Hammerspace – Standards-based Parallel File System Architecture based upon pNFS 4.2 with FlexFiles

Parallel Performance with Standards-based Access:

At the core of Hammerspace architecture and as a key capability to bring to extreme high-performance data environments is Parallel Network File System (pNFS) 4.2 with Flexible Files (Flex Files), which is part of the NFS 4.2 specification.

Much of pNFS 4.2 spec and in particular the innovation of Flex Files originated from developers at Hammerspace, and is now included in standard Linux distributions. Trond Myklebust, the Hammerspace CTO, has also been the Linux NFS Kernel maintainer for more than 20 years.

pNFS 4.2 with Flex Files introduces a high performance parallel file system architecture that is standards-based, and does not require proprietary client software. pNFS 4.2 with Flex Files is in the NFS 4.2 spec which means it is already in place on any Linux server or Linux-based storage.

With the wide adoption of NFS as an industry standard for network file sharing, this integration ensures seamless compatibility of high-performance parallel workloads with standard NFS file access.

PNFS with Flex Files

By default pNFS 4.2 separates the metadata path from the data path. In pNFS 4.2 with Flex Files, a layout is a data structure that describes how a file's data is distributed across multiple servers (or datastores). It provides the client with the information needed to access file parts directly from these servers. Flex Files can non-disruptively recall layouts (metadata about files) to allow data access and data integrity to be maintained even as files are being copied.

Key benefits for high performance use cases:

> **Dynamic Data Placement:** Unlike traditional parallel file systems that may have static data placement, Flex Files allow for dynamic data placement. This means they can adaptively distribute data across different nodes and storage devices, improving performance and storage efficiency.

Layout's Role in Flex Files: Flex Files use layouts to enhance the file distribution process. The layout information lets the client know where different parts of a file are located across the network, enabling it to read or write data in parallel from/to multiple servers.

Efficiency and Performance: By using layouts, Flex Files can optimize data access patterns based on network conditions, server load, and other factors. This leads to more efficient utilization of resources and improved performance.

Improved Scalability and Performance: Flex Files allow for more efficient data distribution and management across multiple servers, enhancing scalability and boosting performance, especially in high-demand environments.

Adaptability: Layouts in Flex Files are dynamic. They can be adjusted as the system changes, like adding new storage resources or when the network conditions vary. This adaptability ensures consistent performance and reliability.

Client-driven Layout Management: Flex Files empower clients with more control over data layout. It uses the layout information to make informed decisions about where to read/write data, leading to better load balancing and reduced server bottlenecks. This ensures more optimized performance for varying workloads and environments. In contrast, other systems typically have server-centric control, which limits flexibility.

Improved Load Balancing: Flex Files offer better load balancing capabilities. Data can be distributed more evenly across servers, reducing hotspots and improving overall system performance.

Greater Flexibility in Storage Types: Flex Files support a wider variety of storage types and configurations compared to some parallel file systems that may be limited to specific hardware or configurations. In this case, any NFS v3 storage from any vendor is supported without modification.

Enhanced Network Efficiency: pNFS with Flex Files is designed to reduce network traffic through more efficient data distribution and retrieval strategies, which is particularly beneficial in environments with large-scale data operations, significantly reducing the network chatter which was prevalent in previous older versions such as pNFS 4.1.

Greater Performance with Less Hardware:

Hammerspace hyperscale customers are achieving extreme performance with less hardware with standards-based pNFS 4.2 with Flex Files, since there is no need for additional client servers or storage hardware. Existing Linux-based clients already contain pNFS with Flex Files, and any NFS v3 storage already supports direct access.

In the case of a large hyperscale customer, this has resulted in performance of 10TB/s parallelized across 800 storage nodes and 2000 client servers with 16k GPU nodes. This was achieved using their existing servers and storage, with only the Hammerspace metadata servers needed.