Overview and requirements

BeeGFS on NetApp with E-Series Storage

NetApp
January 27, 2026

Table of Contents

Overview and requirements

Solution overview

The BeeGFS on NetApp solution combines the BeeGFS parallel file system with NetApp EF600 storage systems for a reliable, scalable, and cost-effective infrastructure that keeps pace with demanding workloads.

NVA program

The BeeGFS on NetApp solution is part of the NetApp Verified Architecture (NVA) program, which provides customers with reference configurations and sizing guidance for specific workloads and use cases. NVA solutions are thoroughly tested and designed to minimize deployment risks and to accelerate time to market.

Design Overview

The BeeGFS on NetApp solution is designed as a scalable building block architecture, configurable for a variety of demanding workloads. Whether dealing with many small files, managing substantial large file operations, or a hybrid workload, the file system can be customized to meet these needs. High availability is built into the design with the use of a two-tier hardware structure that allows independent failover at multiple hardware layers and ensures consistent performance, even during partial system degradations. The BeeGFS file system enables a high-performance and scalable environment across different Linux distributions, and presents clients with a single easily accessible storage namespace. Learn more in the architecture overview.

Use cases

The following use cases apply to the BeeGFS on NetApp solution:

- NVIDIA DGX SuperPOD systems featuring DGX's with A100, H100, H200, and B200 GPU's.
- Artificial Intelligence (AI) including machine learning (ML), deep learning (DL), large-scale natural language processing (NLP), and natural language understanding (NLU). For more information, see BeeGFS for AI: Fact versus fiction.
- High-performance computing (HPC) including applications accelerated by MPI (message passing interface) and other distributed computing techniques. For more information, see Why BeeGFS goes beyond HPC.
- Application workloads characterized by:
  - Reading or writing to files larger than 1GB
  - Reading or writing to the same file by multiple clients (10s, 100s, and 1000s)
- Multi-terabyte or multi-petabyte datasets.
- Environments that need a single storage namespace optimizable for a mix of large and small files.

Benefits

The key benefits of using BeeGFS on NetApp include:

- Availability of verified hardware designs providing full integration of hardware and software components to ensure predictable performance and reliability.

- Deployment and management using Ansible for simplicity and consistency at scale.
- Monitoring and observability provided using the E-Series Performance Analyzer and BeeGFS plugin. For more information, see Introducing a Framework to Monitor NetApp E-Series Solutions.
- High availability featuring a shared-disk architecture that provides data durability and availability.
- Support for modern workload management and orchestration using containers and Kubernetes. For more information, see Kubernetes meet BeeGFS: A tale of future-proof investment.

Architecture overview

The BeeGFS on NetApp solution includes architectural design considerations used to determine the specific equipment, cabling, and configurations required to support validated workloads.

Building block architecture

The BeeGFS file system can be deployed and scaled in different ways depending on the storage requirements. For example, use cases primarily featuring numerous small files will benefit from extra metadata performance and capacity, whereas use cases featuring fewer large files might favor more storage capacity and performance for actual file contents. These multiple considerations impact different dimensions of the parallel file system deployment, which adds complexity to designing and deploying the file system.

To address these challenges, NetApp has designed a standard building block architecture that is used to scale out each of these dimensions. Typically, BeeGFS building blocks are deployed in one of three configuration profiles:

- A single base building block, including BeeGFS management, metadata, and storage services
- A BeeGFS metadata plus storage building block
- A BeeGFS storage only building block

The only hardware change between these three options is the use of smaller drives for BeeGFS metadata. Otherwise, all configuration changes are applied through software. And with Ansible as the deployment engine, setting up the desired profile for a particular building block makes configuration tasks straightforward.

