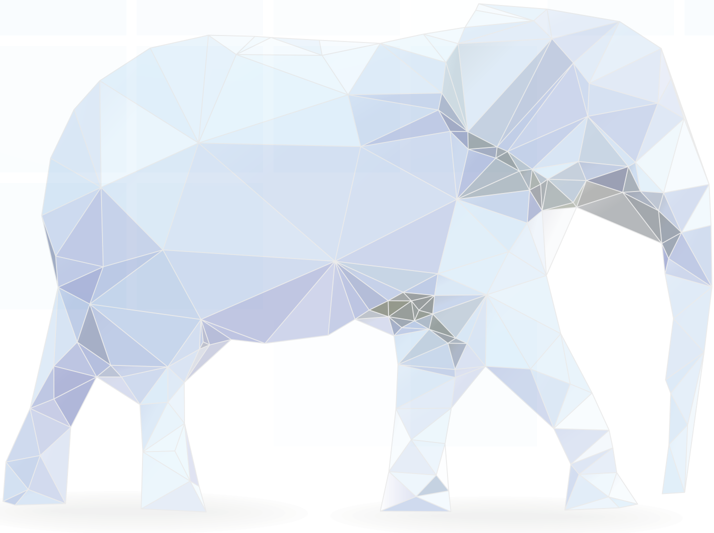
3 IDEAS TO TAKE HADOOP MAINSTREAM

XURMO

Configure Intelligence



Every once in a while, comes an innovation that fundamentally disrupts the way things are done. Hadoop is one such, that has introduced massively scalable, distributed computing to the data-rich enterprise. If nothing else, CIOs can at least imagine an infrastructure where any kind of data can be brought to play, at any scale and without burning through their budgets on specialized infrastructure.

As with any other disruptive innovation, Hadoop too has its growth pangs and its sharply divided groups of supporters, detractors and ambivalent onlookers. At the close of 2014, the last group is by far the largest and probably the most important, since it consists of decision makers and practitioners –people who stake their success on this one buying decision. In my conversations with CIOs, I notice a cautious optimism about Hadoop – optimism for how the business can be transformed with better, timely decisions and caution for whether the technology can deliver all it promises. A *Wikibon* survey bears out this ambivalence – 64% of Hadoop deployments are still in pilot, with CIOs unsure whether their pilot implementations will scale smoothly and maintain performance in production.

There are couple of truisms about Hadoop that set the context for why it is not mainstream yet.

HADOOP IS NOT ONE TECHNOLOGY

It is a collection of tools built on the common theme of distributed computing. At its foundation are the Hadoop Distributed File System (HDFS) - which can store raw data across multiple connected servers that in effect, behave like a single system - and MapReduce - which is the data processing layer that also works on multiple servers but in effect, behaves like a single processing engine. Everything else in 'Hadoop' is outside native storage and processing and falls under 2 broad ecosystem categories: Infrastructure management tools and Data management tools.

The state of the art in the Hadoop ecosystem is already addressing much of the concerns in Infrastructure management. Hadoop 2.0 addresses disaster recovery and high availability of hardware with in-built redundancies. Scalability and performance in Data management/ processing is another story altogether. For example, MapReduce has been proven inadequate for interactive querying and requires skills that are not common. Even batch processing in native MapReduce is slower than acceptable thresholds in many enterprises. Data discovery and governance are not natively solved in Hadoop but get exponentially complex as data variety increases.

HADOOP IS NOT CONTENT AWARE

The original intent for Hadoop was to acquire data quickly, store it cheap and enable parallel processing on it. It was never intended to have systemic understanding of what it contains and whether it is useful or not. Hadoop natively cannot determine which data to save or discard, which data to reveal or hide and which data to select and in what order, to return quick results to users' queries. All that has to be done *post-facto*, once data is acquired in HDFS, using tools which are built for such specialised jobs. And it is a burgeoning list of tools debuting in the ecosystem, each solving a specific problem to try and make Hadoop enterprise-friendly.

This massive retro-fit means more complexity, ambiguity and expenses in working with data stored in HDFS – compared to a traditional Data Warehouse, where a single technology stack works on data conforming to well-defined schema for the specific purpose of analysis.

The bottom-line is that Hadoop and its infrastructure management tools are just not enough to make the cut for widespread enterprise deployments. Hadoop needs some enhancements to make it an enterprise-grade data management infrastructure - where data discovery, governance, scalability and performance issues are comprehensively taken care of.

There are 3 areas that the Hadoop ecosystem should work on:

USE HADOOP TO BUILD A CONTENT AWARE SYSTEM

As discussed earlier, Hadoop is not content-aware, nor is it designed to be. This poses a significant challenge in data management. For comparison, conventional warehousing systems perform the way they do (superbly well!) because they have well-defined schema to place data in. A system that is aware of the data it contains can be configured/ programmed to manage its performance and stability based on well-defined rules.

In the context of Hadoop, the solution is NOT to make it yet another relational database – though that can be done quite easily. Defining schema – even for well-known, structured data sets is not simple. Attempting that on unstructured, dynamic data is suicidal.

The Hadoop-specific path would be to capture as much detail about data that is quickly and painlessly possible, to enable configurable data management. For example in Xurmo, we capture all Cartesian co-ordinates of input data, but not the logical relationships in data. In this architecture Hadoop performs wonderfully as the scalable, inexpensive repository of raw data. Data co-ordinates are also stored in the same system.

A content-aware Hadoop system can address data management issues in ways vanilla Hadoop cannot.

APPLY MACHINE LEARNING ON DATA AND USAGE STATS FOR DATA MANAGEMENT

Machine learning algos offer probabilistic recommendations based on patterns in data. While they don't eliminate the need for human verification/ validation, ML algos certainly save a lot of manual effort. A content-aware system is able to supply data stats to algos which can recommend – and in some cases, autonomously take calls on data management decisions. For example, Xurmo uses stats captured by our content-aware platform to enable search-guided querying, optimized query paths and data governance.

Issues associated with data management in Hadoop can thus be addressed at *asystem* level.

MIGRATE FROM MAPREDUCE TO DISTRIBUTED IN-MEMORY PROCESSING

Native data processing through MapReduce is already passé. Folks at the University of Berkeley invented a new way of distributed in-memory processing, which is now an open source project called Spark. The performance improvement in query responses is frankly breathtaking. At Xurmo, we migrated from MapReduce to Spark in early 2014 and the customer feedback from our production deployments is a testament to this decision.

Hadoop is a wonderful idea and one that deserves a long leash for evolution. It has opened up possibilities for enterprises that were unimaginable just a few years ago. While in its current form, it doesn't do enough to assuage CIO concerns, there is much to be said about its potential to be the data infrastructure for enterprises in the next few years.