CITO Research

Advancing the craft of technology leadership

Five Questions to Ask Before Choosing a Hadoop Distribution

SPONSORED BY

CONTENTS

Introduction

Companies everywhere are excited about harnessing big data and putting it to work. This enthusiasm is understandable: Apache™ Hadoop® has removed all restrictions on how much data a business can store, process, and analyze. It offers the ability to integrate vast amounts of previously underutilized or ignored data in daily operations. However, in terms of enterprise requirements, it must be acknowledged that Hadoop is not a finished product. Businesses too often mistake open source with a low price tag. Without the proper support, adopting Hadoop on its own can end up costing companies more than expected.

Without the proper support, adopting Hadoop on its own can end up costing companies more than expected

To avoid surprises down the line, businesses should go into the Hadoop adoption process with eyes wide open. This CITO Research white paper outlines five key questions to ask before adopting a Hadoop distribution, with an eye toward preventing buyer's remorse.

① What does it take to make Hadoop enterprise-ready?

Apache Hadoop cannot be considered enterprise-ready as-is along several dimensions. Backup capabilities and security are two important areas where Hadoop is deficient on its own.

Backup

Enterprises need backup capabilities for their Hadoop clusters. When you roll new code into production, you need a backup in case something goes awry and you need to roll the system back. Apache Hadoop doesn't offer this capability. Especially when dealing with production systems, enterprises need the ability to roll back to a working version.

The replication process in the Hadoop Distributed File System (HDFS) offers protection from disk failure, but not from human errors. Further, if a file is corrupted, that corrupted file will be automatically replicated across the cluster, exacerbating the problem.

To avoid this type of exposure, businesses should consider a commercial Hadoop distribution with snapshot capabilities. The MapR Distribution fulfills this requirement. Users can take point-in-time snapshots of every file and table. If an error occurs, the snapshot can be restored.

Snapshots created via HDFS are not true snapshots. They do not stay the same size; they continue to grow as more data is copied into them.[1] MapR is able to support real-time snapshots due to its implementation of a POSIX-compliant file system with full random read-write capability. The importance of the read-write file system is explained in more detail later in this paper.

This feature of MapR has been important to its customers.

TechValidate

65% of surveyed IT organizations with experience with other Hadoop distributions rate **MapR data protection and disaster recovery** as significant differentiators

Survey of 48 MapR customers with experience with Apache Hadoop, Cloudera or Hortonworks, 10F-A82-224

Security

Hadoop is lacking in another vital area: security. As one blogger put it, Hadoop is a hacker honeypot. As we put more data into Hadoop, security must increase. The truth is that security is not usually a part of the initial design of an open source project; it's added after the project becomes successful. This holds true for Hadoop, Spark, and Storm. Furthermore, even when available, security features are sometimes not implemented because of concerns that they might slow down or interfere with the business purpose of the project. This type of thinking is shortsighted, particularly at a time when data breaches are eroding consumer confidence.

When securing Hadoop, enterprises want to use the authentication methods they support for other applications. At this time, Apache Hadoop offers only Kerberos authentication. Hadoop does not support Linux Pluggable Authentication Modules (PAM), but the MapR Distribution does, enabling agile authentication services for all applications. In fact, if an authorization scheme works with the Linux file system, it will nearly always work with the MapR Distribution.

Enterprises need multiple levels of encryption, both disk encryption and wire-level encryption to secure data traveling between nodes in the cluster. Apache Hadoop offers neither. The MapR Distribution supports native wire-level encryption implemented using public-private key pairs and offers disk encryption through partners such as Dataguise.

[1] For more information, watch a video demonstrating this phenomenon or read a detailed blog post on the topic of snapshots and mirroring.

② Does the distribution offer scalability, reliability, and performance?

Vanilla Hadoop has limitations in terms of scalability, reliability, and performance. Two factors contribute to these difficulties: Hadoop's reliance on NameNodes and its lack of a read-write file system. We cover these topics briefly to provide context for evaluating their impact on scalability, reliability, and performance.

The impact of NameNodes

In order to keep track of all the files stored within it, each Hadoop cluster must have a NameNode. The NameNode holds metadata about the information stored on all the DataNodes in the cluster. NameNodes create huge data access bottlenecks for businesses. The system running the NameNode requires a substantial amount of memory so that it can handle the entire file system table. Furthermore, NameNodes typically create a single point of failure (see Figure 1), in which a single job processing 10 million files could make an entire cluster inoperable.[2]

