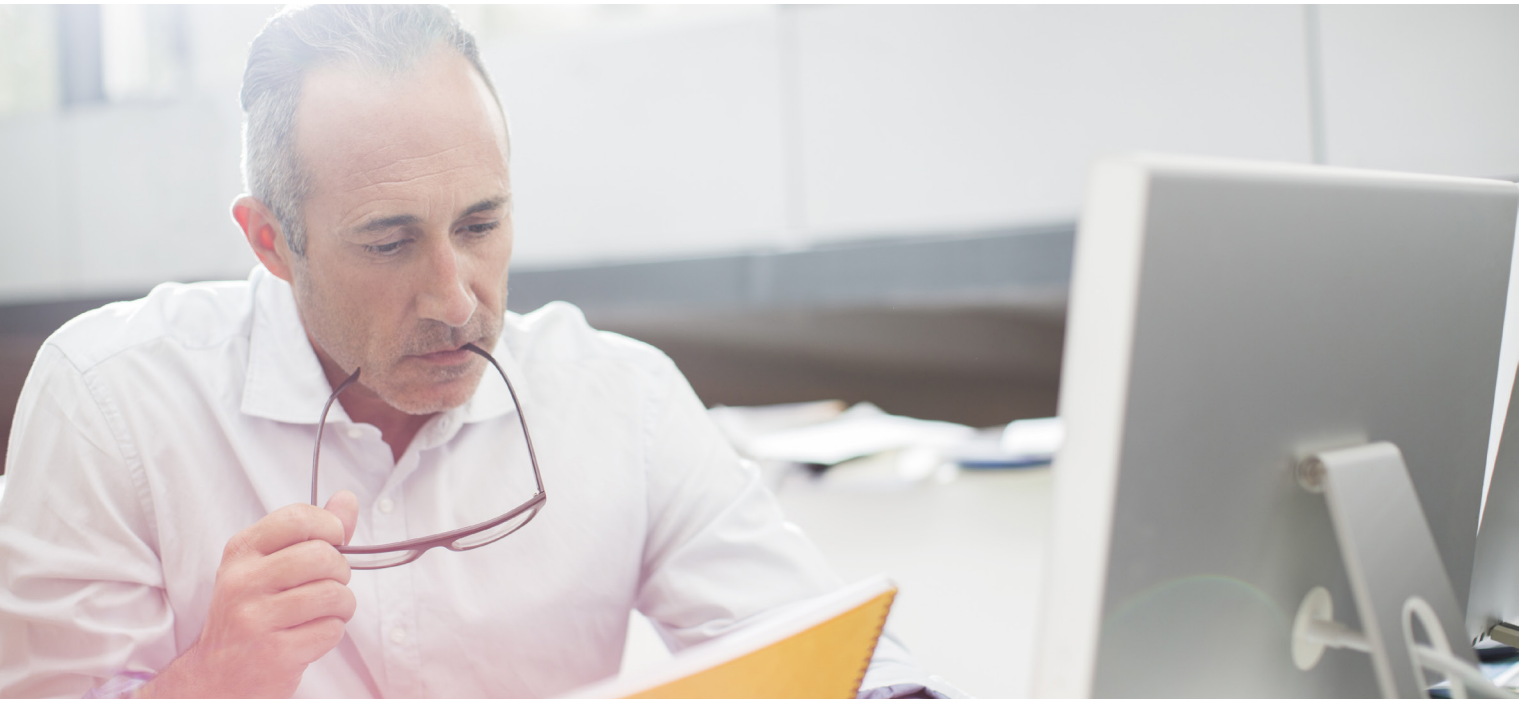
Quest

Deduplication: The hidden truth and what it may be costing you

Not all deduplication technologies are created equal. See why choosing the right one can save storage space by up to a factor of 10.

