Integration developer news | Architecting the Intelligent Enterprise

Making Hadoop Enterprise-Grade

by Jack Norris, Vice President of Marketing, MapR Technologies

There is an increasing focus on Big Data applications across many G2000 IT stakeholders - by architects, developers and IT operations But what would it take to make Hadoop enterprise-grade? For that matter, what does "enterprise-grade" really mean? In this article, an exec from MapR explores Hadoop's data availability, data protection, failover and performance.

What would it take to make Hadoop enterprise-grade? What does "enterprise-grade" really mean?

The growing number of organizations using Hadoop have found it to be an indispensible solution capable of unlocking the value of disparate data sources to improve decision-making and gain a competitive edge. Indeed, Hadoop's many advantages have given rise to an entire ecosystem, which now includes commercial distributions, as well as cloud-based offerings from Amazon, Google and other service providers. Hadoop users have also discovered, however, that most Hadoop distributions have some serious limitations involving data availability and data protection.

However, for enterprises-class adoption, Hadoop distros need to address concerns IT have over its single points of failure and their impact on data availability.

For example, Hadoop was designed with a centralized NameNode architecture where all Hadoop Distributed File System (HDFS) metadata (e.g. namespace, block locations, etc.) are stored in memory on a single node. To enable recovery from a NameNode failure, Hadoop employs a Checkpoint Node (previously called the Secondary NameNode) and a separate Backup Node.

There are two reasons why this approach fails to deliver true enterprise-grade high availability:

- Even when properly implemented, the configuration affords full recovery only from a single failure.
- The recovery itself is not hitless. Any failure, therefore, causes a major disruption, and often results in the need to restart MapReduce jobs, where a similar single point of failure exists in the JobTracker.

One work-around to minimize the disruption is to federate the NameNode by sharding the file metadata across multiple Primary NameNodes. These NameNodes are not redundant, however,

so in effect they create multiple single points of failure. The only advantage is that instead of losing all the data in a cluster, a failure only impacts the data managed by the particular NameNode.

But, an enterprise-grade alternative to the above approach is now available in a select few Hadoop distributions. These newer options will become more crucial as enterprises further articulate their need to assure data availability There are some emerging HA capabilities in the Apache project to provide failover in versions. Further, some commercial versions (such as MapR), support this capability today.

Some of the differences lie in the method of addressing HA (automation, self-healing, etc.) as well as the extent of data protection available. For example, where there are snapshots to provide a rollback point.

For those looking to add HA to their current Hadoop projects, there's more good news. All distributions support the same APIs so no programs need to be rewritten or recompiled, they will run against the upgraded distribution. The migration process is a simple process of freeing up space in the cluster by decreasing the replication factor and performing a rolling upgrade. Further, snapshots and mirroring have no performance impact if the implementation uses a redirect on write. In fact, The method of distributed HA actually increases performance and scale.

The figure below shows a hierarchy of enterprise-grade characteristics that apply to any application. Apache Hadoop offers only rudimentary "high availability" and batch-oriented data replication, as indicated by the grey shading. The additional enterprise-grade capabilities, shown in blue, are now becoming available in some commercial distributions.

The Hierarchy of Enterprise-grade Capabilities

Inside Hadoop's HA-Optimized Architecture for Data Availability, Protection, Recovery

To anticipate this growing list of enterprise requirements, these new Hadoop distros afford automated stateful failover with self-healing by distributing the NameNode function across multiple nodes. Further, the distributed file metadata automatically persists to disk (as with the node's data), and can also be replicated continuously to two or more other nodes to provide hitless self-healing from multiple simultaneous failures. This approach also eliminates the need for separate and dedicated Checkpoint and Backup Nodes, making high data availability achievable without extraordinary effort.

In addition to unplanned downtime, data availability is impacted by planned upgrades and other changes routinely required in the Hadoop software or its configuration in a cluster. With most Hadoop distributions, it is often necessary to make these changes concurrently throughout the entire cluster. The ability instead to make these routine changes to individual nodes or groups of nodes makes for a more manageable and less disruptive rolling upgrade process.

To provide some means of data protection, Hadoop supports basic replication of files among nodes. But such basic data replication does not protect against application or user errors. The corrupted data is simply replicated across the cluster with no ability for roll back or recovery. And HDFS lacks any ability to perform snapshots or to mirror data.

Enterprise-grade data protection requires meeting both recovery point and recovery time objectives through snapshots and mirroring, making it easy to recover from any data issues. Snapshots create regular recovery points and enable users to easily recover volumes or files. Mirroring and wide area replication extend data protection to satisfy recovery time objectives. Local mirroring provides high performance for highly-accessed data, while remote mirroring provides business continuity across multiple data centers, as well as more seamless integration between on-premise and private clouds.

As indicated above, the fundamental hurdle to enterprise-grade capabilities in both the open source and most commercial distributions is the underlying Hadoop Distributed File System. The two biggest limitations with HDFS are its lack of random read/write file access by multiple users or processes, and the requirement for managing and moving all data in batches. These fundamental limitations make it impossible to fully protect all data and provide high data availability.

These limitations inherent to HDFS can be overcome by re-architecting the storage services layer to provide direct access to data via the industry-standard Network File System (NFS) protocol. In addition to making Hadoop more enterprise-grade, direct access via NFS helps make Hadoop more enterprise-friendly by making it easier to integrate into virtually any IT infrastructure.

Any client can simply mount the Hadoop cluster, and application servers can then write data and log files directly into the cluster, rather than writing first to direct- or network-attached storage. Existing applications, utilities, development environments, text editors and other tools

can also use standard NFS to access the Hadoop cluster to manipulate data, and optionally take advantage of the MapReduce framework for parallel processing.

Constant exposures to data loss, inevitable and potentially lengthy disruptions, labor-intensive manual recovery efforts, and cumbersome work-arounds hardly constitute what should be considered "enterprise-grade" data availability and protection. Yet this is the case with most distributions of Hadoop today. Like any other environment, though, Hadoop is maturing and some commercial distributions are now available that do afford the high availability, automated failover, rolling upgrades, and data replication, snapshots and mirroring needed to be worthy of the label enterprise-grade.

Enterprise data architects should become aware of their options. Organizations can support their enterprise needs today with commercial options available today. As the Hadoop ecosystem is a robust community that continues to evolve there will be even more capabilities and options to address Big Data and drive results.

About the Author

Jack Norris is vice president of marketing at MapR Technologies, and brings more than 20 years experience with defining new markets for small companies and increasing sales for larger public companies. He has held posts at Aster Data, Rainfinity (EMC), Brio Technology and SQRIBE.