1

Hadoop Introduction

WHAT'S IN THIS CHAPTER?

- The components of Hadoop
- The roles of HDFS, MapReduce, YARN, ZooKeeper, and Hive
- Hadoop's integration with other systems
- Data integration and Hadoop

Hadoop is an essential tool for managing big data. This tool fills a rising need for businesses managing large data stores, or data lakes as Hadoop refers to them. The biggest need in business, when it comes to data, is the ability to scale. Technology and business are driving organizations to gather more and more data, which increases the need to manage it efficiently. This chapter examines the Hadoop Stack, as well as all of the associated components that can be used with Hadoop.

In building the Hadoop Stack, each component plays an important role in the platform. The stack starts with the essential requirements contained in the Hadoop Common, which is a collection of common utilities and libraries that support other Hadoop modules. Like any stack, these supportive files are a necessary requirement for a successful implementation. The well-known file system, the Hadoop Distributed File System or HDFS, is at the heart of Hadoop, but it won't threaten your budget. To narrow your perspective on a set of data, you can use the programming logic contained within MapReduce, which provides massive scalability across many servers in a Hadoop cluster. For resource management, you can consider adding Hadoop YARN, the distributed operating system for your big data apps, to your stack.

ZooKeeper, another Hadoop Stack component, enables distributed processes to coordinate with each other through a shared hierarchical name space of data registers, known as znodes. Every znode is identified by a path, with path elements separated by a slash (/).

There are other systems that can integrate with Hadoop and benefit from its infrastructure. Although Hadoop is not considered a Relational Database Management System (RDBMS), it can be used along with systems like Oracle, MySQL, and SQL Server. Each of these systems has developed connector-type components that are processed using Hadoop's framework. We will review a few of these components in this chapter and illustrate how they interact with Hadoop.

Business Analytics and Big Data

Business Analytics is the study of data through statistical and operational analysis. Hadoop allows you to conduct operational analysis on its data stores. These results allow organizations and companies to make better business decisions that are beneficial to the organization.

To understand this further, let's build a big data profile. Because of the amount of data involved, the data can be distributed across storage and compute nodes, which benefits from using Hadoop. Because it is distributed and not centralized, it lacks the characteristics of an RDBMS. This allows you to use large data stores and an assortment of data types with Hadoop.

For example, let's consider a large data store like Google, Bing, or Twitter. All of these data stores can grow exponentially based on activity, such as queries and a large user base. Hadoop's components can help you process these large data stores.

A business, such as Google, can use Hadoop to manipulate, manage, and produce meaningful results from their data stores. The traditional tools commonly used for Business Analytics are not designed to work with or analyze extremely large datasets, but Hadoop is a solution that fits these business models.

The Components of Hadoop

The Hadoop Common is the foundation of Hadoop, because it contains the primary services and basic processes, such as the abstraction of the underlying operating system and its filesystem. Hadoop Common also contains the necessary Java Archive (JAR) files and scripts required to start Hadoop. The Hadoop Common package even provides source code and documentation, as well as a contribution section. You can't run Hadoop without Hadoop Common.

As with any stack, there are requirements that Apache provides for configuring the Hadoop Common. Having a general understanding as a Linux or Unix administrator is helpful in setting this up. Hadoop Common, also referred to as the Hadoop Stack, is not designed for a beginner, so the pace of your implementation rests on your experience. In fact, Apache clearly states on their site that using Hadoop is not the task you want to tackle while trying to learn how to administer a Linux environment. It is recommended that you are comfortable in this environment before attempting to install Hadoop.

The Distributed File System (HDFS)

With Hadoop Common now installed, it is time to examine the rest of the Hadoop Stack. HDFS delivers a distributed filesystem that is designed to run on basic hardware components. Most businesses find these minimal system requirements appealing. This environment can be set up in a Virtual Machine (VM) or a laptop for the initial walkthrough and advancement to server deployment. It is highly fault-tolerant and is designed to be deployed on low-cost hardware. It provides high throughput access to application data and is suitable for applications having large datasets.

Hardware failures are unavoidable in any environment. With HDFS, your data can span across thousands of servers, with each server containing an essential piece of data. This is where the fault tolerance feature comes into play. The reality is that with this many servers there is always the risk that one or more may become nonfunctional. HDFS has the ability to detect faults and quickly perform an automatic recovery.

HDFS is optimally designed for batch processing, which provides a high throughput of data access, rather than a low latency of data access. Applications that run on HDFS have large datasets. A typical file in HDFS can be hundreds of gigabytes or more in size, and so HDFS of course supports large files. It provides high aggregate data bandwidth and scales to hundreds of nodes in a single cluster.

Hadoop is a single functional distributed system that works directly with clustered machines in order to read the dataset in parallel and provide a much higher throughput. Consider Hadoop as a power house single CPU running across clustered and low cost machines. Now that we've described the tools that read the data, the next step is to process it by using MapReduce.

What Is MapReduce?

MapReduce is a programming component of Hadoop used for processing and reading large data sets. The MapReduce algorithm gives Hadoop the ability to process data in parallel. In short, MapReduce is used to compress large amounts of data into meaningful results for statistical analysis. MapReduce can do batch job processing, which is the ability to read large amounts of data numerous times during processing to produce the requested results.

