The cloud architect's guide to DistCp

The good, the bad, and the risky

Why effective data migration matters

"There's no free lunch" applies to many things in life, and migrating from on-premises infrastructure to the cloud is no exception. Yet many cloud architects and Hadoop administrators rely heavily on free tools such as DistCp when seeking to modernize their data, analytics, and infrastructure capabilities when migrating to the cloud.

We see the effects of this on a weekly basis at Cirata. Many of our most passionate customers are those who initially anchored their cloud migration plans around free DistCp solutions and commercial products that use these tools. These cloud teams quickly ran headlong into the hidden costs, delays, and unanticipated business disruptions that tend to come with using free tools not built for modern data strategies.

After grappling with such hidden costs and challenges, these cloud leaders sought an automated solution that took advantage of the cloud capabilities they were seeking and allowed them to access their data faster while reducing unforeseen risk and costs, both real and reputational, lurking in manual migration tools.

We want enterprises to benefit from the freedom of data to reside wherever and whenever it is needed so their technology runs faster and more efficiently, all while enabling better business decisions.

This starts by not getting bogged down in complex manual data migrations that add costs, time, and risks to your cloud migration program.

So we created this guide to accelerate the transition to a cloud-based future by lessening cloud architects' learning curve around the pros and cons of manual and automated data migration options. Hopefully it fills key information gaps and helps you migrate to the cloud faster by making better technology decisions.

Cloud architect executive summary

This guide is designed to fill information and analytical gaps that cloud architects typically face when deciding how to migrate data from on-premises environments to cloud platforms such as AWS, Google, and Azure, as well as cloud-based data and analytics platforms such as Snowflake and Databricks. Many architects grapple with understanding if the free tools, and manual data migration approaches typically recommended by Hadoop administrators, are up to the task of building a modern cloud infrastructure that enables data-driven businesses.

DistCp in a nutshell

Distributed copy (DistCp) is a free tool that comes bundled with Hadoop that is used for large inter/intra-cluster copying of data. It uses MapReduce to execute its distribution, error handling and recovery, and reporting.

Why use DistCp?

DistCp is a viable solution when the data does not change frequently and when there is a low volume of data that needs to be copied between Hadoop clusters.

Drawbacks and limitations of DistCp

DistCp is not built for migrating large datasets from on-premises to the cloud, where the data is likely to change during the migration and where application developers need access to the data during migration.

The bottom line

Many firms that are initially drawn both to the familiarity of having used DistCp for intra-cluster data movement and the fact that it's free will very often incur hidden costs, project delays, and some form of business disruption due to the mismatch between the old, on-premises backup use case that DistCp was built for and the new, cloud-based model that cloud architects are moving toward.

Automated solutions allow migrations to occur while production data continues to change, meaning that businesses can perform migrations with no system downtime or business disruption.

Given the benefits of purpose-built, automated solutions, it pays to take one for a test drive if it eliminates risk, increases the likelihood of hitting the migration deadline, and frees up valuable resources for value-added work.

How do we migrate petabytes of business critical customer data to the cloud without causing business disruption while ensuring data consistency across geographically dispersed environments?

The need for streamlined data migration

Migrating from on-premises data storage and infrastructure to a cloud based environment is no small feat, but the payoff is very real. A succinct and poignant example of the drivers for migrating from on-premises to cloud comes from data scientist Ben Podgursky, formerly the data infrastructure lead at LiveRamp:

"While our HDFS cluster was actually a very well-oiled machine by the time we started this migration, it was stressful to maintain and upgrade without downtime or interruption. As our company grew, the downtime windows we could negotiate with the product team got shorter and shorter, until upgrading became functionally impossible. We knew we wanted to use GCS (Google Cloud Service) so we could stay nimble as a dev team."

The ability to get up and running quickly on platforms such as Snowflake and Databricks and execute application workloads on cloud platforms such as Azure, AWS, and Google translate into very real business benefits. But getting to that nimble state from on-premises clusters to cloud services using traditional approaches requires careful planning, analysis, tool selection, dedicated resources, and specialized skills.

