The Bloor Group

Hadoop, Enterprise Ready and Beyond

WANdisco Enables Businesses to Re-Think Hadoop

Robin Bloor, Ph.D.
& Rebecca Jozwiak

WHITE PAPER

Hadoop: The Upside and the Downside

Internet search businesses faced a problem from the get-go. They had to query and manage bigger volumes of data than had ever previously been done and the solution they invented needed to scale effectively, because the data they harvested from the World Wide Web was growing exponentially. It was this that gave rise to Hadoop.

It was a product of research in this area, created by Doug Cutting and Mike Cafarella at Yahoo! in 2005, now a whole decade ago. It became an Apache open source project in 2006 and, Cloudera, the first commercial company to provide support for Hadoop, was founded in 2008. Commercial “early adopter” use of Hadoop never got going until about 2012, but after that enthusiasm for it escalated. It became, as is said, “high fashion among the technorati.”

YARN and the Momentum of Hadoop

Despite its proven scalability, until the release of YARN in 2013, Hadoop was limited in its areas of application – it was used primarily for ETL and for archiving data to the Hadoop Distributed File System (HDFS). Its main constraints were that it ran just one task at a time and that software development on Hadoop was tied to the use of MapReduce. The October 2013 release of Hadoop 2.0, which included YARN, eliminated these limitations entirely. A significant increase in the momentum of Hadoop was an immediate and expected consequence of this.

At that time Hadoop’s software ecosystem consisted mainly of products from related Apache projects, such as: HBase, HCatalog, Pig, Hive, Flume, Sqoop, Nutch, Avro, Oozie, ZooKeeper and a few others. While one or two commercial software vendors were providing add-on Hadoop components, most were simply providing data access to Hadoop and its open source family, treating it as if it were yet another database.

The advent of YARN transformed the situation for simple reasons. Once Hadoop had a scheduling capability it was possible for many applications to share HDFS data concurrently, making the file system far more appropriate for many applications. Given direct access to HDFS, applications could leverage the resources of the large cluster of servers that Hadoop ran on, in any way they chose. They were liberated from the MapReduce algorithm.

Hadoop as the “New Windows”

So new products, some of which had been in preparation awaiting the release of YARN, swiftly appeared. They were many and varied, including security products, ETL products, data cleansing tools, metadata discovery tools, data governance tools, performance monitoring and management tools, programming languages, new development environments, databases which used HDFS as their data store and more. And while some were wholly new products, others were well-established products from the data warehouse world that had been reengineered for Hadoop 2.0.

There are few precedents in the history of IT to explain Hadoop’s current trajectory. One which provides a telling comparison is the arc that Microsoft Windows followed as the PC market took off like a rocket. Windows overwhelmed its competition. It did so by the force of numbers leaving OS/2 and Apple trailing in the dust. Nevertheless in its early releases it was a fairly primitive OS with no security, limited multi-processing capability, no networking and not a particularly rich application ecosystem. So software authors in the PC market flocked to Windows. The deficiencies that Microsoft had failed to address were opportunities and there was an application vacuum waiting to be filled. Suddenly, Windows was bristling with innovative software.

Hadoop provides a clear parallel to Windows in several ways. It manages a processing environment, so it is a fledgling OS. Like Windows, on first release it had only a fairly primitive capability. It was completely deficient in security features. It was a bare bones environment without readily available applications, you had to build them. It was inexpensive to acquire. It ran on mass-produced commodity hardware. It was adopted first by developers, many of whom developed software products for the environment, enriching it and eliminating known weaknesses. It evolved with each new release.

Of course, there are also significant differences between Hadoop and Windows. Hadoop was the product of an open source project rather than a commercial venture. It focuses on enterprise computing rather than personal computing and, at its outset, it was more of a niche development environment than an OS. Nevertheless it has now acquired the kind of magical aura that Windows once had – it is clearly perceived as a "must have" environment that can deliver considerable business benefit.

Hadoop, the OS for Data

It has become clear that Hadoop is not just a "scale-out file system," as often described, but an OS for clusters or grids of computers: the OS for data. That, more than anything else, accounts for its prominence. Successful operating systems always spawn a powerful software ecosystem. The more the ecosystem prospers, the more compelling the OS becomes. That's what happened with MS-DOS, and then with Windows, and then with Linux. It even happened decades ago with OS/360 in the early days of the IBM Mainframe.

