ALLUXIO

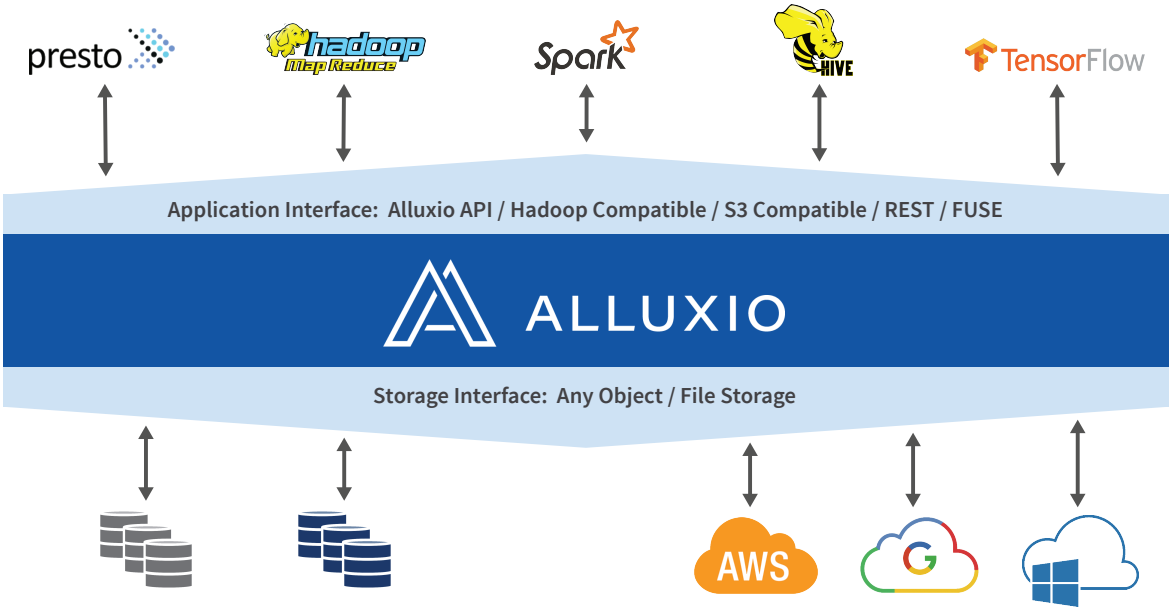
Alluxio Overview

What's Inside

WHITEPAPER

1 / Introduction

Alluxio is an open source software that connects analytics applications to heterogeneous data sources through a distributed caching layer that sits between compute and storage. It runs on commodity hardware, creating a shared data layer abstracting the files or objects in underlying persistent storage systems. Applications connect to Alluxio via a standard interface, accessing data from a single unified source.



2 / Data Access Challenges

Organizations face a range of challenges while striving to extract value from data. Alluxio provides innovation at the data layer to abstract complexity, unify data, and intelligently manage data. This approach enables a new way to interact with data and connect the applications and people doing the work to the data sources, regardless of format or location. Doing so provides a solution to a range of challenges, for example:

- Lack of access to data stored in storage silos across different departments and locations, on-premise and in the cloud
- Difficulty in sharing data with multiple applications
- Each application and storage system has its own interface and data exists in a wide range of formats
- Data is often stored in clouds or remote locations with network latency slowing performance and impacting the freshness of the data
- Storage is often tightly coupled with compute making it difficult to scale and manage storage independently

3 / Benefits of Alluxio

Alluxio helps overcome the obstacles to extracting value from data by making it simple to give applications access to whatever data is needed, regardless of format or location. The benefits of Alluxio Include:

- **Memory-Speed I/O:** Alluxio can be used as a distributed shared caching service so that compute applications talking to Alluxio can transparently cache frequently accessed data, especially data from remote locations, to provide in-memory I/O throughput.
- **Simplified Cloud and Object Storage Adoption:** Cloud and object storage systems use different semantics that have performance implications compared to traditional file systems. For example, when accessing data in cloud storage there is no node-level locality or cross-application caching. There are also different performance characteristics in common file system operations like directory listing ('ls') and 'rename', which often add significant overhead to analytics. Deploying Alluixo with cloud or object storage can close the semantics gap and achieve significant performance gains.
- **Simplified Data Management:** Alluxio provides a single point of access to multiple data sources. For example, If you need to access data stored in multiple versions of HDFS or multiple cloud storage vendors Alluxio also gives applications the ability to talk to different versions of the same storage, without complex system configuration and management.
- **Easy Application Deployment:** Alluxio manages communication between applications and file or object storage, translating data access requests from applications to any persistent underlying storage interface. No application changes are required when accessing different storage systems.

4 / How Alluxio Works

Alluxio is far more than simply a caching solution. A rich set of intelligent data management capabilities ensure efficient use of memory resources, high performance, and data continuity. Data is unified with a single point of access and a standard interface makes data access transparent to applications without any changes. The solution is based on three key areas of innovation working together to provide a unique set of capabilities.

