Dynamic Solutions
INTERNATIONAL

Compression vs. Deduplication

Confused about compression versus deduplication? Let's see if we can help.

What is compression?

Simply put compression is taking an item and manipulating it such that the item takes up less space that it originally did. Think in terms of a propane tank. The gas in the tank has been squeezed into a much smaller space than if it were a free gas. For us in the data processing industry the concept is much the same. We deal with masses of data. That data takes up a lot of room on limited storage resources such as SSD, rotating disk, and tape (physical and virtual). It would be more cost effective and efficient if we could manipulate that data such that it used less storage space. So we want to compress all the bits of data such that a fewer number of bits stores the same data.

Data compression is implemented in many different ways. Generally data compression products are based on the type of data being compressed. For example to compress audio files (music) you might use MP3, AAC, FLAC or many others. These compression formats can be generally defined as lossy or lossless. Lossy compression tends to remove information such that when the file is uncompressed it is not a duplicate of the original. Lossless compression will return a file to its original state when uncompressed. As you might expect lossy compression generally is able to compress a file into a smaller state than lossless compression routines. It is up to the listener whether they can hear a difference in a lossy compressed audio file as opposed to a lossless compressed file.

When dealing with business data lossy compression is not an option. Products such as PKZIP and Gzip are available for X86 based systems which provide lossless compression for data files. Systems that do not use X86 architecture may have a proprietary compression scheme or can send the data to an X86 subsystem for compression.

So how much compression can I expect? As the car dealers say "your mileage will vary". The input data will determine how much compression is seen. Normal lossless compression has an average of a 2 to 1 (2:1) compression across many different data types. Some tape drives state a 2.5:1 compression ratio. Generally if you have a file with lots of repeating data (like lots of space characters) you will get a higher compression as compared to a file with random data. What this means is that textual type of files will compress much better than files that have audio or video information.

Compression
IS TAKING AN ITEM AND MANIPULATING IT SUCH THAT THE ITEM TAKES UP LESS SPACE THAT IT ORIGINALLY DID.

Deduplication
IS THE PROCESS OF FINDING DUPLICATE/REPEATED BLOCKS OF INFORMATION AND ONLY STORING ONE COPY OF THAT BLOCK.

Visit us at DynamicSolutions.com or call +1 303.754.2000 to speak directly with a storage systems specialist.

Dynamic Solutions International
373 Inverness Parkway, Suite 110
Englewood, CO 80112

+1 303.754.2000
Fax 303.754.2009
DynamicSolutions.com

What is deduplication?

Deduplication is the process of finding duplicate/repeated blocks of information and only storing one copy of that block. Sound a bit like compression? It is in that it is a mechanism to reduce the amount of storage required for your data. However the key difference is one of scope. Most compression systems work with a file, a set of files, or possibly a tape in a given instance. Most deduplication systems work over an entire storage environment for an extended period of time.

Deduplication systems are implemented in many different ways but the basic concepts remain. A deduplication system reviews the blocks of data in its environment. For each block of data it calculates a hash value. What is a hash value? A hash value is just a really large number. Due to the way the hash is calculated that number (hash value) should be unique for each unique block of data. The deduplication system then looks in its hash table to see if it has seen this value before. If the value is found the data block is removed and it is replaced by a pointer to the previously seen data block. If the hash value is not found the data block is stored and the hash value added to the hash table. The file itself is now just a string of hash values that can be rebuilt by looking up the blocks that the hash values refer to. Simple?

There are as many different implementations of deduplication systems as there are vendors of deduplication systems. What sizes of blocks are used for the hash value calculation? Are blocks merged to create larger blocks? What hash algorithm is used? Are multiple hash algorithms used to prevent hash value collisions? (A hash collision is where two different data blocks could result in the same hash value. Depending on the hash algorithm unlikely but a mathematical possibility.) How is the hash table indexed? These implementation aspects will affect the performance of the system and the amount of deduplication (compression) seen.

So how much deduplication can I expect? Remember the "mileage will vary"? The pattern of the data and the implementation of the deduplication system will determine the deduplication ratio. (Sound like compression ratio?) Since deduplication works by only storing a block of data once even if seen many times backup and archive applications tend to see the best deduplication ratios. The first time you back up your environment to a deduplication system the deduplication ratio will be very small. Over time as you continue to backup or archive your environment the ratio should improve as the same files are repeatedly seen. Our experience has shown anywhere from 4:1 to 20:1 deduplication ratios in real life. We have seen some vendors claim up to 200:1 but that is with very specific data sets. Even at 4:1 that is twice the storage savings over simple compression.

So if I have deduplication I don't need compression anymore? Not exactly. Again depending on a vendors implementation the stored blocks of data can be compressed for even more space saving. So employing compression and deduplication is the best of both worlds.

Besides storage space reduction are there any other benefits to deduplication? Yes, if you are copying your data over a network to another location. Most deduplication systems that have a remote replication feature will only send new blocks of data rather than a whole file or system. This can dramatically reduce bandwidth requirements.

Finally, for secure environments the data blocks can be encrypted. Encryption generally can be processor intensive. On deduplication systems we are only encrypting blocks we have not been previously stored thus saving on processing overhead.

Conclusions

Deduplication sounds great. Why doesn't everybody use it? Unlike compression which I can use as needed on a single file or a selection of files deduplication requires a commitment to using it across an entire storage environment. If you want to deduplicate your backup environment you deduplicate your entire backup environment. Deduplication also requires special hardware and/or software to perform the deduplication and store the results. How this is implemented will vary from vendor to vendor.

Deduplication can dramatically reduce storage requirements for some storage needs. It does require a commitment and specialized tools. It is not something that a site could implement themselves.