Prior to the introduction of YARN, Hadoop was not really an OS, but once the YARN scheduling capability was added, it clearly qualified as one, albeit a primitive one. At that time, a whole host of software vendors – Actian, SAS, WANdisco, Voltage, RedPoint Global and many others – rushed into the market. As far as we know, all had had products in preparation for quite a while. Some were even able to launch their shiny new offerings as soon as YARN was announced. It was then that the software ecosystem that was already fairly extensive, bristlingwith Apache Hadoop components and a smattering of commercial products, took off like a rocket.

Hadoop is, nevertheless, still in its infancy as an OS. If you browse through Table 1, which summarizes Hadoop's OS capability at the time of writing, it is clear that there is still a great deal of work to be done. What exists in respect to most OS functions can and will be enhanced.

Hadoop is a different species of OS, as its job is to coordinate multiple server environments each of which is running a local OS (in most instances Linux). Linux is a mature OS with proven capabilities in almost all OS functions. So our assessment of Hadoop, shown in the table, is not about the OS capabilities of individual servers, but about the ability to the whole clustered environment to perform in a coordinated manner.

OS Function	Capability
Booting (Initiation)	Yes
Application scheduling	Limited
CPU management	Minimal
Memory management	Minimal
Storage management	Minimal
File management	Yes
Full recovery from local failure	Yes (with 2.0)
Remote back-up	Limited
Cluster management	Limited
Global networking and distribution	None

Table 1. Hadoop's OS Capabilities

Rather than discuss these weaknesses here we will highlight them as we describe and explain the capabilities of WANdisco.

WANdisco and Hadoop

WANdisco is, pure and simple, a distribution platform. It coordinates multiple distributed software environments or applications, whether they are local (i.e., residing within a single data center) or remotely distributed (i.e., in multiple data centers in multiple locations).

The beating heart of the platform is WANdisco's distributed coordination engine (DConE). This is patented software which enables true unblocked peer-to-peer operation between distributed software environments or applications. DConE solves a genuinely hard problem, one which many software engineers may have believed insoluble.

To understand what DConE does you need to consider a situation where two or more servers are attempting to coordinate updates to data. The possibility naturally arises that two users could each access one of the two servers and attempt to change exactly the same item of data which is present on both servers. It would appear that there is no choice but for one server to attempt to lock down the data on the other server while it does the update and then ensure that the data item is updated on the other server before releasing the lock.

This sounds like it might be a simple thing until you consider what may go wrong. What happens if you lose connection between the servers in the middle of the process? What happens if, when you access the other server, the item is already locked? What happens if your server dies right after it places the lock? The situation is more complex than it seems.

The Travails of Data Replication

It would be pleasant if we never needed to replicate data, except to back it up in case of disaster. Then we could keep just a single record, a unique golden source, for every piece of data that mattered to us. If anyone wished to read the data or change the data, we would use whatever computer power was required to ensure that they could do so in a timely manner.

The impossibility of this dream derives partly from the inconveniently slow speed of light, the slightly slower speed of electricity and the need to obey various protocols in connecting software over large or even small networks. In reality, pretty much every connection and every process involved in getting the user to the data consumes precious time. And, of course, you may have thousands of users all clamouring for the data at the same time, creating a workload that no computer could manage. As a consequence, in many situations, some form of data replication is implemented.

Consider, for example, a retail operation with many outlets across a geographically large area. From the point of view of speed and workload, it is far better if the data that users want to access is local. If users only wish to view the data, this can normally be arranged with a little effort. All the data can be held centrally and copies of it replicated to local servers whenever it is updated. This is commonly done and it suits situations where user updates are rare. When an update is necessary it is done centrally and the data is replicated back to the local data store.

When updates to such data are frequent, problems arise. Software architectures that try to cater to this go by the name of "multi-master replication." They involve treating every data store as a master (hence the name multi-master) and trying to keep them in step by various mechanisms of data locking and conflict resolution. Such arrangements work up to a point, but they do not scale well at all. As the frequency of updates increases or the number of masters isincreased they reach a point where they slow down and grind to a halt.