Dawit Alemu, Technical Architecture Lead for the Analytics Center of Excellence at Johnson Controls summarized the promise, and peril, of migrating from on-premises Hadoop clusters to the cloud: "We were intrigued by analytics services on Azure and wanted to use them for making data-driven decisions more effectively. But migrating Hadoop data lakes is a problem we have had for some time."

It's at this point that many cloud teams turn to DistCp.

> **"We were intrigued by analytics services on Azure and wanted to use them for making data-driven decisions more effectively. But migrating Hadoop data lakes is a problem we have had for some time."**
>
> ***– Dawit Alemu, Technical Architecture Lead, Johnson Controls***

DistCp in a nutshell

Simply put, and as defined by the Apache software foundation, "DistCp (distributed copy) is a tool used for large inter/intra-cluster copying. It uses MapReduce to affect its distribution, error handling and recovery, and reporting." Here is how it is typically invoked:

```
bash$ hadoop distcp hdfs://
nn1:8020/foo/bar \hdfs://
nn2:8020/bar/foo
```

In theory, using DistCp is as simple as a Hadoop administrator opening a command prompt window, issuing a few commands, and then copying data from cluster A to cluster B.

Some providers have created user interfaces that open a webpage so administrators can simply select the data they want to replicate, build the schedule they want, and even exclude any sub-directories or specific content types they don't want to replicate. There are even options that if something gets written at a target, the admin can overwrite it and delete it.

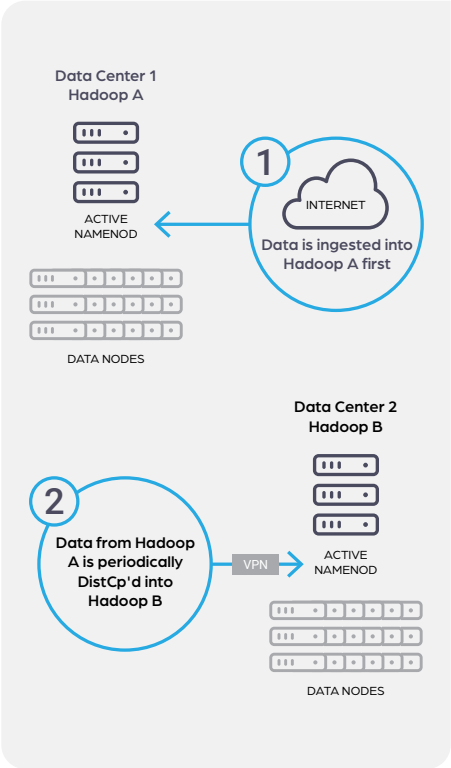
Many Hadoop administrators and cloud architects know of DistCp as the free software tool that comes with their Hadoop environments and is often used in backup and recovery use cases. In Hortonworks, it was recently referred to as Data Plane. Cloudera has BDR and Replication Manager and also uses DistCp under the covers.

These core use cases around backup, disaster recovery, and copying files between on-premises clusters is an important starting point because it outlines the original requirements for using DistCp, the environments it works well in, and the contexts where its applicability injects risks and the potential for business disruption.

In a nutshell, DistCp:

- Supports Hadoop-compatible file systems
- Transfers data at a single point in time
- Transfers data only after a file or object has been created in full at the destination source

Many cloud vendors provide data migration tools that require users to take clusters off-line during backups and data movement. Frank Cohen, CEO of Clever Moe and Founder of Appvance, noted that "using DistCp and hardware migration solutions (like AWS Snowball) require data to be offline for days to weeks."



Expert insight

The challenges with manual migration solutions

by Infosys Senior Architect Mrinal Kumar

Using manual migration solutions is tough. These approaches include native Hadoop utilities like DistCp, ETL tools, or the appliance-based migration approaches where the cloud vendor sends you hard disks to copy all your data and then those are sent back and copied to the cloud. All of these approaches have issues.

