Data Migrator white paper

by Paul Scott-Murphy, CTO, Cirata

Data Migrator white paper

by Paul Scott-Murphy, CTO, Cirata

Cirata Data Migrator

Data Migrator automates the large-scale transfer of data and metadata from existing data lakes, Spark and Hadoop environments to cloud storage and metadata targets. It introduces a novel technique for transferring large datasets while they continue to be acted on and modified by applications at the source, so that data of any scale can be migrated without needing to disrupt usage or modification of it during migration.

It aims to enable datasets to be used immediately in the target environment, providing facilities to ensure that data are available in a usable form, including the potential transformation of data formats when required, by taking advantage of target environment compute capacity alongside knowledge of the content and metadata available at all times.

By providing Live transfer of existing data and the ongoing changes made to it, Data Migrator supports use cases that span from fixed-scope migrations to ongoing, continuous transfer of changing data with minimal RPO. The technology can take full advantage of available bandwidth, provide clear visibility into the progress of data transfer, and offers flexibility of choice in the source and destination environments. It supports all popular public cloud storage services, network-attached storage infrastructure and large-scale distributed storage systems like those used by Hadoop and Spark clusters. Capabilities for Live transfer depend on the source environment's ability to provide appropriate change notification information.

Large-scale data challenges

Large datasets accumulate through the addition of new data, and changes to that data, over time. Datasets of significant size are too large to store in a single physical system, so are held in distributed storage services such as the Hadoop Distributed File System, IBM Spectrum Scale, etc. or object storage services like Oracle Cloud Infrastructure Object Storage, Amazon S3, Azure Data Lake Storage Gen 2, Google Cloud Storage, etc.

Data storage at scale is relatively cheap, because the economies of scale allow efficient use of storage infrastructure, and the architectures of distributed storage services have evolved to provide those capabilities efficiently. However, there remain time costs associated with the interaction between applications and large datasets.

Straightforward information from small-scale datasets such as a directory listing can be obtained almost instantaneously, but obtaining the same information from large-scale datasets can take significant time. There are fundamental differences between Gigabyte-scale datasets and Petabyte- or Exabyte-scale datasets. While distributed storage services provide the functionality to interrogate and interact with data at scale, they come with trade-offs that impact the time required to provide information.

Complicating that, if the storage from which information is needed is under continuous use, potentially by large numbers of clients in an operational cluster, the information provided in response to interrogation of that data may be out of date and perhaps unactionable by the time it is provided. Techniques such as metadata snapshots can help mitigate potential inconsistencies in information, but can only provide a view of data at some time in the past, not information about the current state of that data.

Transferring data at scale

A system that aims to automate the transfer of data from a large-scale, continuously changing data source needs to account for the complexities that emerge from data at scale. In particular, it will need the ability to account for the fact that the data it intends to transfer to a destination may be undergoing modification while that transfer is being performed.

Changes can occur to individual files, modifying their content, or the file system metadata as a result of deleting, replacing or moving content. If a transfer technology imposes no constraints on existing application use of source data, those modifications can occur at any point in time, and may impact that technology's ability to deliver a successful outcome. Accommodating changing data at scale requires more work than best effort attempts to transfer content on the assumption that it is immutable.

Traditional technologies

A naive approach to large-scale data transfer simply ignores the fact that change can occur to the source, and makes no guarantees about the outcome at the destination, effectively delegating that problem to the user of the data. This is the approach taken by some open source tools like Hadoop's distcp utility (and commercial offerings for backup and data replication that are built on it). Such approaches produce an indeterminate outcome at the destination, making it difficult or impossible to rely on them when data consistency is important.

Other approaches avoid change by transferring a snapshot of the source storage system rather than the current data that it holds. While this can help ensure that the destination will reflect that actual full content associated with the source storage snapshot, it means that the data at the destination will not reflect any changes made to the source since the snapshot was taken. The degree to which the data at target are out of date then becomes directly associated with the amount of time it takes to transfer the full snapshot content, which increases with the amount of data.

