KubeCon | CloudNativeCon

THE LINUX FOUNDATION OPEN SOURCE SUMMIT

AI_dev Open Source GenAI & ML Summit

China 2024

CubeFS Boosts Efficiency of AI Production

OPPO Chi.He

Content

01 Introduction to CubeFS

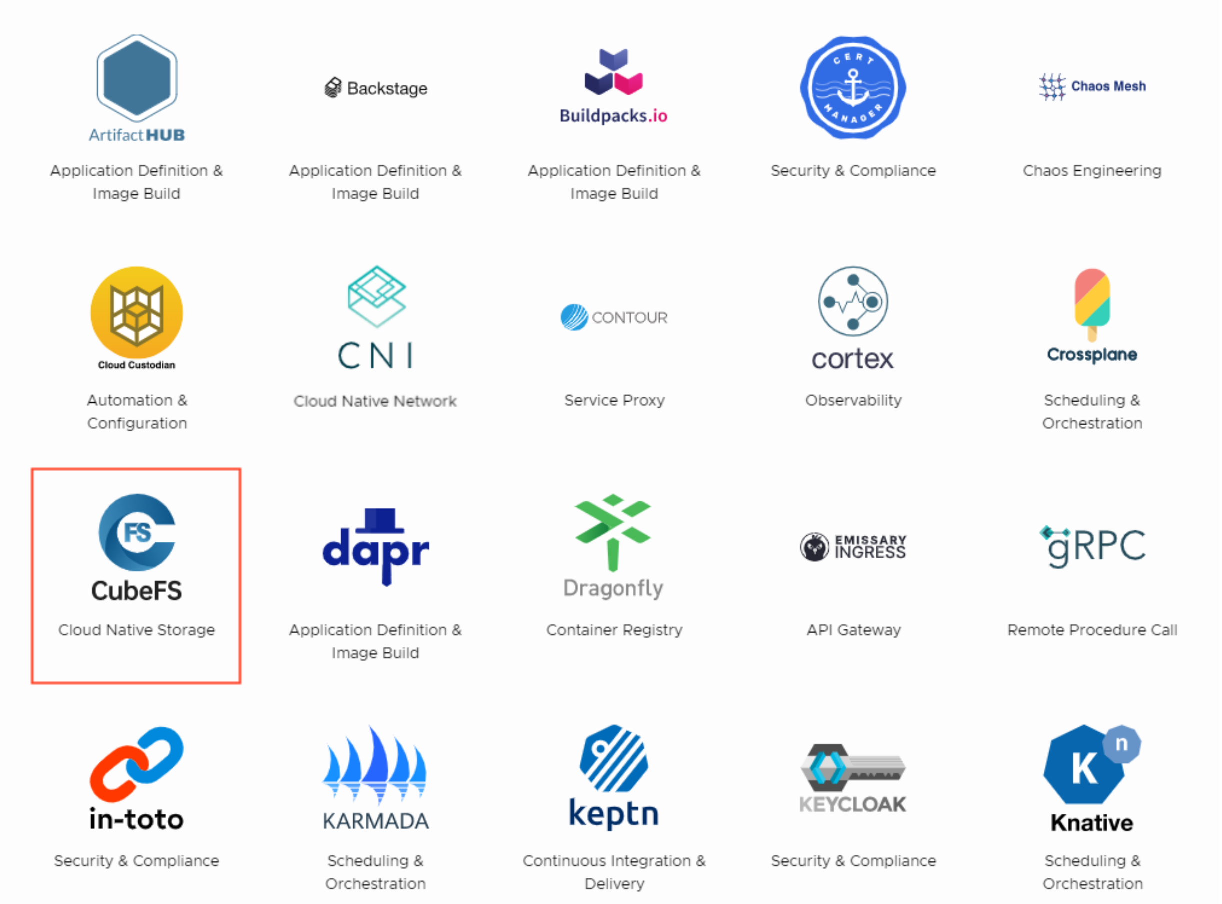
02 The practice of CubeFS on OPPO's machine learning platform

03 Future of CubeFS

Project overview

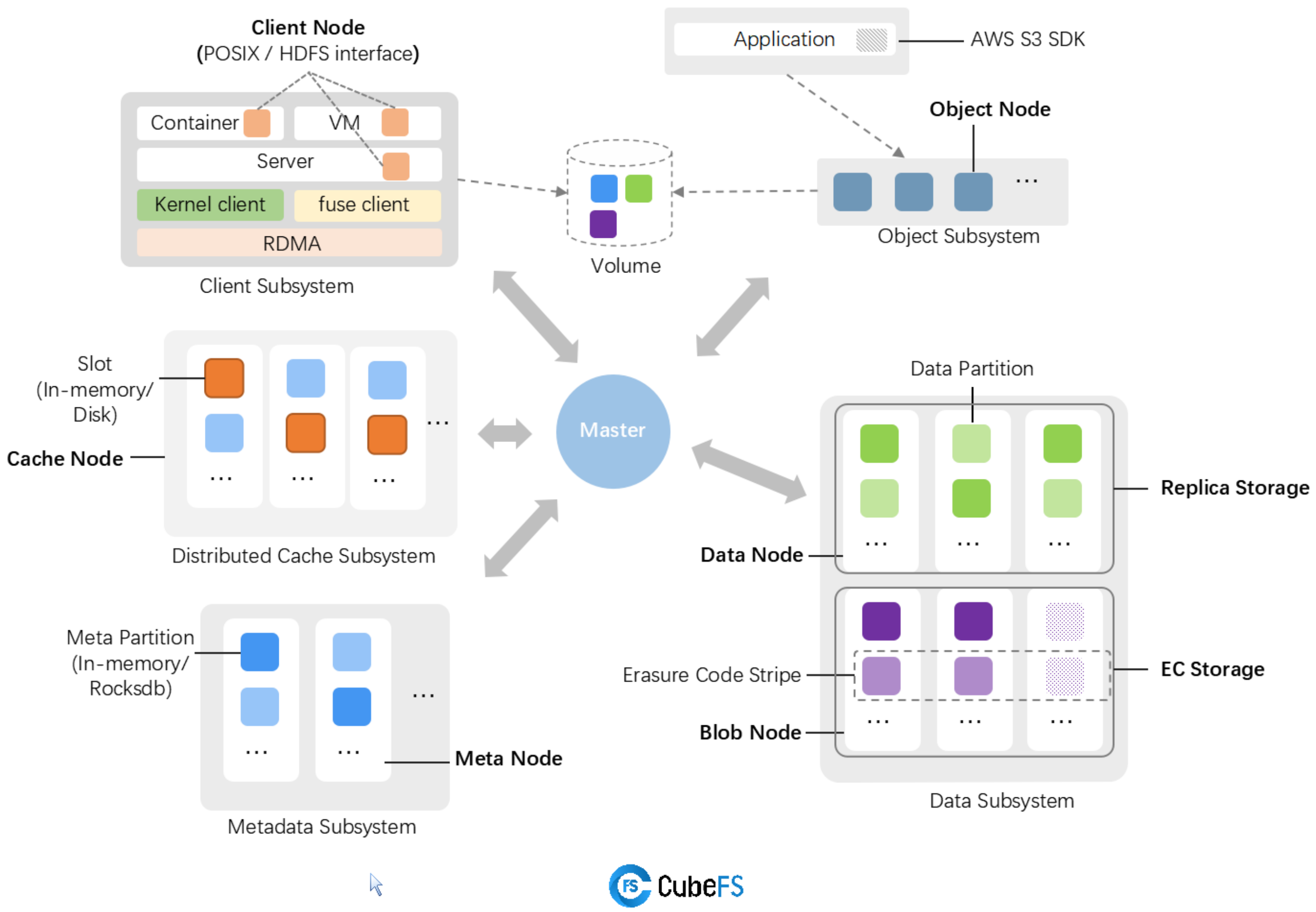
CubeFS is a next-generation cloud-native open-source storage product hosted under the CNCF. It features complete file and object storage capabilities and is currently in the incubation stage.

Official website: https://cubefs.io/



CubeFS

Architecture



Master: Managing resource metadata in the cluster. Ensuring the consistency and high availability of cluster metadata through the Raft protocol.

Volume: A file system that allows clients to access data from containers.

Data Partition: The minimum management unit of file data sharding.

Replica Subsystem: Managing data partition.

Erasure Code Subsystem: Managing erasure code stripe.