"Friends don't let friends run NameNodes."
-MapR

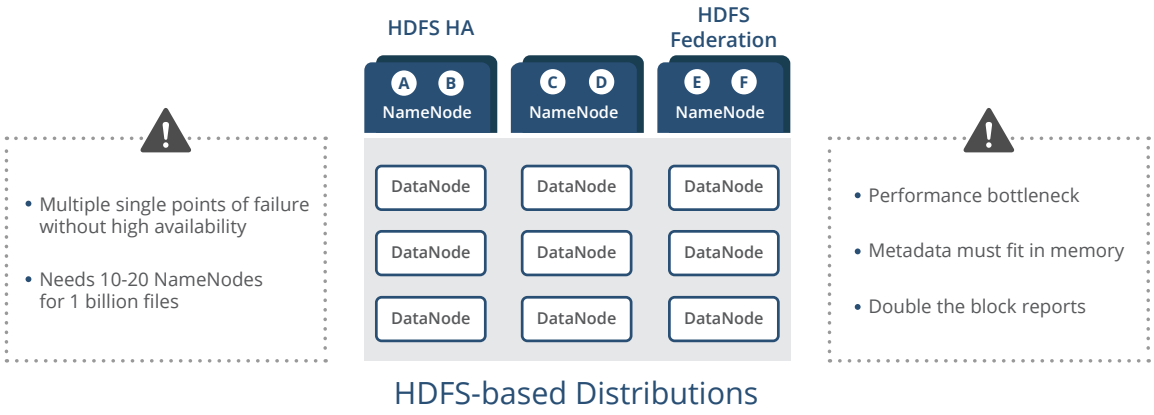

Figure 1: NameNode High Availability in Traditional Hadoop Distributions

The MapR architecture forgoes NameNodes, which improves reliability and scalability. The MapR distributed metadata architecture keeps precise file locations and identifying information because it distributes metadata across the entire cluster. This architecture ensures there is no single point of failure or bottleneck in the platform. The implications are significant (see Figure 2).

[2] By implementing the journaling NameNode concept, you can prevent the NameNode from being a single point of failure. However, it is very difficult to set up and requires additional hardware. With this feature, the NameNode is no longer a single point of failure. It does not mitigate the problems with the NameNode being a processing bottleneck, however.

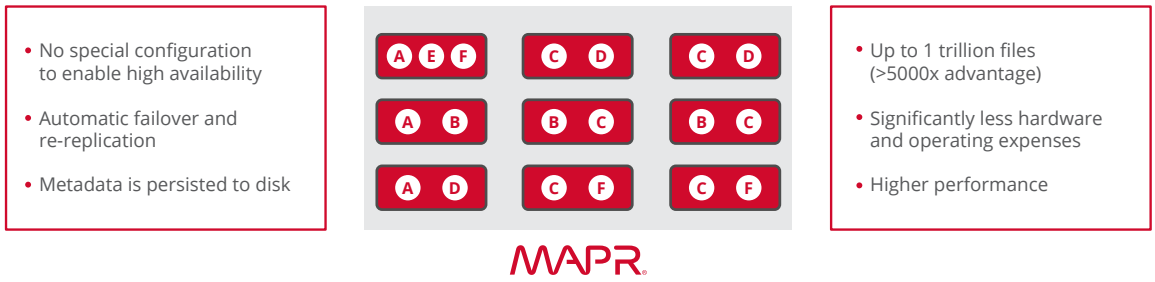

Figure 2: No-NameNode Architecture in the MapR Distribution for Hadoop

The lack of a random read-write file system

HDFS is an append-only file system. Enterprises need a file system that supports fully random reads and writes. An append-only file system forces numerous workarounds with significant implications. An append-only file system is a bit like using a typewriter instead of word processing software. If you need to make changes, you have to retype the whole page.

Because HDFS is an append-only file system, HBase compensates with features called tombstones and compactions. When updates come in, HBase creates a list of tombstones to say that a given entry in the main file is now "dead." After a while, HBase performs a compaction to handle all the updates. In so doing, HBase is working around the fact that HDFS is not a random read-write capable file system.

This is the reason why MapR created MapR-DB as an alternative to HBase (although HBase is also shipped and supported with the MapR Distribution). MapR-DB supports almost all the HBase APIs,[3] but does not need to perform tombstones and compactions because it has the ability to randomly seek and alter the records directly. This enables high performance (an average of 2 to 7 times faster than standard Apache HBase) and consistent low latency for operational applications.

Scalability

Because of Hadoop's ability to store vast amounts of data, many assume that Hadoop scales well. This is a faulty assumption.

