White paper

Digital Preservation White Paper Series

Digital Preservation Storage

Preservica

1. Introduction

Storage of digital content underpins the entire endeavor of Digital Preservation. It refers to the writing of digitally encoded data to some form of persistent media such that it can be read back at a later time. Technologies have evolved, media have become faster, smaller and cheaper over time, but the basic acts of reading and writing data to persistent storage media are something that has been part of computing and information technology for as long as these fields have existed.

This does not, however, imply that this is a completely “solved” problem. There are any number of ways that content written to some persistent medium can be lost over time. In this White Paper, we will discuss some of those, and what needs to be done to mitigate these for effective long term Digital Preservation.

2. Risks to Long Term Storage

There are a number of risks to long term storage. Some risks are dependent on the physical medium of the device itself, some on the degree to which it can be considered to be managed, and some are applicable to all devices.

If we consider two Hard Disk Drives (HDDs) for example. They are likely to have similar risk profiles for mechanical failure. However, if we consider the risk of loss or damage, if one is part of a disk array in a centralized file server and the other is USB pluggable external hard drive, those risk profiles look very different.

Conversely, a HDD that is part of a file server, and a tape cartridge that is part of a tape library are likely to have similar risk profiles for loss, but very different profiles for technological failure.

In all cases, logical access (who can read and write data) presents a risk.

In the rest of this section we’ll consider some of these risks in more detail.

2.1 Complete Loss

This occurs when the physical media itself is lost. Without the media, we are clearly unable to read any of the data it contains.

This loss may be accidental, and in that form is primarily a risk for removable media, like USB sticks and floppy disks, but any form of decentralized storage, like disk drives built into desktop or laptop computers, is susceptible to this.

If we consider the threat of malicious actors deliberately removing media, and not just media being misplaced, then this is applicable to all forms of storage.
Careful asset tracking and management, and physical security measures will act as mitigating factors for this risk.

Taking regular backups of storage media will also mitigate this risk by limiting the amount of information that is only stored on the lost device.

2.2 Small Scale Physical Destruction

This occurs when the physical media itself is damaged or destroyed. Depending on the extent of the damage , we may be able to partially read the data still but will still likely suffer data loss. If the contents of the storage device are encrypted at a device level, even a small amount of damage, potentially something which changes the value of a single bit (see Spontaneous Bit Flips) will likely render the entire contents unreadable.

As with complete loss, this can occur accidentally, and again, removable and decentralized media are at the greatest risk of this kind of information loss.

We do also need to consider the potential of a malicious actor for this kind of information loss, and again, even centralized storage is at risk of that.

Physical security measures will act as mitigation, as will ensuring that removable media are sufficiently robust for the environments that they are intended to be used in.

Taking regular backups of storage media will also mitigate this risk by limiting the amount of information that is only stored on the damaged or destroyed device.

2.3 Large Scale Physical Destruction

This occurs when the environment in which the physical media resides is subject to damage or destruction. This covers a wide range of scenarios with scale varying from a single server room up to entire datacenters. An example of this would be the 2008 fire at Universal Studios, Hollywood, where hundreds of thousands of assets, including both analogue and digital recordings were destroyed [1].

There are accidental, or natural, risk factors for this which can have both internal and external factors.

Internally, faulty plumbing or air-conditioning could lead to flooding, electrical faults could lead to fires, structural faults could lead to collapsing walls or server racks. Generally, internal risks can be mitigated with good room and building planning, and regular maintenance of building services. Physical security measures will also help mitigate the risk of a malicious actor triggering this kind of damage.

External risk factors are harder to mitigate as these can include environmental issues like floods, wildfires [2] or earthquakes (clearly, depending on location). It is much harder, and costlier to mitigate these kinds of risks as they really depend on the infrastructure and location of the building itself.