Meta Partition: The minimum management unit of file metadata.

Metadata Subsystem: Managing meta partition.

Object Subsystem: Object gateway compatible with S3 semantics.

Client: Providing access interfaces for mounting file systems.

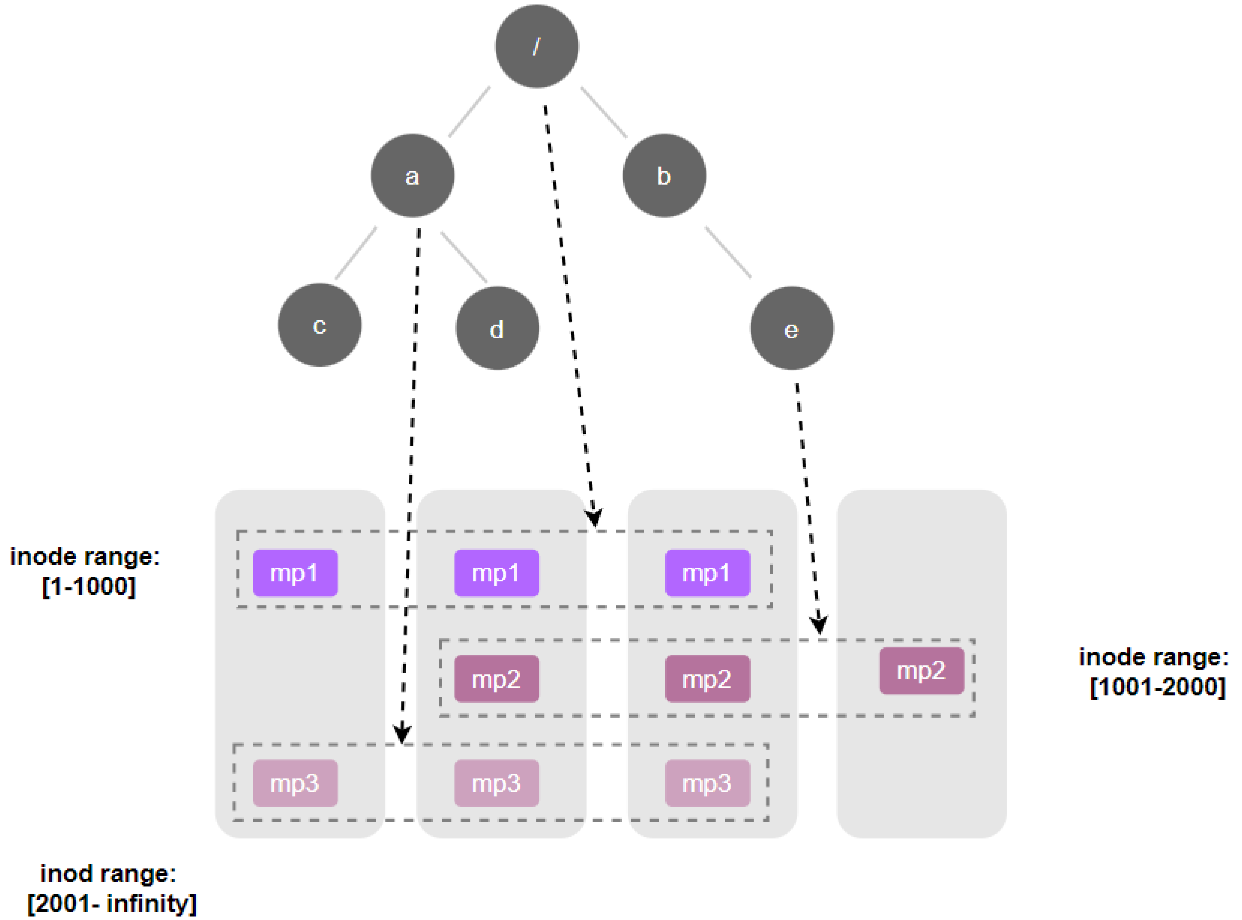
Metadata subsystem

Meta Partition Splitting: Achieving dynamic scaling by splitting the management scope of meta partitions,without triggering data migration tasks.

Full in-memory caching strategy: Improving the access speed of metadata.

Multi-raft: Ensuring strong data consistency and high availability.

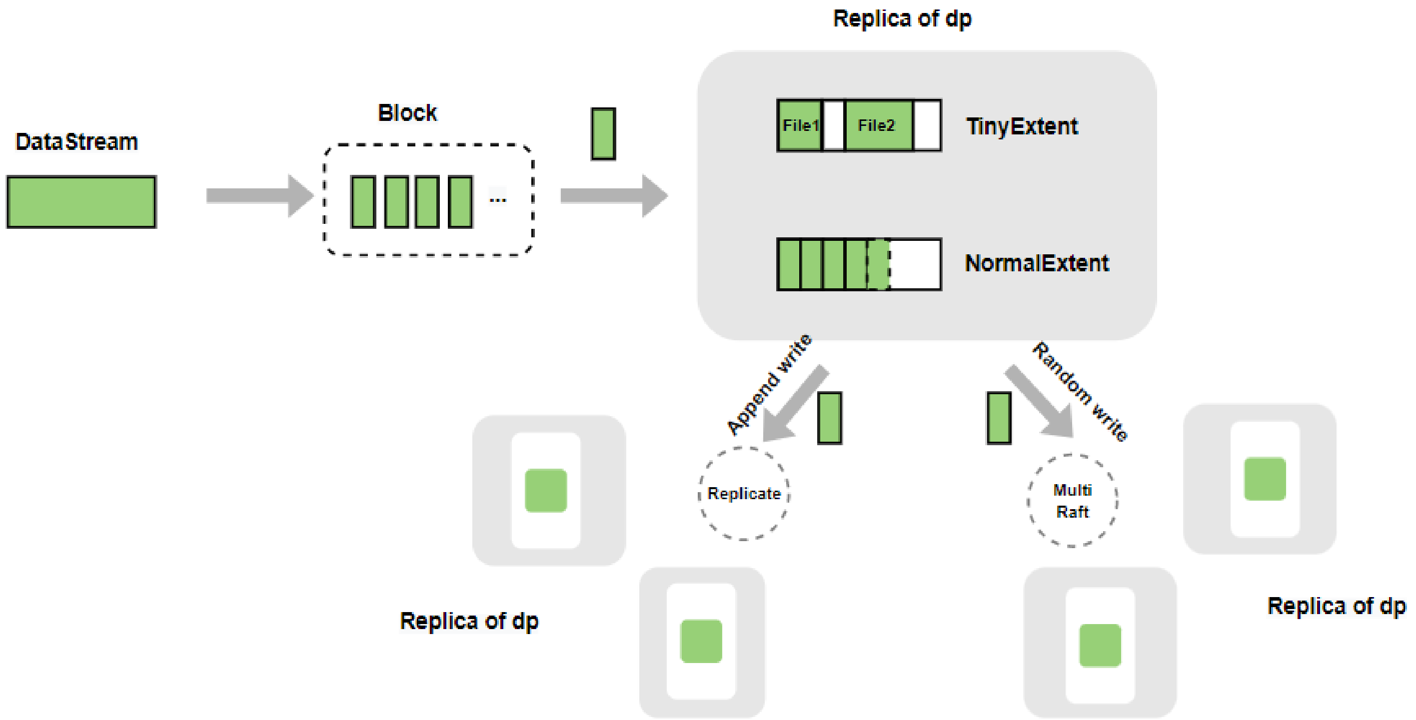
Periodic snapshots: Metadata is periodically persisted to disk on a per-partition basis for backup and recovery purposes.



