# Protecting Mission Data Against Loss

Bernd Holzhauer[1] and Dr. Osvaldo L. Peinado.[2]
*DLR – GSOC, Muenchner Strasse 20, 82334 Wessling, Germany*

**At the end of a mission the final result left will be the data. The spacecraft will be de-orbited. The ground segment will be deployed or used for other missions, but the mission data remains, being the most important asset of the whole mission. This data can be knowledge, pictures, measurement results or whatever … but at the end - this is (just) computerized data. For most of the space missions the data acquired will be stored and used for many, many years. At the end of a mission, the data is all that remains usable. It is valuable and needs to be handled carefully. This data can also be important for follow-on missions and compared as "historical data" in order to evaluate differences or evolutionary changes of the missions.**

**For these reasons keeping the data safe is a real important part of the whole mission. Acquisition of the valuable data normally is very expensive, but many times project planning underestimates both the value of the data and the cost to keep the data usable for a long time.**

**Nevertheless electronically stored data is volatile. So the question is not "IF", the matter is "WHEN" the data will be lost.**

## Nomenclature

| | | |
|---|---|---|
| *Data* | = | All data captured thru a mission |
| *Online data* | = | data direct accessible by a computer |
| *Offline data* | = | data in a tape archive or a vault (need to be retrieved before access) |
| *Data life time* | = | the time from generating data to data deletion or obsolescence – also known as Data Life Cycle |
| *DRP* | = | Data Retention Policy |

[1] SAN System Engineer, Telespazio Deutschland GmbH, c/o DLR – GSOC, Muenchner Str. 20, 82334 Wessling, Germany, no AIAA Member.
[2] Ground Operations Manager, DLR – GSOC, Muenchner Str. 20, 82334 Wessling, Germany, no AIAA Member

## I. Introduction

The best way to store data over a long period of time is to engrave it into stone or in metal plates. But this is not a practical way for nowadays data usage. Also, even stone graved plates will age over the centuries and the data will finally get lost.

The question is not if data will be lost – the question is when data will be not longer available or accessible.

Printed paper is a good way to store data. Paperwork can be clipped into folders and stored in a depot. People doing research within this type of archive do not need to be equipped with special instruments and – as long as ink and paper do not disappear the information is accessible.

**Figure 1 - Engraved Stone**

If data is stored electronically, i.e. by a computer, the researcher will need also the appropriate reading environment. Computer data is easier to operate with but it ages much faster then paper printed information. Due to computer technologies changing extremely fast it is most likely that data from beginning of a space project is not readable by the computers used at the end of the mission.

To avoid problem due to changing computer environment data needs to be migrated from generation to generation of technology.

## II. Mission Data

Comparing Space missions with commercial computer systems, there is a big discrepancy in the definition of a "generation" time. The duration of space missions compared to computer generations shows a big mismatch. A generation within the computer industry is about two to three years. In space science three years is almost nothing. A typical space mission will take much longer – maybe decades – this will include several computer generations.

**Figure 2 - Tape Drive**
(Pertec Inc.)

Figure 2 shows an example of a computer tape drive from 1977 when Voyager mission was started. Meanwhile not just generations have been gone even many complete technology changes have been performed. Spare parts and knowledge to serve such old computer equipment are not longer available.

To keep the mission results alive over a long period of time a lot of data migrations from one technology to newer one will be necessary. The cycle is approximately one migration every 5 years.

Data is a very valuable item. Especially in space operations where collecting data is highly time and cost consuming. For some space missions it will even not be possible to reproduce similar data again.

But not all data generated during a mission needs to be kept over the time. It is part of the mission "design" to define the data formats and the data values, how data have to be stored and which duration they should be kept. Payload data is usually more important then Path-TM or TM/TC. And for Log-Files there is usually no need to keep it over a longer period of time.

At the end of the space mission the only remaining item will be data printed on some paper or more reasonable stored on some computer readable storage media. But mission data is the final value of the mission. It contains all the knowledge gained during the mission time. So it will be a good idea to share it with future science and missions.

### A. Reasons for storing Data

Data is and must be stored for different reasons.

The very first reason is the daily work. Without storing data somewhere it is impossible to work with it. As soon as data comes into a computer (for example by typing at keyboard) it is stored in internal RAM. Powering off the computer will cause this data to be erased. For next day usage data must be stored on some storage media. This may be a local hard drive, a network share or an USB stick.

Next reason is, store the data to use it in near future – means within next weeks or month. This is treaded like the daily usage. The computer and data don't care if the time between two accesses is milliseconds or days or weeks.