For businesses and organizations with large data stores or data lakes, this is an essential component in getting your data down to a manageable size to analyze or query.

The MapReduce workflow, as shown in Figure 1-1, works like a grandfather clock with a number of gears. Each gear performs a particular task before it moves on to the next. It shows the transitional states of data as it is chunked into smaller sizes for processing.

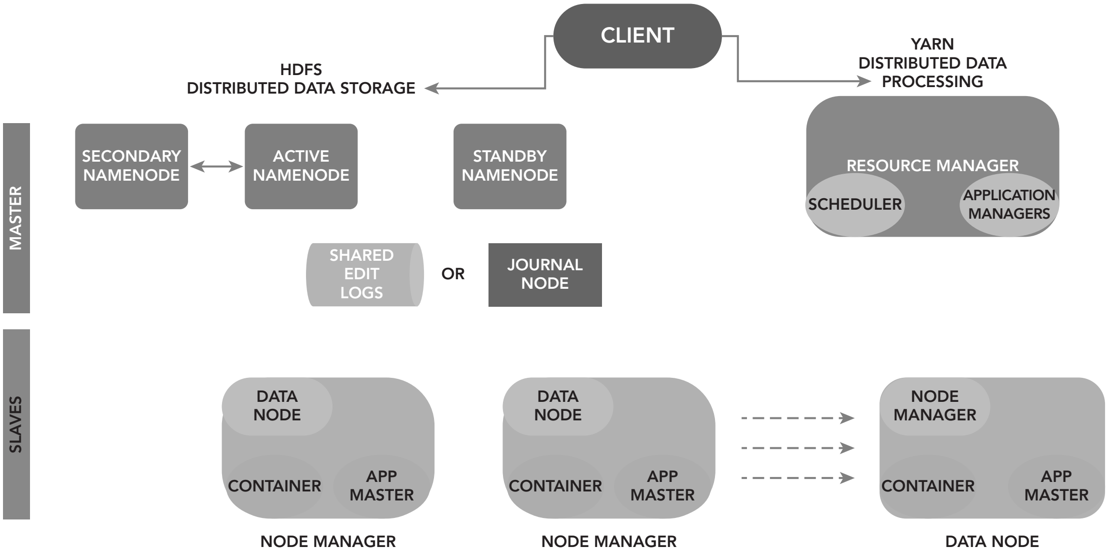


FIGURE 1-1

The capabilities of MapReduce make it one of the most used batch-processing tools. The flexibility of this processor opens the door to use its leverage against existing systems. MapReduce will allow its users to process unlimited amounts of data of any type that's stored in HDFS by dividing workloads into multiple tasks across servers that are run in parallel. MapReduce thus makes Hadoop a powerhouse tool.

With the recent developments in Hadoop, another component, called YARN, is now available that can be used to further leverage your Hadoop Ecosystem.

What Is YARN?

The YARN Infrastructure (Yet Another Resource Negotiator) is the framework responsible for providing the computational resources (memory, CPUs, etc.) needed for executing applications.

What features or characteristics are appealing about YARN? Two important ones are Resource Manager and Node Manager. Let's build the profile of YARN. First consider a two level cluster where Resource Manager is in the top tier (one per cluster). The Resource Manager is the master. It knows where the slaves are located (lower tier) and how many resources they have. It runs several services, and the most important is the Resource Scheduler, which decides how to assign the resources. The Node Manager (many per cluster) is the slave of the infrastructure. When it starts, it announces itself to the Resource Manager. The node has the ability to distribute resources to the cluster, and its resource capacity is the amount of memory and other resources. At run-time, the Resource Scheduler will decide how to use this capacity. The YARN framework in Hadoop 2 allows workloads to share cluster resources dynamically between a variety of processing frameworks, including MapReduce, Impala, and Spark. YARN currently handles memory and CPU and will coordinate additional resources like disk and network I/O in the future.

WHAT IS ZOOKEEPER?

ZooKeeper is another Hadoop service—a keeper of information in a distributed system environment. ZooKeeper's centralized management solution is used to maintain the configuration of a distributed system. Because ZooKeeper is maintaining the information, any new nodes joining will acquire the up-to-date centralized configuration from ZooKeeper as soon as they join the system. This also allows you to centrally change the state of your distributed system just by changing the centralized configuration through one of the ZooKeeper clients.

The Name service is a service that maps a name to some information associated with that name. It is similar to Active Directory being a name service that maps the user id (name) of a person to certain access or rights within an environment. In the same way, a DNS service is a name service that maps a domain name to an IP address. By using ZooKeeper in a distributed system you can keep track of which servers or services are up and running and look up their status by name.

If there is a problem with nodes going down, ZooKeeper has an automatic fail-over strategy via leader election as an off-the-shelf support solution (see Figure 1-2). Leader election is a service that can be installed on several machines for redundancy, but only one is active at any given moment. If the active service goes down for some reason, another service rises to do its work.