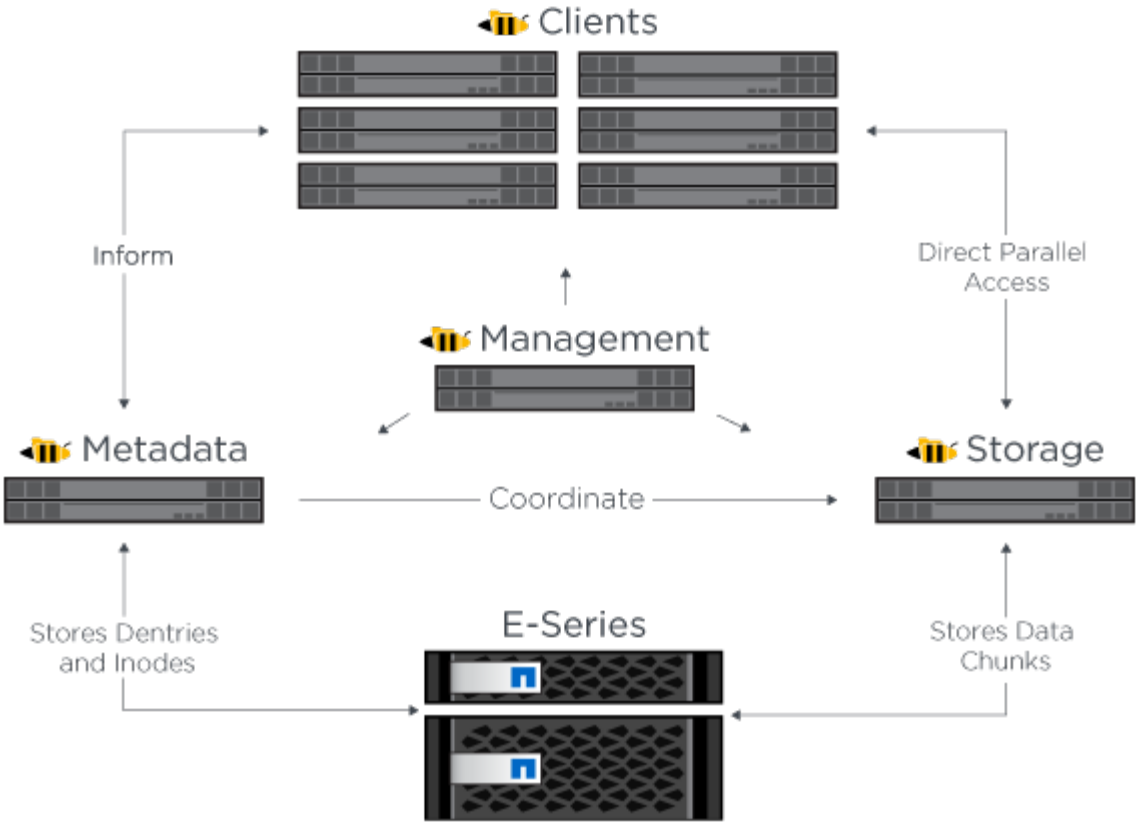
For further details, see Verified hardware design.

File system services

The BeeGFS file system includes the following main services:

- **Management service.** Registers and monitors all other services.
- **Storage service.** Stores the distributed user file contents known as data chunk files.
- **Metadata service.** Keeps track of the file system layout, directory, file attributes, and so on.
- **Client service.** Mounts the file system to access the stored data.

The following figure shows BeeGFS solution components and relationships used with NetApp E-Series systems.



As a parallel file system, BeeGFS stripes its files over multiple server nodes to maximize read/write performance and scalability. The server nodes work together to deliver a single file system that can be simultaneously mounted and accessed by other server nodes, commonly known as *clients*. These clients can see and consume the distributed file system similarly to a local file system such as NTFS, XFS, or ext4.