The same is true when we consider what a “malicious actor” looks like in terms of external threat. Small scale loss or damage is really considering focused acts aimed directly at specific storage infrastructure, but for large scale physical destruction we are really considering major events like terrorist attacks, attacks from hostile nation states, or acts of war.

.Having backups is again a mitigation, but we are now thinking about large scale, off-site/second-site storage.

2.4 Obsolete Media

The risks considered so far assume that content is inaccessible because we either no longer have the physical carrier, or we have it but it is damaged in some way. However, it is possible that we have the physical carrier, intact, in good condition, but still cannot access the content because it relies on some other hardware that we cannot access .

As storage technologies have evolved, we have left behind a vast array of physical devices that were no longer fit for purpose. Examples include:

- Tapes, floppy disks and external disks of various form factors that either relied on discontinued “drives” to read them or outdated connection types to attach them to computing devices, e.g. [3] [4];
- Laser discs [5] and writable optical disks such as CDRW and DVDRW whose drives are no longer routine built-in to hardware such as laptop computers.

Even USB sticks are under threat from increasingly risk averse security mandates which automatically block read/write access to them [6].

If you no longer support or have access to the technology, the challenge with content on media such as this is sourcing third parties to retrieve the content, or provide access to the drives required. In this case risk mitigation is about maintaining networks and knowledgebases around where you might find support when needed.

Looking forward, the mitigation is ensuring that you have copies or backups of the data on alternative technologies.

2.5 Obsolete Technology Layers

When it comes to storage, we tend to ignore the software layer involved and consider obsolescence as a problem of the physical carrier and its physical connections. However, having content on an array of “hard-disks” does not mean that our content all carries the same risk profile.

At a logical level, content is written to file systems which overlay the physical disk. These are essentially a thin software layer, and as with all software, these file systems are subject to evolution and as new systems emerge, older systems gradually fall out of use and support, e.g. [7] [8] [9].

Again here, if you can no longer support or get access to the technology required to read the data on these devices, sourcing third party help is likely to be the best course of action. Maintaining networks and knowledgebases of where you might find that help is a way of mitigating the risk.

Looking forward, this risk is mitigated by ensuring that you have backups of the data on alternative technologies.

2.6 Altering Data

Many storage media are read-write, meaning that we can read back the data that is there, but also change or delete it. This means that all our content is vulnerable to being altered, corrupted or deleted in unauthorized ways.

This can happen accidentally by an unwitting user, or errant or inappropriately applied software, but can also be the action of a malicious actor, deliberately trying to remove, edit or corrupt data.

Whichever vector we consider for this risk, the requirement is access from the user or software to the content. This is typically mediated by several layers of software, down to the Operating System itself, and each of these layers will typically allow for some form of access control. Mitigating this risk involves ensuring that appropriate controls and permissions are in place on all data, and across all software and systems that can interact with it.

This can be further mitigated by having backups of all content stored on separate systems or in separate locations.
We can detect where such changes or deletions have occurred if we have up to date manifests of what we expect to have stored, including some form of checksum for each piece of content.

If we maintain multiple copies, we can even use these to repair damage when we detect it.

2.7 Spontaneous Bit Flips

Some storage media, such as hard-disks, are susceptible to random failures which cause an individual bit, or bits, to spontaneously change value (i.e. changing from a 0 to 1 or vice-versa). This is due to the underlying physical nature of the medium and how this can interact with the environment [10]. For example, hard-disks use magnetic polarity to encode 0s and 1s, and if they interact with a fluctuating magnetic field, these polarities can be reversed. Solid State Disks use transistors to store data, relying on the properties of voltage values across terminals to encode 0s and 1s, these values can be altered if they interact with external charged particles, such as cosmic rays [11].

In many ways, these errors have similar characteristics to deliberately or accidentally altered data, but because they do not depend on logical access to the content, mitigating the risk of their occurrence is not the same. Instead of ensuring we have logical protection around write access to the content, we must ensure we have appropriate physical shielding.