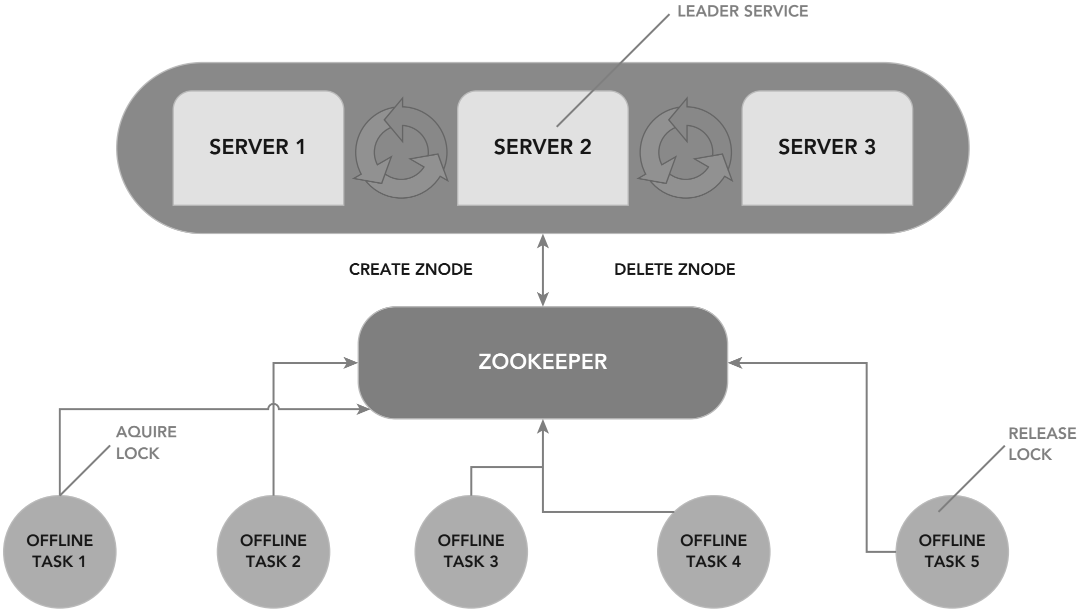


FIGURE 1-2

ZooKeeper allows you to process more data, more reliably and in less time. ZooKeeper can help you build more robust systems. A managed database cluster can benefit from centralized management services in terms of *name services*, *group services, leader election*, *configuration management*, and more. All of these coordination services can be managed with ZooKeeper.

WHAT IS HIVE?

Hive was originally designed to be a part of Hadoop, but now it is a standalone component. It is being mentioned briefly here, because some users find it beneficial to use it in addition to the standard Hadoop Stack.

We can briefly summarize Hive in this way: It is a data warehouse infrastructure built on top of Hadoop for providing data summarization, query, and analysis. If you are longing for the database experience and missing the structure (see Figure 1-3) of a relational environment when working with Hadoop, this might be your solution. Keep in mind this is not to be compared to a traditional database or data structure. Nor can it replace your existing RDBMS environment. Hive provides a conduit to project structure onto this data, and queries the data using a SQL-like language called HiveQL.