CubeFS

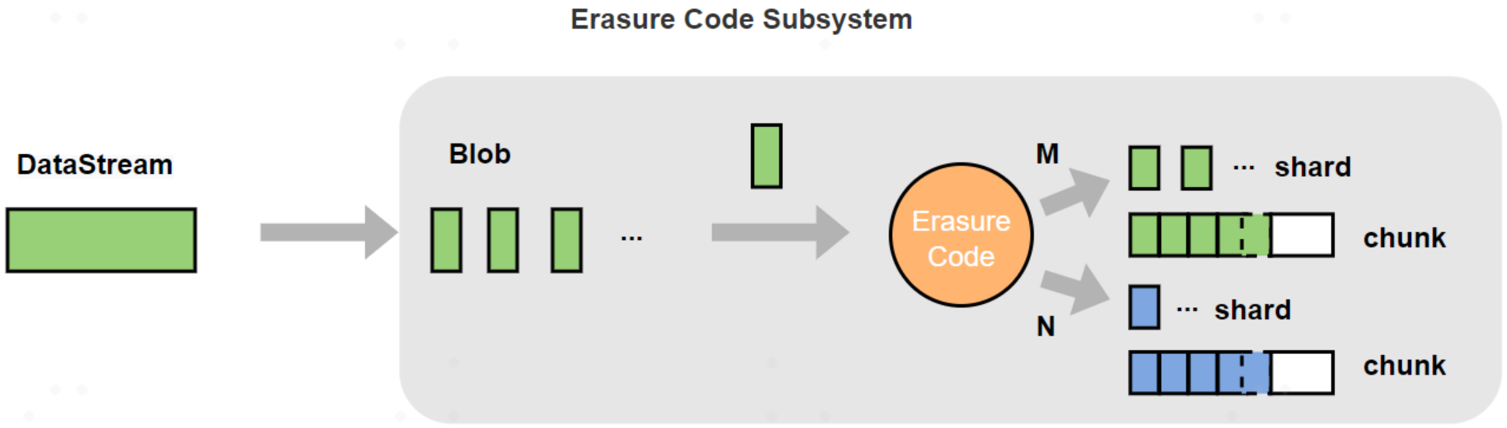
Replica subsystem

Large/Small file storage optimization: Stored through fragmentation/aggregation. Pre-allocated TinyExtents reduce network overhead.
Context-aware replication: Different replication strategies are employed between replicas based on different write patterns to improve replication efficiency.
Automatic decommission for bad disk: The decommissioning process is atomic and does not require manual intervention.
Self-healing for abnormal replicas: Automatically repairing abnormal replicas to ensure high data reliability.



Erasure-coding subsystem

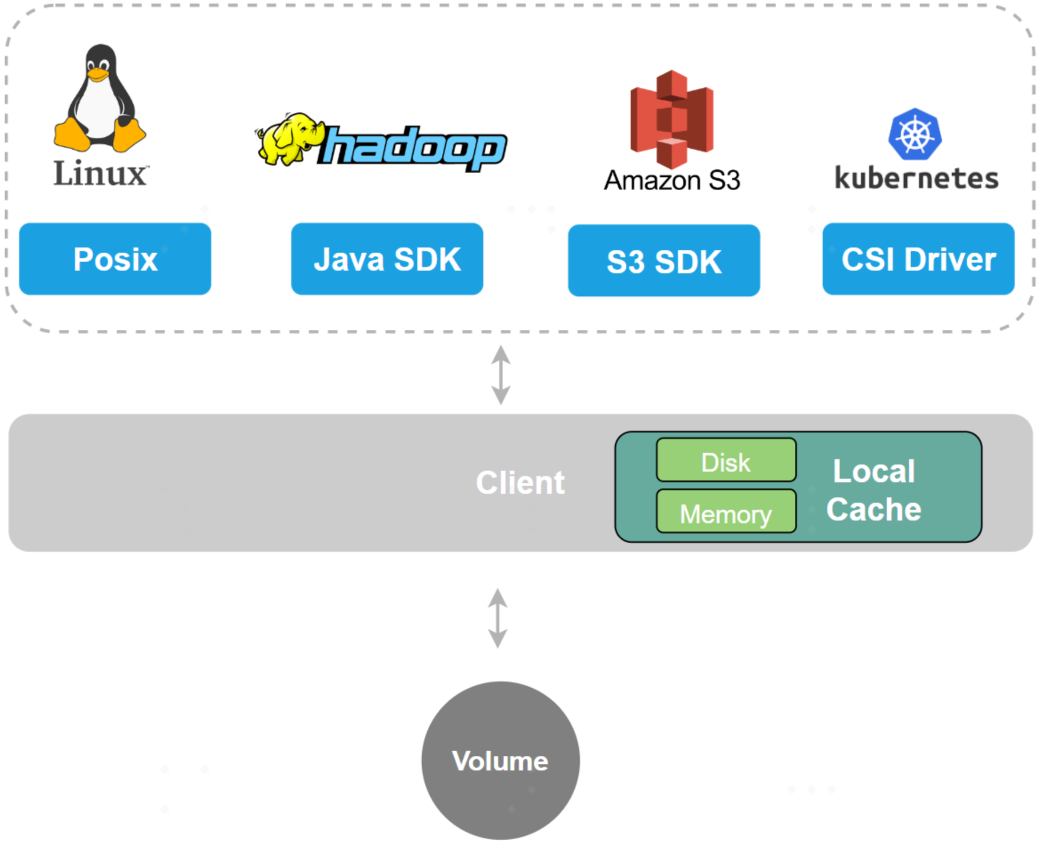
Quorum :Allows for a certain degree of write failures, effectively resolving tail latency issues.
Multi-AZ deployment: Support for 1, 2, 3 AZ deployments, with AZ-level disaster recovery.
Data scrubbing: Ensuring high data availability.



Client

Cross-protocol support: Supporting Posix, HDFS, and other protocols to address different business scenarios and improve data utilization.

Data and metadata caching: Improving data read efficiency.



Summary of features

Multi-protocol
Supports various protocols such as Posix, S3, HDFS, enabling shared business data.

Dual-engine
Flexibly choose between multi-replica or erasure coding storage engines based on business requirements.

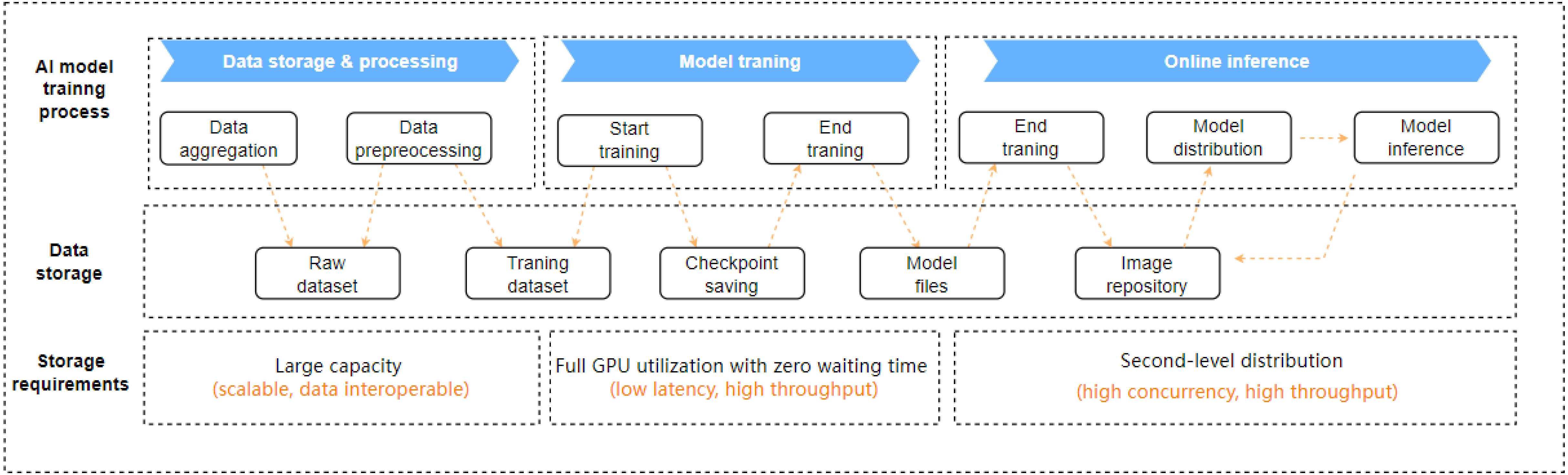
Easily scalable
Meta and data support horizontal scaling, enabling the effortless construction of PB or EB-level storage.

High performance
Full in-memory caching of metadata, with client-side local caching to accelerate access efficiency.

Multi-tenancy
Multi-tenant management, isolating user data and resources.

