

CIRRO

Lot 1 Managed Cloud File System Service



Service Name *3*
Service Overview *4*
What is FlexFS ? 4
Advantages 5
Traditional parallel file system 6
flexFS file system 6
Managed Service Capabilities 6
Security and Compliance 6
Performance and Scalability 7
Cost Efficiency 7
Typical Use Cases 7
Onboarding and Transition 7
Managed Support 7
Exit and Portability 7
Prerequisites 7
Pricing *8*

Service Name

Managed flexFS Cloud File System Service

Taxonomy	
A-202-303-000	File

Service Overview

Our Managed flexFS Cloud File System Service provides organisations with a scalable, high-performance, POSIX-compliant distributed file system delivered as a fully managed service. It enables compute and analytics workloads to access file data efficiently, securely, and cost-effectively by leveraging object storage as the backend with elastic throughput and robust operational support.

The service is designed for data-intensive environments, including high-performance computing (HPC), AI/ML pipelines, large-scale analytics, scientific research, media processing, and enterprise file workloads that demand high throughput, concurrency, and predictable cost control.

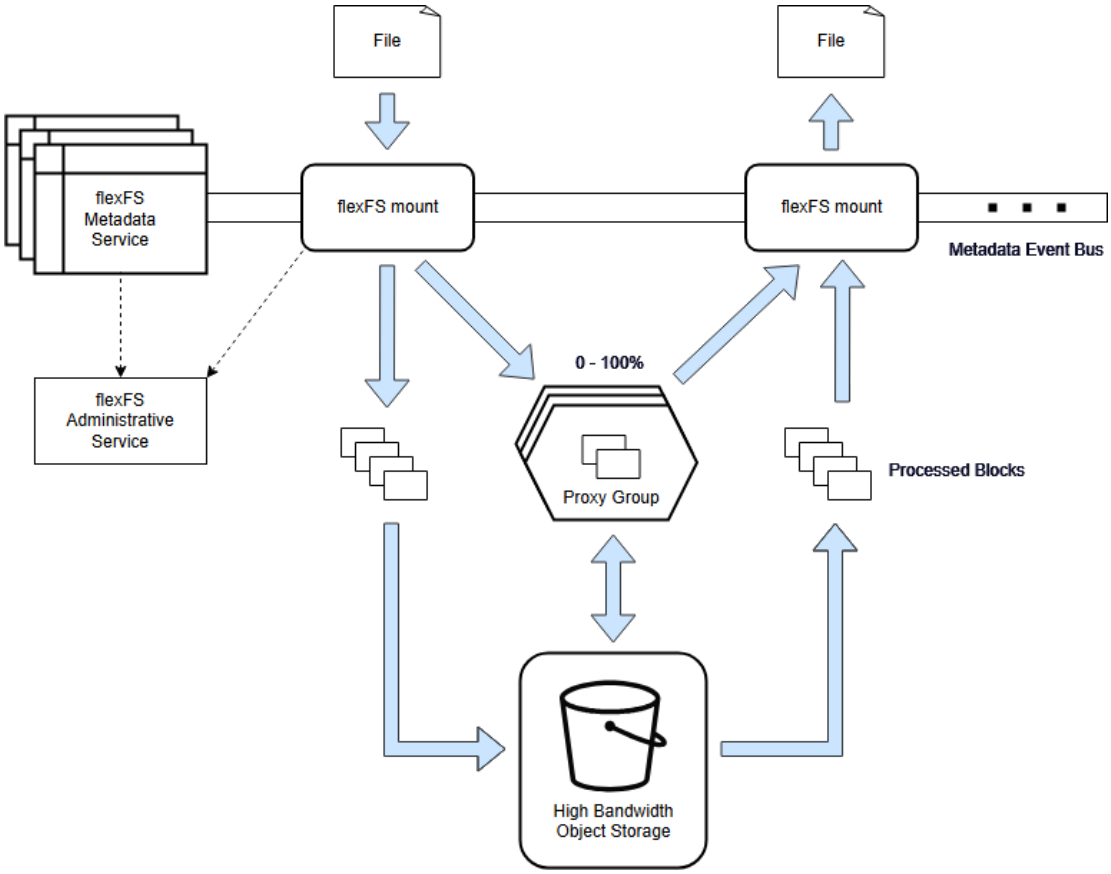
What is FlexFS ?

FlexFS is an innovative file system created by Paradigm4 that can easily transform any network-accessible key-value store with get, put, delete, list operations into a POSIX-compliant network file system which can be mounted across several hosts concurrently. A flexFS volume, when mounted on a host, appears to an end-user as an ordinary folder. FlexFS can be used in a variety of environments: on-premise, in the cloud, or in a hybrid configuration that straddles both. In practice, flexFS is typically backed by an object storage system (e.g. S3, Google Cloud Storage, Azure Blob, Wasabi, MinIO, Scality) and derives performance characteristics of that sytem (e.g. throughput, durability, latency, cost).

FlexFS supports all POSIX-compliant tools and utilities and makes it easy to build reliable, portable, and performant workflows that operate against files in flexFS as if they were located on the host executing the workflow, even though file data will automatically stream to and from the host on-demand. FlexFS automatically handles the difficult and error-prone work of dealing with low-level details such as transient failures, compression, caching, and encryption.

