



Back to Basics: Data Compression



Sandra Moulton

Technical Marketing Engineer
NetApp

This article is the sixth installment of Back to Basics, a series of articles that discusses the fundamentals of popular NetApp® technologies.

Data compression technologies have been around for a long time, but they present significant challenges for large-scale storage systems, especially in terms of performance impact. Until recently, compression for devices such as tape drives and VTLs was almost always provided using dedicated hardware that added to expense and complexity.

NetApp has developed a way to provide transparent inline and postprocessing data compression in software while mitigating the impact on computing resources. This allows us to make the benefits of compression available in the Data ONTAP® architecture at no extra charge for use on existing NetApp storage systems. Since compression's initial release in Data ONTAP 8.0.1, the feedback we've received on it has been very positive. It has been licensed on systems in a broad range of industries. Forty percent of those systems use compression on primary storage, while 60% use it for backup/archiving.

NetApp data compression offers significant advantages, including:

- **Works in conjunction with other industry-leading NetApp storage efficiency technologies.** Compression, coupled with other efficiency technologies such as thin provisioning and deduplication, significantly reduces the total amount of storage you need, lowering both your capital and operating expenses. Total space savings can range up to 87% for compression alone depending on the application. Using other efficiency technologies can make these savings even higher.
- **Has minimal performance impact.** While all compression technologies come with some performance penalty, NetApp has taken great care to minimize the impact while maximizing space savings.
- **Needs no software licensing fees.** The NetApp data compression capability is standard in Data ONTAP 8.1. No license is required, so you incur no additional hardware or software costs when enabling compression.
- **Works for both primary and secondary storage.** You can enable compression on primary storage volumes, secondary storage volumes, or both.
- **Requires no application changes.** Compression is application-transparent, so you can use it with a variety of applications with no code changes required.
- **Space savings inherited during replication and with use of DataMotion.** When you replicate a compressed volume using volume SnapMirror or move a volume with DataMotion™, blocks are copied in their compressed state. This saves bandwidth and time during data transfer and space on target storage while avoiding the need to use additional CPU

Explore

More Back to Basics

Learn the fundamentals of NetApp core technologies by reading other chapters in this series:

- [Chapter 1: Thin Provisioning](#)
- [Chapter 2: Deduplication](#)
- [Chapter 3: FlexClone](#)
- [Chapter 4: Volume SnapMirror](#)
- [Chapter 5: RAID-DP](#)
- [Chapter 6: Data Compression](#)

cycles to compress the same blocks again.

This chapter of Back to Basics explores how NetApp data compression technology is implemented, its performance, applicable use cases, choosing between inline and postprocess compression, and best practices.

How Compression Is Implemented in Data ONTAP

NetApp data compression reduces the physical capacity required to store data on storage systems by compressing data within a flexible volume (FlexVol® volume) on primary, secondary, and archive storage. It compresses regular files, virtual local disks, and LUNs. In the rest of this article, references to files also apply to virtual local disks and LUNs.

NetApp data compression does not compress an entire file as a single contiguous stream of bytes. This would be prohibitively expensive when it comes to servicing small reads from part of a file, since it would require the entire file to be read from disk and uncompressed before servicing the read request. This would be especially difficult on large files. To avoid this, NetApp data compression works by compressing a small group of consecutive blocks at one time. This is a key design element that allows NetApp data compression to be more efficient. When a read request comes in you only need to read and decompress a small group of blocks, not the entire file. This approach optimizes both small reads and overwrites and allows greater scalability in the size of the files being compressed.

The NetApp compression algorithm divides a file into chunks of data called “compression groups.” Compression groups are a maximum of 32KB in size. For example, a file that is 60KB in size would be contained within two compression groups. The first would be 32KB and the second 28KB. Each compression group contains data from one file only; compression is not performed on files 8KB or smaller.

Writing Data. Write requests are handled at the compression group level. Once a group is formed a test is done to decide if the data is compressible. If it doesn't yield savings of at least 25%, it is left uncompressed. Only when the test says the data is compressible is the data written to disk compressed. This optimizes the savings while minimizing resource overhead.