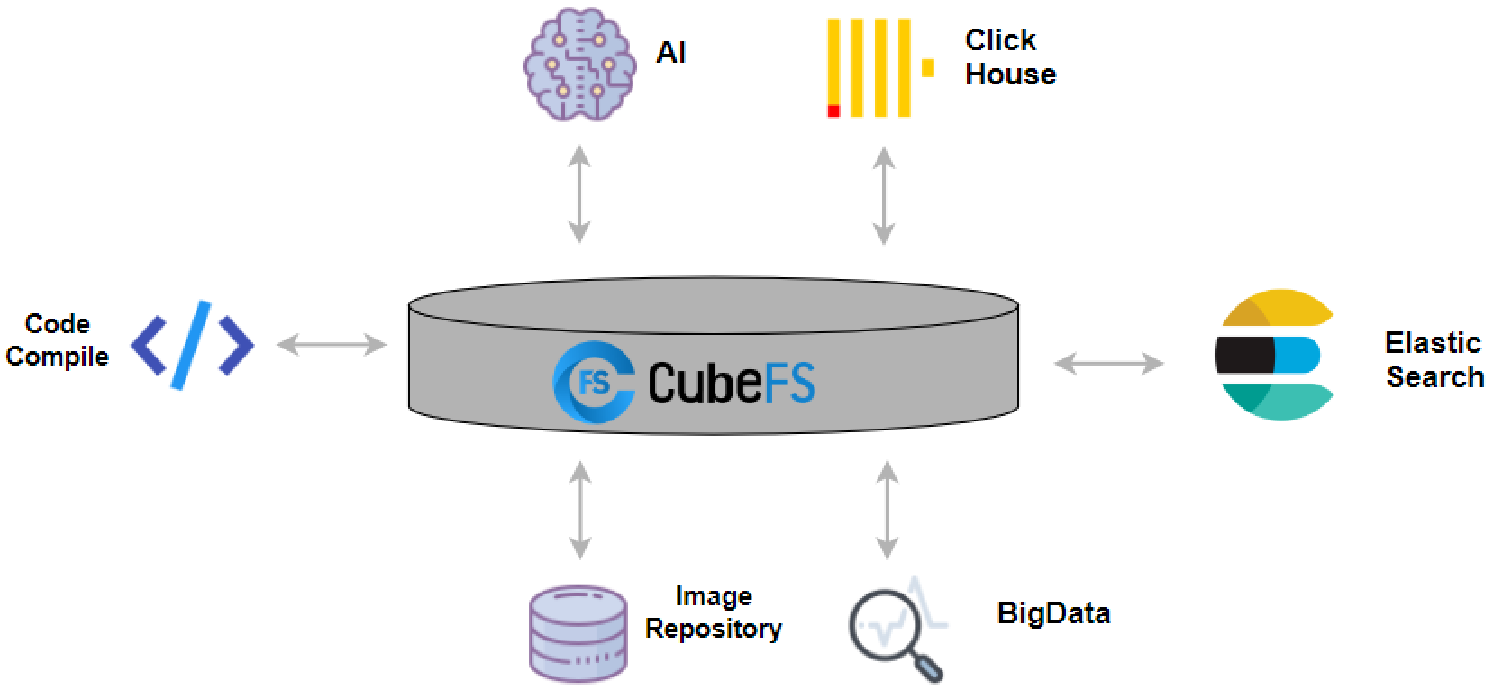
Cloud-native
Using CubeFS on Kubernetes with speed via the CSI plugin.

AI model training process



Data interoperability

Unified storage foundation: The initial landing point for data from different business systems.
Data interoperability: Sharing a set of data across multiple protocols to enhance data circulation efficiency.

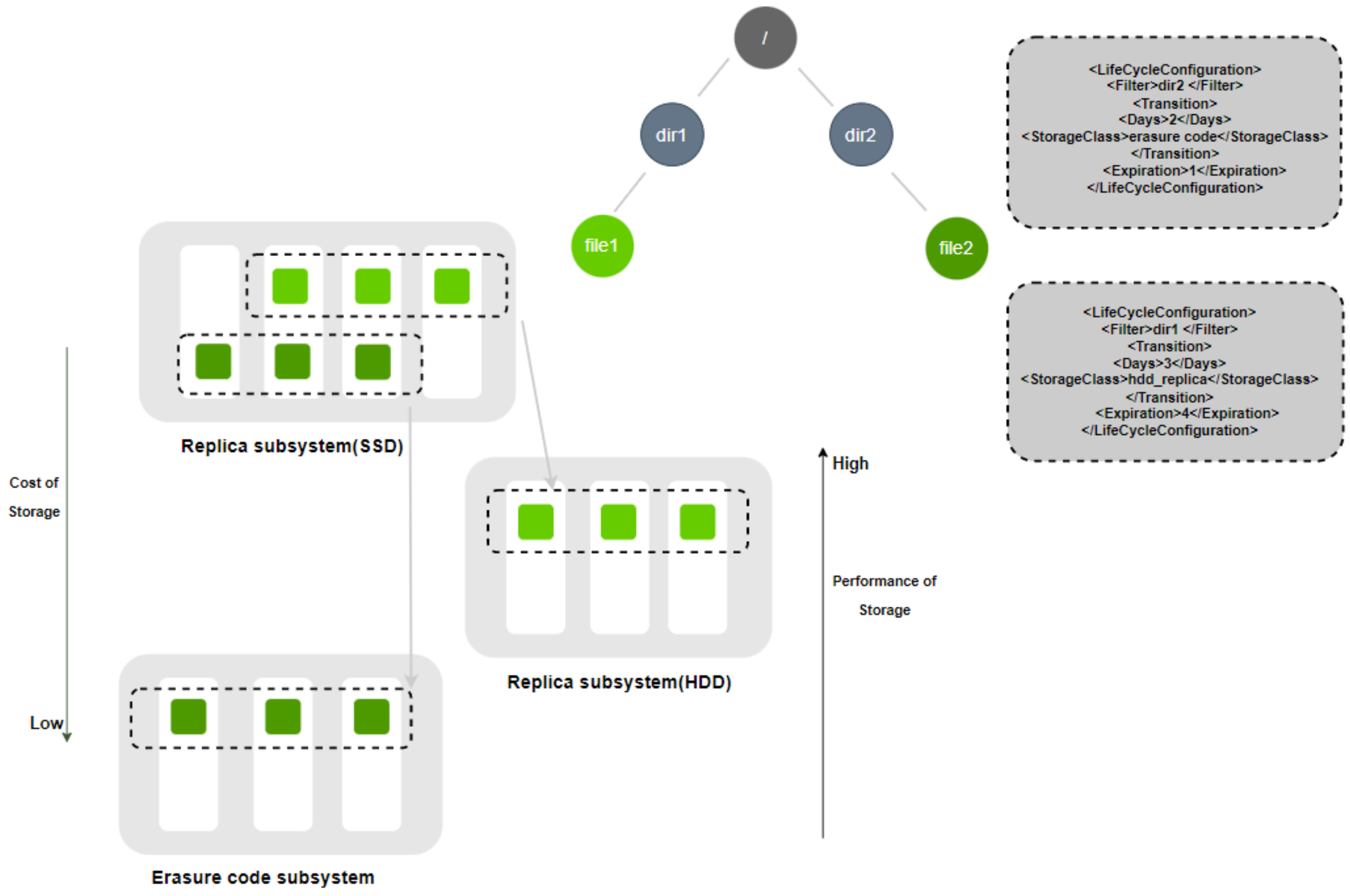


Intelligent lifecycle management

Balancing compute performance and storage costs: Hot data is retained on higher-performance storage media, while cold data is moved to lower-cost storage media.

Configuring lifecycle policies based on directories: Simple operations, automated cold data movement without manual intervention, saving on operational costs.

