

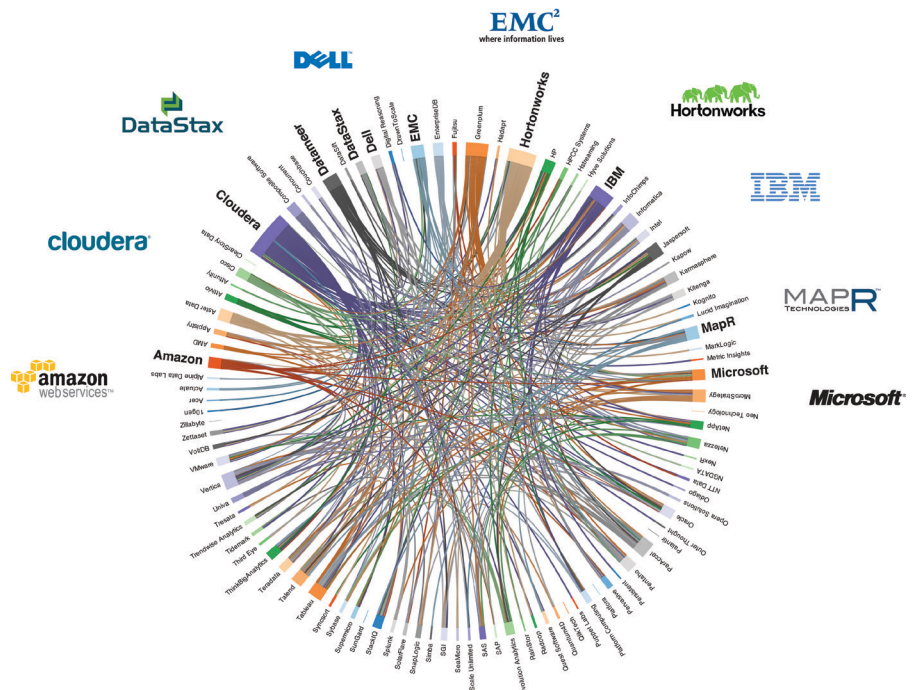
Beyond Hadoop

The leading open source system for processing big data continues to evolve, but new approaches with added features are on the rise.

WHEN A NEW user visits the online music service Pandora and selects a station, the company's software immediately generates a playlist based on the preferences of its community. If the individual creates a Chopin station, for example, the string of songs would consist of the most popular renditions of the composer's music among Pandora's community. Once the new listener inputs a rating, clicking on either the "thumbs up" or the "thumbs down" icon, Pandora factors this preference into future selections. In effect, the service becomes smarter with the vote of each thumb.

Pandora will not discuss exactly how much data it churns through daily, but head of playlist engineering Eric Bieschke says the company has at least 20 billion thumb ratings. Once every 24 hours, Pandora adds the last day's data to its historical pool—not just thumbs, but information on skipped songs and more—and runs a series of machine learning, collaborative filtering, and collective intelligence tasks to ensure it makes even smarter suggestions for its users. A decade ago this would have been prohibitively expensive. Four years ago, though, Bieschke says Pandora began running these tasks in Apache Hadoop, an open source software system that processes enormous datasets across clusters of cheap computers. "Hadoop is cost efficient, but more than that, it makes it possible to do super large-scale machine learning," he says. Pandora's working dataset will only grow, and Hadoop is also designed for expansion. "It's so much easier to scale. We can literally just buy a bunch of commodity hardware and add it to the cluster."

Bieschke is hardly alone in his endorsement. In just a few years, Hadoop has grown into the system of choice for engineers analyzing big data in fields as diverse as finance, marketing, and bioinformatics. At the same time, the



Hadoop ecosystem components as visualized by Datameer.

changing nature of data itself, along with a desire for faster feedback, has sparked demand for new approaches, including tools that can deliver ad hoc, real-time processing, and the ability to parse the interconnected data flooding out of social networks and mobile devices. "Hadoop is going to have to evolve," says Mike Miller, chief scientist at Cloudera, a cloud database service based in Boston, MA. "It's very clear that there is a need for other tools." Indeed, inside and outside the Hadoop ecosystem, that evolution is already well under way.

Distributing Data

Hadoop traces back to 2004, when Google published the second of a pair of papers [see the "Further Reading" list] describing two of the key ideas behind its search success. The first detailed the Google File System, or GFS, as a way of distributing data across hundreds or thousands of inexpensive computers. To glean insights from that

data, a second tool, called MapReduce, breaks a given job into smaller pieces, sends those tasks out to the different computers, then gathers the answers in one central node. The ideas were revolutionary, and soon after Google released the two papers, Yahoo! engineers and others quickly began developing open source software that would enable other companies to take advantage of the same breed of reliable, scalable, distributed computing that Google had long enjoyed.