By Adrian Moir, Senior Consultant, Product Management, Quest Software



Data growth is something we all have to contend with. We have to store more and more data for longer and longer time to satisfy business or regulatory requirements. The global data-sphere will likely grow to 163 zettabytes by 2025, by which time almost half of it will reside in the enterprise.[1]

Protecting your data is a primary business function. It comes with its own set of challenges. Add those to the ever growing and shifting data sets, and it starts to eat into resources quickly.

While protecting data of course is necessary, we do not want it to be a burden. We all look for solutions that can move data from point A to point B as fast as physics will allow! We look towards technology to make the experience better, go faster, store more while forever reducing costs. Quite a challenge indeed.

We have seen many technologies evolve that help smooth the path to secondary data storage optimization, with a view to reducing the resource burdens of backup and recovery. One such technology is deduplication. While deduplication has been available in the market for some time, not all solutions are equal, and it is often useful to understand exactly what the impact of each is likely to have on your available resources and budgets.

NOT ALL THINGS ARE CREATED EQUAL

This is the same for deduplication technologies. Deduplication has become a byword for data reduction; however, using a single word to describe different methodologies can be misleading at best.

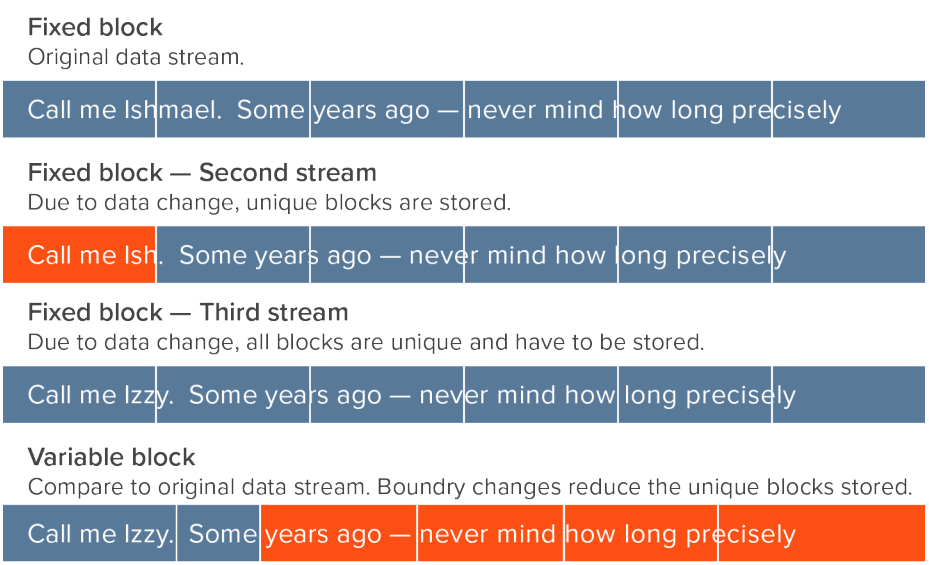


Figure 1: Fixed-block de-duplication limitations.

Fixed-block deduplication is a good start, however it is limited. It only works well on some data types that are stored directly on the file system.

Let us consider some different data reduction technologies:

Compression

A good example is lossless data compression. It relies on exploiting statistical redundancy without losing data, so you can 'undo' the compression and reinstate the data. We have been using these techniques for many years, now found as default in technologies. For example, a GIF image will use LZW (Lempel-Ziv-Welch) compression to reduce the file size of the image without losing information.

Single instancing

Single instancing describes a storage methodology quite well. If I store the same file twice, then I will keep one and reference it for the other. However, this only works if the files are identical. Change a small thing in the file and the whole file is stored again. Now this is great for systems that have their structure populated with the same content, production storage perhaps, but not so well for general data protection.

Fixed-block de-duplication

This is our first foray into what would be described as a proper deduplication methodology. Fixed-block deduplication takes a stream of data and slices it up into blocks of a fixed size. We call these 'chunks' of data.

These chunks are then closely compared using several methods, and if they are deemed the same, then only a single 'chunk' is stored and a reference is kept for each subsequent match. Sound familiar? Think of this as single instancing but at a subfile level, looking at the blocks that make up that file.