A third approach is to attempt to reconcile differences between an initial attempt to transfer the source content, and any changes since made to it. This requires obtaining a description of those differences, and for most distributed storage services, even determining that information can take significant time, preventing the target environment from ever reflecting a reasonably current representation of the source content.

Data Migrator

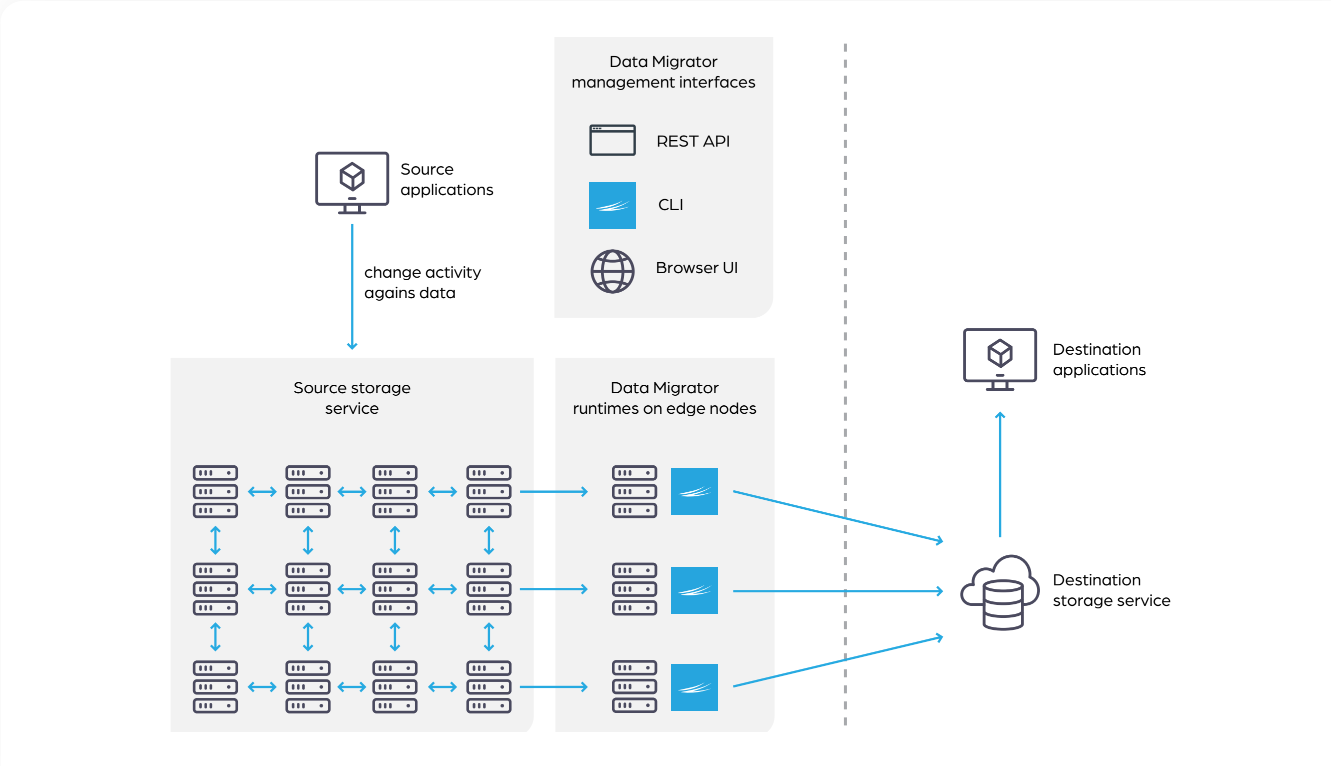
Cirata Data Migrator introduces a novel approach to overcoming the challenges described above by obtaining, coordinating and processing two types of information about the source data concurrently, and performing actions against the transfer destination in response to their combination.

The first stream of information is generated by Data Migrator scanning a location in the source file system for which transfer of data is desired. It uses the interfaces made available natively by the underlying technology to obtain (potentially over an extended period of time), a stream of file system metadata that describes the storage content and their location structure.