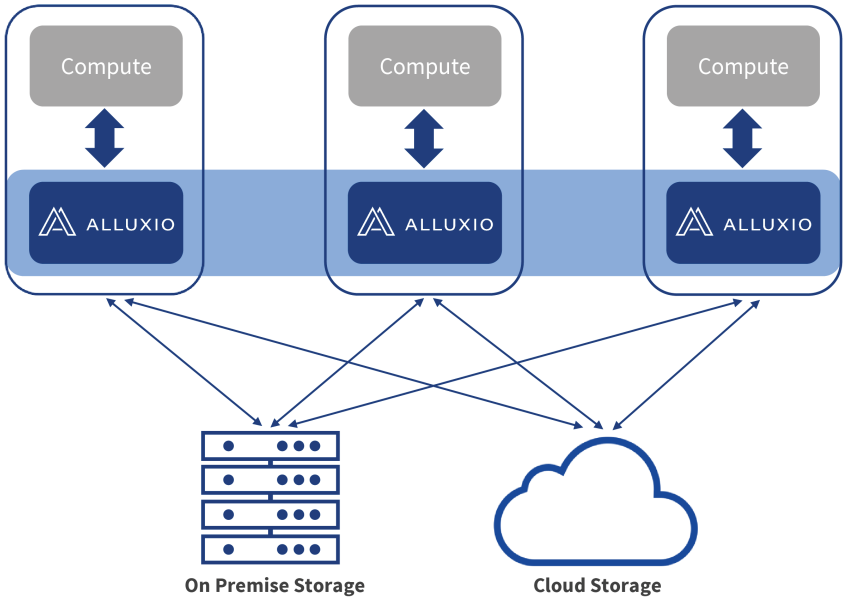
1. **Global Namespace:** A single point of access to multiple independent storage systems regardless of physical location. Alluxio provides a unified view of all data sources and a standard interface for applications.
2. **Server-Side API Translation:** Alluxio transparently converts from a standard client-side interface to any storage interface. Alluxio manages communication between applications and file or object storage, eliminating the need for complex system configuration and management. File data can look like object data and vice versa.
3. **Intelligent Cache:** Alluxio clusters act as a read/write cache for data in connected storage systems. Configurable policies automatically optimize data placement for performance and reliability across both memory and disk (SSD/HDD). Caching is transparent to the user, and uses read/write buffering to maintain consistency with persistent storage.

These innovations at the data layer provide unique benefits and more efficient solutions to modern data problems compared to traditional approaches such as tightly coupled Hadoop clusters, ETL processes, and data lakes. Compared to these alternatives, Alluxio offers:

- Scalability beyond petabytes across storage silos, geographic locations, and cloud providers
- Unified access to a single namespace for all enterprise data, simply identified through a global path
- Concurrent access to shared data sources without modifying applications
- Separation of compute and storage for efficiently managing and scaling resources and adopting cloud architectures
- Storage API independence through support of common storage interfaces including HDFS and AWS S3; applications can access data via their preferred interface regardless of source data API
- Performance through local caching and data placement policies that provide fast local access to frequently used data without maintaining permanent copies.

5 / Enabling Compute and Storage Separation

The primary appeal of a coupled compute-storage architecture is the performance possible by bringing the compute engine close to the data it requires. However, the costs of maintaining such a tight-knit architecture are gradually overtaking the performance benefits. Especially with the popularity of cloud resources, being able to independently scale compute and storage means large cost savings and lower maintenance costs. The reversal of this paradigm puts many data platforms in a tough position, forced to trade off between performance and cost. Alluxio solves this dilemma by providing the same performance of a coupled compute-storage architecture in a decoupled architecture.



Alluxio achieves this by providing a near-client cache when Alluxio is deployed with, or alongside, compute nodes. Applications and compute frameworks send requests through Alluxio, which in turn fetches data from remote storage. Along the way, Alluxio maintains a cached copy of the data in Alluxio in memory or other media (SSD, HDD) available on the Alluxio nodes. Future requests are automatically served through the cached copy. This enables coupled compute-storage architecture performance. However, the key difference is Alluxio doesn't need to hold all data; it only needs to hold the working set. Therefore, Alluxio can match resources to the working data set regardless of total data size. When the working set becomes sufficiently large, Alluxio will provide incremental benefits based on the amount of capacity it has available. This is particularly useful when working with multiple independent storage systems and for accelerating data access to remote locations.

6 / Use Cases

Application Acceleration

Frequently accessed data is served at memory speed from Alluxio. Intelligent caching policies such as replication and local placement ensure highest performance. Remote data can be cached locally eliminating network latency.

Data Unification

Access data across the enterprise as if it were a single source. Alluxio creates a 'virtual data lake' that ties together data regardless of format or location. Eliminate silos and ETL. Access data from different departments, data centers, and multiple cloud providers.

Analytics

Plug and play shared access to large data sets for all popular analytics frameworks. Scale to petabytes of data with memory speed access. Aggregate data across silos and locations regardless of format. Speed up your applications, gain insight faster, and make better decisions.

Separating Compute and Storage

Alluxio allows you to effectively separate compute and storage resources. Data from any source can be accessed locally at memory speed and compute and storage resources can be scaled and managed independently. Cloud adoption is simplified.

Cloud Adoption

Take advantage of cloud flexibility and economics without sacrificing performance and simplicity. Seamlessly unify on-premise and cloud storage in a hybrid environment. Seamlessly access object stores, HDFS, and other file systems simultaneously. Deploy on different cloud providers without changing applications.

Machine Learning

Improve models with access to larger data sets and faster iterations. Mount any storage system as if it were a local file system (with FUSE) and interact with familiar tools and paradigms. Use a common data set for data scientists and in production. Enable self-service data access and train models without starting a new IT project.

7 / Deployment

Alluxio supports a wide range of APIs, storage integrations, and analytics and machine learning applications. Flexible deployment options include bare metal, in Docker containers, and with Mesos, YARN, and Kubernetes.

Supported API's

- Alluxio File System API
- Hadoop Compatible File System API
- REST File System API
- AWS S3 API
- FUSE API

Data Store Integrations

- Public Cloud: SW, GCS, Azure, OSS
- Object Stores: ECS, CLeversafe, Ceph, FusionStor, Minio
- File Systems: HDFS, NFS

Ecosystem Applications

- General Compute: Spark, MapReduce
- SQL: SparkSQL, Presto, Hive
- Streaming: Flink, Spark Streaming
- Key Value: HBase
- Notebook: Zeppelin
- Deep Learning: TensorFlow