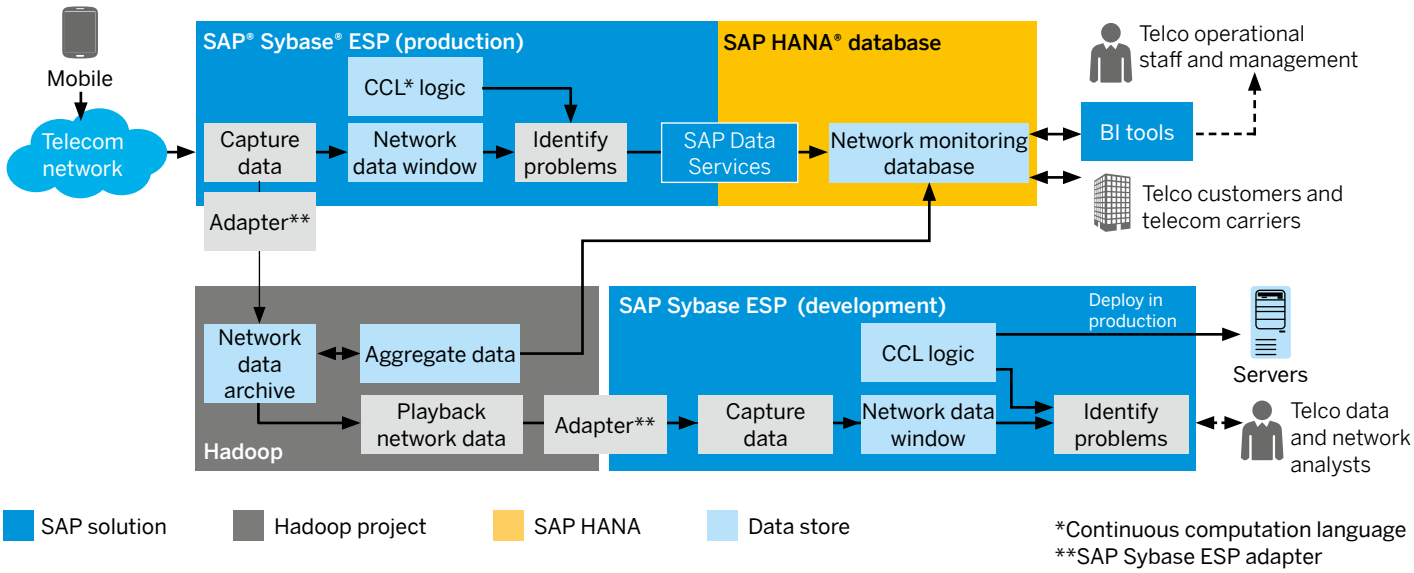
Use Case Implementation

The solution is implemented in three main parts:

- Use SAP Sybase ESP to process network data in real time to identify problems and any patterns in the data that indicate problems are likely to occur.
- Use SAP HANA to record problems identified by SAP Sybase ESP and maintain aggregate statistics and other data that can be used for reporting and analyzing patterns of interest quickly.
- Use Hadoop to keep a historical record of network data captured by SAP Sybase ESP to be used for ad hoc reports, long-term network planning, and improvement of the quality of predictions.

Figure 16 illustrates the implementation and indicates where SAP components are used.

Figure 16: Implementation of Problem Identification in Telecom Operator Network



The top half of the diagram shows how SAP Sybase ESP is used to capture data from the telecom operator network in real time. SAP Sybase ESP then aggregates, processes, and filters the data using an SQL-like code written in continuous computation language (CCL) and sends the results to SAP HANA. In SAP HANA, the results are used to monitor and report on network performance and help handle any problems that are identified.