In practice, these are low probability events, and storage devices are designed with in-built error detection that means if the device is running, it will likely detect and recover from these events automatically. However, these should be considered long term failure modes for storage devices that are passively stored and not in active use.

We can further mitigate these risks by having backups of all content stored on separate systems. Again, if we maintain these multiple copies, and have detailed enough manifests, we can use these to repair this kind of damage when we detect it.

2.8 Technical Failure

All storage technologies have finite expected lifespans [12]. This might be due to mechanical wear and tear accumulating over time, such as failure of spinning platters or actuator arms in a hard disk drive; their physical and/or chemical composition, such as oxidation of the reflective layers of optical discs like CDs and DVDs; or other environmental factors such as bit-flipping in an inactive disk (as mentioned above).

Each technology has its own average lifetime, with some forms expected to last longer than others, but they all also have their own failure modes, with some failing “gracefully”, with warning over time, and some failing completely and suddenly.

Complicating this picture, each individual device may fail long before we would typically expect for any number of reasons. This might be due to the exact history of a single device, or may be due to something more systematic, like an error in the manufacturing process which means that whole batches of devices are liable to premature failure.

To mitigate against these risks, we need to ensure that we understand the expected lifetimes of the storage devices we use, and monitor their actual elapsed lifetime against this. Ideally, storage devices should be decommissioned before they are significantly older than their expected lifetime.

We can mitigate these risks by maintaining multiple copies of our content on devices with different underlying technologies.

2.9 Organizational Failure

We may opt to outsource our storage provision to a third party, signing up for contracted Service Level Agreements that guarantee the durability of storage, the redundancy (number of copies stored) and the diversity of underlying technologies used.

This moves much of the risk mentioned so far onto the third party who should provide Service Level Agreements documenting the contractually agreed levels of risk. However, it creates a risk on the provider themselves. If they are to cease trading, or fail as an organization in some way, all the data, including all copies, we have stored with them would be at risk of inaccessibility.

We can mitigate these risks by maintaining multiple of copies of our content with different third party providers.

3. Technical Risk Mitigation

In the previous section we outlined some of the risks to long term storage of content and briefly discussed in each case how we might mitigate these. In this section, we will describe the technical aspects of mitigation in a bit more detail. Specifically, we will look at how bit-level validation and duplication can help mitigate risks.

3.1 Bit Level Validation

Bit level validation is a means of being able to check and assert that the stream of digital 0s and 1s that we read today is exactly the same as what we originally received.

In order to do this we generate some metadata called a checksum, which is dependent on the values of all those 0s and 1s. This is achieved using a checksum algorithm that takes a copy of the digital content as its input, performs some operation on it, and results in a (generally) much smaller block of data that we can store independently of the content.

Once we have these checksums for all of our content, validating our storage is a case of periodically re-creating them to compare to the original stored version.

This is typically described as a “tamper-evident” mechanism; if the newly calculated checksums do not match the originals, we can see that the content was changed, but not necessarily how. If we perform this at regular intervals, we will have a window during which the content might have changed, but not exactly when. With a single copy of the content, we generally will not have enough information to repair any damage we detect.

3.1.1 Checksum Algorithms

Checksum algorithms are usually designed such that a small change in the input, e.g. changing a single 0 to a 1, will result in a significant change to the result, meaning that it should be easy to spot that two inputs are different.

Algorithms range from the simple “Parity Bit” (indicating whether there is an odd number of 1s in your input), right the way through to complex cryptographic hashes.

The nature of checksum algorithms are that they (tend to) reduce large inputs to a much smaller output. A file can have any number of bits, the checksum itself tends to have a fixed, small number. This means that there are more possible inputs than possible outputs, and so inevitably some (different) inputs will produce identical outputs. This means that there are always cases where a checksum algorithm will fail to distinguish between different inputs. These are called checksum “collisions”.

As a general rule, the more complicated the checksum algorithm, the fewer “collisions” are theoretically possible, and the more reliable the checksum is as a guarantee of fixity. This is traded off with higher computational costs to generate the checksum, and longer checksum values.