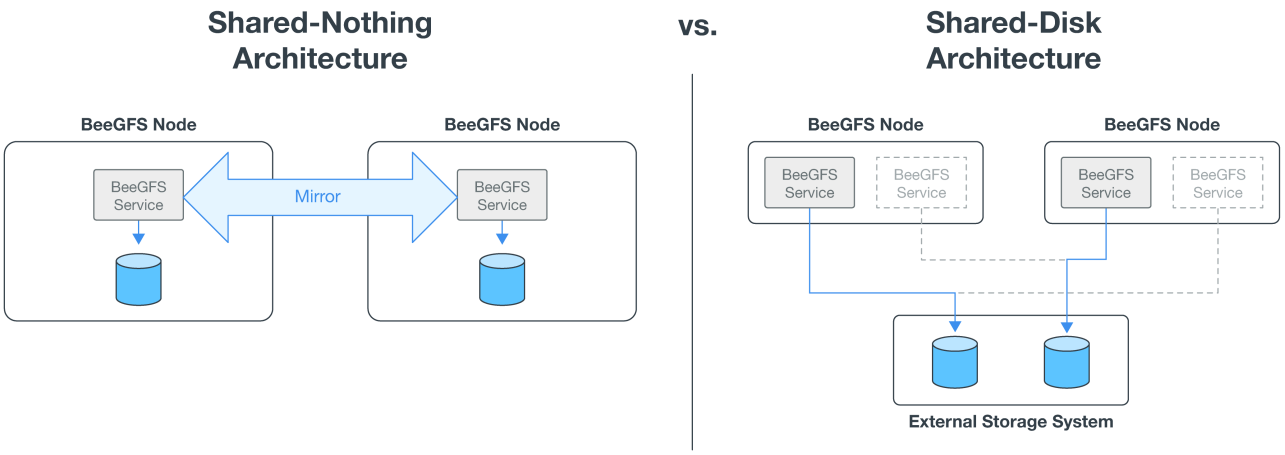
The four main services run on a wide range of supported Linux distributions and communicate via any TCP/IP or RDMA-capable network, including InfiniBand (IB), Omni-Path (OPA), and RDMA over Converged Ethernet (RoCE). The BeeGFS server services (management, storage, and metadata) are user space daemons, while the client is a native kernel module (patchless). All components can be installed or updated without rebooting, and you can run any combination of services on the same node.

HA architecture

BeeGFS on NetApp expands the functionality of the BeeGFS enterprise edition by creating a fully integrated solution with NetApp hardware that enables a shared-disk high availability (HA) architecture.

> While the BeeGFS community edition can be used free of charge, the enterprise edition requires purchasing a professional support subscription contract from a partner like NetApp. The enterprise edition allows use of several additional features including resiliency, quota enforcement, and storage pools.

The following figure compares the shared-nothing and shared-disk HA architectures.



For more information, see Announcing High Availability for BeeGFS Supported by NetApp.

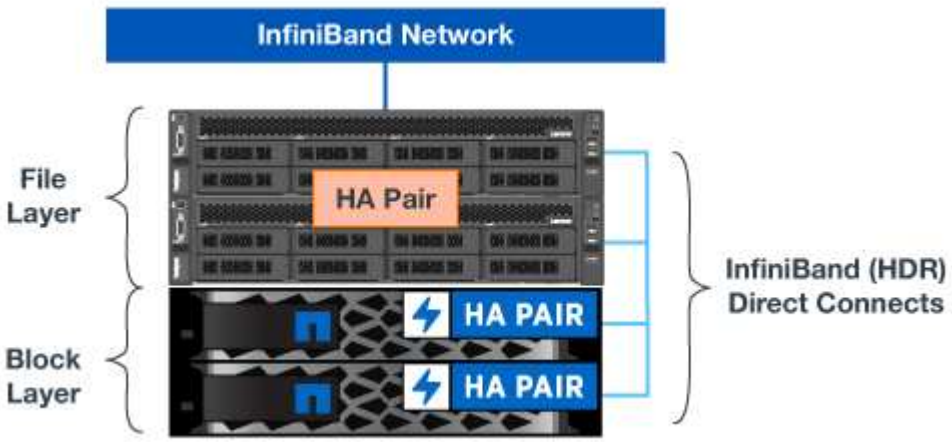
Verified nodes

The BeeGFS on NetApp solution has verified the nodes listed below.