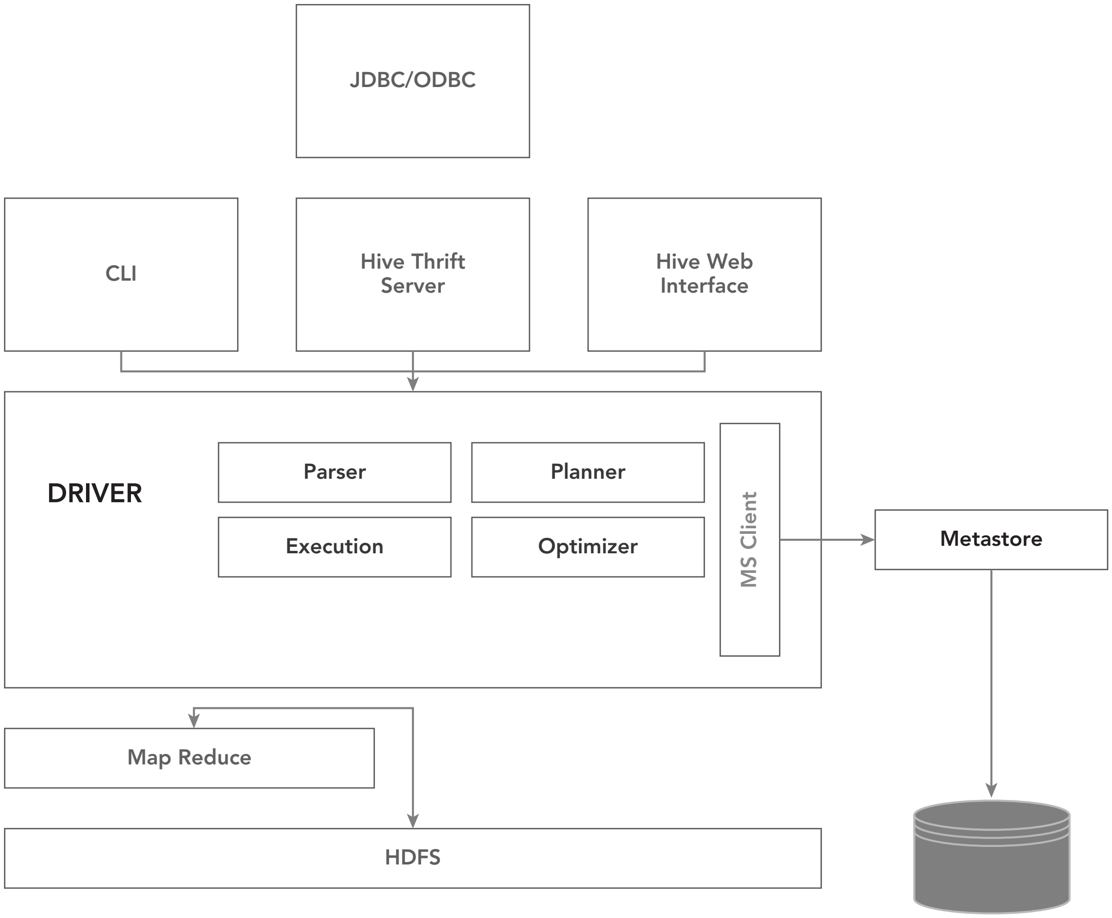


FIGURE 1-3

INTEGRATION WITH OTHER SYSTEMS

If you work in the technical field, you are well aware that integration is an essential part of any successful implementation. Generally, through some discovery process or planning session, organizations can pinpoint a need to manage big data more effectively. Subsequent steps involve making the determination as to how you will be implementing Hadoop into your existing environments.

Organizations implementing or considering Hadoop are likely introducing it into an existing environment. To gain the most benefit it is important to understand how Hadoop and your existing environment can work together, and what opportunities are available to leverage your existing environment.

To illustrate, consider a well-known building toy that allows you to create new toys based on connecting bricks together. There are endless possibilities of what you can create by simply connecting bricks together. The key component is the connector dots that exist on every brick. Similar to the toy bricks, vendors have developed connectors to allow other enterprise systems to connect to Hadoop. By using the connectors, you will be able to leverage your existing environments by bringing Hadoop into the fold.

Let's review some of the components that have been developed to integrate Hadoop with other systems. You should consider any leverage that you may gain by using these connectors within your environment. Clearly when it comes to integration, you must be your own SME (Subject Matter Expert) regarding the systems within your environment.

These connectors for Hadoop will likely be available for the latest release of the system within your environment. If the systems you would like to leverage with Hadoop are not on the latest release for your application or database engine, you need to factor in an upgrade in order to use the full features of this enhancement. To avoid disappointment, we recommend a complete review of your system requirements to avoid frustration and disappointment. The ecosystem of Hadoop brings everything together under one technical roof.

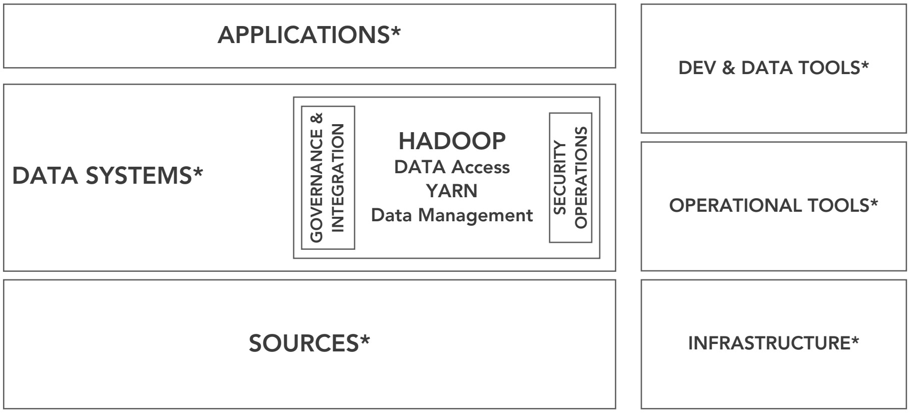
The Hadoop Ecosystem

Apache calls their integration an ecosystem. The dictionary defines an ecosystem as a community of living organisms in conjunction with the nonliving components of their environment (things like air, water, and mineral soil) interacting as a system. The technology-based ecosystem has similar attributes. It is a combination of product platforms defined by core components made by the platform owner and complemented by applications made by autonomous (machines that act independently from humans) companies in the periphery (surrounding a space).

Hadoop's open source and enterprise ecosystem continues to grow based on the wide variety of products available from Apache, and a large number of vendors providing solutions for integrating Hadoop with enterprise tools. HDFS is a primary component of the ecosystem. Because Hadoop has a low commodity cost, it is easy to explore the features of Hadoop either through a VM or setting up a hybrid ecosystem within your existing environment. It is an excellent way to review your current data methodologies with Hadoop solutions and its growing vendor pool. By leveraging these services and tools, Hadoop's ecosystem will continue to evolve and eliminate some of the road blocks associated with the analytics processing and managing of large data lakes. Hadoop integrates into the architectural layers of the data ecosystem by using some of the tools and services discussed in this chapter.

One ecosystem is the Horton Data Platform (HDP). HDP helps you get started with Hadoop by using a single-node cluster in a virtual machine, as illustrated in Figure 1-4. Because Hadoop is a commodity (little to no additional cost) solution, HDP gives you the ability to deploy to the cloud or within your own data center.

HDP gives you the data platform foundation to build your Hadoop infrastructure, including a long list of Business Intelligence (BI) and other related vendors. The platform is designed to deal with data from many sources and formats, allowing you to design your own custom solution. The list of resources is too large to define here, but it is highly recommended that you obtain this information directly from the vendor. The beauty of selecting a product like HDP is that they are one of the leading committers with Hadoop. This opens more doors for using Hadoop with multiple database resources.