If we only concern ourselves with the possibility that some digital content has changed accidentally (either through inadvertent user action, or some bit-flipping process), then simpler checksums that are quicker to compute are likely to reliable enough. Some widely implemented algorithms in this class include Adler-32 and Cyclic Redundancy Checks (CRCs).

However, we typically also need to guard against the possibility of malicious changes to the data, and in this case, even complex cryptographic checksums are not guaranteed to be reliable. Cryptographic checksums in common use for fixity checking include MD5, SHA-1 and SHA-2, in increasing degree of complexity and thus computational cost. Vulnerabilities have been published which make it possible to intentionally craft two files with identical MD5 values, or identical SHA-1 values. This would allow a malicious actor to swap those two files without the change being detected. So far, there is no equivalent demonstrated vulnerability in SHA-2 algorithms.

These algorithms are used in cryptography for operations such as signing encrypted messages, which underly all secure communications, so the economic and political incentives to cracking them are strong. However, in that function, they tend to be used in isolation. One way to benefit from the quicker computation time of a vulnerable algorithm such as MD5 is to calculate multiple checksums using multiple algorithms. A collision attack that produces two different files with identical MD5 values and identical SHA-1 values is a lot less incentivized and as such, far less likely to be developed.

Preservica supports calculating checksums in any combination of MD5, SHA-1, SHA-256 (a 256 bit variant of SHA-2) and SHA-512 (a 512 bit variant of SHA-2). Checksums can be provided as part of a submission to Preservica, in which case Preservica will always check content against these and refuse to ingest any content that does not match its supplied checksum. If checksums are not provided, Preservica will calculate them as part of the ingest workflow and use these values for on-going validation.

Preservica supports on-going Fixity Checking that can be configured to periodically check every file in the repository.

3.2 Timestamping

Timestamping provides an additional layer of provenance to an assertion that digital content has not been changed on the basis of its checksum.

A message is sent to a Timestamping Authorit y containing some details of the digital content (potentially file names, but at least some persistent identifier) and the checksums calculated for it. A response is issued repeating the message and the time at which it was received.

At some later point, we can produce the digital content, calculate a checksum, and demonstrate that not only is the checksum unchanged, but that it is the same as it was at the point of timestamping.

Digital Timestamping and Timestamping Authorities are described in a number of standards, such as ANSI X9.95 [13], or the European Union eIDAS Regulations [14].

Digital Time Stamping authorities generally rely on a chain of trust including technologies like SSL Certificates, which are themselves issued by trusted authorities. This means they are subject to the same long-term considerations such as dealing with revocation of certificates, and the longevity of the authorities, which tend to be corporations, and what to do if they no longer exist.

Although we would normally assume that the Timestamping Authority is a trusted third party, technologies such as blockchain can provide this kind of evidence in situations of zero trust. Publishing details including the checksum onto an immutable ledger such as a blockchain can provide the same evidence of provenance, without having to trust any single person or organization.

3.3 Adding Resilience

Validation, and timestamping can provide evidence of whether contents have or have not changed, but on their own they cannot guard against change. The only protection against change, damage or loss of content from a particular storage device is another undamaged copy of the content. This is referred to as redundancy.

Having multiple independent copies of your content enables self-healing storage capabilities. When a checksum verification turns up a mismatch, the system can check other copies of the same content for a copy that matches, and use this to repair the damaged copy. The more copies that exist, the more storage devices have to fail or have been corrupted before there is no remaining good copy. Obviously, this comes at the cost of storing the same data multiple times, so there is clearly a trade-off to make.

When considering the risks detailed in the previous section, it is clear there are also considerations around how storage is likely to have failed, and what constitutes “independent” in this context.

Having multiple copies of a file on the same hard-drive (or SSD, or USB stick, or tape) is likely to protect you against accidental modification or deletion, but it does not protect you against the failure of the device as a whole.