The bottom half of the diagram shows how Hadoop is used to capture raw network data that has been streamed to it by SAP Sybase ESP using a custom-developed adapter. The data is then used in two ways:

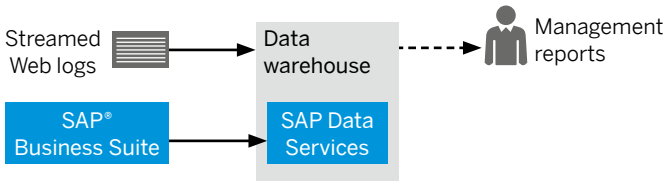
- Historical data aggregation and analysis for use in new or ad hoc reports developed for SAP HANA. Without storing the raw data in Hadoop, this would not be possible.
- Better network problem prediction that enables data and network analysts to improve prediction of problems by examining historical network data in Hadoop, which is then used to help develop new "patterns of interest" written in CCL. The effectiveness of the new patterns in indicating issues can be tested by playing back network data, stored in Hadoop, into SAP Sybase ESP.

Note that playing back data from Hadoop into SAP Sybase ESP can occur much faster than in real time, allowing data and network analysts to rapidly evaluate alternative patterns written in CCL.

USE CASE: PETABYTE-SCALE DATA WAREHOUSE MIGRATION

A high-tech company implemented a multipetabyte data warehouse using non-SAP software. The data warehouse captures data from multiple sources. Besides data from SAP Business Suite, captured data includes click stream data and data from the Web properties, which are the main ways the company interacts with its customers. Most of the workload on the current data warehouse consists of analyzing the raw streams of Web log data, for which the data warehouse is not ideally suited.

Figure 17: Current Implementation



The company now wants to migrate away from the data warehouse – without disrupting the feeds of management information the existing data warehouse produces – to a combination of Hadoop, which would hold the raw streaming data and other data, and SAP HANA, for producing new and replacement management information reports.

Use Case Implementation

The transition to SAP HANA is implemented in a phased approach:

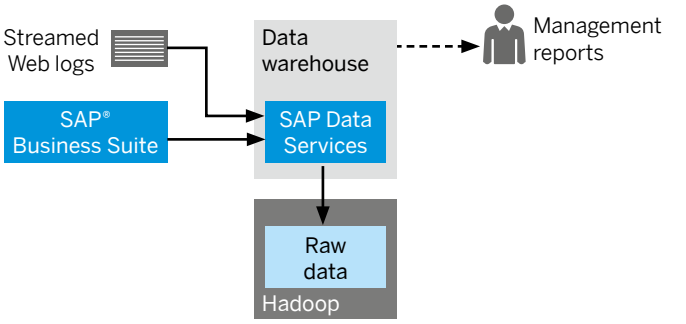
1. Replicate data in Hadoop
2. Aggregate data in Hadoop
3. Copy aggregate data to SAP HANA
4. Replace reporting by SAP HANA

Each of these is described in the following text.

1. Replicate Data in Hadoop

The first step is to replicate data in Hadoop. This is done by adapting SAP Data Services, which is already used to push data from SAP Business Suite to the existing data warehouse, to capture the streamed Web logs as well and then push both into Hadoop (see Figure 18).

Figure 18: Replicate Data in Hadoop

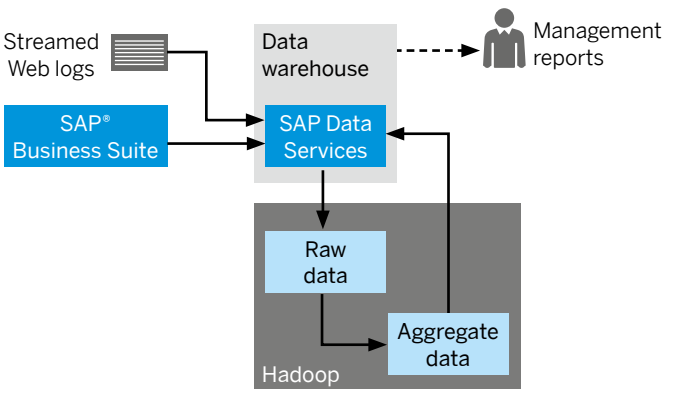


2. Aggregate Data in Hadoop

The second step is to carry out in Hadoop the data aggregation that was previously done in the data warehouse, then push the results back into the data warehouse using SAP Data Services so that reporting from the existing data warehouse can continue (see Figure 19).

This approach has the beneficial result of significantly reducing the workload on the existing data warehouse, enabling it to provide results faster.