This methodology sounds better, however it is limited. It works well on some data types that are stored directly on the file system, since they are byte-aligned and have their file systems written in 4k, 8k, 32k chunks, like virtual machines, for example.

In this case, fixed-block solutions can be very effective. However, this does not work well on a mixture of data where those boundaries are not consistent or if they are backed up through different types of software, which changes the alignment. We know data is anything but consistent, and depending on its type, will have a different make up, block size, byte alignments and content.

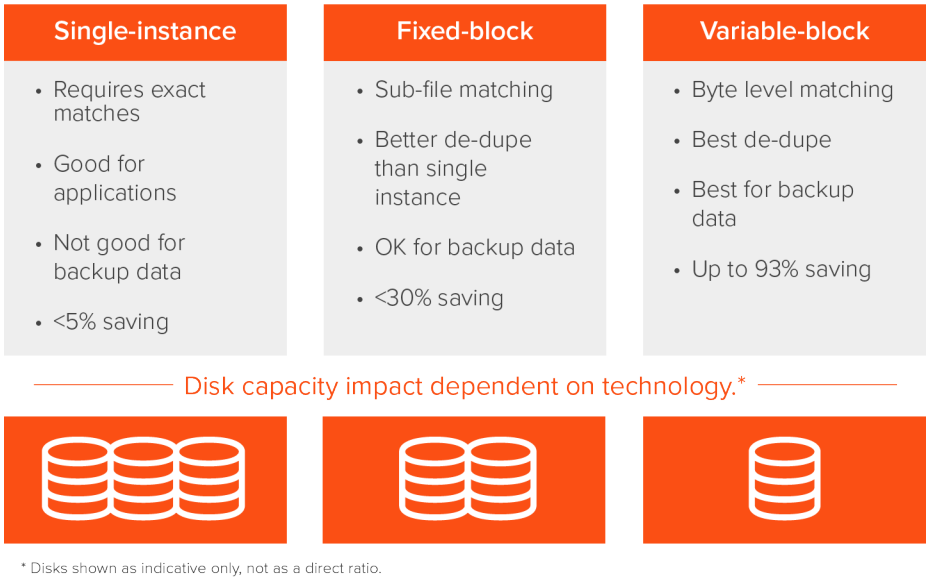


*Figure 2: **Variable block de-duplication results in the best disk space and cost savings.***

Variable-block deduplication

Again, we now look to technology to solve the issues surrounding fixed–block deduplication and be able to handle different data types and still get the data reduction we are looking for. Enter the mathematicians!

How do you determine what data in a file will match any data you have stored, without having to inspect every bit. The need to match a changing data block size into a new 'chunk' that can be matched again for a different data type is a big challenge. To overcome this, the use of a sliding window, variable chunk dedupe with Rabin fingerprinting is used.

To find duplicates, the data is fed through a Rabin fingerprinting algorithm and a chunk is created when a unique set of bytes are found. Because the data is variable and is calculated over a sliding window, that same set of bytes of data (variable chunks of data) can be identified again and again, no matter where it is in the stream of data. This prevents the need to have the data nicely aligned to catch duplicates, like with fixed-block dedupe systems.

It does not matter if data is added before or after in the stream of data. Once the chunk is identified, a SHA 1 hash is generated and stored in the dedupe dictionary. Any future occurrences of the data will be found, since the same chunk will be identified. The SHA1 hashes will match, and the data will be deduplicated.

The hashes are then checked in a bloom[2] filter to provide a faster check to see if that hash is already known and if the 'chunk' needs to be stored or referenced. Continually matching hashes allows 'chunks' to be created when no matches are found and references to be created when matches are found.

Overall, the sliding window provides more matches and therefore reduces the unique data that has to be stored, saving significant storage space over other dedupe technologies.

Content-aware variable-block deduplication

There is a next step that can provide even further savings, being aware about the content of the stream itself. For example, Quest® QoreStor™ adds a content-aware algorithm to its variable-size chunking. The algorithm identifies patterns in the data — in spite of the shifting that results from additions or deletions in the data stream — then aligns the block start- and endpoints to duplicate chunks, while identifying only the changed chunks as unique.