Having copies on different machines in the same building (servers in the same data-center) protects against a machine failure, loss or destruction, but not against a site-wide fire, or bomb-strike.

Having copies in different building protects against site-wide damage, but if both buildings are in the same geographic region, they may be damaged by the same flood or earthquake.

And it’s not just a case of providing increasing geographic separation. Two data-centers on opposite sides of the world might be using hard-disks supplied by the same manufacturer, with the same systemic defect. Or they might use different underlying technologies, disk and tape perhaps, but be operated by the same third-party provider, and be subject to the same risk of that provider’s bankruptcy or organizational failure.

True independence here really means having different environmental, technological, geo-political and organizational risk factors for different copies. Of course, that comes at a cost, in terms of having to pay for multiple times the volume of storage, losing “economy of scale” that you might be able to get with a single provider, and in the maintenance of ensuring that you have sufficiently different underlying technologies and technology providers. In practice, this is all subject to a cost-risk trade-off, at some point, the cost of the storage is greater than the benefit of guaranteeing storage and/or the cost of losing the content. In some scenarios, these costs might be easily calculable in terms of possible fines, penalties or contractual liabilities should the content be lost. In many others, the costs and benefits of the content are more intangible, but they should always be considered.

Preservica supports writing to and reading from multiple storage devices. Preservica supports Amazon S3 at various tiers of storage (hot/cold etc), Azure Blob Storage and Generic third party S3 services. Enterprise on-Premise customers can also use any form of storage that can be mounted as a POSIX drive, which includes hard-disks, SDDs, and Hierarchical Content Management systems ultimately backed by tape libraries.

Where content exists on multiple devices known to Preservica, the on-going Fixity Check can be configured to automatically repair damaged or missing files from another known good copy.

4. Permanent Storage Media

For most of this paper, we have been discussing storage media that are essentially commodity products not specialized for long term storage and archiving needs. There are however a number of technologies at various stages of maturity that seek to address at least some of the risks described.

4.1 M-Disc

M-Disc (https://www.mdisc.com) is a form of optical disc that shares a form factor with CD and DVD and is compatible with existing CD/DVD hardware. Instead of using a layer of organic dye into which the data is burned, M-Disc uses a "rock-like" layer into which the data is burned. This creates a physical rather than chemical change in the data layer, which enables a much longer life span (claimed to be up to 1,000 years using standardized testing).

4.2 Paper/Film

Physical long-term preservation processes are longer established than digital because we have been dealing with physical storage for a lot longer. This lends itself to approaches that leverage those processes, effectively "printing" digitally encoded data onto physical carriers such as paper, film or microfiche.

These include approaches where the content is encoded in some of barcode/QR code structure, alongside details of the algorithm required to write the software to decode that again. Such an approach means that the ability to re-read the content is not coupled to the existence of specific hardware, generally the ability to magnify an image on the film/paper, scan or image it, and then process it on some form of computer. The complexity of reading content from systems like these mean that whilst they are good for long-term storage of infrequently accessed data, they are not so well-suited to providing fast and frequent access.

Services such as Piql (https://piql.com/) and Eupalia (https://eupalia.com/en/) offer storage solutions based on this approach.

4.3 DNA Storage

DNA storage involves encoding digital data on to synthesized strands of DNA. This provides very high storage densities in a medium that is known to have incredibly long stable life times, albeit at the cost of slow read and write times. The current nature of DNA storage means that reading content back is essentially a destructive act. The strands of DNA are stored in a physical container, and both that and the contents must be destroyed as part of reading back. This also means that all data in a single device must be read at the same time, even when retrieval is intended for only a subset of the data.

The destructive nature of reading means that multiple copies of the data must be stored across multiple devices, and although the storage density means that this is not really an issue in terms of physical volume, it does add to the write time. This is a still advancing field of study, so this is likely to improve over time.