DistCp doesn't report on errors, and it's really difficult to track where an error has occurred during the migration phase. Handling the delta – or changing data – is definitely a big challenge. All of these approaches have a significant handicap in that they don't track or handle data changes. The burden of tracking and handling changing data then falls on the migration team as part of the project implementation.

Tracking the different streams of copies that are running through manual tools is complicated. If there is a failure during the migration, tools like DistCp won't even tell you that a failure has occurred, and you'll realize it late in the game once you are in the validation phase.

"If there is a failure during the migration, tools like DistCp won't even tell you that a failure has occurred, and you'll realize it late in the game once you are in the validation phase."

– Mrinal Kumar,
Senior Architect, Infosys

How is DistCp used?

Since the growth of cloud computing and the resulting need for data migration from on-premises to cloud environments, many admins have taken to using DistCp for copying files from on-premises clusters to their object stores in a cloud instance because it's free and they are familiar with it from executing disaster recovery work.

Hadoop admin usage of the basic DistCp command line is relatively straightforward. But IT, data, and cloud migration projects are rarely straightforward, and when administrators start getting into schedules, exceptions, and similar realities of data migration, a simple use of DistCp gets very complicated very quickly.

An example from a Fortune 500 insurer makes this clear:

Case Study

How a leading insurance company lost 9 months trying to use DistCp

A Fortune 500 property and casualty insurer sought to use DistCp for migrating data from their on-premises Hadoop clusters to Amazon S3 target object stores. The company's Hadoop administrators and consultants wrote custom scripts for DistCp, but instead of targeting a HDFS cluster as is the original use case for DistCp, the target was an Amazon S3 bucket.

The infrastructure team quickly ran into problems because they had to keep doing their own schedules and writing and rewriting their own scripts to modify how DistCp worked with the target Amazon S3 cluster. When migration of a certain block of data failed, it was very difficult for the team to figure out what went wrong or where it went wrong. The IT team had to invest a significant amount of time just to figure out what went wrong. The team ended up being 9 months behind schedule trying to deal with the complexity and challenges of using DistCp to migrate data to a cloud target environment.

The team ended up being 9 months behind schedule trying to deal with the complexity and challenges of using DistCp to migrate data to a cloud target environment.

Where it makes sense to use DistCp

Backing up data and predictable recovery points

DistCp provides a predictable recovery time and recovery point, in the sense that if an admin is looking for a recovery time or recovery point of six hours or more, then DistCp can provide that predictability. Disaster recovery was one of the original use cases for DistCp. However, DistCp recovers from a point in time that may not be in sync with the business needs.

Low data volume

When data volumes are relatively small – say in the <100TB range – then DistCp can be an acceptable solution, assuming other limitations such as rate of data change and accessibility of data are accounted for. There is no hard and fast rule here, but if you've got 20–30TB of data, then you can probably find a way to migrate it with DistCp.

Minimal data change during migration

If you only have historical data to migrate, then DistCp could work fine. If the data sets to be moved are not undergoing rapid change during migration (roughly defined as <50–100 events per second), then DistCp may be a suitable and low-cost data migration technology. However, DistCp starts to breakdown quickly as the change event volume increases.

Avoid common data migration traps

Starting with DistCp as the go-to solution because it is familiar leads many cloud architects to fall into one of a handful of data migration traps, including common ones such as:

1. Assuming that approaches that work at a small scale will work for big data
2. Missing updates that are made to data at the source after the migration has begun
3. Programming, maintaining, and managing in-house custom, big data migration scripts

To avoid these traps and execute a successful migration so the business can benefit from cloud-based services sooner, it's important to understand the hidden costs and risks in terms of complexity and business disruption when it comes to manual migration strategies using DistCp.

