Preserving Digital Information Forever

Andrew Waugh, Ross Wilkinson, Brendan Hills, and Jon Dell'oro

CSIRO Mathematical and Information Sciences
723 Swanston Street, Carlton, VIC, 3052, Australia
Tel: +61 3 8341 8200
E-mail: {andrew.waugh, ross.wilkinson, brendan.hills, jon.delloro}@cmis.csiro.au

ABSTRACT

Well within our lifetime we can expect to see most information being created, stored and used digitally. Despite the growing importance of digital data, the wider community pays almost no attention to the problems of preserving this digital information for the future. Even within the archival and library communities most work on digital preservation has been theoretical, not practical, and highlights the problems rather than giving solutions. Physical libraries have to preserve information for long periods and this is no less true of their digital equivalents. This paper describes the preservation approach adopted in the Victorian Electronic Record Strategy (VERS) which is currently being trialed within the Victorian government, one of the states of Australia. We review the various preservation approaches that have been suggested and describe in detail encapsulation, the approach which underlies the VERS format. A key difference between the VERS project and previous digital preservation projects is the focus within VERS on the construction of actual systems to test and implement the proposed technology. VERS is not a theoretical study in preservation.

KEYWORDS: Digital preservation, archiving

INTRODUCTION

Surprisingly little attention is given to the preservation of digital information over long periods. While there are small groups of active researchers in the library, archival, and space science communities, much of their work is theoretical; highlighting the difficulties of digital preservation and suggesting approaches to overcoming them, rather than experimenting or building systems. Within the computing community, almost no attention has been paid to the problem; designers and builders of computer systems seem to implicitly assume that information will be kept in their 'system' and 'preserving' digital information equates to ensuring that the data in the system is 'backed up' regularly.

Yet digital information is relatively fragile and can easily be lost due to a variety of problems. Loss may be caused by technical failures, including physical deterioration of the media on which the information is stored, the inability to read the media because the media readers are no longer supported, and loss of the software that interprets the stored information. In addition to these technical failures information can be effectively lost by a failure to preserve aspects of the information that make it useful to an individual or organization. These aspects include the information's status, its ownership, its reliability, its authenticity, and its retrievability.

The fragility of digital information is becoming of concern as the world increasingly moves towards storing information digitally instead of on paper or film. In many organisations the point has already been reached where most internal business is conducted electronically, and hence information is almost certainly being lost. We risk a 'dark age' where it is impossible to reliably state what occurred or why because the information that documents the business has been lost.

For this reason, we believe that practical preservation of electronic records must start now and this paper presents a strategy developed from experimentation. We believe that the approach presented in this paper will preserve electronic records. We do not make any claim that this is the ultimate solution, but it will provide interim preservation until a better solution can be developed.

The fundamental principle behind any digital preservation strategy must be 'do minimal harm'. Because there is little experience in long term preservation of digital information it is quite possible that we may adopt poor preservation methods. If this occurs, the preserved information must not be so damaged by the preservation technique that it would have been preferable not to have applied the technique in the first place.

We believe the keys to long term preservation of digital information are:

- Encapsulation; that is, wrapping the information to be preserved within descriptive metadata and keeping it at a single location.
- Self documentation; that is, the ability to understand and decode the preserved information without reference to external documentation.

- Self sufficiency; that is, the minimization of dependencies on systems, data, or documentation.
- Content documentation; that is, the ability of a future user to find or implement software to view the preserved information.
- Organization preservation; that is, the ability to store information that allows an organization to actually use the preserved information.

This paper describes a digital record format and associated practices that is designed to preserve information indefinitely. The format was developed as part of the Victorian Electronic Record Strategy (VERS) project [14]. Although focussed on records, we believe the format we developed is equally applicable to other digital information such as digital images and databases, and it can be described in terms of the Open Archival Information Service (OAIS) model [5].

We will begin this paper by describing the VERS project. We will then consider why preserving digital information is so challenging and summarize the various approaches that have been suggested for overcoming these challenges. We will finish by describing the preservation format developed for the VERS project, describing how it fulfils the keys to long term preservation described above.

THE VICTORIAN ELECTRONIC RECORD STRATEGY

The Victorian Electronic Record Strategy (VERS) project is tasked to build systems to capture and preserve electronic records permanently in a government environment. It is being sponsored by the Public Record Office Victoria and funded by the Victorian State government. Unlike many similar projects, VERS is not a theoretical study or paper analysis. Instead, development of the strategy involves implementation of actual systems.

Stage 1 (1995-6) was a background investigation. The report [13] investigated whether there were any available systems that could preserve records indefinitely. Having concluded that there were no such systems, the report then considered whether there were any techniques that could form the basis of a system to preserve electronic records. The report identified encapsulation as a suitable technique.