Overall, variable-block deduplication provides more matches and therefore reduces the unique data that has to be stored.

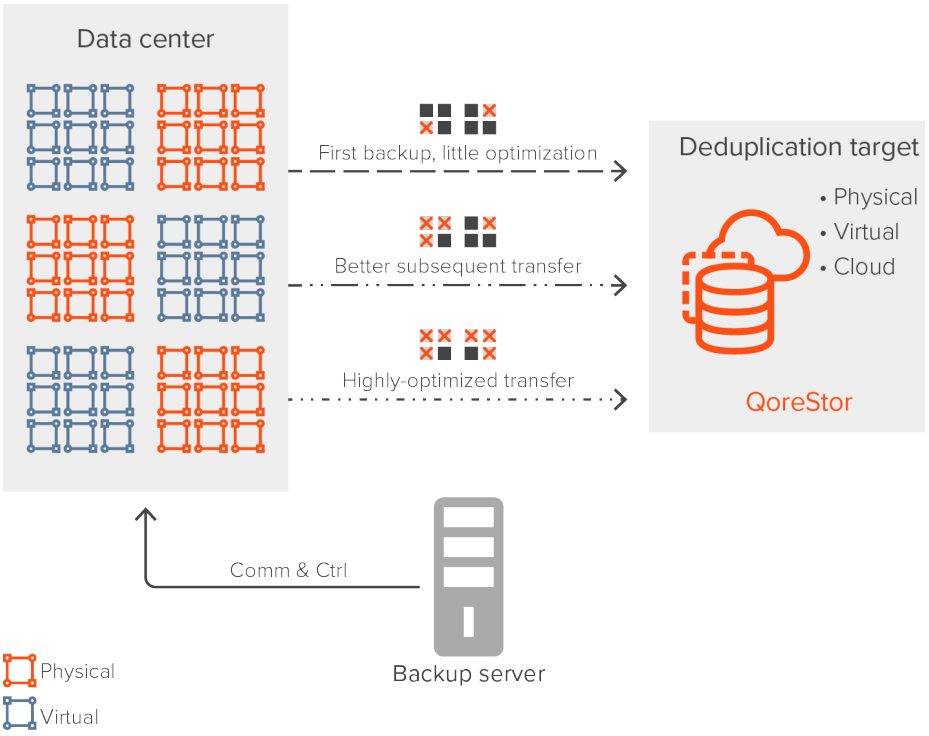


*Figure 3: **Direct-to-target transfer technologies offer better source-side deduplication.***

With source-side matching, there is less data being sent over the wire, allowing you to run more backups in parallel.

ALL GOOD THINGS START AT THE SOURCE

As we have already seen, deduplication technologies come in various forms. Another key attribute is to consider where the deduplication takes place and what impact this might have. We have only viewed technologies that are commonly used on a target storage solution.

Let us consider the issue of backing up multiple systems, applications and data in parallel.

We all want to move as much data as possible with the least impact to our users. This being the case, we define a window in time where this can take place, commonly known as the 'backup window.' This is often outside of business hours to minimize impact due to the nature of moving large quantities of data over networks – the same ones users use to access the applications.

In today's world, that data transfer is becoming an issue where more and more businesses operate on a 24/7 basis, and there is very limited room for impacts caused by backups. There is actually more than one issue here. The first is the application server impact on saturating its network connection. The second is the target device having its network connection saturated with all the data from multiple machines.

Because time is a finite quantity, you get to a point where you become limited in what you can achieve in your set backup window.

Improvements can be made by utilizing some resources on each of the source application servers that are being backed up. Looking at the first issue, by matching hashes of data in the backup stream at the source server, only unique data needs to be sent over the network. While this creates a little overhead on the source application servers, it's nowhere near the impact of a flooded network connection.

