DesignMind

SQL on Hadoop: Which solution is right for you?

Most adopters of Hadoop quickly find the Hive data warehouse system to be their most important and frequently used Hadoop application. It makes sense—the Hive Query Language, which implements most of the SQL-92 standard (and more recently the windowed analytic functions of SQL-99) plus numerous extensions, provides a familiar and easy to use way for analysts to become productive on Hadoop using existing skills, and Hive is also compatible with standard JDBC / ODBC clients like Excel and Tableau.

The scalability of the Hadoop platform and the potential for low cost data storage and processing with open-source technology allows analysts access to many terabytes and even petabytes of data for analysis. Hive is also a powerful tool for data transformation, capable of reading delimited text files (among other formats), applying SQL-based transformations with a facility for user defined functions, and writing back to compressed column-store files (RCFile, Parquet) to improve performance in frequently run analytical queries typical of data warehouses. It's no wonder it's the backbone of most Hadoop-based data warehouse projects.

While Hive has matured enough to be considered a mainstream data warehouse platform, it has numerous limits:

- Its query language only supports a subset of the functionality of other modern data warehouse query languages (no scalar subqueries, for example). The gap is closed with the SQL-99 functions introduced in Hive 0.11 and later, but many users can't or won't upgrade if their distribution doesn't support this. Cloudera's CDH5, for example, with support for Hive 0.12, was only introduced in May 2014.
- Similar to the last point, there's a lack of data transformation functions used in common ELT / ETL operations such as householding, change data capture, and other similar functions, not to mention libraries of functions for dealing with custom data formats in common use in various industries. This can mean custom Java software development for functions already supported in high-level languages by other tools.

About DesignMind

DesignMind uses leading edge technologies, including SQL Server, SharePoint, .NET, Tableau, Hadoop, and Platfora to develop big data, data science, and business intelligence solutions, data warehouses, and custom applications. Headquartered in San Francisco, we help businesses leverage their data to gain competitive advantage.

DesignMind A white paper by **Mark Kidwell**, Principal Big Data Consultant, DesignMind | September 2014

- Hive's processing engine, Hadoop's venerable MapReduce, is a highly scalable method for processing large quantities of data, but is fundamentally batch oriented and adds considerable latency for small queries. Techniques exist to improve Hive performance (see my upcoming Hive tuning article for more info), and alternative backends are implemented or planned including Tez and Spark (HIVE-7292) respectively. But Hive is still less appropriate for interactive, user-facing queries requiring sub-second response.
- Security at all levels remains a challenge. Simple concepts in existing DBMS platforms, such as granting access to a view or a particular function are more difficult to implement in Hive and Hadoop, but getting easier. Features like ACLs to cell-level encryption / decryption functions (critical for storing PCI/PII sensitive data in Hadoop) are absent from all Hadoop tools.

"Security at all levels remains a challenge. Simple concepts in existing DBMS platforms, such as granting access to a view or a particular function are more difficult to implement in Hive and Hadoop, but getting easier. Features like ACLs to cell-level encryption / decryption functions (critical for storing PCI/PII sensitive data in Hadoop) are absent from all Hadoop tools."

A number of database vendors have taken notice of these gaps, and worked to provide some alternative to SQL on Hadoop via Hive. The vendors approach the problem in several ways:

- Provide connectors from an existing DB engine / platform to Hadoop using some external data interfaces, streaming in data on demand and processing it locally. While this approach allows a richer set of functionality than Hive, it defeats the purpose of Hadoop - leveraging cheap compute nodes with local disk for massively parallel processing of large data sets.

- Similar to Hive, transform a high-level language such as SQL into a low-level MapReduce (or Tez / Spark) job that runs on the cluster. Tools like DataMeer adopt this approach, adding a huge library of useful functions for analysts (DataMeer also adds great data viz and a spreadsheet-like UI). Unfortunately for applications that require interactive performance, using MapReduce doesn't help.
- Run the DB engine right on the cluster (on multiple / all datanodes), reading directly from HDFS and leveraging locality of data (processing of a slice of data occurs on the same node holding the data). This architecture is often the most costly for DB vendors to implement as it can require them to start from scratch, but typically produces the best performance. Cloudera Impala, Rainstor, Actian Vector for Hadoop, Splice Machine and Vertica for MapR (among others) take this approach.

"Most companies spending a lot on appliances would still take advantage of the chance to cut their data management costs by 50% or more."

Of the options above, the third is truly the goal for most organizations, turning a Hadoop cluster running on commodity hardware into the equivalent of more expensive MPP data warehouse appliance. The question then becomes what features separate these, and how much do they cost? At the low (free) end, Cloudera Impala, Apache Drill, Metamarket's Druid and Facebook Presto are all open-source, highly-scalable interactive query engines that leverage inter/intra-query parallelism, compression, column-oriented storage and other modern techniques to improve performance. But they are relatively immature, often lagging behind Hive and proprietary systems in SQL functionality and security features by several years or more. At the higher end, the other database products from Rainstor, Actian and Vertica add better SQL functionality, security, higher concurrency, and even better performance. Actian Vector claims up to 30x performance improvements over Impala through its use of vector (SIMD) style processing, Vertica pre-materializes multiple projections of tables in sorted order, and Rainstor is known for extreme compression and data security suitable for compliance with financial data storage requirements.

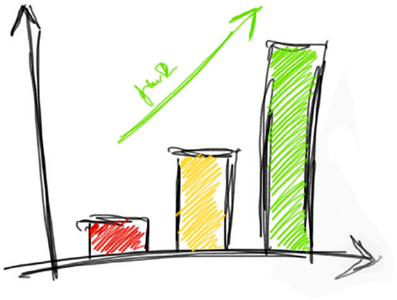
All of the additional features of the proprietary solutions come at a cost, however. Pricing for these products can run in the several thousand dollars per node or thousands per terabyte loaded. Added to the cost of building and supporting a Hadoop cluster, and managing the database itself, and they can start to approach the same order of magnitude as data warehouse appliances. Even then, most companies spending a lot on appliances would still take advantage of the chance to cut their data management costs by 50% or more.

Knowing the above choices, which solution is right for you?

- If you're already using Hadoop, but not yet using a data warehouse appliance, you're ready to start benefitting right away with an high-performance open-source query engine like Impala or Drill. Just be aware that their functionality won't approach higher-end systems, or even Hive for some use cases.
- If you're already paying for and using a data warehouse appliance that's nearing its end-of-life, you can probably save money on the next generation with a Hadoop-based high-end solution. It's time to start your next-gen SQL on Hadoop pilot project ASAP!
- If you're already using and just paid for a data warehouse appliance, you of course will wait until it's time to add capacity or replace the appliance before buying another platform. That's OK, as it gives you more time to deploy Hadoop and gain operational experience running and using it for other analytic and storage use cases, and it lets the SQL on Hadoop technology bake even more.

The Hadoop platform has opened amazing opportunities for companies to improve their data processing and get new insights from ever-larger data sets at a fraction of the cost of a only a few years ago. It's certain that Hadoop technology will continue to improve over the next several years, and everyone will benefit from the lower cost and improved functionality that comes with that improvement.

> *"The Hadoop platform has opened amazing opportunities for companies to improve their data processing and get new insights from ever-larger data sets at a fraction of the cost of only a few years ago."*



Mark Kidwell specializes in designing, building, and deploying custom end-to-end Big Data solutions and data warehouses.