WANdisco's Distributed Coordination Engine, DConE, implements a specific distribution algorithm that solves this problem entirely without scalability issues surfacing. Since the inability to perform efficient replication has been a constraint on IT applications for decades, there are reasons why we should be impressed. However, we are also entitled to ask "What difference does it make?"

The DConE Implemented on Hadoop

Because the problem of distributed replication is pervasive throughout distributed IT systems, there are many situations where WANdisco could have chosen to implement the DConE capability – if intelligently deployed it could and would make some difference to almost every distributed computing situation where there is contention for data. WANdisco chose

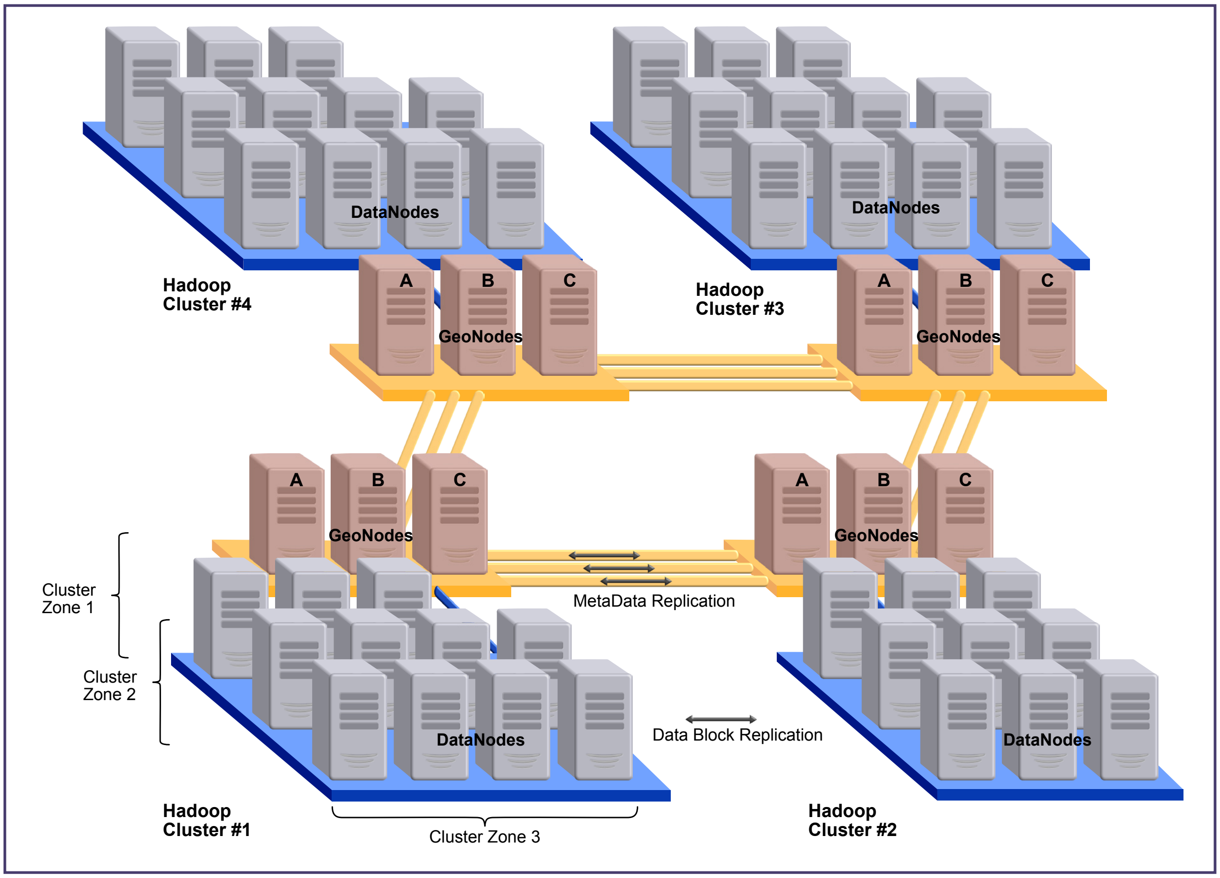

Figure 1. WANdisco, DConE and Hadoop

to implement on Hadoop because it could significantly improve the possibilities of Hadoop itself. WANdisco's architects, based on their experience building Hadoop from the earliest days, realized that extending Hadoop's data awareness from nodes and racks to clusters and data centers opens up new possibilities for scalability and reliability.