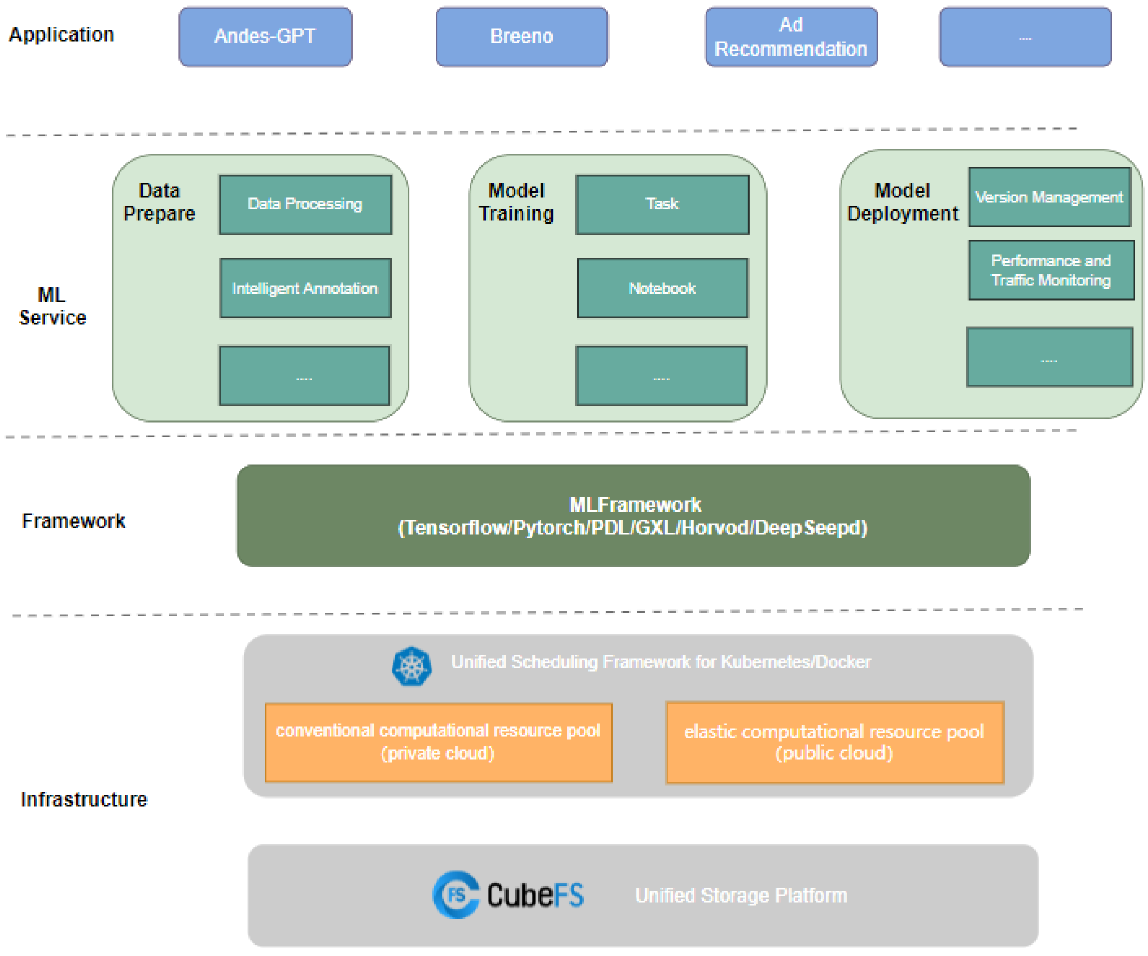
Lease-based cold data strategy: Cold data movement does not affect business access to storage.



CubeFS

Architecture for StarFire

KubeCon CloudNativeCon THE LINUX FOUNDATION OPEN SOURCE SUMMIT AI_dev China 2024



StarFire is OPPO's self-developed end-to-end machine learning service platform

Availability

Meta subsystem adopts a three-replica strategy, increasing availability from 99.9% to 99.99%.。

Operational Costs

With a streamlined architecture, significant reductions in operational costs are achieved, making scaling simpler.

Meta Performance

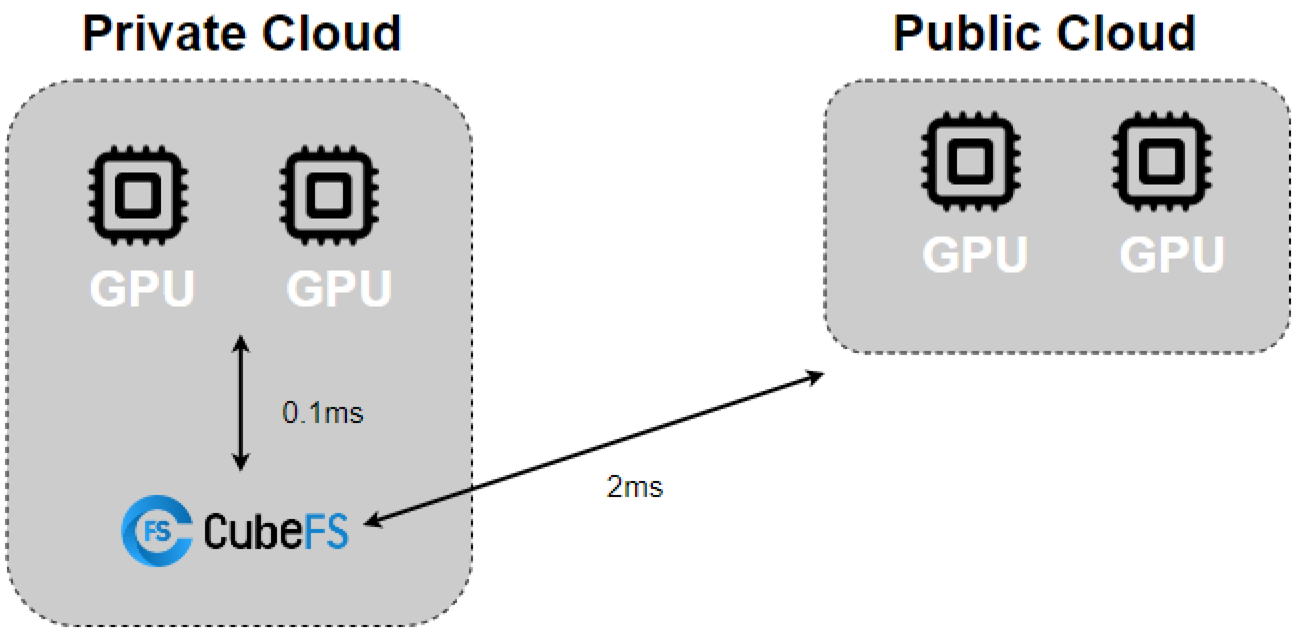
With a full in-memory metadata strategy, the average latency is reduced to 1ms.

Challenges by cross clouds

Decreased storage access performance: The network latency between public cloud GPU resources and private cloud storage resources is around 2ms, leading to low GPU utilization and impacting AI training efficiency.

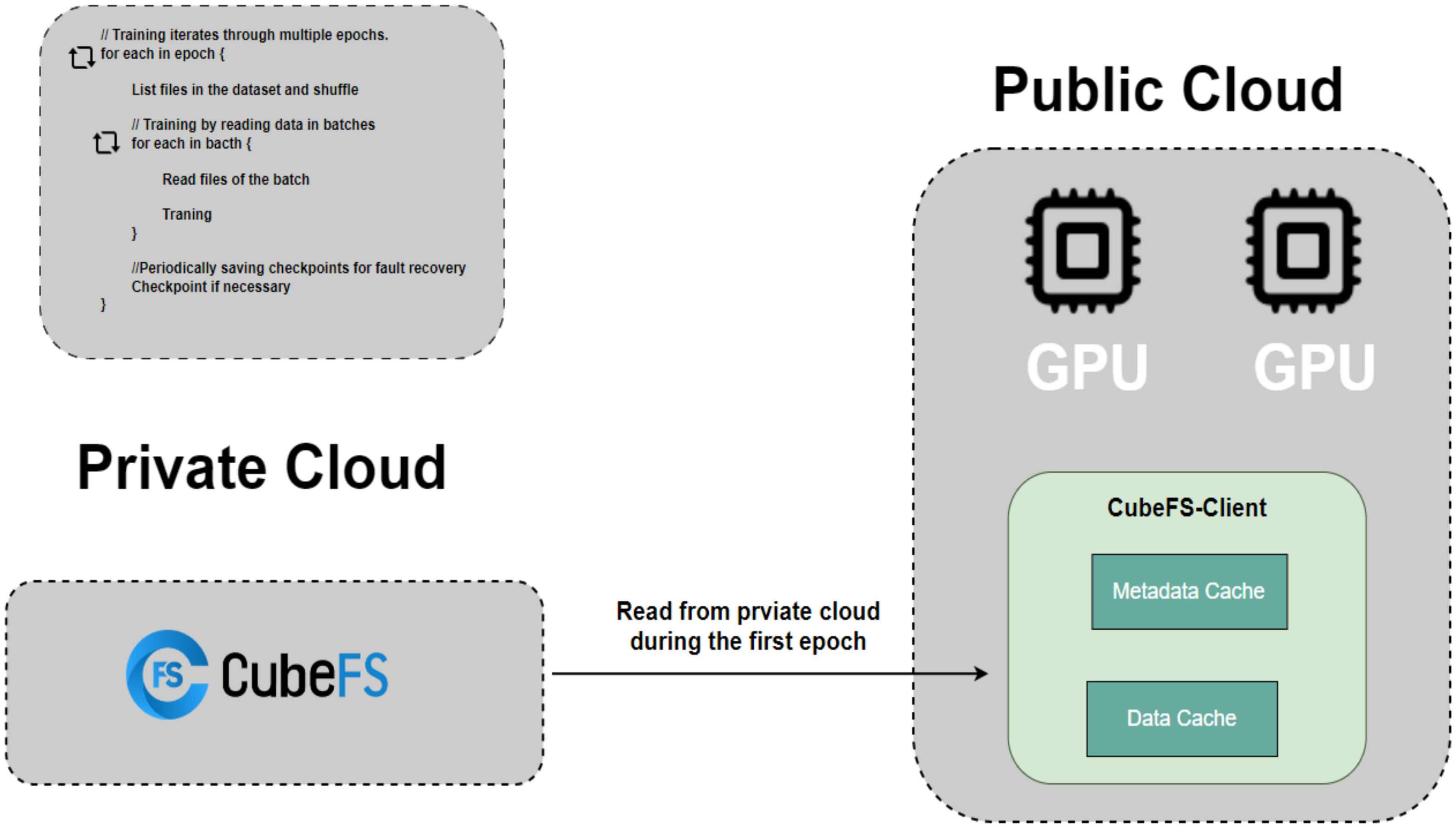
Storage is relatively harder to scale elastically: Migration costs are high, and ensuring data consistency between public and private clouds is challenging.