The second stream of information is generated by the source storage service itself in response to application activities performed against the data that it holds – a change notification feed. This type of change data is obtained by Data Migrator at the same time as it performs the scan of the source file system.

The combination of these otherwise time-independent streams is used to establish a relative order of information in each stream. This is done by injecting benign activities in the source file system, effectively placing markers in the change data stream. The order guarantees of that stream (which differ by storage service) can then be relied on to ensure that sets of notifications marked by Data Migrator existed after a reference point in the stream of information from the source file system scan. By creating and tracking information about the relative order of events between the streams, Data Migrator can plan and perform the actions required against the destination to achieve a migration or continuous transfer goal in a consistent, efficient and predictable manner.

The planning/scheduling of actions to perform against a target is a dynamic process, influenced by both the streams of information. Data Migrator uses a forest of directed acyclic graphs to represent the event dependencies used to schedule action execution, and aims to perform work concurrently where possible to benefit from the scale out capabilities of source and target storage services. It also enables Data Migrator to distribute certain types of work to adjunct processes to better take advantage of aggregate hardware resources, including compute and network capacity.



Simplified runtime architecture

Data Migrator operates as one or more runtime processes that are typically deployed as regular client applications of the source storage system, installed and operating on virtual or physical hosts close to the source storage service. By operating as a regular, application-level client of the interfaces to that storage platform, it does not impose any change on the storage, can be introduced while the storage remains in constant operational use, and has no direct impact on other applications' use of the storage.

Additionally, Data Migrator's runtime processes require access to the interfaces of the destination storage system, and act as regular client applications against it. There is no other intermediate storage or processing; data flows directly from source storage to Data Migrator, and from there to the destination storage.

Because there can be multiple instances of the Data Migrator runtime operating concurrently to service the transfer of data from a source to a target, the product also provides a single point of control to act against that distributed deployment, providing one form of horizontal scalability.

A second form of horizontal scalability (not represented in the diagram above) is the distribution of content transfer among multiple processes. By delegating the responsibility of content transfer across multiple hosts, the system can benefit from their aggregate network capacity.

Information processing

The intelligence embodied in the implementation of Data Migrator is to decide what is the most efficient action to take against the intended target of data transfer to optimize the use of limited resources:

- bandwidth between source and destination,
- the cost of consuming metadata and content from the source file system,
- the cost of consuming metadata (and potentially content) from the target, and
- the cost of any computation performed against this information.

Its goal is to migrate user-selected pre-existing data from the source and make it available at the destination, and also to modify that destination storage to reflect the most current state of the source storage.

Scale and correctness

Actions taken against the destination by Data Migrator to perform data transfer need to be able to scale with the demands of operational goals. The product takes advantage of a sophisticated determination of dependencies between actions that have the potential to be performed concurrently in order to ensure that:

1. Operations that can be performed independently of one another are performed concurrently where it benefits the overall throughput, and
2. Operations that must be performed in a well-defined order to ensure consistency of outcome at the destination follow that order.

Data Migrator schedules and executes actions in response to changing activity occurring at the source environment. It requires considerable intelligence to maintain efficiency during continued operation against large-scale source data, and Data Migrator's strategies for that scheduling and execution have been tuned to the demands of large-scale environments operating under load.