4.4 Other Permanent Storage

Research and development is ongoing on various other durable storage media, such as diamond and silica crystal. These require specialist laser systems to control the structures of the crystals, and while they are a long way from mass commercialization, they promise very high storage densities and very long, stable and durable lifespans.

“5D” glass storage involves using lasers to manipulate nano-structures within glass-like crystals. The storage densities and expected lifetimes are very high, at the cost of long write times. This is a still advancing field of study and so this is likely to improve over time.

4.5 Issues

There are some key issues that these share.

Genuine archival/long term storage represents a small fraction of the overall demand for storage, and so these technologies are, almost by definition, niche. This means that there is little economic incentive to really drive a competitive marketplace that would unlock economies of scale, making the technologies cost-effective, ensuring a diversity of supply, and building confidence in the longevity of the physical form factors, which ultimately determines our ability to read back the content.

The intended long term nature of these technologies means that they tend to be WORM-like (Write Once, Read Many). This is a benefit in many ways as it removes an entire class of risk (the data being changed once written), but it means that genuine use cases for deletion of data are hard to support. If you are required to delete only part of the data written to a particular physical device, you can only do this by copy all the data you need to keep to a second device and then physically destroying the first.

Finally, although we might have confidence in the long term durability of the media, we still need to be able to read it back. This is not too much of a concern for the paper/film approaches as the key technologies involved (magnification, scanning/imaging) are likely to continue to be exist, and visible, non-encoded instructions can clearly indicate the purpose and encoding of the data.

DNA sequencing is another technology that has uses beyond storage and which is likely to continue to exist long into the future, although without knowing why the strands were synthesized it’s reasonable to question whether future readers would understand that there was anything to decode, let alone how to decode it.

By utilizing existing commodity optical storage technologies, M-discs have immediate commercial viability, but the downside is that the long-term ability to read them back is coupled to the lifetime of those technologies.

More exotic storage technologies that rely on more complex technologies, e.g. specialist laser systems, and that have limited other uses are certainly at risk of achieving very durable storage with no equivalent long-term assurances that the data can be accessed.

5. Data Structure

Having decided on the storage media to use, there are also potentially decisions to take around storage structure. Most storage media will enable some form of logical structuring such that relationships between content can form part of the presentation to end users, although in some cases it might be necessary, or desirable to utilize external indexes or databases to track those relationships.

5.1 Object Storage & File Systems

In a true object store, each piece of digital content (typically, each file) is considered to be completely independent and thus is just written to storage alongside some metadata about it, using an identifier that can be later used to retrieve it. The identifier is typically an abstract identifier such as a GUID, meaning that there are no issues around collisions of files having the same name.

When implementing such object storage, logical relationships between content need to be recorded and maintained in a separate index or database. These relationships are essentially metadata, and subject to change as archival practice, and understanding of the content evolves over time, by separating content storage (which should be immutable) from metadata storage (which should not) the risk of inadvertently altering content can be reduced.

Most users of computers are familiar with a File & Folder/Directory paradigm describing digital storage. In this view, related content (files) can be grouped into folders or directories where they are presented to the user together. These directories can then be hierarchically arranged, allowing for logical organization of the content. This is particularly useful for discoverability in situations where there is no external or embedded index that enables users to search for content.

Directory structures can mimicked in object stores by using common prefixes on the object ids to act as directories, this enables UIs to recreate a folder/file view of the object store.

It is important to remember, that for most storage media, the file system, and directory structures are actually software-level abstractions and not descriptions of how content is actually ordered and stored on the hardware. The binary data for files in the same folder are not guaranteed to be located anywhere near each other on the underlying medium.

Preservica implements its storage layer as object stores, writing content to a variety of different underlying storage technologies, alongside some fixed identifying metadata. Descriptive and Preservation metadata is separately stored in relational databases. This enables us to support a variety of storage technologies, some of which do not themselves support a traditional File/Folder based File System, to separate editing of metadata from editing anything on the storage device, to optimise performance for metadata reads (which are the most frequent interactions with the repository) and to ensure that for both content, and metadata there is a single source of truth about what should be stored.