Data privacy and security: Storing data in public clouds poses a risk of data leakage.



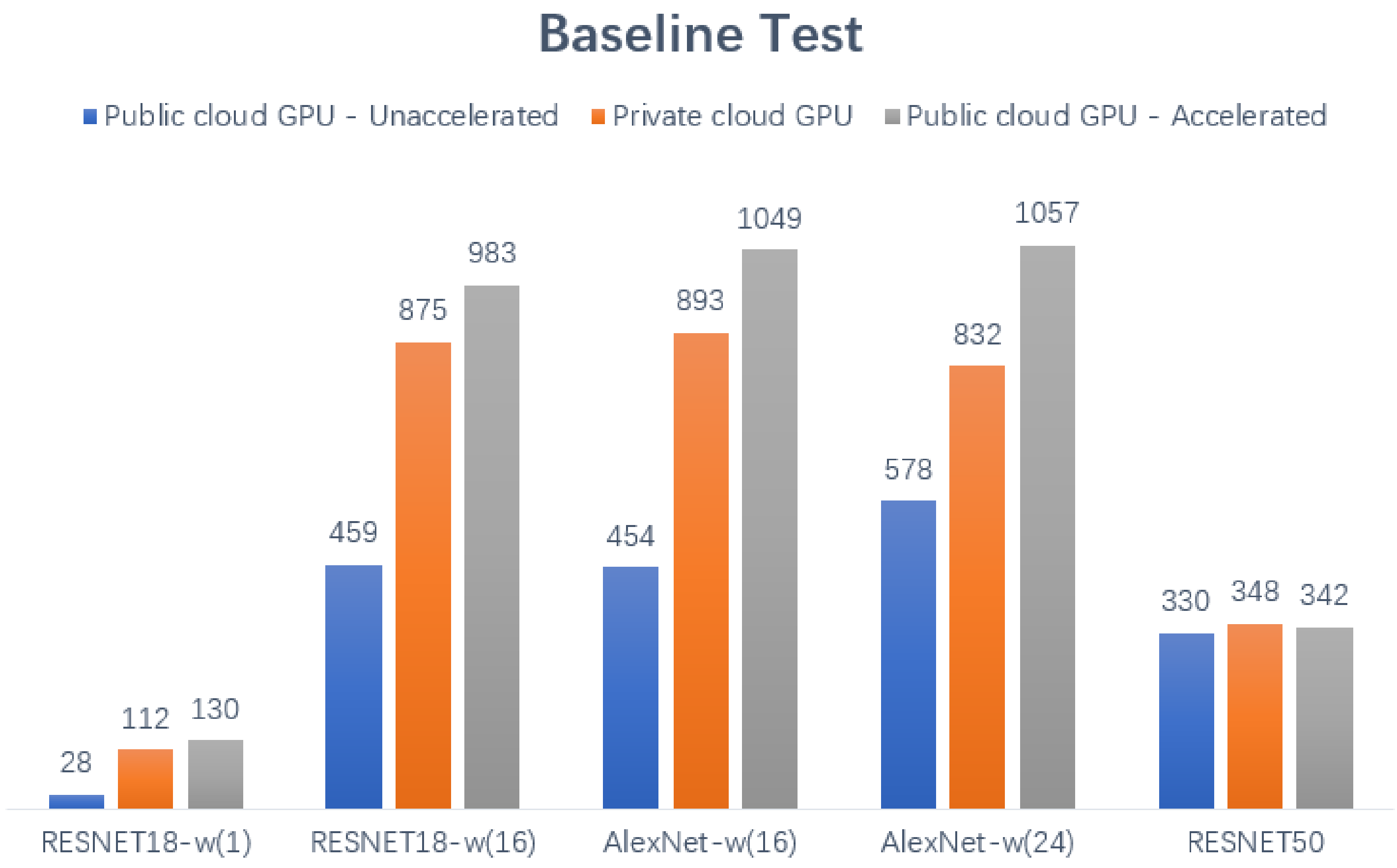
Caching acceleration solution

Shuffle training data: Involves readdir meta operations.
Reading training data: Involves open/close metadata operations and read data operations.
Training characteristics: Both single-machine and multi-machine training involve iteratively executing epochs on the same batch of data.
Metadata/data caching: Maximizing the available memory and disk space on compute nodes to enhance training efficiency.



CubeFS

Benefits of local caching acceleration



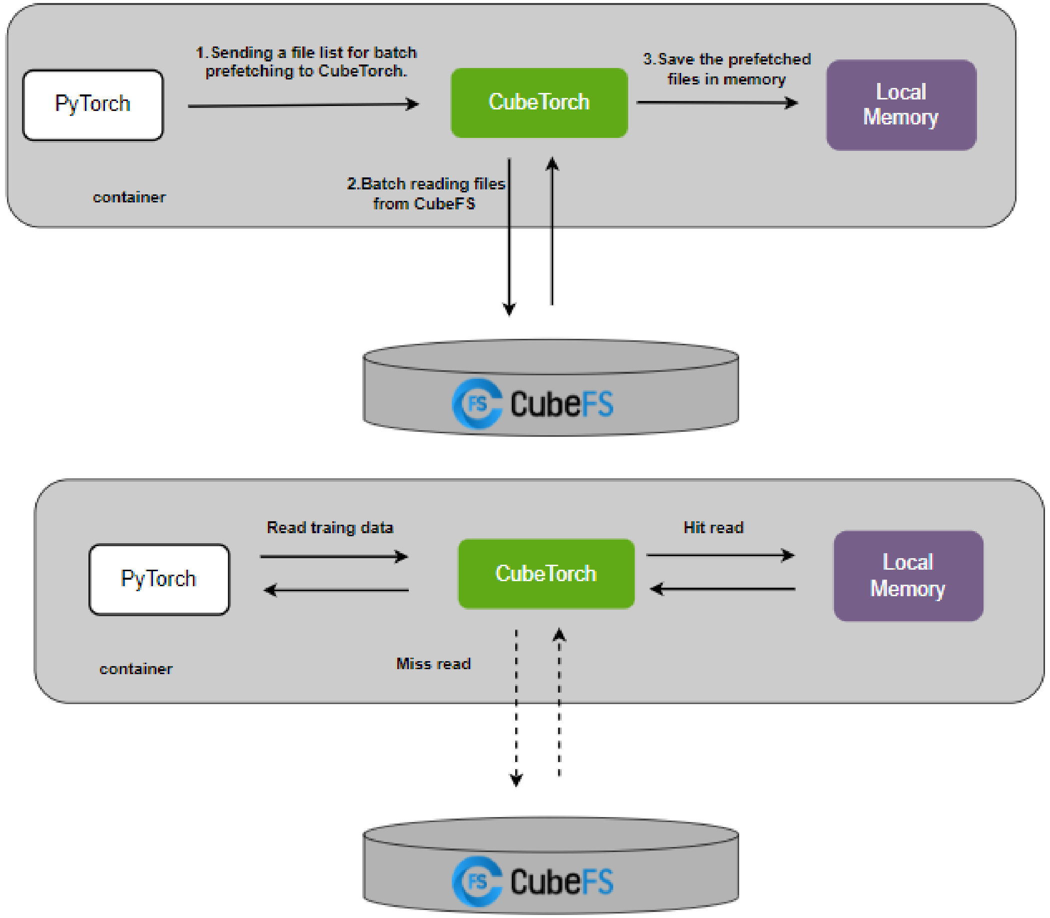
For RESNET18 with Dataloader workers set to 1 and 16, performance increased by 360% and 114% respectively.