But if data should be stored for longer time – i.e. more then 2 years it will be better to consider some important points. Before going in to depth let's have a look, why data losses occurs.

### B. Reasons for Data losses

Beside accidental data losses due to fire, burglary or natural catastrophes the main problem is aging. In contrast to archived paperwork, electronically stored data will age much faster and also in different ways.

1. Data stored on physical media (magnetically or optical) presents problems related to the substrate itself. With magnetic information, the signal on the substrate tends to attenuate fairly rapidly. That problem can be solved by refreshing (rewriting) the media on a regular base.
2. The storage media technology itself will be aging. It will run out of service. Raising the need to be replaced by a newer one. These cycles are between 5 to 10 years - sometimes even shorter, especially when consumer style media is used.
3. Updates of application software may change the data format. This will influence the capability reading the older data. This causes a possible inaccessibility of data written by older software version in the years before.
4. On private data formats, especially if binary data streams are stored directly, the correlating description for the bit stream may be lost. Without such a description the data is still available but without meaning. That's similar to a loss of data. This can be avoided by using common standard file formats or doing some "Inline Documentation" like XML tagging.
5. Overwriting files with wrong content is not really an aging problem, but one of usual errors made, and this risk rises with long duration storage.

When a mission is planned, the format and the life cycle of the data need to be defined. The experience shows is better to use standard file formats. Proprietary formats will cause additional cost during data migration later on.

### C. Data loss problems will raise with Higher Density

Data density is growing. Currently all two to three years the capacity of hard disks, computer tapes and other storage media is nearly doubling. Higher density means at least: More data resides on a single media. If a medium is broken, more data will be lost.

But higher density media also becomes more sensitive to environmental parameters. Less surface, i.e. magnetic particles or electrons are used per bit. That means data becomes more sensitive to environmental influences and a single bit failure becomes more likely. Error correction mechanisms take care about such single bit errors, but with higher density the number of bit errors will raise and if the error correction cannot handle the amount of failed bits, the data is lost.

## III. Data Retention Policy

All space missions are seriously planed and long term prepared. Since data is the final result of a space mission the data format and data handling should be planned, too. This should be described in a so called Data Retention Policy (DRP).

A DRP is typically used to setup legal and security items. Therefore a DRP requires classification of the data. It should be easy to enhance a DRP to have a classification of the data. Defining rules of how to store and for which period's in time (data life time) the data must be stored, makes data handling for other people transparent and saves not just data but also saves money at the end.

Possible data classes are:

- Payload data
- Path-TM data
- TM/TC
- Log-Files
- Monitoring Files

Each class has its own priority and retention time on storage media. Even storage may to seem infinite (from the user point of view), it does not make sense to keep all the data for the total lifetime. For example, log-files and status information are nice to have for tracking purposes during daily work; nobody will look into these files after a certain time. So there is no need to keep it all the time nor to refresh or to migrate this information to new storage media.

It will also be a good idea and not a big effort to sort and separate data identified by classes into different files and appropriate directories; this allows later on data movement and deletion by just a few commands and/or simple scripting.

Beside data classification the DRP should also describe the following points:

**D. Flat file directories vs. structured directory tree**

**Never ever store data files in a flat structure!**

Storing all files in a single directory or just a few directories (flat structure) will work fine during software development and the testing phases. But it will cause a lot of problems during long term data handling. It is best practice to use the capability of a tree structured file system for presorting data. Let's for example look at a 10 year mission writing a data file every 5 minutes:

The mission generates

- 288 file s per day, which is acceptable for a single directory
- 8.640 files a month, this hits the top limit for a single directory
- 105.120 files per year. This is far over good practice. It may work, but …

… a file operation, processing 100 files per second, will take 18 minutes to complete. The typical "ls" (list) on a Linux/UNIX system handles approximately 10.000 files per second. So the system will be quiet for about ten (10) seconds before start printing to screen. The user may think the system is crashed.

Depending on operating system version even a simple "rm *" (remove all files in current directory) may fail on directories containing more than 50.000 files. It could be a strong and time consuming manual work to clean up such files from storage.

Best practice: Using the capability of directory trees as preselection for different data classes and also as a "time stamp" when data is created.

Note: Verify time stamps in received data streams before generating file names out of it. Wrong date stamps in data streams will otherwise generate "time-leaped" files on storage. And this will definitely cause problems in the future.

Define and document the directory structure within the DRP like:

```
/archive/2011/052/14/solar_experiment01_00-09.data
             /052/14/solar_experiment01_10-19.data
             /053/01/…
        /2012/001/01/…
             /002/01/…
```

This example uses "/archive/year/DOY/hour/" as preselection path in the directory tree structure. The file name contains the recorded minutes in its name (00-09). This is easy to understand for everybody involved in data handling. Also it makes data selection for processes like archiving or researching through specific date ranges pretty simple.