Failures in external systems

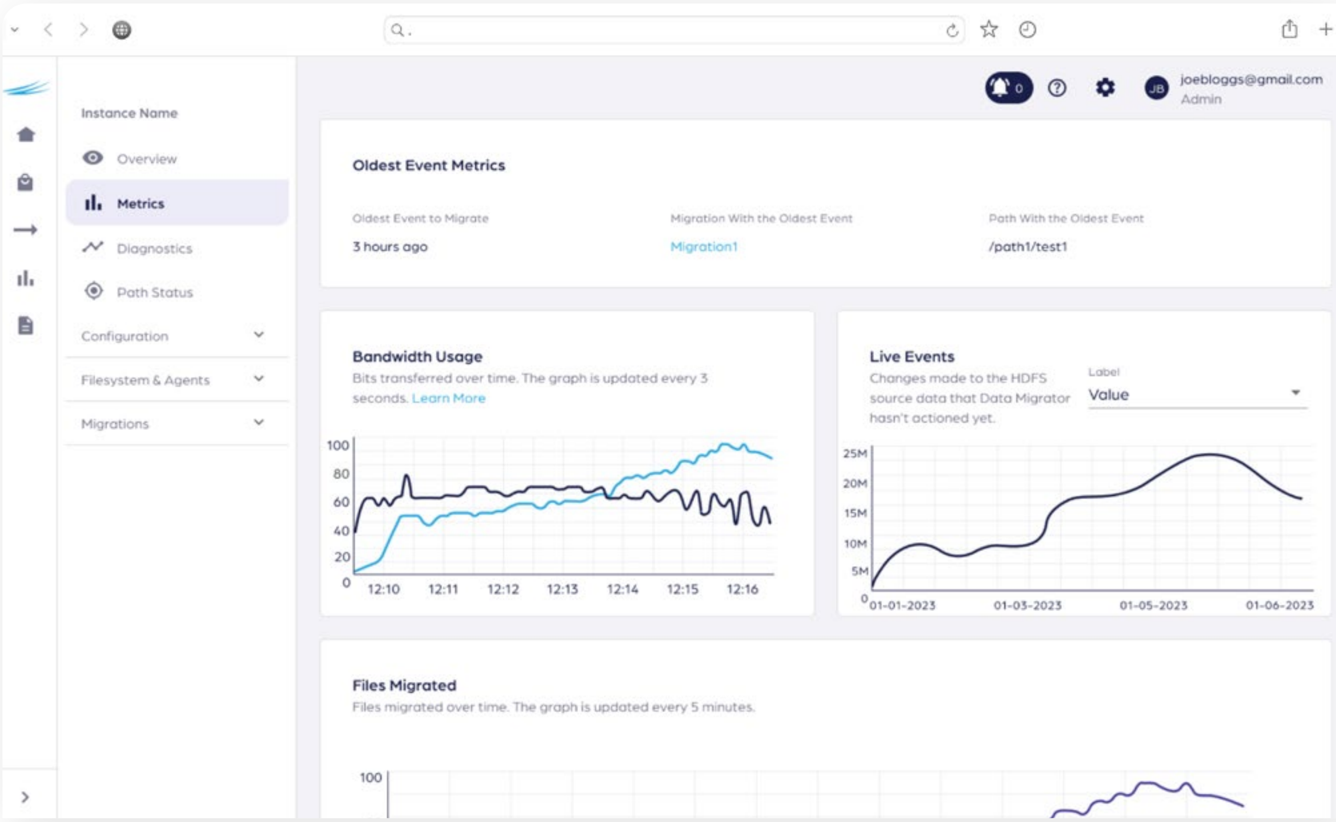
In addition to scanning source data, consuming change notification feeds, and performing actions against the destination storage service, Data Migrator must behave correctly under failure conditions. Its response to transient and permanent failures at scale focuses on automating corrective actions that require no user input. Data Migrator only resorts to administrator notification when there is no correct automatic response to environment, network or system failures.

By making data and metadata migration as simple as selecting the datasets needed in that target storage, organizations can continue to operate all of their existing data infrastructure without any disruption while introducing Cirata Data Migrator, through which they can select the data and metadata that needs to be available in the target.

Automation

Cirata's technology automates the migration of changing datasets at scale, and is designed and proven to scale to multi-petabyte data lake migrations with ease. Key innovations that simplify the administrator's tasks in operating the product include:

- Live data migration, which is the ability to migrate actively changing data without needing to perform repeated scans of the source datasets. This allows migration to begin and complete without any change to source application, and without needing to restrict data movement to periods of downtime or reduced usage at the source. It can help eliminate costly planning and modifications to source systems just for the purpose of migrating or transferring data.
- Selective data migration, to avoid the overhead of moving data that is not needed. Temporary files, staging locations, intermediate representations and more can be bypassed to ensure that only the required data is delivered to the destination.
- Selective metadata migration, which makes the information describing source data available to a target metastore while data migration is underway so that it can be accessed immediately, simplifying the processing of structured or semi-structured data.
- Data consistency, by providing comprehensive facilities to ensure that data arrives in the destinations in the same form present at the source, Cirata Data Migrator provides confidence that analytic outcomes at the destination are wholly accurate and comprehensive.
- Broad compatibility with a wide range of different source and target data infrastructure, whether they are on-premises, in public clouds, or built on platforms like Hadoop or Spark. Unlike data migration technologies that are constrained to a single public cloud provider's environment, Data Migrator can also transfer data to more than one destination at the same time.