Limitations of DistCp

At its heart, DistCp was designed for copying files between clusters and not for large-scale migrations of actively-changing "hot" data from on-premises clusters to the cloud. DistCp is script-driven and requires a batch-mode operating model that relies on "big-bang" migrations. DistCp migrations contribute to one of the biggest pain points for cloud architects and their stakeholders: application developers and data scientists need to wait until "after the bang" to start working with applications to deliver business value.

The other challenge of a migration that uses DistCp is that there is no way to guarantee data consistency if data is actively changing during that big bang migration. Either production clusters need to be taken offline during migration, or repeated scans need to be performed, which introduces significant business disruption and resource requirements.

DistCp limitations at a glance

- It is labor-intensive to manually reconcile differences as a result of data changes made since the last DistCp run. This results in higher costs and often leads to delayed and failed projects.
- Multiple scans are required to capture ongoing data changes made between DistCp runs. Depending on the size of the dataset and amount of changes occuring, it may be impossible to ever catch up with all changes.
- DistCp runs as a standard MapReduce job competing for resources with other processes and requires you to have open firewalls across all nodes in the cluster, posing security issues.

DistCp migrations introduce risk and business disruption. application developers and data scientists need to wait until "after the bang" to start working with applications to deliver business value.

Implications of scaling DistCp from 500tb to 2.5pb for data migration

Data context

One client wanted to initially migrate 500TB (out of 2.5 PB) of data representing over 8.6 million files from their Apache Hadoop 2.8 cluster in their own datacenter with 800 nodes to Amazon S3. File system events–of–interest–per–second (the rate of change) averaged roughly 50 events per second (though they spiked to 10 times that during peaks.)

"Traditional" DistCp approach

A traditional migration approach using DistCp that does not take changes into account during operation would need to attempt to determine what differs between the source Hadoop cluster and the target Amazon S3 bucket by comparing each file/object and transferring any differences after an initial transfer has occurred. Migrating 500TB of unchanging data at 3 Gb/s transfer rate will take 15.4 days.

But datasets will continue to change during that 15–day window, and things look very different when you start to scale the migration.

Traditional approaches to data migration don't scale well

If there is no information available about exactly what data have changed, the source and target need to be compared. That comparison needs to touch every file (8.6 million), and if performed in sequence, will take 55 hours. But during that time, more data will continue to change, and without a mechanism to capture those specific changes, the team would need to return again to the data and compare it in total again.

Using the 50 events per second change rate, during that 55 hours of change scans, the client team could expect roughly 10.1 million changes to files, and about 2 TB of data to have been modified.

Changes increase 42,776% at scale

The change rate climbs exponentially as the data volume to be migrated increases: what was 50 events per second for 500 TB of data became 21,388 events per second for 2.5 PB of data—a 42,776% increase!

The comparison of all data will need to touch 5 times as many files, now 43 million, and if performed in sequence, will take about 10 days, even before transferring any changed data. At the change rates for our example, this would be another 10TB of data, which itself will take more than 7 hours.

And on and on the process goes, tying up valuable IT resources to ensure changed data is brought over.

DistCp warning signs

The previous example is already being felt by data and infrastructure teams. In the 2021 Hadoop Data Migration Benchmark Report, we asked Hadoop administrators and cloud architects to name their top drivers of data migration costs, and their answers map closely to how they would handle the above scenario:

1. Complexities of handling on-premises data changes during migration
2. IT resources required to perform and manage the data migration
3. Custom code development and maintenance of data migration scripts/programs

DistCp migrations introduce risk and business disruption

"Big bang," DistCp-based migrations rely on taking a snapshot of the data at a point in time. If you're using a distributed copy approach to move huge quantities of vastly-changing data, you're signing up for a long-term game of Whack-A-Mole trying to locate the data changes and ensure the target environment is up to date.

Distributed copy jobs are "lumpy," meaning that you're pushing the data out in one large batch at a time, and that is impacting not only the memory and the processing resources of the Hadoop cluster, but also the other applications and workloads in that data center. The email systems, web servers, etc. in that data center are going to be negatively impacted just by the simple act of running these jobs.