Figure 19: Aggregate Data in Hadoop



SAP solution Hadoop project Data store

3. Copy Aggregate Data to SAP HANA

The third step executes in two parts:

- Copy the aggregated data that was being pushed from Hadoop into the existing data warehouse into SAP HANA as well
- Copy historical aggregated data in the existing data warehouse (that was created before the migration had started) into SAP HANA

Both parts of this step use SAP Data Services. The end result is that, effectively, SAP HANA contains a complete copy of the data in the existing data warehouse, with both copies kept up-to-date by Hadoop (see Figure 20).

4. Replace Reporting by SAP HANA

The fourth and final step is to implement reporting functionality in SAP HANA using SAP BusinessObjects BI solutions (see Figure 21). The objective is to replace and improve the reporting performed by the existing data warehouse. This will enable the existing data warehouse to be decommissioned once the new reporting functionality in SAP HANA is in productive operation.

Note that with this approach, the raw data in Hadoop is also available for analysis.

Figure 20: Copy Aggregate Data to SAP HANA

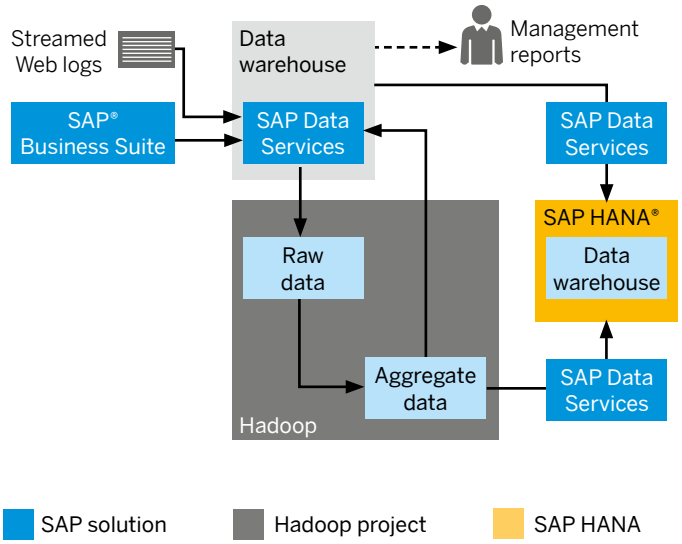
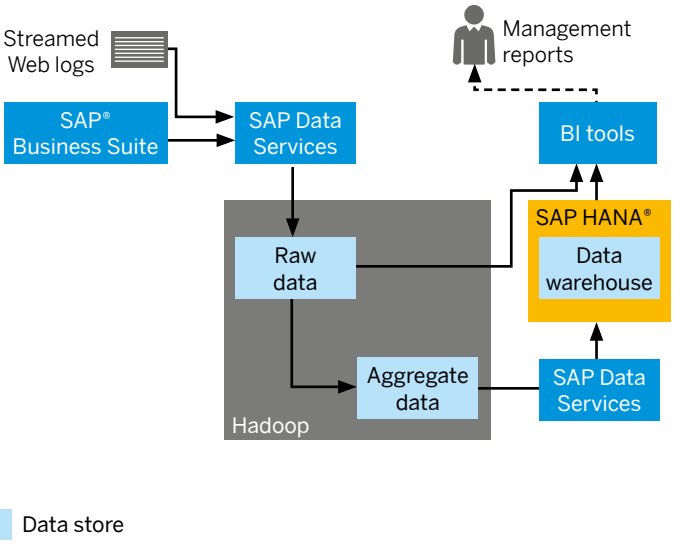


Figure 21: Replace Reporting by SAP HANA



Hadoop Implementation Guidance

This paper explains the main features of Hadoop and demonstrates ways in which Hadoop can deliver business value in key scenarios and sample use cases. The reference architecture provides an overview of how Hadoop can be integrated into an SAP software landscape in a general way. But the last piece of the jigsaw puzzle is Hadoop implementation, where generalities become specific. This is covered by the present section, which provides guidance on:

- General principles – Good ideas and principles to bear in mind during a Hadoop implementation
- Hadoop implementation – Steps to follow when planning a Hadoop implementation

GENERAL PRINCIPLES

General principles are primarily just good ideas to follow when implementing Hadoop. They should be factored into any Hadoop implementation you carry out.