Data Migrator operation

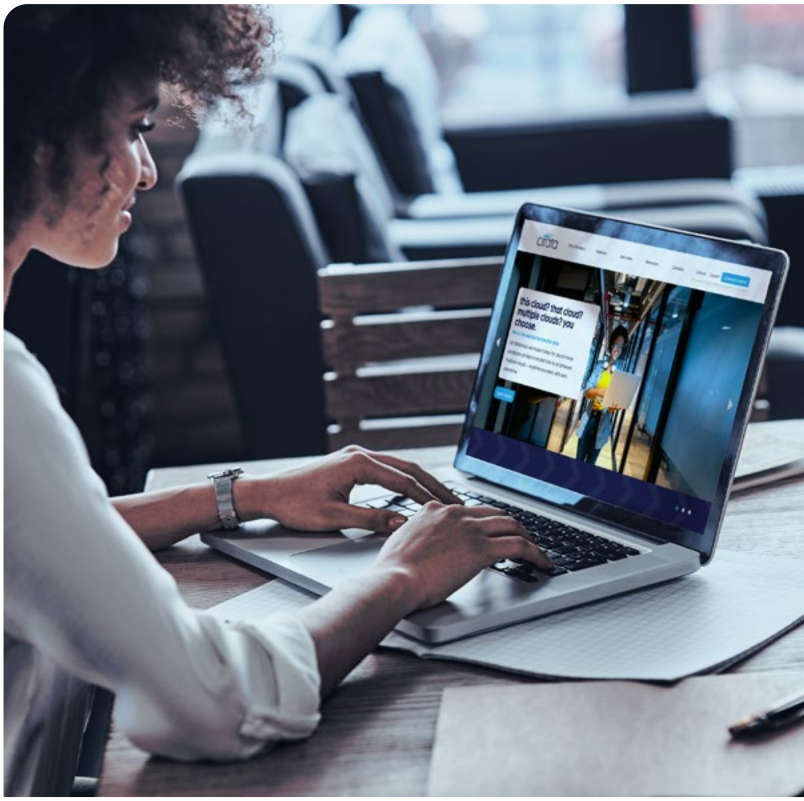
By migrating existing and new data without disruption to source environments operating on-premises or in other clouds, Data Migrator greatly reduces the cost and effort required to leverage all of the functionality and benefits of the destination environment. Users can benefit from the shared data, analytics, governance, security and scalability of the destination environment even while the migration of existing data and workloads is underway.

Migrating data and metadata using the product can be as simple as just three steps:

1. Install Data Migrator on an edge node of the source environment, without any service restarts or application changes,
2. Specify target storage and metadata locations,
3. Start data and metadata migrations by selecting the content to migrate.

All of the additional capabilities of Data Migrator to ease management and operation for migrations at any scale are then available to tune the data and metadata transfers while in progress. You can monitor and validate migration outcomes while continuing to operate the source platform without any change to application behavior or disruption to service.

About Cirata



Unleash your data powerhouse to fuel AI and analytics.

Leveraging our patented DConE® technology and trusted by global brands and industry leaders, Cirata specializes in the integration of high-value datasets into leading cloud platforms enabling game-changing AI and analytics.

With Cirata, data leaders can leverage the power of AI and analytics across their entire enterprise data estate to freely choose analytics technologies, avoid vendor, platform, or cloud lock-in while making AI and analytics faster, cheaper, and more flexible. Cirata's portfolio of products and technology solutions make strategic adoption of modern data analytics efficient, automated, and risk-free.

For more information about 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