**E. Self-explaining File Names and Extensions**

Program developers like to use cryptic file names, and as long as these files are just processed by application software the file name is not important. As soon as files are handled by a third party, it will be good to have self-explaining (speaking) file names. This will help for example SAN engineers to identify files. It also helps the software engineers during changes in software or the subsystem engineers if something is accidentally misoperated in their file systems. It is much easier by a human to identify the meaning and maybe importance of files, if the files are named properly. For example in standard office software it is accustomed, to use different file name extensions to classify document types. Logical naming and consequent used file extensions make things a lot easier.

Define and describe the file name and the extensions in DRP. Discuss the DRP with the storage administrator. Use his knowledge about file handling from storage and archive point of view. He knows how to detect obsolete data and how to setup cost saving backup and migration strategies.

Using self-explaining file names and extensions will easily show up, when a file is copied by accident to a wrong directory and will not overwrite other valid data there.

### F. File sizes

There are a few pros and cons for small and/or large files.

Generally speaking: small files are easier to analyze and so far smoother in handling. Storing a certain amount of data can be done in a lot of small files or in a fewer number but larger files.

To judge about file sizes please keep in mind: Data is stored on media in junks, so called blocks. Disk manufacturers talk about 512 Bytes (or multiplies of it) per block which means physical disk addressing. But SAN storage is addressed via an operating system and this block (or chunk) sizes vary nowadays typically from 4 k to 64 k per block. To use files smaller then the block sizes defined at the storage will waste disk space, because the blocks will be allocated completely to their full size anyhow.

A lot of small files will slow down the operation if data will be written to tapes. Tapes are designed to store sequentially large amounts of data. Writing small files directly to tape may drop down performance drastically. Writing to tape is usually done via caching systems to keep tape drives streaming. This works usually fine in backup solutions. In a HSM system reading a lot of files from tape, it may take hours to complete. For this reason within HSM environment larger but less files are preferable.

On the other hand, selecting just small parts out of large files will cause a significant delay during reading. Much more data will be read and transferred than actually processed. This will unnecessarily stress the local network by transferring the 'overhead' between disk, server and workstation.

### G. Databases and their files

Databases are special cases of structured data. Databases like Oracle or MySQL are able to serve billions of data sets. This should be enough also for long term operations. The database internal structures and access mechanisms allow fast and easy access to each record written in the past, but ... the application is depending on a third party product, which may become obsolete. At least, someday the database supplier will stop the support for the used release and will force an upgrade to a newer version. That means a very special data migration will be required. This is usually supported by database vendor but may mean an operational outage of the database itself. Also it may be necessary to modify the application program, depending on release changes.

Databases handle most of its data in internal memory cache to increase access speed. To precede a backup the data in the database files need to be consistent. This requires a memory 'flash to disk' to secure this files as consistent. That makes it a bit trickier for the storage administrator to backup database files. Popular databases are supported by the major backup manufacturers with additional backup modules, but usually on extra money. Some of this software will stop the database for a short time. Therefore these database backups should be scheduled during low traffic times. The regular approach is to run a backup on a fixed schedule but this may conflict with the orbiters EOS/LOS time plan.

Low traffic times in space business means LOS periods, but unfortunately they are not clock synchronous. So it would be a good approach to have application software sending a trigger signal at start of a longer LOS phase instead of using a time scheduled backup.

Another database related item is, MySQl database tables for example are stored in 3 files per table. Creating a fresh new table on small time gaps will raise number of files on disk drastically and will slow down the restart of the database itself. Generate new tables only where it make sense.

### H. Inline Documentation

As long as somebody is working with the data (daily work) the data format is known and maybe handy. Nevertheless the data format needs to be well documented.

The best kept file has no value when it contains bit stream data and the description of the stream is lost. In this case the data is still available but it's meaningless. During defining the mission documents about data and file formats will be written. Everything is fine at this point in time. But what if this documentation is lost later on or after mission completion?

During data transmission from vessel to ground of course bandwidth is a limiting factor so data need to be stripped down and compressed or for security reasons it will be even keyed in addition. It is not the best idea to write such data streams directly to the storage system. To be used by spacecraft operator the data needs to be converted into readable formats. Disk space on ground segment is not as limited as bandwidth during communication. So it is a good idea to place human readable marks into the data streams before writing the data to disk.