- **Consider all the options** – Although this paper focuses on Hadoop as a solution for handling Big Data, other software solutions may offer a better balance of features, such as better development tools, security, or availability. So consider all the options. Don't assume that Hadoop is the only or best answer.
- **Develop a plan and strategy for how to leverage Hadoop** – Short-term, small-scale projects can help you understand the value Hadoop can bring, but a longer-term plan for Hadoop is needed to realize its full benefits. Ideally, the plan should be just another part of your IT strategy and business plan. It should consider the role SAP Real-Time Data Platform could play.
- **Persist everything** – The relatively low cost of data storage offered by Hadoop permits consideration of a strategy where raw data is never thrown away. As a result, you can carry out unanticipated data analyses at a later date, even though the value of the data persisted was not clear at the outset.
- **Make Hadoop an integral part of data management and governance** – Just because Hadoop can be used to persist anything and everything, it doesn't mean the quality of the data being kept in Hadoop should be ignored. Decisions are likely to be based, in part, on data in Hadoop, therefore understanding and improving its quality is important. This means that data in Hadoop should be included in any data quality and data governance processes that are used by the business. Again, SAP Real-Time Data Platform should be able to help.
- **Perfection can be the enemy of the good** – Although data quality is important and should not be ignored, getting error-free data can be expensive, if not impossible, especially given the quantity of data that can be stored in Hadoop. In practice, a balance must be struck. Quality improvements should be made to data in Hadoop where it is practical and worthwhile. Where it is not, the lower quality of the data should be communicated to users so they can factor it in to decisions they make based on that data.
- **Minimize data transfer** – Hadoop can store massive volumes of data. Moving this data around, especially over networks, is time-consuming and resource-intensive. For these reasons, you should design a Hadoop implementation that moves data only when necessary. For example, send only summary results from Hadoop to a data processing platform such as SAP HANA. Data stored elsewhere, for example, in SAP Business Suite, should be replicated just once in Hadoop in advance of its use to minimize repetitive data accesses to SAP Business Suite when Hadoop jobs are running. Again, SAP Real-Time Data Platform can help as it provides one place for managing data, whether the data is stored in SAP HANA, SAP Sybase IQ, or Hadoop.
- **Leverage Hadoop's "store now, understand later" approach** – Hadoop applies a schema when data is read, for example, by a MapReduce job. This is unlike a conventional RDBMS, which forces data to fit into a predefined structure when it is stored. Hadoop's approach provides great flexibility in how the data in Hadoop can be used. To fully benefit from this, data in Hadoop should be kept in its original raw form wherever practical, so that the original information is always available for new analyses.
- **Choose the "right" Hadoop data storage architecture** – Although Hadoop can store data in its raw form, it is still a good idea to consider the alternatives that Hadoop offers, such as sequence files or Avro, which offer performance or space-saving benefits. You can then choose the most appropriate one for the problem to be solved. Discussion of these storage architectures lies beyond the scope of this paper.
- **Invest in BI tools that leverage Hadoop effectively** – Writing MapReduce programs to run on Hadoop is not easy to do. It requires experts who understand how to write them and have knowledge of the programming languages that can be used on Hadoop. Writing queries that cross multiple data stores, not just Hadoop, is harder. To address this, BI tools should be used to enable power users, and ideally end users, to develop applications. SAP Real-Time Data Platform is designed to make this easier to do.