DistCp migrations burden network and IT resources

MapReduce requires processing from every node in the cluster when it's running. Whether it's doing analytics that a business is deriving value from or copying data from a cluster to another using distributed copy, it's using every processor from all of the nodes and it's using memory from all the nodes. And that can consume up to 15% or 25% of the resources of the cluster. That means running DistCp jobs will negatively impact the applications the business is actually trying to derive value from.

DistCp migrations impact firewall configurations

Also, there may be firewalls between the Hadoop clusters that govern the flow of traffic, and each one of those lines would represent an individual firewall rule because every data node in the source data node needs to have a network connection to every data node in the target. This is a function of HDFS block replication. You won't know where the blocks are going to be when you need to replicate them, so you'll need to open up all the possible paths. Imagine being the firewall administrator that has to set all of that up... It's quite time intensive and not conducive to getting up and running quickly.

Is DistCp right for you?

Use this checklist to see if DistCp is the right solution based on the pros and cons outlined above. If you answered no to any of the questions, you'll need to develop customer scripts, manage increased resource workloads, wait for data to be available after the migration is complete, and rescan the data to identify and bring over data changes.

Checklist for DistCp suitability

Technical Environment for Migration	Is DistCp a Viable Option?	
	Yes	No
We have less than 100TB of data to migrate.	☐	☐
Our data is not "hot" and the change rate is likely to be <50–100 events per second.	☐	☐
We are ok if the data that is migrated is not updated in real time.	☐	☐
We can absorb significant increases in cluster utilization during the migration.	☐	☐

Automated solutions are purpose-built for streamlined data migration to the cloud

A better way: the role of automated solutions in a modern data strategy

If you answered "No" to any of the above decision criteria, then automated solutions are worth exploring. Automated solutions, such as Cirata's Data Migrator, allow migrations to occur while production data continues to change, meaning that businesses can perform migrations with no system downtime and no business disruption, and can easily handle data changes that happen during migration. Automated solutions are designed specifically for data migration and allow further configurability, such as network bandwidth usage, so as not to impact current production activity.

Automated solutions have a number of distinct advantages over manual migration solutions such as DistCp:

- Confidently hit migration deadlines by eliminating manual workarounds and knowing exactly how much time the migration will take.
- Extract business value from the data sooner because processes happen in the background without having to do a point-in-time migration.
- Focus on making sure the applications that are running in the cloud do what they need to do.

The path forward

While there are specific use cases that lend themselves to using DistCp, such as low volume datasets that undergo minimal change during migration, more and more companies require a more modern solution that accounts for larger volumes of live or "hot" data that needs to be available immediately in the cloud for technical users.

The labor-intensive nature of adapting DistCp to these modern data architectures and cloud-based strategies means using DistCp-required custom script development, manual migration management, and complex reconciliation of changes, which tend to add hidden costs and effort to the total migration project.

Given the benefits of purpose-built, automated solutions, why wouldn't you take one for a test drive?

There may be no such thing as a free lunch, but a free trial of an automated solution may be even better if it eliminates risk, increases the likelihood of hitting the migration deadline, and frees up valuable resources for value-added work.

Given the benefits of purpose-built, automated solutions, why wouldn't you take one for a test drive?

cirata.com/contact-us

About Cirata

Welcome to Cirata – a new company with over 45 patents and 15+ years of data science expertise in rapidly integrating high value datasets to leading cloud platforms for game changing AI activation and analytics insights.

We accelerate data-driven revenue growth by automating data transfer and integration to modern cloud analytics and AI platforms without downtime or disruption.

For more information on Cirata, visit www.cirata.com.

5000 Executive Parkway, Suite 270, San Ramon, CA 94583
US +1 877 (926-3472) EMEA +44 (0) 114 3039985
APAC +61 2 8211 0620 All other +1 925 380 1728
www.cirata.com

eb-cagd-230912