Since compressed data contains fewer blocks to be written to disk, it can reduce the number of write I/Os required for each compressed write operation. This not only lowers the data footprint on disk but can also reduce the time needed to perform backups.

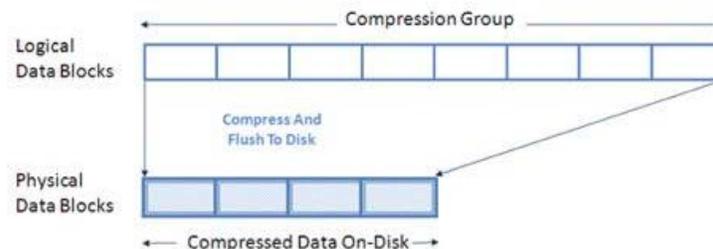


Figure 1) Files are divided into chunks of data called compression groups, which are tested for compressibility. Each compression group is flushed to disk in either a compressed or an uncompressed state depending on the results of the test.

Reading Data. When a read comes in for compressed data, Data ONTAP reads only the compression groups that contain the requested data, not the entire file. This can minimize the amount of I/O needed to service the request, overhead on system resources, and read service times.

Inline Operation. When NetApp data compression is configured for inline operation, data is compressed in memory before it is written to disk. This can significantly reduce the amount of write I/O to a volume, but it can also affect write performance and should not be used for performance-sensitive applications without prior testing.

For optimum throughput, inline compression compresses most new writes but will defer some more performance-intensive compression operations—such as partial compression group overwrites—until the next postprocess compression process is run.

Postprocess Operation. Postprocess compression can compress both recently written data and data that existed on disk prior

to enabling compression. It uses the same schedule as NetApp deduplication. If compression is enabled, it is run first followed by deduplication. Deduplication does not need to uncompress data in order to operate; it simply removes duplicate compressed or uncompressed blocks from a data volume.

If both inline and postprocess compression are enabled, then postprocess compression will only try to compress blocks that are not already compressed. This includes blocks that were bypassed during inline compression such as partial compression group overwrites.

Compression Performance and Space Savings

Data compression leverages the internal characteristics of Data ONTAP to perform with high efficiency. While NetApp data compression minimizes performance impact, it does not eliminate it. The impact varies depending on a number of factors, including type of data, data access patterns, hardware platform, amount of free system resources, and so on. You should test the impact in a lab environment before implementing compression on production volumes.

Postprocess compression testing on a FAS6080 yielded up to 140MB/sec compression throughput for a single process with a maximum throughput of 210MB/sec with multiple parallel processes. On workloads such as file services, systems with less than 50% CPU utilization have shown increased CPU usage of ~20% for datasets that were 50% compressible. For systems with more than 50% CPU utilization, the impact may be more significant.

Space savings that result from the use of compression and deduplication for a variety of workloads are shown in Figure 2.

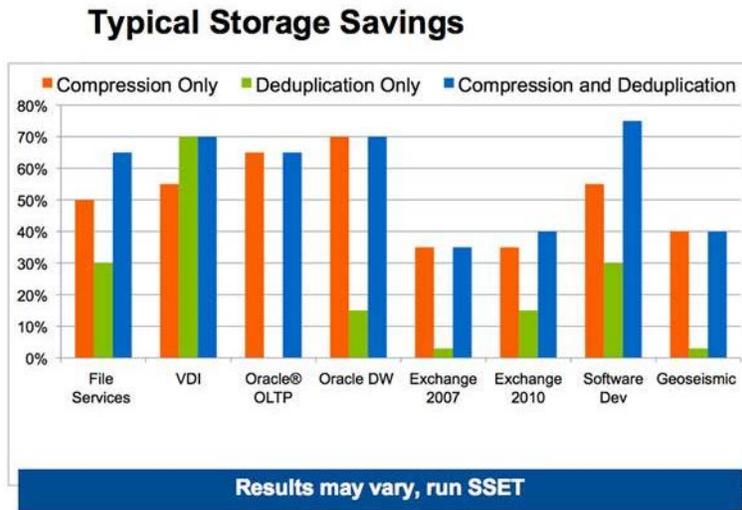


Figure 2) Typical storage savings that result from using compression, deduplication, or both.