5.2 Structured File Systems

There are a number of standards that utilize the file/folder nature of file systems to describe even more structured means of storage. Standards like OCFL (https://ocfl.io/) and BagIt (https://datatracker.ietf.org/doc/html/rfc8493) describe how content and metadata can be arranged within folders using fixed conventions to logically bring together all elements of the Information Package to be preserved.

These standards remove the dependency on the archival/preservation software to describe the archival holdings, although, as stated above, because files and folders are software-level abstractions in the file system software, they still do not completely remove software dependencies.

Access speeds to read metadata from storage like this can be slow when compared to reading from a read-optimized database or index, and so this approach will often necessitate duplicating or "de-normalizing" the storage of the metadata, which has additional storage costs, and additional software complexity for ensuring that the separate copies of the metadata remain in sync as they are edited over time.

5.3 Multi-Level or Hierarchical Storage

Many systems support being able to write to multiple different storage media, and some enable policies to determine which content is stored to which subset of available devices. This may be using fixed metadata such as hierarchical/structural information, or using calculated metrics such as access frequency.

Most tape storage systems for example are actually examples of Hierarchical Storage Management (HSM) systems. In these, a large tape library actually interfaces to a computer/server via a smaller hard-disk, which acts for a cache. When content is written by the computer, it is written to this disk, and additional software then copies it to tape asynchronously. When content is requested for read, the tape software copies from the tape back to this disk. This disk can thus act as a readily available cache for frequently accessed material, cutting read times considerably. This idea of a cache is common in many software applications, and even where the underlying storage is itself "on-line", it may be remote so caching can reduce latency in serving the read request.

This idea of multi-layer storage can be used to ensure that frequently accessed content is fast to retrieve while infrequently accessed content is placed on more cost-effective, but also to ensure that more expensive to generate, or valuable content is stored with more redundancy than content that is readily reproducible, less valuable, or already held elsewhere by other institutions.

Preservica supports granular storage policies to enable different content to be stored on different sets of storage devices. For example, allowing large preservation master files to be stored to cheaper cold storage while smaller access copies are stored on more expensive hot storage. Or allowing some higher-value content to be written to more devices.

6. Managing Change

All storage devices have finite lifespans, and care should be taken to ensure content exists on alternative devices before the end of life of a given device. This means that planning for the retirement of storage devices is part of the management of the storage lifecycle.

Copying large volumes of data is a time-consuming and expensive process, and is always subject to likely failures at some stage, whether that's a system crashing mid-way through, or a temporary communication failure. However, in principle, it is a relatively simple operation; it means copying content from one device to another and verifying the transfer using a checksum (similar to the validation discussed above), before finally removing it from the original (or destroying the original).

Device retirement is not the only time we may wish to move or copy content between storage devices however. Storage policies should be reviewed and revised on an on-going basis to ensure that changes in budget or risk appetite are reflected in how storage resources are used, or to take advantage of new or evolving technologies. Again, in principle, this will follow the same relatively simple operation as described above, but again, this should be carefully managed to cope with the risk of errors or failures.

Preservica supports adding and removing/retiring storage devices, as well as managing the movement of content between storage devices based in a granular storage policy. Storage policies can be described within the system and then automatically applied to existing content using built-in functionality. Preservica will perform the copy and verification before removing content from existing devices where relevant.

If you are using a particular system (be that a Content Management, Digital Asset Management, or Preservation System), you will also need to consider how you will access your content if and when you choose to exit from that system.

The same high-level "copy, verify, remove" process will be in use here, but you may need to consider that copying will require accessing remote systems via some interface, and may come with download/retrieval costs, or built-in rate limits.

Preservica has well-defined exit-plan arrangements and will work with customers to ensure timely access to all their content.