IMPLEMENTING HADOOP

Once a business decides to implement Hadoop, it must then decide where to start. This section suggests a good approach to follow, in three phases:

- Planning for Hadoop
- The initial Hadoop project
- Hadoop evolution

Planning for Hadoop

Like any project, it is worthwhile spending time planning what to do. Consider the following:

- **Is Hadoop the best solution for the problem?** Earlier sections of this paper have described the importance of using the best balance of data technologies to help solve a business problem. So the first step is to decide whether Hadoop is the right technology to use.
- **Understand how your industry uses Hadoop.** Be aware of how other businesses in your industry use Hadoop to gain business value. Talk to Hadoop experts at SAP to learn what might be possible. SAP offers Big Data bundles, which provide predefined industry-specific scenarios and implementations for Hadoop.
- **Don't simply copy others' ideas.** Don't aim for a "me-too" Hadoop implementation. Although examining what others in your industry are doing can be a good source of ideas, every business is different. An implementation that just copies others may not provide the best value.
- **Get cross-functional input.** A good way to avoid a "me-too" implementation is to assemble a team from different parts of your business for a brainstorming session on just how Hadoop could be used. This cross-functional team is more likely to identify valuable new ways of analyzing combinations of disparate data that Hadoop can handle. Consider transactions, e-mail correspondence, Web logs, phone calls, supply chain information, and public data on the Web. For added benefit, include an SAP expert who understands how to leverage Hadoop from a business perspective.
- **Define the initial Hadoop project** – Based on the analysis of data technologies and the cross-functional brainstorming session, identify and define the scope of the first Hadoop project. Develop estimates of the business benefits expected once implementation is complete.
- **Further Hadoop projects** – Don't focus solely on the first project. Spend time identifying and documenting additional uses for Hadoop that could be implemented in the future.

The Initial Hadoop Project

Once the initial Hadoop project is identified, the next step is implementation. Factors to consider include:

- **Hadoop data sources and sizing.** Think about:
  - **Data sources.** Identify the data sources that need to be kept in Hadoop. See "Hadoop as a Flexible Data Store" section for examples.
  - **Data volumes.** Estimate data volumes for those data sources, including how long the data will be kept. As mentioned earlier, the cost of storage is reducing so fast that persisting data "forever" may be a valid option.
  - **Data quality.** How will data in Hadoop be used? Is the quality of the data sufficient? Does it need to be processed or analyzed before it can be used, for example, to map the way customers are identified in different data sources? How could or should SAP Data Services be used to assess and improve the quality of data in Hadoop? A balance needs to be struck between the cost of improving data quality and the benefits improved data quality will bring.
  - **Data enhancements.** Does the data in Hadoop need to be enhanced, for example, by using SAP Data Services to access data in SAP Business Suite? Is there any benefit, for example, in enhancing product codes with additional data about the product, such as product categories?
  - **Storage requirements.** Use the data volumes to estimate the amount of storage required to keep the data in the Hadoop DataNodes.
  - **Workload.** Identify the major types of queries, analytics, and processes that will be run on Hadoop both for the initial Hadoop project as well as later. Will these workloads have peaks and troughs? Are they seasonal?
  - **Security,** system, and operational management. Is there any security or data privacy legislation that will influence how or where Hadoop can be deployed? Are there any needs specific to managing or operating the system?
  - **Service levels.** What service levels will the Hadoop implementation require in terms of response times, system availability, and redundancy?

- **The Hadoop system design.** Assuming that the decision has been made that Hadoop is the best solution for all or part of the problem being solved, decide what technology is needed to implement the project. Consider:
  - **Hadoop software.** SAP has partnerships with Intel Corp., Cloudera Inc., and Hortonworks Inc.
  - **Hardware for running Hadoop.** SAP has partnerships with IBM and HP.
  - **Analytic database and BI tools.** Identify the SAP solutions that can be used to present the results of analyses run on Hadoop. Refer to the "Reference Architecture" section as a guide.
  - **Data source integration.** Identify what types of data will be captured in Hadoop (see section on "Hadoop As a Flexible Data Store") and the technology that can be used to capture it, for example, such as SAP Data Services. Again, use section "Reference Architecture" as a guide.
  - **Design the Hadoop system.** Use all the above information to design the initial Hadoop system to be deployed and implemented alongside SAP solutions. As computer hardware is continually changing, refer to guidelines from software and hardware manufacturers that support Hadoop as well as SAP experts to help design and size the Hadoop system.
- **Initial Hadoop project implementation.** Once the Hadoop system has been designed, implementation can start.
  - **Define implementation phases.** Develop a plan to implement the project in phases. Include hardware and software acquisition and installation, software development, initial implementation, and running the operational software. It is usually better to start with small volumes and limited scope to prove the solution works, then increase the scope and scale later.
  - **Resource the project and execute.** Procure the technology and people required to implement, then execute. Normal good project management practices should apply.
  - **Learn lessons.** At each stage of the initial project, look for lessons to be learned about implementing Hadoop.

