HADAPT

The Adaptive Analytical Platform™
for BIG Data

Technical Overview Whitepaper

March 2011

Introduction

There are three major trends in the market for analytical platforms.

(1) The amount of data that enterprises want to an analyze is exploding. Data is coming from more sources, is being produced at increasingly rapid rates, is being analyzed at an increasingly granular level, and more historical data is being kept around for analysis. At the same time, the types of analytics that enterprises want to run is becoming increasingly complex, with intricate mathematical models, statistical extrapolations, and machine learning being added on top of traditional aggregations and summarization tasks.

(2) The "cloud" is becoming an increasingly popular environment in which to perform the analysis. This includes both public clouds and private clouds. The advantage of running the analysis in the cloud is that resources are virtualized, allowing for increased utilization of the hardware and a flexible deployment option where the resources needed for a particular analysis are allocated only for the duration of the analysis and can be released once it is finished. This is an enormous shift from the traditional approach of being forced to buy hardware devoted exclusively to the analytical database. These appliances sit completely idle during quiet periods, and significant capital expenditure must be allocated to over-provision the platform to handle peak load. In addition, the public cloud allows enterprises to eliminate the burdens of managing their own infrastructure and frees them up to focus on their core competencies. However, some practical considerations such as perceived security limitations and the high costs of loading large datasets into public clouds have caused some enterprises to prefer the private cloud to the public cloud.

(3) Hadoop has commoditized the market for unstructured data processing, transformation, and analysis. Hadoop's low price (it is free and open source under an enterprise-friendly Apache license), proven track record (see http://wiki.apache.org/hadoop/PoweredBy for details about which companies are using Hadoop, and how they are using it), and rapidly emerging ecosystem, have all caused the technology to become extremely popular for unstructured data processing tasks.

Starting at the beginning of the millennium, the first of the above three trends was already clearly apparent, and several well-funded start-ups built shared-nothing MPP analytical platforms designed precisely to overcome the scalability problems that this trend was causing for traditional vendors. These "second generation analytical platforms" did extremely well in solving many of the scalability problems, and several of them were acquired by the traditional vendors over the past few years. However, trends (2) and (3) above started to emerge after the wave of second generation analytical platforms were founded, and therefore these platforms were originally architected without these trends in mind.

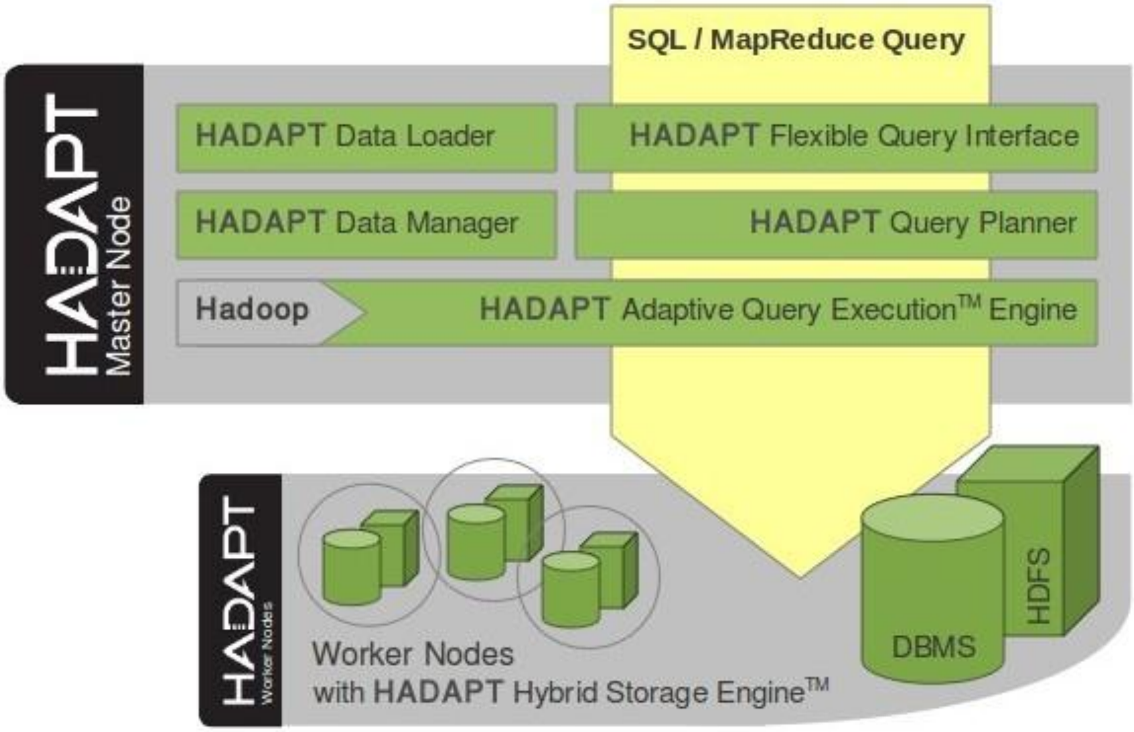
Of course, the second generation vendors did not want to be left behind, so they created stop-gap solutions in their architecture in an attempt to deal with these emerging trends. Many of them released "cloud" versions of their software, and many of them released special "connector" code, that facilitates the shipping of data back and forth between their platform and Hadoop. However, due to the fact that these solutions were bolted on long after these systems were designed and engineered, they are far from optimal. For example, clouds are well-known for drastic performance fluctuations due to the multi-tenant nature of cloud environments. Furthermore, virtual machines in the cloud are generally supposed to be disposable (i.e. they should be able to be started and stopped at any time). Unfortunately, second generation analytical platforms plan and optimize analysis completely in advance, and are unable to adapt on-the-fly at query time as performance fluctuates and virtual machines involved in processing are killed.