The result, Apache Hadoop, consists of two main software modules. The Hadoop Distributed File System (HDFS) is similar to a file system on a single computer. Like GFS, it disperses enormous datasets among hundreds or thousands of pieces of inexpensive hardware. The computational layer, Hadoop MapReduce, takes advantage of the fact that those chunks of data are all sitting on independent computers, each with its own processing power. When a developer writes a program

to mine that data, the task is split up. “Each computer will look at its locally available data and run a little segment of the program on that one computer,” explains Todd Lipcon, an engineer at Palo Alto, CA-based Hadoop specialist Cloudera. “It analyzes its local data and then reports back the results.”

Although Hadoop is open source, companies like Cloudera and MapR Technologies of San Jose, CA, have found a market in developing additional services or packages around making it easier to use and more reliable. MapR, for example, has helped ancestry.com use Hadoop to carry out pattern matches on its enormous library of DNA details. After a customer sends in a saliva sample, the company can extract the basic biological code and use a Hadoop-based program to search for DNA matches—that is, potential mystery relatives—across its database.

Analyzing Networks

For all its strengths in large-scale data processing, however, experts note MapReduce was not designed to analyze data sets threaded with connections. A social network, for example, is best represented in graph form, where in each person becomes a vertex and an edge drawn between two individuals signifies a connection. Google’s own work supports the idea that Hadoop is not set up for this breed of data: The company’s Pregel system, publicly described for the first time in 2009, was developed specifically to work with graph structures, since MapReduce had fallen short.

Along with a handful of students, University of Washington network scientist Carlos Guestrin recently released a new open source processing framework, GraphLab, that uses some of the basic MapReduce principles but pays more attention to the networked structure. The data distribution phase takes the connections into account. “If I know that you and I are neighbors in the graph, there will be some computation that needs to look at your data and my data,” Guestrin explains. “So GraphLab will try to look at our data on the same machine.”

The trick is that GraphLab partitions this data in a novel way. The standard method would have been to split the data into groups of highly connected

“Hadoop is going to have to evolve,” says Mike Miller. “It’s very clear that there is a need for other tools.”

vertices. In social networks, however, a few persons have a disproportionate number of connections. Those people, or vertices, could not be stuffed into single machines, which would end up forcing numerous computers to communicate. To avoid this inefficiency, Guestrin says GraphLab’s algorithm partitions data according to the edges, so that closely linked edges are on the same machines. A highly connected individual such as the pop star Britney Spears will still live on multiple pieces of hardware, but far fewer than with the standard technique. “It minimizes the number of places that Britney Spears is replicated,” Guestrin explains.

Hadoop, on the other hand, is agnostic to the structure of the data, according to Guestrin. Two pieces of information that should be analyzed on the same computer might end up in different clusters. “What ends up happening is that they have to move data around a lot,” Guestrin says. “We can be smart about how data is placed and how data is communicated between machines.”

Hadoop can often complete the same tasks as GraphLab, but Guestrin says his more efficient approach makes it much faster. On several common benchmark tests, such as a name recognition task in which an algorithm analyzes text and assigns different categories to words, GraphLab has completed the job 60 times faster, using the same hardware.

Running in Real Time

In some cases, though, this is not fast enough. Today’s companies often want results in real time. A hedge fund might be looking to make a snap decision based on the day’s events. A

ACM Member News

DANIEL SPIELMAN WINS MACARTHUR ‘GENIUS’ AWARD



If you received a \$500,000 grant for being a “genius,” how would you react? In the case of

Yale University computer scientist Daniel Spielman, he did not tell a soul other than his wife.

“I learned about the award in mid-September,” explains Spielman, who recently received a MacArthur Fellowship award. “But the MacArthur Foundation people asked me to keep it a secret until Oct. 1. Fortunately, I received an email before speaking to them. Otherwise, I would have tipped off my entire department with my screaming.”

As for spending the money? Spielman has no immediate plans. “It’s given in 20 installments over five years,” he says, “so I can’t do anything too crazy with it. But I plan to use it to provide more time to work on research.”

That is good news for the scientific community—and society. Spielman has devoted his career to the pursuit of abstract questions that address how to measure, predict, and regulate the environment and behavior. He helped develop error-correcting codes that are used in satellite video broadcasts, as well as optimization algorithms that support computational science and machine learning on massive datasets, among other applications.

“Most of my research is about the design of faster algorithms. They don’t just ‘speed things up.’ They change what is reasonable to do. You wouldn’t, say, browse the Internet on a phone if it took 10 minutes to load a page. Sophisticated algorithms compress the communications of the phone to enable it to transmit reliably without using too much power. Web pages are rarely delivered to your phone through the shortest route in the Internet. Rather, algorithms are used to manage information flow and prevent information traffic jams.”