Since Inline documentation means marks and metadata is within the data file, data description cannot go loss. A good example for Inline Documentation is the EXIF Data within JPG files. This EXIF contains the photographic details, date and time stamps and may also contain GPS data if camera supports this.

When data does not fit to a standard file format and a private format is prefered, it will be a good idea to place XML-tags into the data stream. Even if an XML "container" includes binary data the engineer in the future has a good chance to understand the meaning of this data.

### I. Standard Formats

For many types of data standard file formats are available. Use it! Examples for common file formats are PNG and JPG for pictures, MP3 for audio and MOV or AVI for video. It is preferable to use these common standards.

Almost everybody is able work with such standard files independently from the operating system used. Especially when standards are changing software tools will be available to convert an old standard format into a new one. For important data it would be a good idea to combine data migration from one technology to next with the converting the files from older to newer standard.

## IV. Data Storage

As soon as data is stored on a network share or a central storage, the storage administrator will take care of it. But storage administrator just sees files by their names and sizes. Storage administrators are not interested in file content and tread all files with same priority. A Data Retention Policy will help the storage admin to handle files more effectively.

Basically data can be stored "online" (on disk) and is directly accessible or "offline" (in tape archive) which means data needs to be retrieved before it can be accessed. Both methods have their advantages and disadvantages.

### J. Online

Online storage means the data resides on direct access media. This could be a local hard disk, a local electronic storage device (like an USB Stick or a Solid State Disk), a network share or a SAN device. However the data can be immediately accessed which is a direct and fast method. Online storage is good for daily work and easy to handle.

The disadvantages of this method are:

1. Online storage is limited in capacity.
2. Data can be deleted, accidentally overwritten or damaged by human errors.
3. Online storage is (ever) more expensive than offline storage, it consumes power and it is not really good for archiving data

Data can be kept online, i.e. the data resides on a local disk or on a storage system somewhere in a network. This is fine for daily work and fast access, but for large amounts of historical data it is inefficient to store the data on hard disks.

### K. Offline (Archive)

Offline storage is mainly build by tape archives. But it's possible to use optical disks or USB devices as offline storage, too. The major advantage of this method is the capability to put the data far away from the computer. So it can not be overwritten or deleted by accident. The carrying media can also be stored into a vault or an environment which does protect the media for longer period. The second advantage is the nearly unlimited capacity in number of tapes and/or disks.

Since data is not connected to the workspace computer it can not be overwritten by accident. Also by selecting a very smooth environment the media can be kept longer before a refreshing cycle will be necessary.

### L. Cloud Storage

"Cloud" is a new buzzword in storage industry. It is mainly used for internet connected storage. To understand what it really means it is necessary to specify "Cloud" more closely. Cloud may mean:

*1. Private Cloud*

Private Cloud is noting else then a network storage rented from a large provider. It can be also a company owned SAN/NAS storage with some connection to the internet. Connections to this cloud can be provided via fast private WAN links or just via the internet (usually more slowly).

Private Cloud provides usually a high security and a controlled access from almost everywhere. Capacity and data availability including backup and archive is part of the service contract (SLA) with the cloud storage provider.

*2. Public Cloud*

Public Cloud is a cheap way to share data in the internet. Because of security reasons it should not be used for serious mission data. For open (public) results of a mission it can be a usable alternative. Backup and archiving is usually not specified/garanteed.

Cloud storage can be good solution for sharing data. If a provider accepts a strong SLA it may be an alternative because the provider has to take care about the changing in technology and the necessary data refreshing cycles and migrations. But the provider may later on go out of business or is taken over by another company.

Also storing data in a cloud may conflict with legal regulations. For example the transfer of person related data to other companies is forbidden in the EU. That means a special contract with an EU provider is the only way to store such data in a cloud. The provider has to guaranty the data is never ever transferred to servers outside the EU.

**M. Tape as a offline storage**

Figure 2 (Page 2) shows a tape drive a little older then 30 years. It is a "Real to Real" tape drive where today no more service is available nor the interfaces to modern computer systems exist.


**Figure 3 – LTO cassette**

Meanwhile a lot of tape families where born and gone. Today's quasi standard is the LTO (Linear Tape Open) family. This standard was defined by Hewlett-Packard, IBM and Seagate and was first sold in 2000.

Regardless other tape standards exists LTO is today's favorite tape format. Since computer and storage development is an ongoing business LTO is not kept as a single standard. LTO Ultrium is a whole family of tape generations.