As another example, an architecture that includes a connector with Hadoop, where data is shipped back and forth between Hadoop and the analytical platform can cause data silos, performance degradation during transfer, and increased complexity of maintaining multiple systems. It would be far more optimal to more closely integrate the analytical platform into Hadoop, so that the combined system can handle both structured and unstructured query processing under the same roof.

In this whitepaper, we discuss the design of Hadapt, whose goal is to architect a third generation analytical platform that takes into account, and optimizes for, all three emerging trends from the beginning. By starting from scratch with these goals in mind, we are more able to closely integrate with the emerging Hadoop platform, and adapt in an optimized fashion to unpredictable cloud environments, while still maintaining the scalability (or even improving upon it) of second generation analytical platforms.

Architecture

The Hadapt platform is a hybrid analytical system designed to tackle massive amounts of data in cloud environments at high speed. Hadapt integrates closely with Hadoop by enhancing it with optimized structured data storage and querying capabilities. Petabyte datasets and thousands of machines are no longer a problem. Impressive performance



and scalability is built into the Hadapt platform at every level.

The main components of Hadapt platform include:

- *Flexible Query Interface* which allows queries to be submitted via a MapReduce API and SQL either directly or through JDBC/ODBC drivers.
- *Query Planner* that implements patent-pending Split Query Execution technology to translate incoming queries into an efficient combination of SQL and MapReduce to be executed in parallel by single-node database systems and Hadoop respectively.
- *Adaptive Query Execution*™ which provides automatic load balancing and query fault tolerance, and makes Hadapt perfect for large clusters and the cloud.
- *Data Loader* that performs partitioning of data into small chunks and coordinates their parallel load into the storage layer while replicating each chunk for fault tolerance and performance.
- *Data Manager* which stores various metadata on schema, data, and chunk distribution. It also handles data replication, backups, recovery, and rebalancing.

- *Hybrid Storage Engine* that combines a standard distributed file system (HDFS) with a DBMS layer optimized for structured data.

Flexible Query Interface

The Hadapt platform offers a highly flexible query interface. Data may be queried using both SQL and MapReduce (MR), and a combination of both. For queries that access data completely stored in relational storage, SQL is the recommended query language since its declarative nature gives the query planner more flexibility to improve performance. However, MapReduce remains a valid interface, and in fact allows for data analysts to implement even the most advanced analytical algorithms. Embedding SQL inside MR or vice versa helps achieve ultimate flexibility. Queries can be executed directly via the command-line, via a web interface, or remotely through the JDBC/ODBC drivers shipped with Hadapt. The JDBC/ODBC drivers are critical for customer-facing business intelligence tools that work with database software and aid in the visualization, query generation, result dash-boarding, and advanced data analysis. These tools are an important part of the analytical data management picture, and by supporting a more robust version SQL, Hadapt enables these tools to work much more flexibly in Hadoop deployments.

Query Planner

The SQL interface of Hadapt accepts queries and analyzes them during the query planning phase. It takes into account data partitioning and distribution, indexes, statistics about the data to create an initial query plan. Our patent-pending Split Query Execution technology ensures that the amount of work done inside the DBMS layer is maximized (for performance benefits). All work that cannot be pushed down to the underlying DBMS layer is performed by Hadoop.

Adaptive Query Execution™

As more and more data is added to a shared-nothing MPP analytical platform, the number of machines (or "nodes") across which data is partitioned also increases. It is nearly impossible to get homogeneous performance across hundreds or thousands of nodes, even if each node runs on identical hardware or in an identical virtual machine. For example, part failures, individual node disk fragmentation, software configuration errors, and concurrent queries all reduce the homogeneity of cluster performance. The problem is much worse in cloud environments, were concurrent activities performed by different virtual machines located on the same physical machine or sharing the same network can cause massive variation in performance. If the amount of work needed to execute a query is equally divided among the nodes in a shared-nothing cluster, then there is a danger that the time to complete the query will be approximately equal to time for the slowest compute node to complete its assigned task. A node with degraded performance would thus have a disproportionate effect on total query time. Since Hadapt was designed from scratch to work well in cloud environments, Hadapt adaptively adjusts the query plan and the allocation of query processing tasks to worker nodes on-the-fly during query processing to deliver much improved query speeds and increased node utilization.

Furthermore, virtual machines in the cloud are designed to be highly disposable. For example, many users of public clouds will kill and request new virtual machines when a system monitor indicates that a particular virtual machine is seeing reduced performance due to "stolen cycles" or other types of resource-hogging by other tenants on the same physical hardware. Virtual machines can also fail for more traditional reasons, and these failures are more frequent as the number of nodes involved in processing scales. When a node failure occurs or virtual machines are killed, Hadapt shifts the work to healthy machines on the fly and continues execution without restarting queries.

Data Loader

Hadapt loads data using all machines in parallel and thus achieves great speed on extremely large data sizes. For maximum query performance and fault tolerance, data is partitioned into small chunks and is replicated across the cluster.

Data Manager

Hadapt stores metadata on the schema, data, and chunk distribution. This information helps the Query Planner choose the best execution plan. The Data Manager also handles data replication, backups, recovery, and rebalancing chunks across the cluster.

Hybrid Storage Engine

A DBMS engine installed on each node complements a standard distributed file system (HDFS). While the DBMS layer is optimized for structured data, HDFS handles unstructured information.

For questions or more information about Hadapt, please email us at: info@hadapt.com