—Dennis McCafferty

global brand might need to respond quickly to a trending topic on Twitter. For those sorts of snap decision-related tasks, Hadoop is too slow, and other tools have begun to emerge.

The Hadoop community has been building real-time response capabilities into HBase, a software stack that sits atop the basic Hadoop infrastructure. Cloudera's Lipcon explains that companies will use Hadoop to generate a complicated model of, say, movie preferences based on millions of users, then store the result in HBase. When a user gives a movie a good rating, the website using the tools can factor that small bit of data into the model to offer new, up-to-date recommendations. Later, when the latest data is fed back into Hadoop, these analyses run at a deeper level, analyzing more preferences and producing a more accurate model. "This gives you the sort of best of both worlds—the better results of a complex model and the fast results of an online model," Lipcon explains.

Cloudant, another real-time engine, uses a MapReduce-based framework to query data, but the data itself is stored as documents. As a result, Miller says, Cloudant can track new and incoming information and only process the changes. "We don't require the daily extraction of data from one system into another, analysis in Hadoop, and re-injection back into a running application layer," he says. "That allows us to analyze results in real time." And this, he notes, can be a huge advantage. "Wait-

GraphLab, a new open source processing framework, uses some of the basic MapReduce principles, but pays more attention to the networked structure.

ing until overnight to process today's data means you've missed the boat."

Miller says Cloudant's document-oriented store approach, as opposed to the column-oriented store adopted in HBase, also makes it easier to run unexpected or ad hoc queries—another hot topic in the evolving Hadoop ecosystem. In 2009, Google publicly described its own ad hoc analysis tool, Dremel, and a project to develop an open source version, Drill, just launched this summer. "In between the real-time processing and batch computation there's this big hole in the open source world, and we're hoping to fill that with Drill," says MapR's Ted Dunning. LinkedIn's "People You May Know" functionality would be an ideal target for Drill, he notes. Currently, the results are on a 24-hour delay.

"They would like to have incremental results right away," Dunning says.

Although these efforts differ in their approaches, they share the same essential goal. Whether it relates to discovering links within pools of DNA, generating better song suggestions, or monitoring trending topics on Twitter, these groups are searching for new ways to extract insights from massive, expanding stores of information. "A lot of people are talking about big data, but most people are just creating it," says Guestrin. "The real value is in the analysis." **C**

Further Reading

Anglade, T.

NoSQL Tapes, <http://www.nosqltapes.com>.

Dean, J. and Ghemawat, S.

MapReduce: Simplified data processing on large clusters, *Proceedings of the 6th Symposium on Operating Systems Design and Implementation*, San Francisco, 2004.

Ghemawat, S., Gobioff, H., and Leung, S.

The Google file system, *Proceedings of the 19th ACM Symposium on Operating Systems Principles*, Lake George, NY, Oct. 19–22, 2003.

Low, Y., Gonzalez, J., Kyrola, A., Bickson, D., Guestrin, C., and Hellerstein, J.M.

GraphLab: A new parallel framework for machine learning, *The 26th Conference on Uncertainty in Artificial Intelligence*, Catalina Island, CA, July 8–11, 2010.

White, T.

Hadoop: The Definitive Guide, O'Reilly Media, Sebastopol, CA, 2009.

Gregory Mone is a Boston, MA-based writer and the author of the novel *Dangerous Waters*.

© 2013 ACM 0001-0782/13/01

Milestones

Supercomputing Visionaries Honored

ACM and the IEEE Computer Society honored high-performance computing innovators at the recent SC12 conference in Salt Lake City, UT. Among those honorees were the inventor of the first multicore processor, biomolecular modeling researchers, and an expert in managing software security flaws.

University of Notre Dame computer science and engineering professor Peter Kogge received the Seymour Cray Computer Engineering Award. Kogge

developed the space shuttle I/O processor, invented the Kogge-Stone-Adder process for adding numbers in a computer, and helped create the first multicore processor (EXECUBE) at IBM. He recently spearheaded DARPA's initiative to investigate a super-computer capable of a quintillion operations per second.

Klaus Schulten and Laxmikant Kale, professors at the University of Illinois at Urbana-Champaign, received the Sidney Fernbach Award for

their contributions to the development of "widely used parallel software for large biomolecular systems simulation." Schulten directs the Center for Biomolecular Modeling and was the first to demonstrate that parallel computers can be used to solve the "many-body" problem in biomolecular modeling. Kale directs the Parallel Programming Laboratory; his work has focused on enhancing performance and productivity via adaptive run-time systems.

Mary Lou Soffa of the University of Virginia received the ACM-IEEE Computer Society Ken Kennedy Award for her work in detecting and managing software security flaws. Soffa developed software tools for debugging and testing programs to eliminate or reduce false alarms and improve operating efficiency. Her research produced automatic, practical solutions in software engineering, and systems and programming languages for improving software reliability, security and productivity.