Node	Hardware	Details
Block	NetApp EF600 Storage System	A high-performance, all-NVMe 2U storage array designed for demanding workloads.
File	Lenovo ThinkSystem SR665 V3 Server	A two-socket 2U server featuring PCIe 5.0, dual AMD EPYC 9124 processors. For more information about the Lenovo SR665 V3, see Lenovo's website.
	Lenovo ThinkSystem SR665 Server	A two-socket 2U server featuring PCIe 4.0, dual AMD EPYC 7003 processors. For more information about the Lenovo SR665, see Lenovo's website.

Verified hardware design

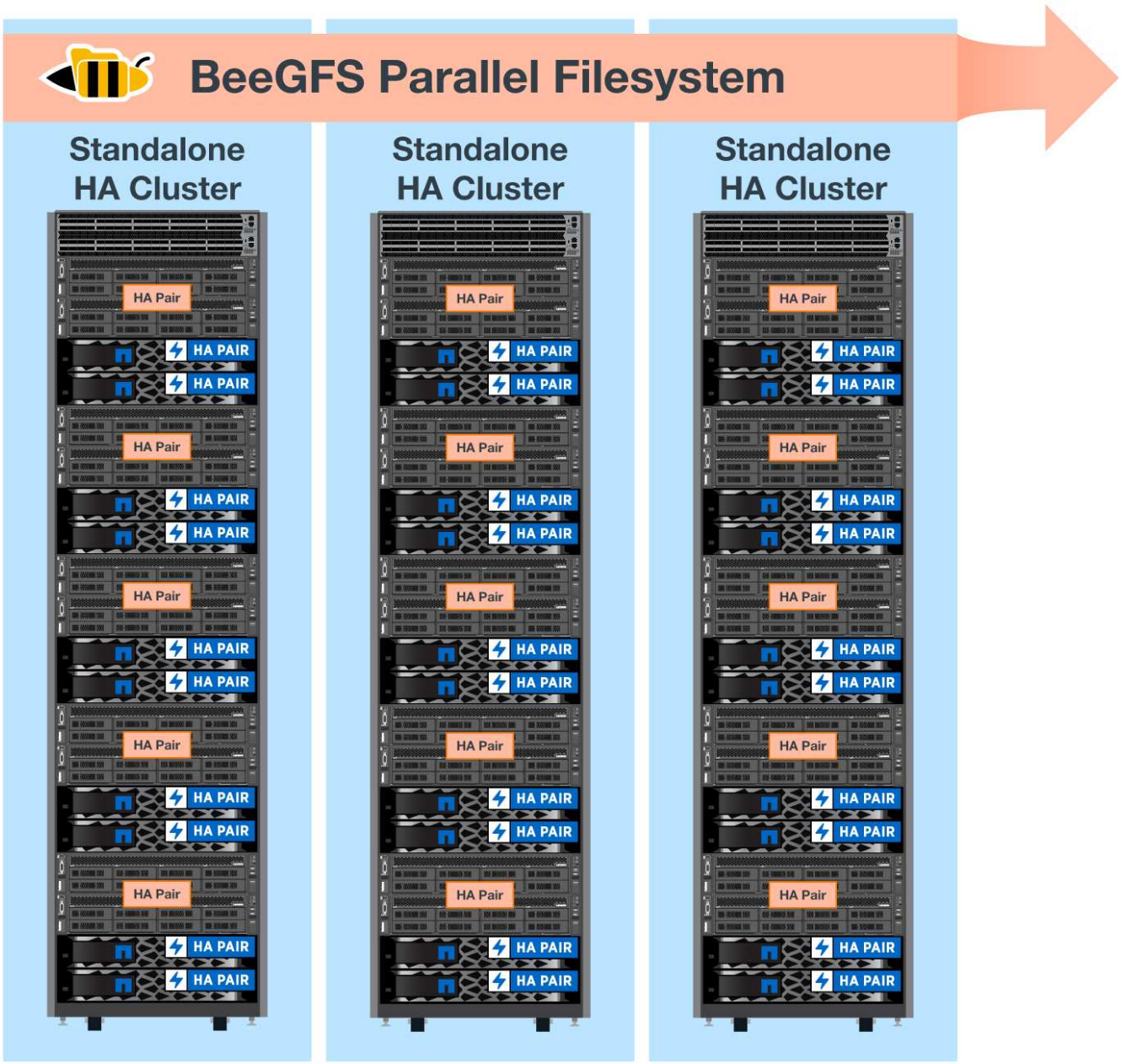
The solution's building blocks (shown in the following figure) uses the verified file node servers for the BeeGFS file layer and two EF600 storage systems as the block layer.