For AlexNet with Dataloader workers set to 16 and 24, performance increased by 130% and 80% respectively.

Compared to private cloud deployment, there is also a performance improvement of 12% to 27%.

CubeTorch

Data loader: Parallelizes the data loading process with model training, thereby enhancing GPU training efficiency.
Training data preloading: By batch downloading, preload the next batch of training data into memory; bypass the kernel.



CubeFS

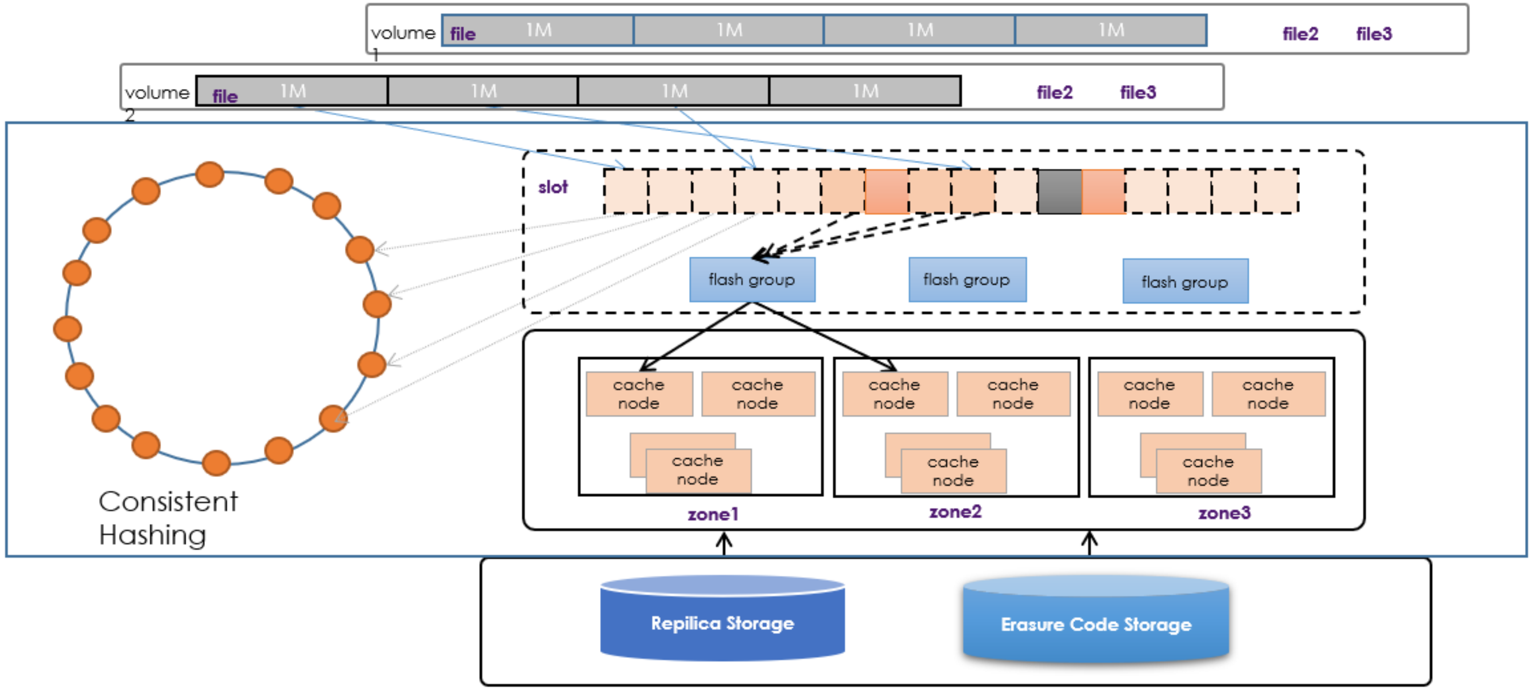
Distributed cache

Distance awareness: Selecting cache nodes with the lowest network latency for access from compute nodes.

Dynamic scaling: Providing users with elastic scalability for throughput capacity.

Elastic replicas: Configured according to business requirements to balance access requests for hot data.

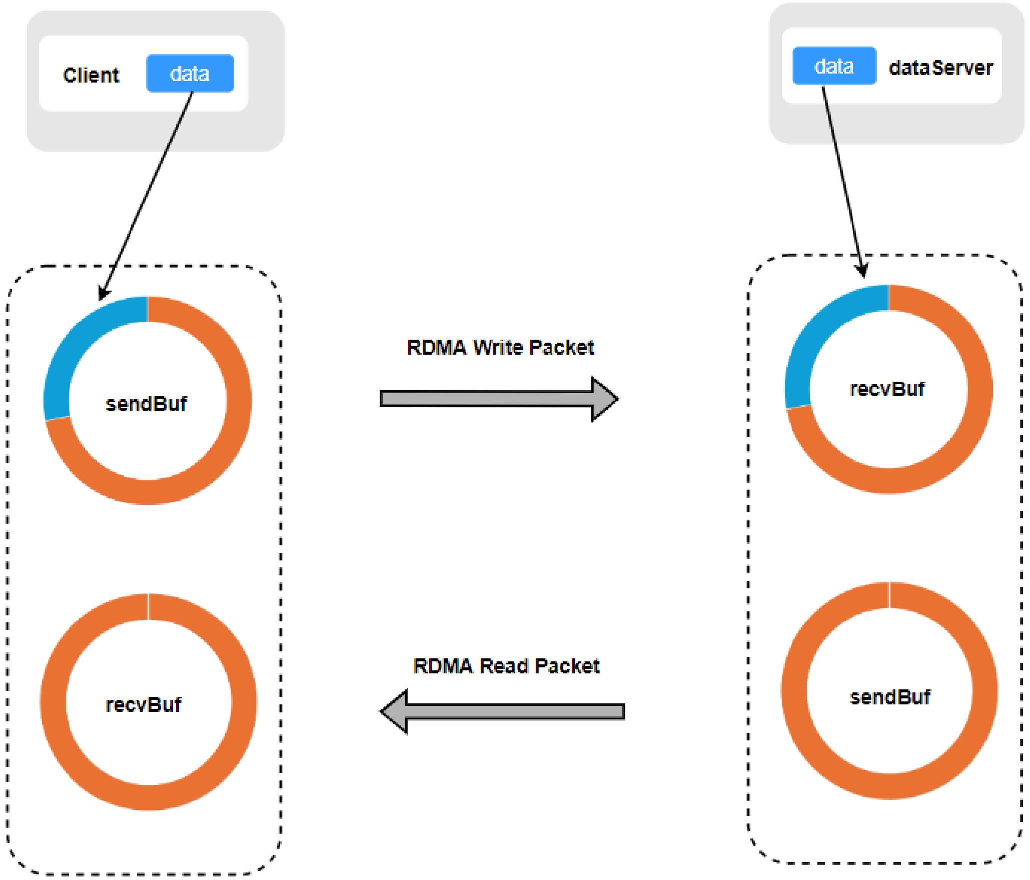
Multi-level caching: Combining with local caches on compute nodes to form a multi-level cache, further enhancing access performance.