Use Cases

As I've already discussed, choosing when to enable compression or deduplication involves balancing the benefits of space savings versus the potential performance impact. It is important to gauge the two together in order to determine where compression makes the most sense in your storage environment.

Database backups (and backups in general) are a potential sweet spot for data compression. Databases are often extremely large, and there are many users who will trade some performance impact on backup storage in return for 65%+ capacity savings. For example, one test backing up four Oracle volumes in parallel, with inline compression enabled, resulted in 70% space savings with a 35% increase in CPU and no change in the backup window. Most of us would probably choose to enable compression in such a circumstance given the significant savings and assuming the CPU resources are available on target storage. When sizing new storage systems for backup, you may want to verify that adequate CPU is available for compression.

Another possible use case is file services. In testing using a file services workload on a system that was ~50% busy with a

dataset that was 50% compressible, we measured only a 5% decrease in throughput. In a file services environment that has a 1-millisecond response time for files, this would translate to an increase of only 0.05 ms, raising the response time to 1.05 ms. For a space savings of 65%, this small decrease in performance might be acceptable to you. Such savings can be extended even further by replicating the data using NetApp volume SnapMirror® technology, which saves you network bandwidth and space on secondary storage. (Secondary storage inherits compression from primary storage in this case, so no additional processing is needed.) In this scenario you would have:

- 65% storage capacity savings on primary storage
- 65% less data sent over the network for replication
- 65% faster replication
- 65% storage capacity savings on secondary storage

There are many other use cases in which compression makes sense, and we have a number of tools and guides that can help you decide which use cases are best for your environment. For primary storage, consider using compression for the following use cases:

- File services
- Geoseismic
- Test and development

For backup/archive storage, consider using compression for the following use cases:

- File services
- Geoseismic
- Virtual servers
- Oracle OLTP
- Oracle Data Warehouse
- Microsoft® Exchange 2010

Using Compression

NetApp data compression works on all NetApp FAS and V-Series systems running Data ONTAP 8.1 and above. Data compression is enabled at the volume level. This means that you choose which volumes to enable it on. If you know a volume contains data that is not compressible, you shouldn't enable compression on that volume. Data compression works with deduplication and thus requires that deduplication first be enabled on the volume. A volume must be contained within a 64-bit aggregate—a feature that was introduced in Data ONTAP 8.0. Starting in Data ONTAP 8.1, there are no limits on volume size beyond those imposed by the particular FAS or V-Series platform you use. You can enable and manage compression using command line tools or NetApp System Manager 2.0.

Before enabling compression, NetApp recommends that you test to verify that you have the required resources and understand any potential impact. Factors that affect the degree of impact include:

- The type of application
- The compressibility of the dataset

- The data access pattern (for example, sequential versus random access, the size and pattern of the I/O)
- The average file size
- The rate of change
- The number of volumes on which compression is enabled
- The hardware platform—the amount of CPU/memory in the system
- The load on the system
- Disk type and speed
- The number of spindles in the aggregate

In general, the following rules of thumb apply:

- Compression performance scales with the type of hardware platform.
- More cores deliver more throughput.
- Faster cores mean less impact on throughput.
- The more compressible the data, the lower the impact on performance.

Choosing Inline or Postprocess Compression

When configuring compression, you have the option of choosing immediate, inline compression in conjunction with periodic postprocess compression, or postprocess compression alone. Inline compression can provide immediate space savings, lower disk I/O, and smaller Snapshot™ copies. Because postprocess compression first writes uncompressed blocks to disk and then reads and compresses them at a later time, it is preferred when you don't want to incur a potential performance penalty on new writes or when you don't want to use extra CPU during peak hours.

Inline compression is most useful in situations in which you aren't as performance sensitive and can accept some impact on write performance, and have available CPU during peak hours. Some considerations for inline and postprocess compression are shown in Table 1.

Table 1) Considerations for the use of postprocess compression alone versus inline plus postprocess compression.