7. Further Reading

The Preservica White Paper Library covers all aspects of Digital Preservation showing the strategies and solutions required to ensure information is available in the future.

The following are also sources of information on the topics discussed:

[1] J. Rosen, "The Day The Music Burned," 11 June 2019. [Online]. Available: https://www.nytimes.com/2019/06/11/magazine/universal-fire-master-recordings.html.

[2] MSN, “Theosophical Society Historical Landmark in Altadena Destroyed by Eaton Fire: World’s Largest Archive of Theosophy Burns Down,” 13 01 2025. [Online]. Available: https://www.msn.com/en-us/society-culture-and-history/history/theosophical-society-historical-landmark-in-altadena-destroyed-by-eaton-fire-world-s-largest-archive-of-theosophy-burns-down/ar-BB1rpdcs. [Accessed 30 01 2025].
[3] Wikipedia, “Jaz Drive,” [Online]. Available: https://en.wikipedia.org/wiki/Jaz_drive. [Accessed 30 01 2025].
[4] Wikipedia, “Castlewood Orb Drive,” [Online]. Available: https://en.wikipedia.org/wiki/Castlewood_Orb_Drive. [Accessed 30 01 2025].
[5] Wikipedia, “LaserDisc,” [Online]. Available: https://en.wikipedia.org/wiki/Laser_disc. [Accessed 30 01 2025].
[6] Gartner, “Consult the Board: USB Blocking Policies and Challenges,” 24 07 2023. [Online]. Available: https://www.gartner.com/en/documents/4560399. [Accessed 30 01 2025].
[7] Wikipedia, “Extended File System,” [Online]. Available: https://en.wikipedia.org/wiki/Extended_file_system. [Accessed 30 01 2025].
[8] Wikipedia, “Apple DOS,” [Online]. Available: https://en.wikipedia.org/wiki/Apple_DOS. [Accessed 30 01 2025].
[9] Wikipedia, “Hierarchical File System (IBM MVS),” [Online]. Available: https://en.wikipedia.org/wiki/Hierarchical_File_System_(IBM_MVS). [Accessed 30 01 2025].
[10] A. Gordon, “What is a Bit Flip: Causes, Consequence and Prevention,” 02 04 2023. [Online]. Available: https://www.itsupportguides.com/blog/what-is-a-bit-flip-causes-consequences-and-prevention/. [Accessed 30 01 2025].
[11] BBC, “The computer errors from outer space,” 12 10 2022. [Online]. Available: https://www.bbc.com/future/article/20221011-how-space-weather-causes-computer-errors. [Accessed 30 01 2025].
[12] ArcServe, “Data storage lifespans: How long will media really last,” [Online]. Available: https://www.arcserve.com/blog/data-storage-lifespans-how-long-will-media-really-last. [Accessed 30 01 2025].
[13] ANSI, “ANSI X9.95-2022: Trusted Time Stamp Management and Security,” [Online]. Available: https://webstore.ansi.org/standards/ascx9/ansix9952022. [Accessed 30 01 2025].
[14] European Union, “eSignature FAQ,” [Online]. Available: https://ec.europa.eu/digital-building-blocks/sites/display/DIGITAL/eSignature+FAQ. [Accessed 30 01 2025].

8. Other papers in the Preservica expert series

Digital Preservation Overview

Automated File Format Preservation

Preserving Multi-Part Information Assets

Digital Preservation Metadata

Digital Preservation Policy Creation

About Preservica

Preservica is transforming the way organizations around the world protect and future-proof critical long-term digital information. Available in the cloud (SaaS) or on-premise, our award-winning Active Digital Preservation™ archiving software has been designed from the ground up to tackle the unique challenges of ensuring digital information remains accessible and trustworthy over decades.

It's a proven solution that's trusted by thousands of businesses, archives, libraries, museums and government organizations around the world, including the UK National Archives, Texas State Library and Archives, MoMA, Yale and HSBC.

preservica.com/about