In practice, WANdisco delivers five specific kinds of capability. We use Figure 1 above to explain each of these in outline. This diagram shows four Hadoop clusters (possibly in four different locations) each running WANdisco. The blue and yellow platforms represent WANdisco running on each server. DConE is installed as a transparent plug-in to the NameNode servers at each location. These are labelled in the diagram as GeoNodes, because, with WANdisco running, they are NameNodes for a specific location/cluster within a network of peer-to-peer Hadoop clusters. Looked at simply, as the diagram indicates, WANdisco keeps the metadata in step and replicates data blocks between clusters as any new data or new files are added to any cluster.

Availability, Performance, Scalability

First of all, WANdisco will enhance the availability, performance and scalability of multiple Hadoop clusters that share workloads and data, addressing concerns that have surfaced over the past years as larger institutions evaluate Hadoop.

A normal Hadoop cluster spreads data across tens or hundreds of servers and uses a metadata manager (the NameNode) to coordinate access. WANdisco's DConE provides consistent access to that metadata across many clusters, possibly in many locations. The actual content is efficiently replicated in the background.

Imagine running a single Hadoop cluster and discovering that the cluster will not support the workloads that need to access the data it contains. This concern is real, not theoretical. Painful experience has shown that certain Hadoop applications can cause degradation of service if not isolated into separate clusters, while other applications need to run close to primary data sources.

With WANdisco it will be possible to build another cluster in the same data center and replicate the data to that cluster and thus share the workload between the two clusters, say, Cluster #1 and Cluster #2 (see Figure 1). If later it becomes apparent that the same data needs to be used at other data centers, then it will be possible to instantiate other clusters, say, Cluster #3 and Cluster #4, in the other data centers. WANdisco will keep everything in step.

This means that if workloads escalate beyond the capabilities of a specific instance of Hadoop,they can be scaled up by creating another instance or even several instances, either locally or geographically dispersed. If it is not a capacity issue (too many workloads), but a performance issue (the jobs are taking too long) the same applies.

Via WANdisco's data distribution and geographical dispersion, the highest levels of availability and bulletproof disaster recovery can be assured. Continuous hot backup that enables automated disaster recovery emerges as a side effect of replication, without any impact on performance. This is in sharp contrast to typical Hadoop backup and recovery solutions based on DistCp that prevent other applications from running during a backup. These solutions only make it practical to run backups outside of normal business hours, risking loss of any data generated since the last backup when downtime occurs. In fact, real-world experience shows that there is a risk of substantial data loss and lengthy service interruptions without WANdisco's capabilities.

Resource Efficiency

WANdisco removes the need for standby servers and clusters. It has an active/active architecture, in the sense that each Hadoop instance is active in respect to all other Hadoop instances it interacts with. As a consequence, all servers in every instance and at every location are readable and writable – there are no read-only servers.

Consider Figure 1. In a typical Hadoop environment, one of the remote clusters would be designated as a backup cluster. When using WANdisco's Nonstop Hadoop, all clusters are fully active and available for application use. The total number of required processing nodes can thus be spread across two or more clusters that provide data redundancy. Considering that a typical cluster costs over $1 million per petabyte of storage, WANdisco can deliver significant cost savings.

Virtual Clusters or Cluster Zoning

Cluster zoning divides a Hadoop cluster into virtual clusters. This WANdisco capability is particularly useful when you need to run a very mixed set of workloads on a Hadoop cluster, such as when batch analytics activity needs to run at the same time as some OLTP applications. Rather than introduce high-end servers into the cluster at considerable expense, or have one workload degrade the other, as would happen if YARN were used to schedule the two activities, WANdisco makes it possible to configure virtual clusters within the Hadoop cluster.

Figure 1 provides a simple example, showing a cluster of 11 servers that has been segmented into three cluster zones, with four, four and three servers. A workload running within one virtual cluster does not impact workloads running in any other virtual cluster. Peak demand for extra processing resources can be easily accommodated with a burst-out capability into a new virtual cluster. This is a particularly attractive data center capability because it enables a more efficient use of hardware resources and a much more creative approach to buying, deploying and repurposing servers.