Goal	Recommendation
Minimize Snapshot space.	Inline compression will minimize the amount of space used by Snapshot copies.
Minimize disk space usage on qtree SnapMirror or SnapVault® destinations.	Inline compression provides immediate savings with minimal impact on backup windows. Further, it takes up less space in the snapshot reserve.
Minimize disk I/O.	Inline compression reduces the number of new blocks written to disk.
Avoid performance impact for new writes.	Postprocess compression writes the new data to disk uncompressed without any impact on the initial write performance. You can then schedule when compression occurs to recover space.

Minimize impact on CPU during peak hours.

Postprocess compression allows you to schedule when compression occurs, minimizing the impact of compression during peak hours.

Data Compression and Other NetApp Technologies

NetApp data compression works in a complementary fashion with NetApp deduplication. This section discusses the use of data compression in conjunction with other popular NetApp technologies.

Snapshot Copies. Snapshot copies provide the ability to restore data to a particular point in time by retaining blocks that change after the Snapshot copy is made. Compression can reduce the amount of space consumed by a Snapshot copy since compressed data takes up less space on disk.

Postprocess compression is able to compress data locked by a Snapshot copy, but the savings are not immediately available because the original uncompressed blocks remain on disk until the Snapshot copy expires or is deleted. NetApp recommends completing postprocess compression before creating Snapshot copies. For best practices on using compression with Snapshot copies refer to [TR-3958](#) or [TR-3966](#).

Volume SnapMirror. Volume SnapMirror operates at the physical block level; when deduplication and/or compression are enabled on the source volume, both the deduplication and compression space savings are maintained over the wire as well as on the destination. This can significantly reduce the amount of network bandwidth required during replication as well as the time it takes to complete the SnapMirror transfer. Here are a few general guidelines to keep in mind.

- Both source and destination systems should use an identical release of Data ONTAP.
- Compression and deduplication are managed only on the source system—the flexible volume at the destination system inherits the storage savings.
- Compression is maintained throughout the transfer, so the amount of data being transferred is reduced, thus reducing network bandwidth usage and the time to complete the transfer.
- SnapMirror link compression is not necessary, since the data has already been compressed with NetApp data compression.

The amount of reduction in network bandwidth and SnapMirror transfer time is directly proportional to the amount of space savings. As an example, if you were able to save 50% in disk capacity, then the SnapMirror transfer time would decrease by 50% and the amount of data you would have to send over the wire would be 50% less.

Qtree SnapMirror and SnapVault. Both qtree SnapMirror and SnapVault operate at the logical block level; source and destination storage systems run deduplication and data compression independently, so you can run them on either or both according to your needs. This allows you to compress and/or deduplicate your qtree SnapMirror and/or SnapVault backups even when the source data is not compressed or deduplicated. Postprocess compression and dedupe automatically run after a SnapVault transfer completes unless the schedule is set to manual.

Cloning. NetApp FlexClone® technology instantly creates virtual copies of files or data volumes—copies that don't consume additional storage space until changes are made to the clones. FlexClone supports both deduplication and compression. When you enable compression on the parent volume of a clone, the savings are inherited on the clone. Or you can enable compression on a clone volume so that new data written to the clone benefits from compression without affecting the parent copy.

Conclusion

NetApp data compression technology is an important storage efficiency tool that can be used to optimize space savings on both primary and secondary storage. For complete information on all the topics discussed in this chapter and more, refer to [TR-3958: NetApp Data Compression and Deduplication Deployment and Implementation Guide: Data ONTAP 8.1 Operating in 7-Mode](#) and [TR-3966: NetApp Data Compression and Deduplication Deployment and Implementation Guide: Data ONTAP 8.1 Operating in Cluster-Mode](#).

› **Got opinions about Compression?**

Ask questions, exchange ideas, and share your thoughts online in NetApp Communities.



By Sandra Moulton, Technical Marketing Engineer

Since joining NetApp two years ago, Sandra has focused almost exclusively on storage efficiency, specializing in deduplication and data compression; she has been responsible for developing white papers, best-practice guides, and reference architectures for these critical technologies. Sandra has over 20 years of industry experience, including performing similar functions at other leading Silicon Valley companies.



Quick Links

- › [Tech OnTap Community](#)
 - › [Archive](#)
 - › [User Groups](#)
 - › [PDF](#)
-