In general, flexFS is a drop-in replacement for general-purpose, POSIX-compliant file systems offered by cloud providers (e.g Amazon EFS, FSx for Lustre) and other vendors. One notable difference, however, is that flexFS is designed to provide high elastic throughput by default, and at a low cost-point, whereas most competing file systems provide low-latency by default and charge extra to add throughput capacity. This difference has major cost and time-to-answer implications

when performing large-scale analyses, working with large files, and sharing file volumes with many users at once. And those are the realms where flexFS demonstrates dominance.



Advantages

What primarily sets flexFS apart from traditional parallel file systems is its ability to directly target object storage as a native block storage. Object storage systems typically have thousands of always-on servers, high aggregate throughput, low usage costs, and high durability. Since flexFS mount clients can load / store file data blocks directly to an object storage service, without passing all I/O operations through a storage service, it is possible for flexFS to have as much aggregate throughput as the underlying object storage service itself.

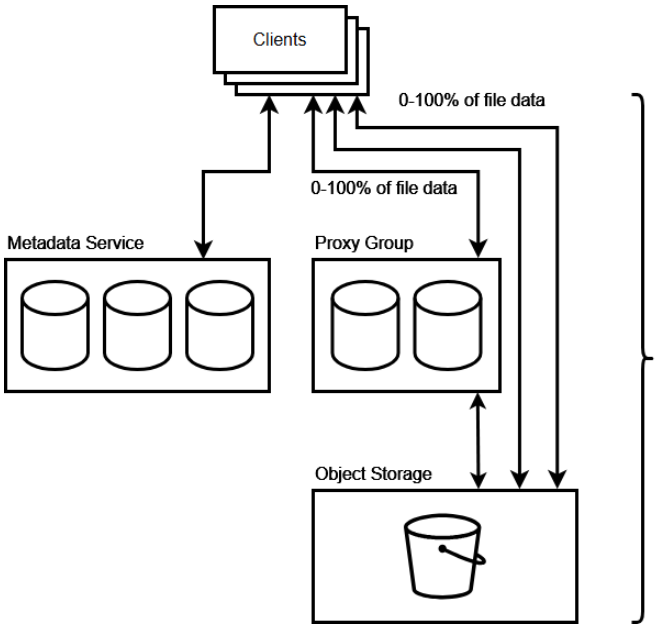
A potential drawback of this direct throughput optimized mode, however, is that object storage services tend to have relatively high latency. To help mitigate that latency when reading and writing small files, flexFS has an optional ability to route only the first n blocks (where n is configurable) through intermediate writeback caching clusters called proxy groups. The end result is a file system with high throughput and durability that requires minimal infrastructure. And with minimal infrastructure comes lower TCO and operational overhead. Meanwhile, there is also a great deal of flexibility in tuning the latency versus throughput characteristics of flexFS.

Traditional parallel file system

Traditional parallel file systems were originally designed and built prior to the popularization of public clouds, and their mount clients are not capable of communicating directly with object storage. In general, all I/O operations are routed to a storage service (usually a cluster of instances with ephemeral NVMe drives). Such a service is costly to run / maintain, and can easily result in a throughput-limiting bottleneck.

flexFS file system

The flexFS file system, however, was built from the ground-up to be able to directly leverage bulk storage services such as object storage. Any unfavourable latency characteristics of those bulk storage services can be mitigated by making use of proxy groups that function as writeback caches for only a configurable fraction of the file data blocks.



Benefits

- Lower TCO than mainstream alternatives
- Low overhead (small, typically fixed-size infrastructure)
- Architected for tunable throughput and latency profiles
- Proxied connections to object storage support low latency
- Direct connections to object storage support high throughput
- Some blocks (e.g. first 10) can be proxied and the rest direct
- High elasticity (throughput can be independent of infrastructure)

Drawbacks

- Throughput-optimized profile (no proxy service) has higher latency

Managed Service Capabilities

We provide end-to-end management of the flexFS platform, including infrastructure provisioning, configuration, monitoring, patching, and optimisation. This removes operational complexity while ensuring resilience and performance.

Security and Compliance

The service includes encryption at rest and in transit, secure access controls, audit logging, and alignment with cloud and government security best practices. Security is integrated by design and supported throughout the service lifecycle.

Performance and Scalability

flexFS delivers elastic throughput that scales with demand, supporting high-concurrency workloads and large data volumes. Proxy caching and performance tuning ensure balanced throughput and predictable access.

Cost Efficiency

By using object storage and elastic scaling, the service avoids over-provisioning and reduces infrastructure costs. Customers pay only for storage and performance consumed.

Typical Use Cases

Use cases include HPC and scientific research, AI and machine learning pipelines, large-scale analytics, media processing, and enterprise shared file workloads.

Onboarding and Transition

Onboarding includes assessment, configuration, secure integration with cloud storage, workload validation, and transition to live service.

Managed Support

The service includes monitoring, incident management, upgrades, performance optimisation, and regular reporting.

Exit and Portability

On exit, we support data portability, documentation handover, and transition to alternative platforms or operating models.

Prerequisites

This service requires access to a provisioned server (on premise or cloud) and access to an Object store – both available under other Cirro GCloud15 services.

Pricing

We offer the service as a true pay per use model of as a pre provisioned capacity purchased in 5TB increments.

	Per GB per month
Metered Billing	£0.06
Pre-provisioned in 5TB units	£0.045