*Check with vendor. Resources may vary.

FIGURE 1-4

HDP is considered an ecosystem because it creates a community of data, bringing Hadoop and additional tools together.

Cloudera (CDH) creates a similar ecosystem for its data platform. Cloudera sets the stage with the ability to integrate structured and unstructured data. Using the platform-delivered unified services, Cloudera opens the doors to process and analyze several different data types (see Figure 1-5).

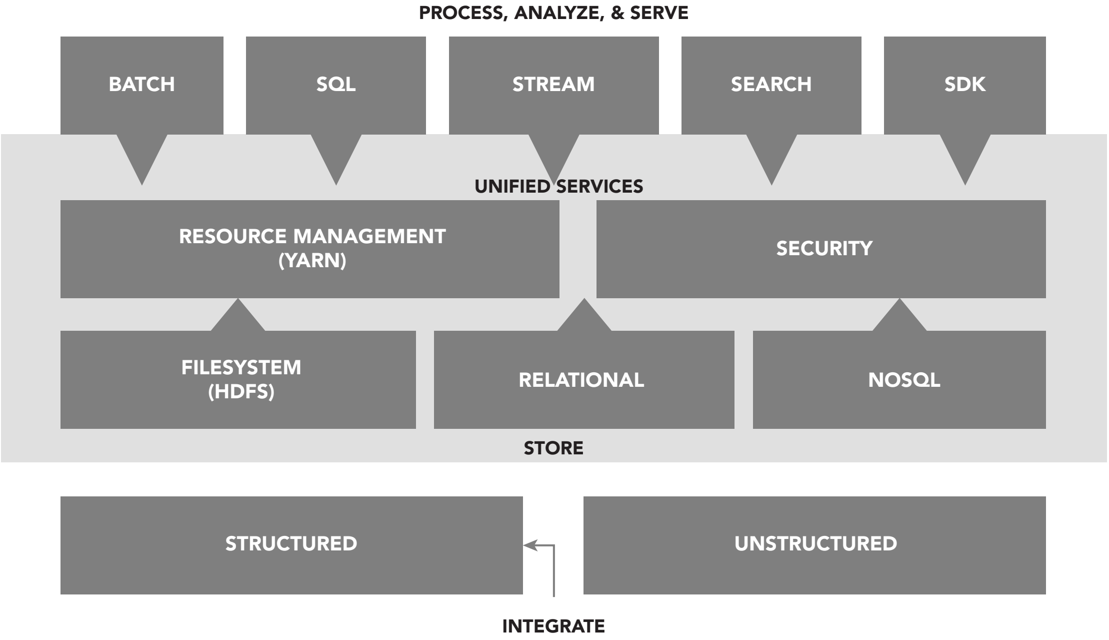


FIGURE 1-5

Data Integration and Hadoop

Data Integration is a key step in the Hadoop solution architecture. A number of vendors use open source integration tools to easily connect Apache Hadoop to hundreds of data systems without having to write code. This is a definite plus if you are not a programmer or developer by trade. Most of these vendors use a variety of open source solutions for big data integration that natively supports Apache Hadoop, including connectors for HDFS, HBase, Pig, Sqoop, and Hive (see Figure 1-6).