**Table 1 - LTO Family**

| Generation | Capacity<br>native/compressed | Speed<br>native/compressed | Date<br>to market |
|---|---|---|---|
| Ultrium 1 | 100/200 GB | 20/40 MB/s | 2000 |
| Ultrium 2 | 200/400 GB | 40/80 MB/s | 2002 |
| Ultrium 3 | 400/800 GB | 80/160 MB/s | 2004 |
| Ultrium 4 | 0.8/1.6 TB | 160/320 MB/s | 2007 |
| Ultrium 5 | 1.5/3.0 TB | 140/280 MB/s | 2010 |
| Ultrium 6 | 3.2/8.0 TB | 210/525 MB/s | 2012? |
| Ultrium 7 | 6.4/16 TB | 350/788 MB/s | |
| Ultrium 8 | 12.8/32 TB | 472/1180 MB/s | |

The LTO family today contains quite some generations. Since newer drives are backward compatible at least two generations, it is possible to skip a generation during upgrading the storage system. That means if today Ultrium 4 is in use the drives (and tape cassettes) maybe exchanged/upgraded to Ultrium 6 in 2013 or 2014.

What is after Ultrium 8? How long will it be supported? Newer members of the family may be defined – but also a complete new technology may show up.

## V. Refreshing Data

Magnetically stored information will get week after a certain time. The signal to noise ratio during reading the tape will drop over the storage time and the error correction has finally to do a good job.

Differently to analog music tapes the quality of a freshly generated copy of a digital data tape is like a brand new recording. Refreshing a tape is an action to take place before the original tape gets unreadable. This copying process is called "Refresh" or "Refresh Cycle". For long term archiving refreshing is mandatory. Best practice is to refresh tape recordings at least every 2 years. For some kind of critical data (i.e. financial files) it may be regulated by law down to much shorter periods.

Refreshing the tapes can be automatically done. Modern archiving software usually supports automatic tape refreshing on defined periods.

## VI. Migrating Data

If data is copied from older to newer technology the process is called “migration”. Migration can be combined with or replace a refresh cycle to save handling cost. Migration is not just another style of copying. Migration may also include changing file format to be compatible to newer formats and software.

During data lifetime (data life cycle) storage technology will change, forcing the migration of the data to the newer technology. Depending on total data lifetime this may happen several times.

Depending on the technology changes migration can be an easy copy process or if also software and/or format changes are required a more complex copy & change process might be necessary.

If standard file formats are in use this changes may be done by COTS software, i.e. standard programs. If private data formats are in use migration may also require some software engineering.

Migration and refreshing may be combined together. Suppose a 2 year refreshing cycle and a 4 year migration in tape technology, one refreshing cycle can be combined with the migration. Migration and refreshing are both basically copy processes.

## VII. Conclusion

Data is not just the remaining “leftover” from a mission. Data is the final and valuable result of a mission. To protect this data over a long period in time a lot effort will be necessary. So it makes sense to qualify and quantify the data and its formats so that engineers in the future can understand and use it for other missions.

This means:

- Define a Data Retention Policy right at the beginning of the mission
- Describe classes, retention times, file-names and directory structure, which will easy allow doing repetitive jobs by program scripts
- Use a logical self explaining directory structure
- Use self explaining “speaking” filenames
- If possible, use standard file formats
- Use “Inline documentation”
- Avoid plain binary or encrypted files
- Do not mix different data classes in files or directories
- Communicate often and frankly with the storage administrator

Remember: All data will be lost in time – but some simple logical thoughts written down in a Data Retention Policy will protect the value of the data, save cost during arching (refreshing and migration) and make the data reusable for future use.

Please keep in mind: storage cost will rise on long term projects. Storage will be replaced at least once or even more times during the defined project lifetime and data needs to be migrated. So each file, which does not make sense to be stored over long term, should be eliminated as soon as possible.

This reduces not just the required capacity it also reduces the number of files stored. In addition it reduces the access speed to a single file and the migration time if storage is changed.

Long term in space business means: there is a daily addition/multiplication over many thousands of days.

## Appendix A
## Acronym List

| | |
|---|---|
| **SAN** | Storage Area Network |
| **NAS** | Network Attached Storage |
| **WAN** | Wide Area Network |
| **SLA** | Service Level Agreement |
| **DRP** | Data Retention Policy |
| **DOY** | Day Of Year |
| **AC&S** | Attitude Control and Stabilization |
| **ACA** | Attitude Control Assembly (Space Station) |
| **ACAD** | Attitude Control and Determination |

## References

[1]Holzhauer, B. and Peinado, Dr. O.: *"Saving Cost with the Right Software Design for Long term Operations",* AIAA-2012-2081

[2]SNIA Archive: *"practitioners confirm lon-term information retention crisis",* http://www.snia.org

[3]IDC: *"Storage Watch",* http://www.idc.org