RDMA

KubeCon CloudNativeCon THE LINUX FOUNDATION OPEN SOURCE SUMMIT AI_dev China 2024

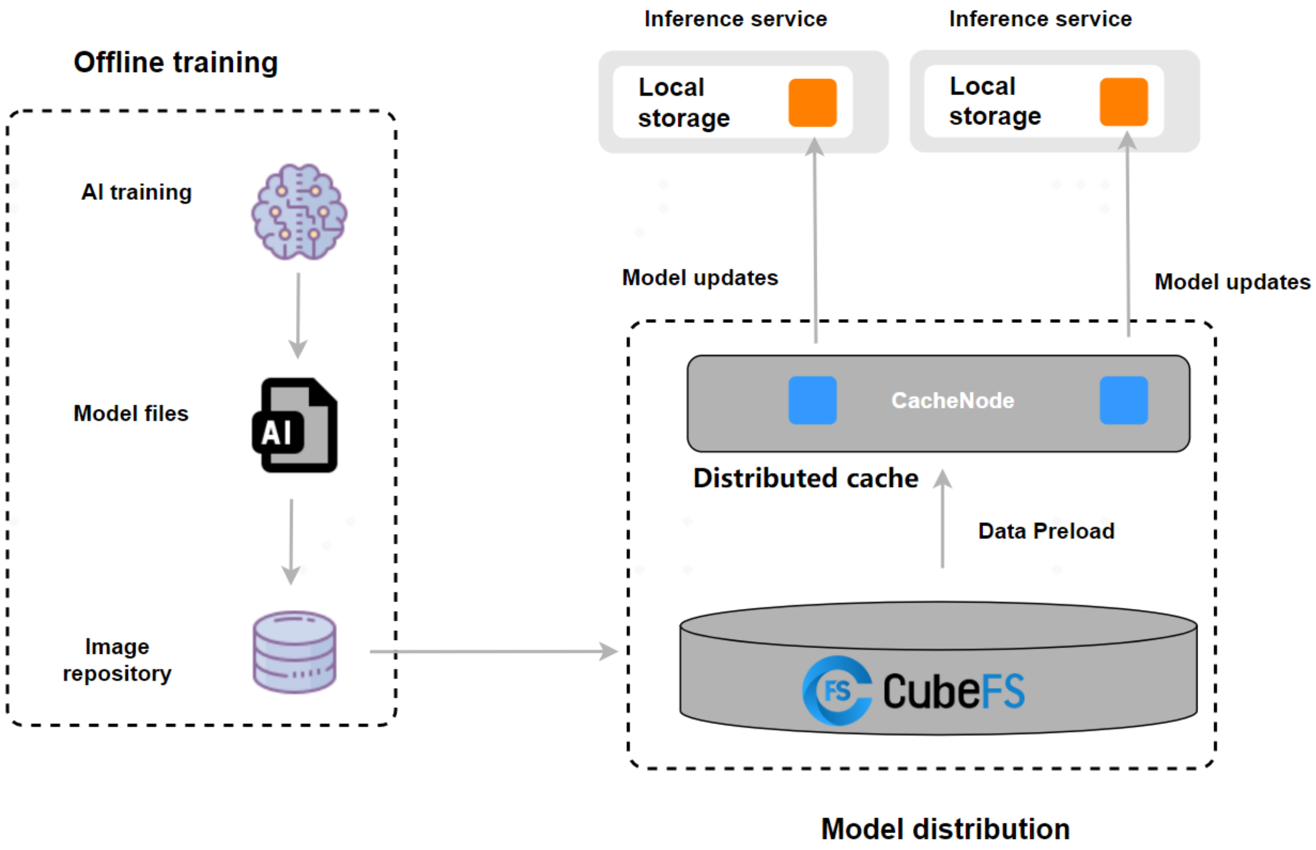
Shared memory ring: Data is copied to sendbuf, then efficiently written to recvBuf via RDMA write.
Higher data write efficiency: The data transfer process bypasses data copying between the kernel and protocol layers, with no CPU involvement throughout.



CubeFS

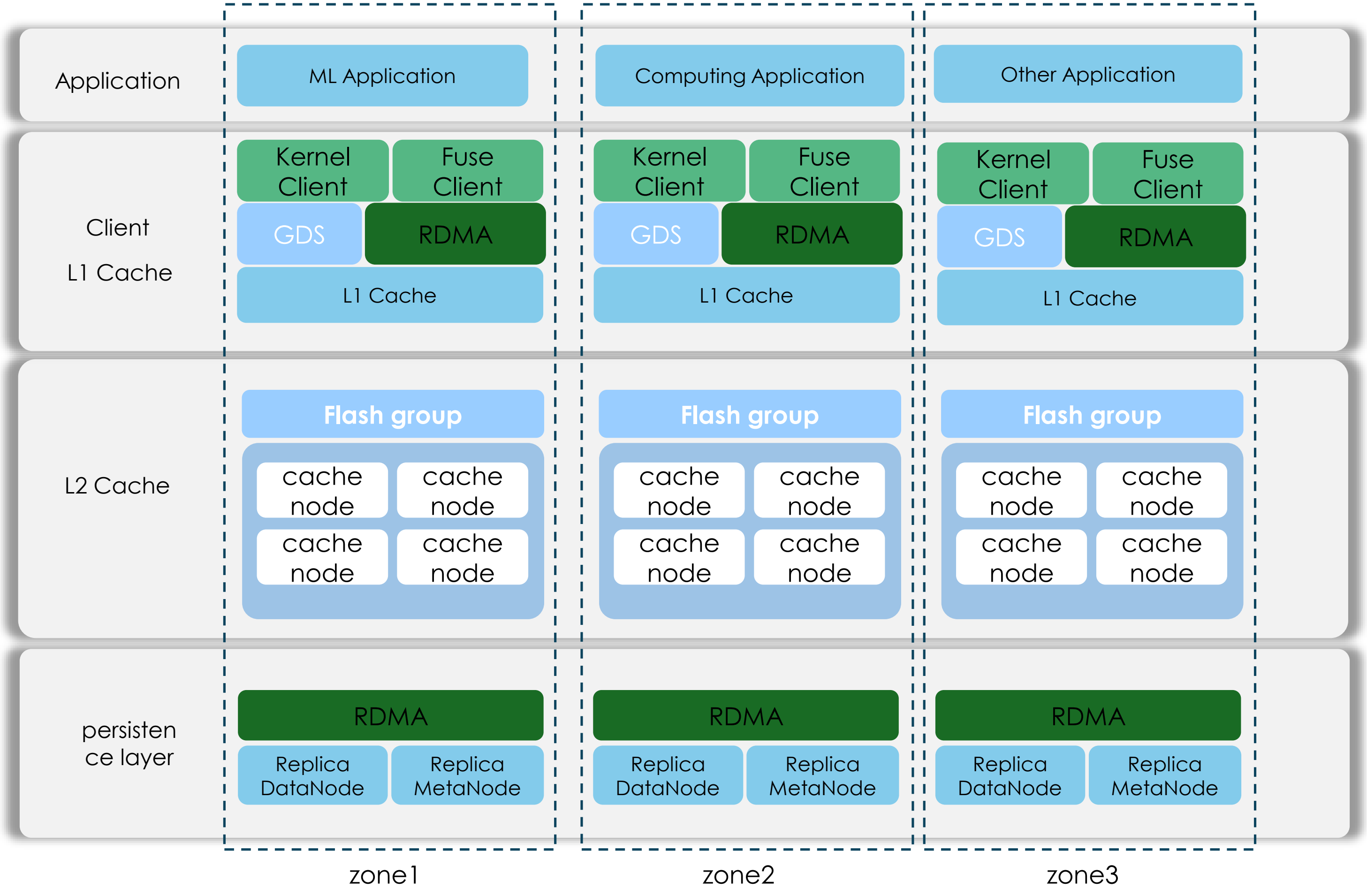
Model distribution

Data preloading: Preloading model files into CacheNode.
Region awareness: Reading cached model file data from nearby locations.
Elastic replicas: Increasing the throughput capacity of model files.



CubeFS

Future prospects



GDS

Enable GPU direct storage at client

Kernel Client

Build kernel filesystem

CubeFS community

Ecosystem Engagement

Prometheus
Kubernetes
Vitess
MySQL
Helm
Harbor
Elastic Search
APACHE HBASE
Spark
ClickHouse

Contributors and Users

Contributors as well as users:

oppo
網易 NETEASE
xiaomi
JOYBUY 京东
贝壳 找房大平台
BIGO

Users:

邮政科学研究规划院 POSTAL SCIENTIFIC RESEARCH AND PLANNING ACADEMY
pits does matter
LinkSure
MEIZU
唯品会 vip.com
360
中科软科技 Sinosoft Co.,Ltd.
SmartMore 思谋科技
瑞为技术 RECONOVA

KubeCon

CloudNativeCon

THE LINUX FOUNDATION

OPEN SOURCE SUMMIT

AI_dev

Open Source GenAI & ML Summit

China 2024