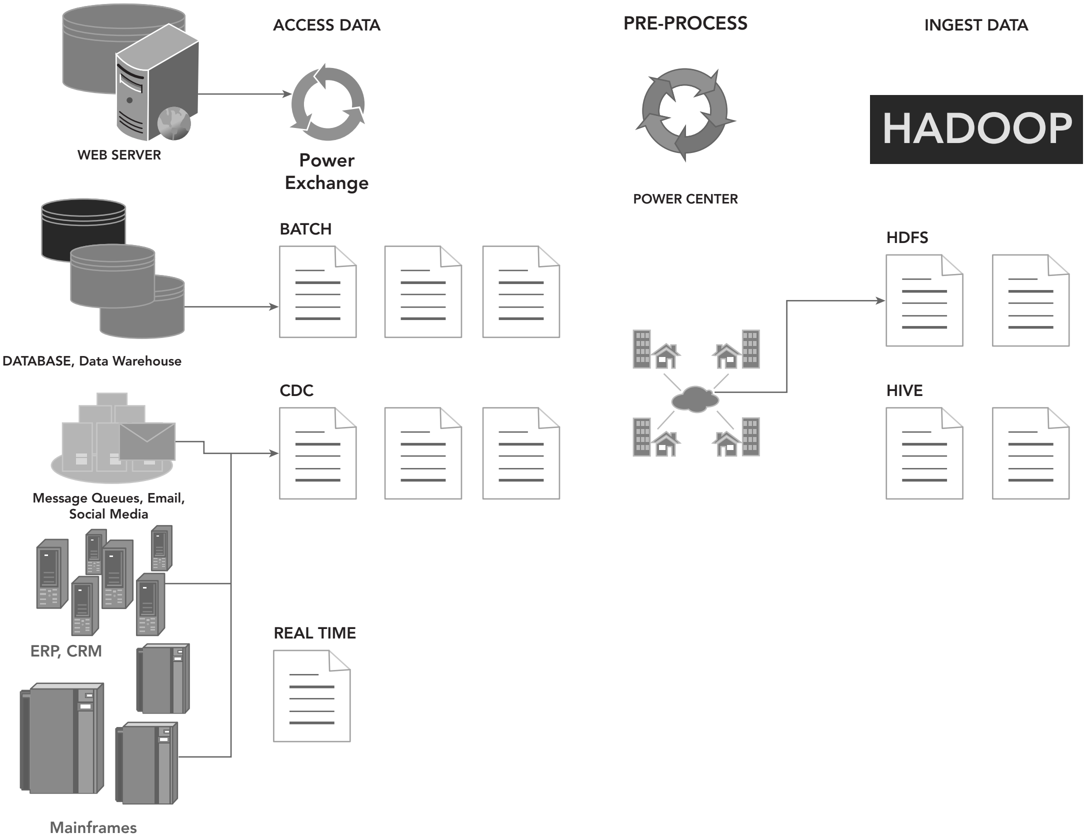


FIGURE 1-6

Hadoop-based applications are well balanced and have the ability to focus on the Windows platform and integrate well with the Microsoft BI tools such as Excel, Power View, and PowerPivot, creating unique ways for the easy analysis of massive amounts of business information.

This does not mean that Hadoop or the other data platform solutions do not run in a non–Windows based environment. It would be prudent to review your current or planned environment to determine the best solution. A data platform or data management platform is just what it says it is. It is a centralized computing system for collecting, integrating, and managing large sets of structured and unstructured data.

In theory, either HortonWorks or Cloudera could be the platform you have selected along with the RDBMS connector that works with your current data environment and Hadoop. Most vendors have highly detailed information regarding system requirements. In general, a significant number of tools will mention a Windows operating system or a Windows-based component, because of the breadth of Windows-based BI tools available. Microsoft SQL Server is the leading Windows tool for database services. Organizations using this enterprise tool are no longer limited by big data. Microsoft has the ability to work and integrate with Hadoop by providing flexibility and enhanced connectivity for Hadoop, Windows Server, and Windows Azure. Informatica software, using the *Power Exchange Connector* along with Hortonworks, optimizes the entire big data supply chain on Hadoop, turning data into actionable information to drive business value.

The modern data architecture, for example, is increasingly being used to build large data pools. By combining data management services into a larger data pool, companies can store and process massive amounts of data across a wide variety of channels including social media, clickstream data, server logs, customer transactions and interactions, videos, and sensor data from equipment in the field.

Hortonworks or Cloudera Data Platforms, along with Informatica, allows companies to optimize their ETL (Extract, Transform, Load) workloads with long-term storage and processing at scale in Hadoop.

The integration of Hadoop along with enterprise tools allows organizations to use all of the data internally and externally for an organization to achieve the full analytical power that drives the success of modern data-driven businesses.

Hadoop Applier, another example, provides real-time connectivity between MySQL and Hadoop's Distributed File System, which can be used for big data analytics—for purposes like sentiment analysis, marketing campaign analysis, customer churn modeling, fraud detection, risk modeling, and many others. Many widely used systems, such as Apache Hive, also use HDFS as a data store (see Figure 1-7).

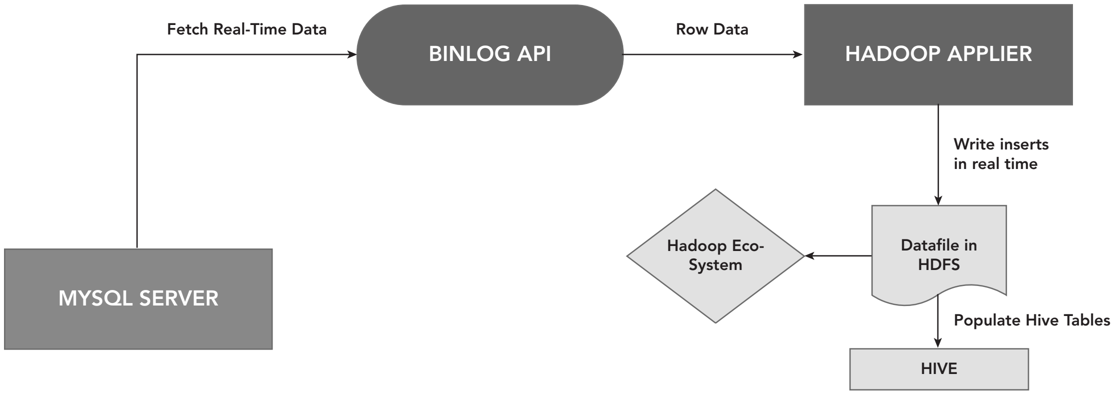


FIGURE 1-7

Oracle has developed an offering for its flagship database engine and Hadoop. It is a collection of useful tools to assist with integrating Oracle's services with the Hadoop stack. The Big Data

Connectors Suite is a collection of tools that have the ability to provide a deep dive into the information discovery waters of analytics and a fast integration of all the data stored within your infrastructure. All tools are considered scalable, which fits nicely into your environment if you are a current or future Oracle customer. Oracle has several tools in their suite, but we will only feature a few of them in this chapter.