The virtue of virtual clusters is born out by WANdisco customers, some of whom achieve a hardware cost reduction of over 35% in large deployments, while also maintaining high service levels. It is achieved by the combination of WANdisco's very high utilization of compute resources with its virtual cluster capability.

Multi-Data Center Ingest – Effective Replication

Replication has particular utility for distributed businesses, especially global businesses. The computer resources can be distributed close to the user population and data sources, improving response times for user updates and queries.

The advantage that WANdisco active/active architecture confers here is not confined to OLTP activity. It will work equally well with the batch ingest of data and the ingest of data streams, as may be required in "Internet of Things" and "The Industrial Internet" applications. The reality is that the data ingest workload, in whatever form it takes, can be distributed. It can even be spread across multiple virtual clusters in multiple locations. WANdisco automatically replicates the designated data between each instance of Hadoop.

It is difficult to overstate the importance of this capability. None of the traditional database products can do this. For decades, applications and systems have been built that were constrained in their ability to replicate data. By removing this awkward and often expensive limitation, WANdisco enables a large number of Hadoop use cases that would otherwise be impossible or at best slow and inefficient. And it can also be deployed to enhance systems that were previously hampered by the constraints of old replication technologies.

Global Hadoop Operation

Surprisingly, many Hadoop deployments start with several clusters at different locations, particularly when data locality laws are in play. That poses problems of security and efficiency if the data is to be treated as part of a logical data lake, rather than as silos of information.

WANdisco's selective data replication facilities allow information officers to centrally control where data resides. Data can also be made available on demand to conserve network bandwidth.

The Potential for WANdisco and Hadoop

We have explained why we believe Hadoop has become the OS for data. The circumstantial evidence supports our view. Like other operating systems before it, it has generated a swiftly expanding software ecosystem and with the advent of YARN it has become capable, to some degree, of managing processes.

It may not have started life as an operating system but it has become one – and unlike any OS that preceded it, it can span thousands of servers. It is a distributed OS by any reasonable definition. However, at the moment it suffers from clear limitations.

Currently, Hadoop has, with YARN, a very limited scheduling capability. Its ability to manage the resources under its control (CPUs, memory, I/O channels, spinning disk or SSD) is limited. Thus far its focus has been on scaling out rather than seeking to maximize the efficient usage of the resources under its control. While it needs no local backup capabilities within a cluster it is not easy to organize managed backup (replication) of data between Hadoop instances, and it has no disaster recovery capability at all. While it manages clusters of servers after a fashion, its capability here is minimal and it has no distributed capabilities for operations between Hadoop clusters, local or remote.

The beauty of WANdisco is that it resolves all of these problems, and in doing so, it transforms Hadoop into an OS that meets all the enterprise benchmarks for scalability and reliability.

Beyond the Horizon

There has been particular enthusiasm for Hadoop in specific application areas, notably:

- As the corporate staging area for data (the so-called data lake)
- As a working environment both for analytics and serving data via ETL
- As an archiving environment for data of any kind

Note however, that Hadoop embodies a versatile scale-out file system. It has been used as a database file system by a clutch of database products (not just HBase, but also Splice Machine, SQL on Hadoop from Actian, Impala from Cloudera and others). If HDFS can serve as a file system for database products it can no doubt serve as a file system for almost any application.

It has already been used for collaborative video editing, for 3D rendering and for many areas of high performance computing that are data-heavy. As far as we know, it has not yet been used much as a data store for email systems or as a general document store (like Microsoft SharePoint) or for digital asset management or OLTP business applications or as a games platform. Neither has it been used to manage virtual machines or for communications applications.

But these applications will inevitably appear. Hadoop is an OS and developers of every kind will eventually populate it with applications of every kind.

In our view WANdisco will likely have a role to play in this. It fills in many critical gaps in Hadoop's capability and allows the resources within the Hadoop environment to be managed in an extraordinarily versatile fashion. Organizations that are investigating Hadoop would be wise to also investigate WANdisco.

About The Bloor Group

The Bloor Group is a consulting, research and technology analysis firm that focuses on open research and the use of modern media to gather knowledge and disseminate it to IT users. Visit both www.TheBloorGroup.com and www.InsideAnalysis.com for more information.

Austin, TX 78720 | 512-524–3689