



HBase: Hadoop's Next Big Data Chapter

HBase still has a few 'rough edges,' but that hasn't kept this NoSQL database from becoming one of the hottest pockets within the white-hot Hadoop market.

By Doug Henschen, [InformationWeek](#)

May 08, 2012

URL: <http://www.informationweek.com/software/information-management/hbase-hadoops-next-big-data-chapter/232901601>

The Apache Hadoop Framework has many components, including the MapReduce distributed data-processing model, Hadoop Distributed File System (HDFS), Pig data-flow language, and Hive distributed data warehouse. But the part stealing much of the attention these days--and arguably maturing most rapidly--is HBase.

HBase is Hadoop's NoSQL database. Patterned after Google BigTable, HBase is designed to provide fast, tabular access to the high-scale data stored on HDFS. High-profile companies including Facebook and StumbleUpon use HBase, and they're part of a fast-growing community of contributors who are adding enterprise-grade reliability and performance upgrades at steady clip.

Why is HBase so important?

First and foremost, it's part of the [Hadoop Framework](#), which is at the epicenter of a movement in which companies large and small are making use of unprecedented volumes and varieties of information to make data-driven decisions. Hadoop not only handles data measured by the tens or hundreds of terabytes or more, it can process textual information like social media streams, complex data like clickstreams and log files, and sparse data with inconsistent formatting. Most importantly, it does all this at low cost, powered by open source software running on highly scalable clusters of

inexpensive, commodity X86 servers.

[**Want more on NoSQL databases? Read [Amazon DynamoDB: Big Data's Big Cloud Moment](#).]**

The core data-storage layer within Hadoop is HDFS, a distributed file system that supports MapReduce processing, the approach that Hadoop practitioners invariably use to boil down big data sets into the specific information they're after. For operations other than MapReduce, HDFS isn't exactly easy to work with. That's where HBase comes in. "Anybody who wants to keep data within an HDFS environment and wants to do anything other than brute-force reading of the entire file system [with MapReduce] needs to look at HBase," explains Gartner analyst Merv Adrian. "If you need random access, you have to have HBase."

HBase offers two broad use cases. First, it gives developers database-style access to Hadoop-scale storage, which means they can quickly read from or write to specific subsets of data without having to wade through the entire data store. Most users and data-driven applications are used to working with the tables, columns, and rows of a database, and that's what HBase provides.

Second, HBase provides a transactional platform for running high-scale, real-time applications. In this role, HBase is an ACID-compliant database (meeting standards for Atomicity, Consistency, Isolation, and Durability) that can run transactional applications. That's what conventional relational databases like Oracle, IBM DB2, Microsoft SQL Server, and MySQL are mostly used for, but HBase can handle the incredible volume, variety, and complexity of data encountered on the Hadoop platform. Like other NoSQL databases, it doesn't require a fixed schema, so you can quickly add new data even if it doesn't conform to a predefined model.

Life sciences research firm NextBio uses Hadoop and HBase to help big pharmaceutical companies conduct genomic research. The company embraced Hadoop in 2009 to make the sheer scale of genomic data-analysis more affordable. The company's core 100-node Hadoop cluster, which has processed as much as 100 terabytes of data, is used to compare data from drug studies to publically available genomics data. Given that there are tens of thousands of such studies and 3.2 billion base pairs behind each of the hundreds of genomes that NextBio studies, it's clearly a big-data challenge.

NextBio uses MapReduce processing to handle the correlation work, but until recently it stored the results--now exceeding 30 billion rows of information--exclusively on a MySQL database. This conventional relational database offers fast storage and retrieval, but NextBio knew it was reaching the limits of what a conventional database like

MySQL could handle in terms of scale--at least without lots of database administrative work and high infrastructure cost.

"With MySQL we started down the typical database route of sharding and partitioning in order to scale, but we've known for some time that it would be much more effective to use HBase," says Satnam Alag, NextBio's VP of engineering. With HBase NextBio can just add nodes to a cluster in order to scale, spreading the storage and compute load across more servers.

Trusting that HBase could deliver "optimal write and read performance" and "orders of magnitude" higher scalability than MySQL, Alag says NextBio late last year implemented a small cluster exclusively for HBase work. It used a Hadoop distribution from MapR, which provides software as well as enterprise support and training. It took six months to bring the deployment into production, in large part due to the learning curve associated with HBase, says Alag.

NextBio's primary Hadoop system for MapReduce processing runs entirely on Apache open-source software, and the company does not use any third-party support. Where HBase was concerned, NextBio wanted MapR's help. The support is necessary, says Alag, because HBase still has "rough edges." For example, NextBio has had trouble with servers going down, and when that happens, it loses access to some of its data until those servers can be recovered.

"We're trying to figure out whether it's our code or the way we're using it that causes servers to crash," Alag says. "We hit HBase pretty hard, and we're constantly adding new data."

[Want more on NoSQL databases? Read [2 Lessons Learned Managing Big Data In Cloud](#).]

For the time being, NextBio is still running its MySQL-based correlation database as a backup until it can be more confident about HBase reliability. Alag says his company plans to test HBase release 0.92, a recent Apache software upgrade that promises greater resiliency and recoverability. For example, active replication has been added so live replicas of HBase clusters can be set up in separate data centers. This supports failover between sites in the event of a failure. The release also supports distributed log splitting, which speeds recovery after server failures.

Alag and analysts like Adrian of Gartner are confident the HBase reliability problems will be solved fairly quickly.

There are NoSQL alternatives to HBase, even within the Apache family. [Apache Cassandra](#), for example, is a highly scalable, NoSQL database that's gaining popularity, but it has different strengths than HBase, according to Charles

Zedlewski, VP of products at Hadoop distributor and training and support firm Cloudera.

"HBase is known for consistency, whereas Cassandra's strong suit is availability," Zedlewski explains. "HBase will always will always give you the right answer, but sometimes portions of the data go offline. Cassandra always supplies an answer, but it's not always up to data with the very latest information."

In recognition of these strengths, HBase tends to get used for write-heavy workloads. As an example, Cloudera customer eBay is using Hbase as the basis for a new search engine, code-named Cassini, that will handle search and search indexing across eBay's massive sites.

"If you have millions of merchants updating what products they have for sale on a daily basis, that's a write-heavy workload," Zedlewski says.

In contrast, Cassandra is usually favored when low-latency reading is of utmost importance but the content might not be updated all that frequently, Zedlewski says.

HBase is crucial to the Hadoop community because most developers want to write to databases, not file systems, says Zedlewski. Tabular access to data is what they're used to, and that's what HBase give them--all with big-data scale and NoSQL flexibility.

Copyright © 2012 [United Business Media LLC. All rights reserved.](#)