Oracle XQuery for Hadoop (see Figure 1-8) runs a process, based on transformations expressed in the XQuery language, by translating them into a series of MapReduce jobs, which are executed in parallel on the Apache Hadoop cluster. The input data can be located in a filesystem accessible through the Hadoop Distributed File System (HDFS), or stored in Oracle's NoSQL Database. Oracle XQuery for Hadoop can write the transformation results to Hadoop files, to the Oracle NoSQL Database, or to the Oracle Database.

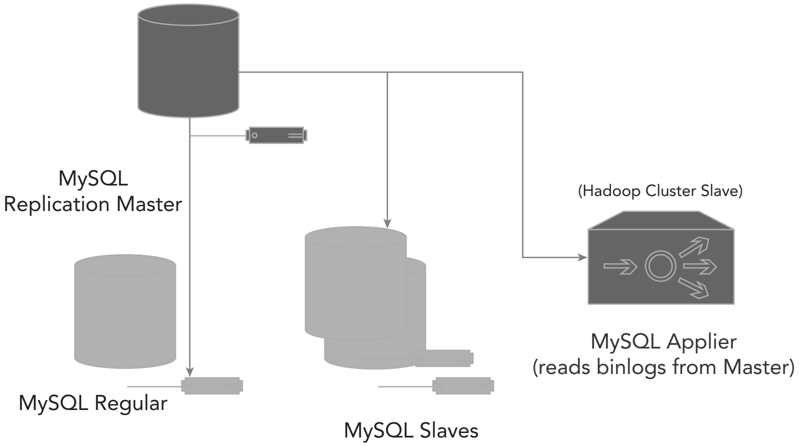
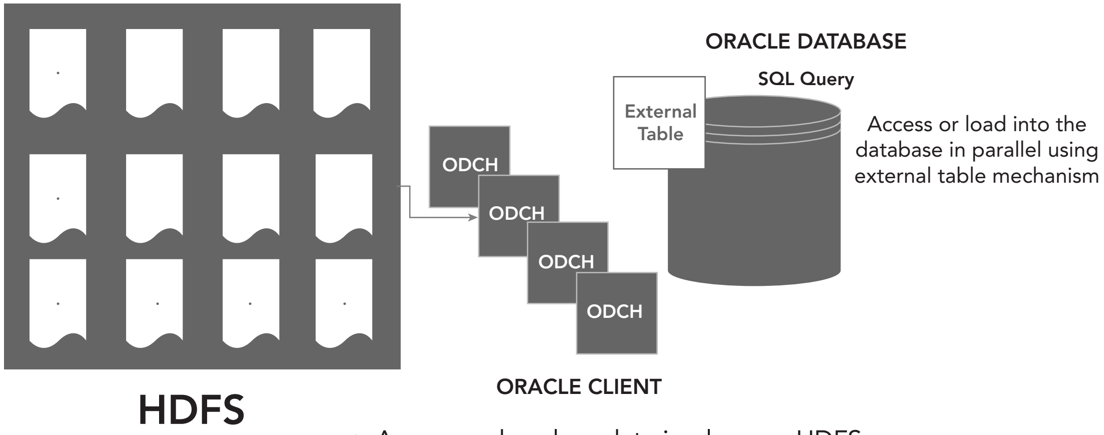


FIGURE 1-8

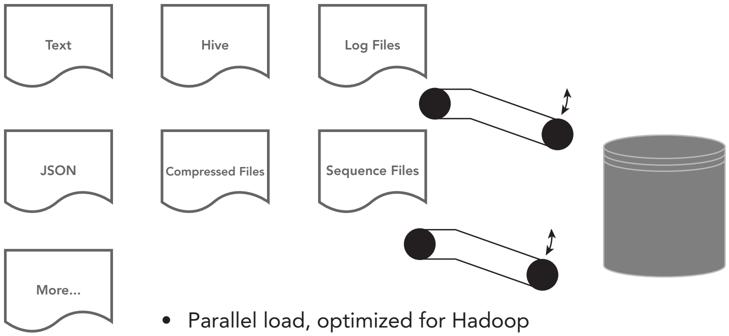
Oracle SQL Connector for the Hadoop Distributed File System (HDFS) is a high speed connector for loading or querying data in Hadoop with the Oracle Database (see Figure 1-9). Oracle SQL Connector for HDFS pulls data into the database; the data movement is initiated by selecting data via SQL in the Oracle Database. Users can load data into the database, or query the data in place in Hadoop, with Oracle SQL via external tables. Oracle SQL Connector for HDFS can query or load data in text files or Hive tables over text files. Partitions can also be pruned while querying or loading from Hive-partitioned tables.

Another Oracle solution, the Oracle Loader for Hadoop, is a high performance and efficient connector to load data from Hadoop into the Oracle Database. Oracle Loader for Hadoop pushes data into the database as data transfers are initiated in Hadoop (see Figure 1-10). Oracle Loader for Hadoop takes advantage of Hadoop compute resources to sort, partition, and convert data into Oracle-ready data types before loading. Pre-processing data on Hadoop reduces database CPU usage when loading data. This minimizes the impact on database applications and alleviates competition for resources, which is a common issue when ingesting large data volumes. It makes the connector particularly useful for continuous and frequent loads.