Hadoop Evolution

Many lessons will be learned from the initial Hadoop project implementation. However, the initial implementation is just the start. Steps need to be put in place for Hadoop's ongoing support. For example:

- **Check that identified business benefits are being realized.** Where benefits are not being realized, identify any actions that need to be taken to rectify this.
- **Check operational effectiveness.** Check that Hadoop is performing and operating technically as expected and that service levels are being met. If not, take corrective action.
- **Perform ongoing reviews.** Reconvene the cross-functional team periodically to get its feedback on how well Hadoop is working from a business perspective. The review should include data sources, data volumes, data quality, and data enhancements.
- **New uses for Hadoop.** Periodically use the same cross-functional team to identify new data sources and new uses to which Hadoop could be applied.
- **Monitor Hadoop and related technology developments.** Hadoop is a new technology that is rapidly evolving. Understand and monitor how Hadoop and these new technologies will be used by SAP Real-Time Data Platform.

Future Trends

Hadoop is new technology that is still evolving. Many start-ups and open-source initiatives that form part of the Hadoop ecosystem (see the section with this name) are developing new ways and new technologies that complement the core Hadoop technologies and provide Hadoop with new features. This section provides insight into the current status of these developments as of early 2013.

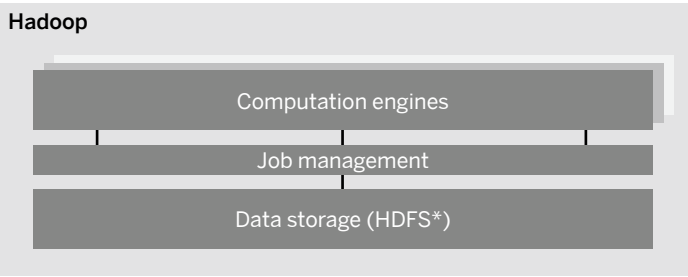
Please note that the mention of any product in this section does not imply any endorsement or recommendation by SAP as to its suitability for use. Nor does it imply that SAP will provide support for it in its solutions. The goal of this section is to simply inform readers on how Hadoop is evolving so that they can better understand the need to continue monitoring it.

At a high level, Hadoop is evolving in three main ways:
- **New computation engines.** These represent new and different ways of processing data in Hadoop.
- **Hadoop optimizations.** These are ways in which Hadoop can be made to perform more quickly using fewer resources.
- **Management tools.** These are tools and especially technologies that make Hadoop implementations, which can consist of hundreds or thousands of servers, easier to manage.

NEW COMPUTATION ENGINES FOR HADOOP

Figure 22 shows the software architecture of Hadoop.

Figure 22: Hadoop Software Architecture



*Hadoop Distributed File System

This software architecture is one of the key features of Hadoop. It allows new computation engines to be built that leverage the raw data stored in the HDFS.

Some computation engines already exist, for example:
- Hive provides read-only access to data in Hadoop using a language that is very similar to SQL. Hive also provides a Java Database Connectivity (JDBC) interface, which allows it to be treated as an external database.
- HBase turns Hadoop into a fast, massively scalable and available data store for random read/write access to any data associated with a key. This is often used by Web sites, for example, to store large images, videos, or other documents for rapid retrieval and display.

Other computation engines are being developed, such as:
- Impala and Hadapt combine disk-based relational databases and Hadoop data on a Hadoop DataNode.
- Giraph turns Hadoop into a "graph store." Graph stores are a highly flexible way of storing relationships between things. They can be used to connect data of different types: customer data held in customer relationship management software, data on the Web, social data (products they bought), call center contacts, and service contacts.

An Apache project called Yarn is in development. It involves updating the job management and resource scheduling part of Hadoop so that it is no longer restricted to running just MapReduce jobs. This will allow data stored in Hadoop and HDFS to be processed in new and different ways. It will also make it easier to build new computation engines on top of Hadoop. Yarn is part of what will eventually become Hadoop 2.0, which is intended to make the HDFS file system more reliable.