The BeeGFS on NetApp solution runs across all building blocks in the deployment. The first building block deployed must run BeeGFS management, metadata, and storage services (known as the base building block). All subsequent building blocks can be configured through software to extend metadata and storage services, or to provide storage services exclusively. This modular approach enables scaling the file system to the needs of a workload while using the same underlying hardware platforms and building block design.

Up to five building blocks can be deployed to form a standalone Linux HA cluster. This optimizes resource management with Pacemaker and maintains efficient synchronization with Corosync. One or more of these standalone BeeGFS HA clusters are combined to create a BeeGFS file system that is accessible to clients as a single storage namespace. On the hardware side, a single 42U rack can accommodate up to five building blocks, along with two 1U InfiniBand switches for the storage/data network. See the graphic below for a visual representation.

A minimum of two building blocks are required to establish quorum in the failover cluster. A two-node cluster has limitations that might prevent a successful failover from occurring. You can configure a two-node cluster by incorporating a third device as a tiebreaker; however, this documentation does not describe that design.



Ansible

BeeGFS on NetApp is delivered and deployed using Ansible automation, which is hosted on GitHub and Ansible Galaxy (the BeeGFS collection is available from Ansible Galaxy and NetApp's E-Series GitHub). Although Ansible is primarily tested with the hardware used to assemble the BeeGFS building blocks, you can configure it to run on virtually any x86-based server using a supported Linux distribution.

For more information, see Deploying BeeGFS with E-Series Storage.

Technical requirements

To implement the BeeGFS on NetApp solution, ensure your environment meets the technology requirements outlined in this document.

Hardware requirements

Before you begin, ensure that your hardware meets the following specifications for a single second-generation building block design of the BeeGFS on NetApp solution. The exact components for a particular deployment may vary based on customer requirements.

Quantity	Hardware component	Requirements
2	BeeGFS file nodes	Each file node should meet or exceed the specifications of the recommended file nodes to achieve expected performance. **Recommended file node options:** • **Lenovo ThinkSystem SR665 V3** ◦ **Processors:** 2x AMD EPYC 9124 16C 3.0 GHz (configured as two NUMA zones). ◦ **Memory:** 256GB (16x 16GB TruDDR5 4800MHz RDIMM-A) ◦ **PCIe Expansion:** Four PCIe Gen5 x16 slots (two per NUMA zone) ◦ **Miscellaneous:** ▪ Two drives in RAID 1 for OS (1TB 7.2K SATA or better) ▪ 1GbE port for in-band OS management ▪ 1GbE BMC with Redfish API for out-of-band server management ▪ Dual hot swap power supplies and performance fans
2	E-Series block nodes (EF600 array)	**Memory:** 256GB (128GB per controller). **Adapter:** 2-port 200Gb/HDR (NVMe/IB). **Drives:** Configured to match desired metadata and storage capacity.
8	InfiniBand host card adapters (for file nodes).	Host card adapters may vary based on the file node's server model. Recommendations for verified file nodes include: • **Lenovo ThinkSystem SR665 V3 Server:** ◦ MCX755106AS-HEAT ConnectX-7, NDR200, QSFP112, 2-port, PCIe Gen5 x16, InfiniBand Adapter
1	Storage network switch	The storage network switch must be capable of 200Gb/s InfiniBand speeds. Recommended switch models include: • **NVIDIA QM9700 Quantum 2 NDR InfiniBand switch** • **NVIDIA MQM8700 Quantum HDR InfiniBand switch**

Cabling requirements

Direct connections from block nodes to file nodes.