ORACLE DATABASE
SQL Query
External Table
Access or load into the database in parallel using external table mechanism
ODCH
ODCH
ODCH
ODCH
ORACLE CLIENT
HDFS

- Access and analyze data in place on HDFS
- Query and join data on HDFS database resident data
- Load into the database using SQL if required
- Automatic load balancing to maximize performance

FIGURE 1-9



- Parallel load, optimized for Hadoop
- Automatic load balancing
- Convert to Oracle format on Hadoop —save database CPU
- Load specific hive partitions
- Kerberos authentication
- Load directly into in-memory table

FIGURE 1-10

Oracle R Connector for Hadoop enables rapid development with R-style debugging capabilities of parallel R code on user desktops, supported by simulating parallelism (see Figure 1-11). The connector enables analysts to combine data from several environments—client desktop, HDFS, Hive, Oracle Database, and in-memory R data structures—all in the context of a single analytic task execution, thus simplifying data assembly and preparation. Oracle R Connector for Hadoop also provides a general computation framework for the execution of R code in parallel.

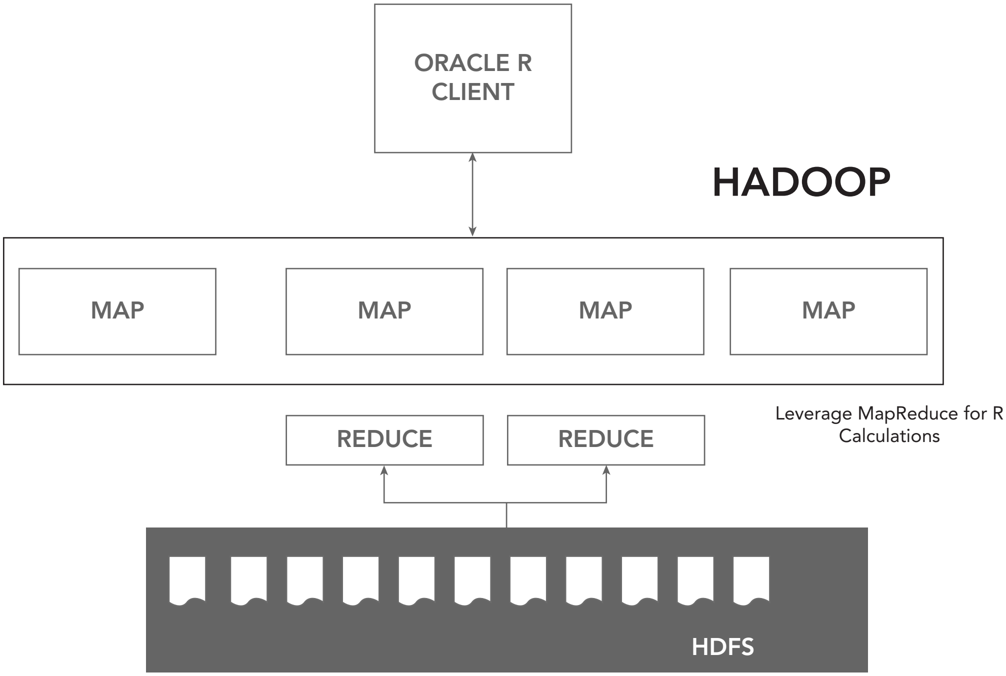


FIGURE 1-11

If Oracle is your organization’s tool of choice, you have a suite of tools to choose from, as described in this section. They have partnered with Hadoop, and the Oracle site is well documented and allows you to download any of the previously mentioned connectors as well as configure them to work with the Hadoop ecosystem.

SUMMARY

By using the Hadoop Stack, you leverage the best practices in enterprise Hadoop, combined with a mix of programming and high-level tools. Most clusters are on your premises today, but service providers are giving even more options for data to exist in the Cloud. SQL, relational, and non-relational data stores can now leverage functionality using Hadoop.

Hadoop has established itself for the long haul when it comes to data. This is very fitting, because data continues to grow over time. It uses pre-existing enterprise systems that can expand into Hadoop’s data platform. Companies and developers within the open source community are designing and defining the best practices for Hadoop-based large scale enterprise data. The businesses, as well as the IT community, are deeply concerned with scalability for all data types. With Hadoop, companies are no longer confined to expensive enterprise solutions or pricey warehouse appliances.

Hadoop is not a replacement for the existing data rich environments that populate most organizations. When you consider Hadoop, it is important to consider aspects like MapReduce or YARN, which are making huge strides in deep data analysis and advanced analytics. Hadoop provides real-time processing of big data, which can provide an immediate impact on decisions that can affect your bottom line. Various industries, from finance to healthcare, can get immediate benefits from using the Hadoop Stack, or any of its related components. It pushes the limit of what was previously thought to only be achieved with a data mining tool. It literally makes you look at data differently. Hadoop has provided the bridge that does not replace but improves how organizations look at data. Hadoop removes limitations and continues to cover new ground in all aspects of development.

Understanding Hadoop's storage system allows you to leverage data integration and business analytics to consolidate large data lakes and analyze all data types, which are not dependent on their current source. Having a complete understanding of Hadoop's platform allows its users to process a vast amount of scalable data in real time delivering optimum analytics. The beauty of Hadoop's storage process is that there is no additional storage or computing expense. There are only gains, such as increased data accuracy and analytics. The next chapter will detail the aspects of Hadoop's storage.