Using the source-side matching technique also impacts the second issue. Because there is less data being sent over the wire, there is an ability in being able to run more backups in parallel. If this is done, then either the backup window can be reduced, or

more backups can be done in the same timeframe, greatly improving throughput. This is a technology known as source-side deduplication.

THE COST OF COMPROMISING ON TECHNOLOGY

The impact of choosing the wrong deduplication technology can be best explained by the size and associated cost. Let us consider a data set size of 100TB. To keep this simple, let's say you have weekly backups that you wish to keep to 12 weeks. Each week would add 100TB onto the target storage. Without any data reduction technology, after three months. 1.2PB of storage would be consumed before the initial backups begin to expire.

Let us consider a fixed-block deduplication technology that would give you a 30% reduction. That would reduce the storage requirement to 840TB. That is a good savings, but it is still sizable, and probably a costly solution.

Now let us consider a variable, content-aware de-duplication technology. With up to a 93% reduction, this would bring your storage requirement to just 84TB. As you can see, choosing the right deduplication technology can save storage space by up to a factor of 10.

Now consider the storage cost difference between 840Tb and 84Tb. Factoring in the cost for power and maybe rack space, if you are using a co-located data center, the potential savings becomes apparent very quickly. Choosing the wrong technology has many implications beyond just disk space. Costs are always being squeezed and need to be carefully considered. Choose the right technology and you can reduce your costs as well as your data footprint.

ONE OF THESE THINGS IS NOT LIKE THE OTHER

When considering an optimized storage solution for your backup needs, research carefully so that you have a good understanding of what technologies can be applied to your environment. Over the years, there has been a lot of emphasis on deduplication appliances, fixed hardware and software designed to provide a solution.

While this has been a good approach and has served well, the nature of infrastructure is changing. We now find ourselves in a world that requires us to be more agile in terms of deployment, infrastructure and location.

Cost implications are becoming more prevalent too. Having to refresh the storage every 3-5 years and paying for the software element of an appliance all over again is not ideal. This is one of the areas where Quest QoreStor is different.

QoreStor is a true software-defined secondary storage technology. It is software that can be installed directly on physical server hardware, in a virtual machine or in a public or private cloud environment, public or private. Because QoreStor is not tied to any specific hardware platform (like dedupe appliances), you can choose the backup software you would like, from your favorite vendors, with your normal discounts. When it comes time to refresh the hardware, do just that, only pay for new hardware and keep the existing QoreStor software license.

Being software defined does not mean that QoreStor is limited either. It utilizes variable-block deduplication technology and offers inline and source-side deduplication capabilities. On top of that, it also compresses the deduplicated 'chunks' and then encrypts them for security.

QoreStor also offers data separation using its 'Storage Group' technology, allowing differing compression and encryption settings for different data sets or users. All of this is included in the QoreStor license, along with the ability to replicate from one QoreStor instance to another, leveraging secure, optimized data transfer.

If you are looking to optimize your secondary storage for backup and recovery, hold optimized multiple copies of data in other locations, including the cloud, in a cost effective and secure way, then look no further than Quest QoreStor.

Quest QoreStor can be installed directly on hardware, in a VM or in a cloud environment, and works with your existing backup software.

1 Reinsel, David; Gantz, John; Rydning, John; "Data Age 2025: The Evolution of Data to Life-Critical," IDC, April 2017.

2 A Bloom filter is a space-efficient probabilistic data structure, conceived by Burton Howard Bloom in 1970, that is used to test whether an element is a member of a set.

ABOUT QUEST

At Quest, our purpose is to solve complex problems with simple solutions. We accomplish this with a philosophy focused on great products, great service and an overall goal of being simple to do business with. Our vision is to deliver technology that eliminates the need to choose between efficiency and effectiveness, which means you and your organization can spend less time on IT administration and more time on business innovation.

If you have any questions regarding your potential use of this material, contact:

Quest Software Inc.
Attn: LEGAL Dept
4 Polaris Way
Aliso Viejo, CA 92656

Refer to our website (www.quest.com) for regional and international office information.