The difficulty again lies with the reliance on NameNodes. Single clusters can only scale as large as their NameNode—a limiting factor. Larger clusters require adding more NameNodes. NameNodes are extremely restrictive from an engineering perspective, as companies have to ensure that the block size they have on tap is sufficient for the size of the cluster and the amount of memory in the NameNode. If a company has too many small files for its block

[3] There is one exception: MapR-DB doesn't support coprocessors. This is for security reasons. In an enterprise deployment, you do not want random engineers deploying code directly into the HBase engine.

size, because of the NameNode limitation, the entire system can come to a standstill. For instance, if Hadoop is configured with 2 GB blocks, but a user writes a 1 MB file, nearly the entire block is lost on a tiny file. Because of this problem, some companies will resort to workarounds in which they either compact small files or create elaborate sequencing models in order to better use their blocks. But while this solves one problem, it creates another, as it requires a great deal of engineering experience, time, and resources. Every custom file format that is written must also be able to be read, which places an additional burden on any resource that wants to perform data analysis.

The alternative is using a distribution like MapR in which the block size limitation is eradicated. Hadoop is limited to between 50 and 100 million total files; with the MapR Data Platform, companies can access trillions of files.

Reliability

Reliability is integral to successful scaling. And reliability is predicated on constant availability and efficiency. MapR, utilizing the read-write file system and therefore not having to fallback on compactions or layers of code, is able to achieve high availability and low latency. With MapR, bigger is better. With HBase, bigger is significantly worse, as an ill-timed compaction can essentially shut down an entire cluster. And even if it doesn't go down entirely, HBase can become so bogged down from compactions that the system is basically out of commission. Companies cannot afford these types of outages or slowdowns, especially if they're relying on the data to perform real-time analytics to inform decision-making.

Performance

The MapR Distribution increases efficiency by allowing companies to achieve high levels of performance with smaller amounts of hardware than if they had stand-alone Hadoop. In fact, MapR broke a record for the MinuteSort benchmark, processing 1.65 terabytes of data in one minute on a 298-node cluster, easily besting the previous record of 1.6 terabytes on 2200 nodes. This demonstrates not only performance, but also efficiency, discussed in the next section.

Fastest Hadoop on the market

TechValidate

77% of surveyed IT organizations that had experience with another Hadoop distribution selected the MapR Distribution including Hadoop because of **better performance**.

Source: Survey of 52 MapR customers with experience with Apache Hadoop, Cloudera or Hortonworks, TVID: 0D7-262-5C0

③ Is the distribution efficient when it comes to TCO and ease of administration?

Many companies assume that as an open source product, Hadoop is free, or at least significantly cheaper than alternatives. This is not true if you look at its total cost of ownership (TCO). MapR requires far less hardware than other distributions of Apache Hadoop. Higher performance on less hardware means that more can be accomplished with a single cluster, again supporting a lower TCO. For an example, see Table 1.

Table 1. MapR is 5 times more efficient

	Number of jobs per day	Number of nodes	Average jobs per node
Yahoo Y!Grid	1 million	32,500	About 31
Leading Ad Tech MapR customer	65,000	400	162

Doing more with less hardware means reduced hardware, administration, and power costs. One media company was able to consolidate eight HBase clusters down to a single cluster running MapR.

Scale Infinitely

TechValidate

"MapR-DB requires about half the machines compared to other [NoSQL] platforms. This dramatically reduces the cost of a new system."

Source: Atzmon Hen-Tov, VP R&D, PONTIS, TVID: 5CC-B3A-A34

Adopting a solution like MapR helps companies drive down their Hadoop TCO. With MapR-DB, companies can perform operations and analytics on the same platform, minimizing the impact on production. Users can take a snapshot of a table and then immediately perform analytics on it, something that is impossible to do in HBase.

MapR-DB is a zero administration database. This means that it automatically splits regions when it needs to with zero impact on the platform when it does so.

MapR supports volumes, which make large amounts of data easier to manage. Administrators can specify where particular volumes should be backed up and when. Essentially, this means you need fewer people to manage an equivalent size MapR cluster as compared with Apache Hadoop.

TechValidate

"The ability to leverage Volumes gave us a method to reduce large shuffles and allowed us to take a daily process from 36 hours of run time to 3 hours—with no additional hardware required."

Source: Mike Brown, Chief Technology Officer, comScore, Inc., TVID: F47-395-8A7

"When I joined LinkSmart in 2012, it took 24-28 hours to process reports. I rewrote the reporting platform using MapR at its core and was able to process the same reports in 20 minutes."

Source: Manny Puentes, Chief Technology Officer, LinkSmart, TVID: FFD-95E-273

④ How flexible and open is the distribution?

Too often, companies looking to adopt Hadoop mistakenly believe that using a commercial distribution will force them to forego Hadoop's open source benefits. Companies fear that as soon as they implement a distribution, they will be locked in.