HADOOP OPTIMIZATIONS

Hadoop's MapReduce approach, although very scalable, has a number of performance bottlenecks such as:

- **Use of Linux operating system jobs.** Each Hadoop MapReduce step is executed as a Linux job. This incurs significant overhead in starting and stopping those jobs. It does, though, allow Hadoop to leverage the native multi-programming and multithreading features of Linux to realize very high levels of parallelism.
- **Using disks for storing intermediate files.** The intermediate files output by a map job are stored on disk. They then need to be read again by a reduce job before the final result can be produced. Given the volumes of data stored and processed in Hadoop, these files can take a substantial amount of time not just to write and read, but also to transmit between DataNodes over a network.
- **Batch-only operation.** Hadoop usually works in two separate steps. The first step is to capture and store data in HDFS, the second is to analyze or process the data that has been stored. This makes immediate processing of very rapidly arriving or streaming data impractical.

There are initiatives under way to address all of these bottlenecks. For example:

- MapR is a Hadoop distribution with a faster, more reliable, enterprise-ready MapReduce implementation.
- Spark and Shark. Spark is a high-performance Hadoop HDFS–compatible MapReduce engine that stores intermediate results in memory. Shark is a high-performance implementation of Hive running on top of Spark.
- HStreaming is a real-time data analytics platform to analyze, process, store, and archive streaming data on Hadoop.

These are just a few examples. There are many other initiatives. The point is that it is important to recognize that Hadoop and its related technologies are continually and rapidly evolving.

MANAGEMENT TOOLS

Hadoop implementations often consist of hundreds and perhaps thousands of servers acting as DataNodes. Each DataNode needs to be managed – commissioned, monitored, repaired, and retired – just like the other systems and servers a business uses. The base open-source Hadoop solution provides few tools to do this. Generally there are two approaches to meeting this important need:

- **Hadoop distribution-specific management tools.** In this case, providers of open-source Hadoop distributions, such as Intel, Cloudera, or Hortonworks, provide additional software that can be used to manage Hadoop clusters. This additional software is available for purchase and is not open-source software.
- **General purpose management software.** Other software companies provide separate tools for managing Hadoop. Zettaset Inc., for example, provides tools to cover Hadoop administration, security, and analytics.

Final Words

You should now have an idea how Hadoop can be leveraged by businesses. You have seen how it can be introduced into a software landscape alongside SAP solutions, such as SAP Business Suite and SAP HANA. You also saw how it can be used to deliver business value in new and novel ways. Hadoop, though, is still evolving. New developments will continue to be made at a rapid pace, at least in the short term. Nevertheless, the potential benefits of Hadoop for an enterprise are real and available now. At the same time, leveraging the opportunities that Hadoop provides while the technology is still rapidly evolving can present challenges. This is where SAP can help. Our commitment is to support our customers and work with partners to deliver tangible business benefits.

First, SAP is integrating Hadoop alongside other SAP technologies, such as SAP HANA and SAP Sybase IQ, to create SAP Real-Time Data Platform. This will make Hadoop easier to manage and the data that crosses multiple platforms easier to analyze and manage.

Second, SAP will continue to work both on its own and with partners to identify and evaluate Hadoop developments. The goal is to make new Hadoop-related technologies available to SAP customers through SAP Real-Time Data Platform when they are technically mature and their business value is proven. One example is the partnership between SAP and Intel, which has the goal of helping organizations readily adopt an enterprise-class Big Data solution.[13]

FIND OUT MORE

To learn more about SAP's approach to implementing Hadoop, or to discuss with SAP experts how your enterprise can benefit from a Hadoop implementation, please contact us at SAP_Product_Architecture_Communications@sap.com.

SAP is committed to support its customers and work with partners to deliver the **tangible business benefits that Big Data can bring**.

13. See press release: www.sap.com/corporate-en/news.epx?PressID=20498

www.sap.com/contactsap

SAP®

The Best-Run Businesses Run SAP™