Quantity	Part Number	Length
8	MCP1650-H001E30 (NVIDIA Passive Copper cable, QSFP56, 200Gb/s)	1m

Connections from file nodes to the storage network switch.
Select the appropriate cable option from the following table according to your InfiniBand storage switch. The recommended cable length is 2m; however, this may vary based on the customer's environment.

Switch model	Cable Type	Quantity	Part Number
NVIDIA QM9700	Active Fiber (including transceivers)	2	MMA4Z00-NS (multimode, IB/ETH, 800Gb/s 2x400Gb/s Twin-port OSFP)
		4	MFP7E20-Nxxx (multimode, 4-channel-to-two 2-channel splitter fiber cable)
		8	MMA1Z00-NS400 (multimode, IB/ETH, 400Gb/s Single-port QSFP-112)
	Passive Copper	2	MCP7Y40-N002 (NVIDIA passive copper splitter cable, InfiniBand 800Gb/s to 4x 200Gb/s, OSFP to 4x QSFP112)
NVIDIA MQM8700	Active Fiber	8	MFS1S00-H003E (NVIDIA Active Fiber cable, InfiniBand 200Gb/s, QSFP56)
	Passive Copper	8	MCP1650-H002E26 (NVIDIA Passive Copper cable, InfiniBand 200Gb/s, QSFP56)

Software and firmware requirements

To ensure predictable performance and reliability, releases of the BeeGFS on NetApp solution are tested with specific versions of software and firmware components. These versions are required for implementation of the solution.

File node requirements

Software	Version
Red Hat Enterprise Linux (RHEL)	RHEL 9.4 Server Physical with High Availability (2 socket). **Note:** File nodes require a valid Red Hat Enterprise Linux Server subscription and the Red Hat Enterprise Linux High Availability Add-On.
Linux Kernel	5.14.0-427.42.1.el9_4.x86_64
HCA Firmware	**ConnectX-7 HCA Firmware** FW: 28.45.1200 PXE: 3.7.0500 UEFI: 14.38.0016 **ConnectX-6 HCA Firmware** FW: 20.43.2566 PXE: 3.7.0500 UEFI: 14.37.0013

EF600 block node requirements

Software	Version
SANtricity OS	11.90R3
NVSRAM	N6000-890834-D02.dlp
Drive Firmware	Latest available for the drive models in use. See the E-Series disk firmware site.

Software deployment requirements

The following table lists the software requirements deployed automatically as part of the Ansible-based BeeGFS deployment.

Software	Version
BeeGFS	7.4.6
Corosync	3.1.8-1
Pacemaker	2.1.7-5.2
PCS	0.11.7-2
Fence Agents (redfish/apc)	4.10.0-62
InfiniBand / RDMA Drivers	MLNX_OFED_LINUX-23.10-3.2.2.1-LTS

Ansible control node requirements

The BeeGFS on NetApp solution is deployed and managed from an Ansible control node. For more information, see the Ansible documentation.

The software requirements listed in the following tables are specific to the version of the NetApp BeeGFS Ansible collection listed below.

Software	Version
Ansible	10.x
Ansible-core	>= 2.13.0
Python	3.10
Additional Python packages	Cryptography-43.0.0, netaddr-1.3.0, ipaddr-2.2.0
NetApp E-Series BeeGFS Ansible Collection	3.2.0

Copyright information

Trademark information