MapR supports all of the same APIs as standalone Hadoop. Put in simple terms, MapR is the Hadoop API. Users can download open source software and then run it on MapR without recompiling it. MapR supports multiple versions of many Hadoop components (other distributions support only a particular version of Hive, Spark, and Flume) with each release of its platform. MapR ensures all of these products work together on the same platform. This has significant benefits when companies perform rolling upgrades; one component can be upgraded independently of the others. Decisions about what components to use are business decisions, not the decision of the Hadoop vendor. Rather than dictating which tools and which versions it supports, MapR allows businesses to decide for themselves.

TechValidate

Customers Choose MapR for High Availability

81% of surveyed IT organizations that had experience with another Hadoop distribution selected MapR because of **high availability features** (e.g., no-NameNode architecture, rolling upgrades).

Source: Survey of 52 MapR customers with experience with Apache Hadoop, Cloudera or Hortonworks, TVID: CEB-82E-5D5

This extends to data lock-in as well. With MapR, there is no "special" data format. Files are files.

For companies who are using Hadoop and want to begin running Apache YARN, MapR provides a seamless upgrade path. With the MapR pluggable architecture, either MapReduce version 1 or YARN (AKA MapReduce version 2) applications can be run simultaneously in a cluster. With MapR, companies are enabled to migrate at their own pace. MapReduce v1 jobs can be migrated to benefit from YARN when they are ready, instead of being forced into a schedule on other platforms. With other Hadoop distributions, some users have required a complete shutdown of their entire cluster during this upgrade.

Many applications and development tools that business users have in place work in conjunction with NFS. MapR supports full NFS with a read-write file system, which enables those tools to be used with Hadoop by virtue of this NFS compatibility. NFS compatibility is such an integral part of integration into the business that MapR now offers this feature in its free community edition, and no other Hadoop distribution is capable of this.

TechValidate

"I couldn't live without MapR-DB tables – no compaction, no region servers. I also depend on NFS access every day."

Source: Engineer, Medium Enterprise Financial Services Company, TVID: B42-9F4-303

⑤ What additional workforce expertise does a company need to run Hadoop?

Upon adopting Hadoop, many companies quickly realize that they simply do not have the in-house expertise to make it run smoothly, let alone enterprise-ready. To run Hadoop without support, companies need an administrative staff that understands block size configurations, knows what to do if they lose a NameNode, and comprehends the ins and outs of HBase and its delicate dance with HDFS.

The MapR Distribution greatly simplifies Hadoop, improving scalability and performance in the process and freeing administrators from many low-level details.

Conclusion

Potential customers should thoroughly educate themselves about Hadoop's capabilities and limitations before selecting a distribution and understand what they'll need to do to make Hadoop enterprise-grade.

CITO Research recommends that businesses try out the MapR Distribution. Because there is a free version of the platform, if companies decide they no longer want to pay for MapR enterprise features, they can fall back to the free version.

Customers that don't implement an enterprise-grade Hadoop distribution are essentially reconciling themselves to getting less out of Hadoop than out of other IT systems. There's no reason to make big data the exception.

Using contributions from companies like MapR does not mean a business is negating the value of open source. In fact, MapR is a leader in several open source projects such as Apache Mahout and Apache Drill, and is a contributor to many others including HBase, Hive, Zookeeper, MapReduce, Parquet, and YARN. The capabilities in MapR result in higher performance and reliability for all the Hadoop projects that run against the HDFS API. In truth, using MapR means the business will have a Hadoop distribution that easily integrates with the rest of the IT infrastructure—one that is scalable and lives up to enterprise standards. MapR makes Hadoop the best version of itself.

How MapR can increase productivity

A leading international clothing retailer adopted Hadoop to help handle the massive amounts of data it was collecting on its customer base. The business soon ran into issues as its existing workforce lacked the skills to make Hadoop fully operational. Users were experiencing constant downtime during NameNode bottlenecks and upgrades. Many analysts opted not to use Hadoop because of these challenges, meaning that all the data in Hadoop was going to waste.

The company opted to implement MapR and the improvements were immediate. By essentially outsourcing the process of making Hadoop enterprise-quality, the retailer was able to get its workforce focused on its core competencies of tracking market trends and getting the right products into stores, rather than trying to configure Hadoop. Prior to adopting MapR, the company estimated it was using less than 4% of its big data; it is now able to use (or discard) over 75% of it. Users are satisfied that they're making data-driven decisions and campaigns can be much more accurately tracked.

Related Resources

Download the Hadoop Buyer's Guide for insights on how to get started with Hadoop

Get started with the free MapR Sandbox for Hadoop, including point-and-click tutorials

This paper was created by CITO Research and sponsored by MapR

CITO Research

CITO Research is a source of news, analysis, research and knowledge for CIOs, CTOs and other IT and business professionals. CITO Research engages in a dialogue with its audience to capture technology trends that are harvested, analyzed and communicated in a sophisticated way to help practitioners solve difficult business problems.

Visit us at http://www.citoresearch.com