Stage 2 (1998) implemented a prototype archival system that demonstrated the capture, encapsulation, archiving, and retrieval of electronic records. A demonstration system was constructed to built to capture, store, and make available records. The records were document type records and included text, images, and drawings. Records were captured from common desktop applications such as Microsoft Word and email, and particular attention was paid to the degree to which this capture could be performed automatically. In addition, government recordkeeping and archival processes were analyzed to ensure that the prototype reflected actual government processes. The purpose of Stage 2 was to demonstrate that preservation of electronic records by encapsulation was technically feasible. The result of this stage is documented in [14].

Stage 3 started in 1999 and is scheduled to continue to 2001. This stage involves implementing a VERS system on every desktop within the Department of Infrastructure, a medium sized Victorian Government department. The purpose of this stage is to refine the techniques developed in Stage 2 and confirm that they can be economically implemented within a real organization. In addition, two other Australian agencies are working on including elements of the VERS within new systems.

Although VERS focussed on electronic records, records are only a specific instance of the broader problem of preserving digital information. The remainder of this paper summarizes the lessons we have learnt about preserving digital information and the VERS preservation format itself.

PRESERVATION CHALLENGES

Levy [22] suggests that the first question to be asked is what is to be preserved and why. The focus of VERS was to preserve records, that is, “recorded information, in any form, […] created or received and maintained by an organization or person in the transaction of business or the conduct of affairs and kept as evidence of such activity” [24]. The preservation of a record does not, necessarily, require the preservation of the artifact that originally represented the record.

Preserving digital information has three aspects: physical preservation; functional preservation; and organizational preservation.

Physical preservation involves preserving the bit stream that forms the digital information against the physical deterioration of the media and against the obsolesce of the media readers. Physical preservation will not be considered further in this paper as the only practical method for physical preservation is the periodic transfer of the bits to new media (refreshing). Refreshing is a standard and widely practiced computer technique. It is also widely practiced to preserve conventional paper based information (e.g. microfilming books). Although digital media currently needs to be refreshed far more frequently than microfilm, it has several advantages. Refreshing digital information does not result in a loss of quality, can be completely automated, and results in higher density recording.

Functional preservation is the preservation of some (or all) of the functions of the original software environment. Merely preserving the bits is of no use if the bits cannot be decoded and the information used. Preserving some or all of the functions of the original application is consequently the next layer of preservation. Note that preserving the functions does not require or imply preserving the application. Functional preservation is a significant challenge as software is fragile and the program and data are often tightly coupled. Fortunately, it is often not necessary to preserve the full functionality of the application that originally created the information. For example, it often sufficient to preserve the ability to view and extract the information, and not the ability to modify the preserved information.

Organizational preservation is the preservation of sufficient supporting information to enable an organization to use the preserved information to support its business. This is different to preserving the technical ability to physically view the information. The requirements for organizational preservation will vary, but typically it is necessary to be able to find the information, to be able to understand its context (particularly in relation to other preserved information), to be confident of its authenticity, and to know its ownership. For example, one means of demonstrating authenticity of information is to document its provenance; this requires documenting everything that has happened to the information over its lifetime. Two studies which consider this question specifically in the context of electronic records can be found in [8, 12].

APPROACHES TO PRESERVING ELECTRONIC RECORDS

Several general approaches to the preservation of electronic digital information have been identified. These include: system preservation, emulation, migration, standardization, and encapsulation. These approaches are not mutually exclusive. One approach may include aspects of other approaches.

System Preservation

The simplest approach to preserving electronic records is to preserve the computer system on which the record is created and stored. Although a simple solution, the cost of keeping obsolete computer hardware operational precludes this preservation approach in all but exceptional circumstances.

Emulation

Emulation allows the original application software to be used without requiring the original system to be maintained [4, 18].

Although emulation is in widespread use within the computing industry to prolong the life of legacy applications, there are significant practical challenges in using emulation to preserve digital information over a long period. The Y2K bug has shown that the applications themselves may contain bugs that may cause the loss of information over time. The original application may not capture or preserve the knowledge necessary for organizational preservation (for example, it may not be possible to prove that the information has not been altered). Emulation depends on preserving a significant amount of information. A hardware emulation solution, for example, assumes the preservation of the emulator, the operating system, the application and the data. Not only is it often difficult to identify exactly what must be preserved (particularly with modern software written in a modular fashion using ‘plug-ins’), but the loss of any of these components means the effective loss of the information. The emulator is a software application itself, and will need to be preserved, either by emulating the system on which it runs, or by periodic re-implementation. Accurate renewal may become difficult once familiarity with the system being emulated is lost.

Emulation is a useful approach if the goal to preserve the software as an artifact itself. However, if the goal is to preserve access to the information, emulation is likely to prove counter productive. Why would future researchers wish to use archaic software to access information; software that they have no training on, or experience of?

However, keeping the original application running using emulation is the only feasible preservation approach if the organization preserving the information lacks sufficient knowledge to understand the format of the digital information.

Migration

An alternative to preserving the original application is to migrate the digital information to a new, replacement, system. Again, migration is widely used within the computer industry to transfer data from one system to its replacement; particularly when replacing database systems.

Migration has the benefit of eliminating the need to retain the original application. This benefit is so significant that we expect that most successful long term preservation strategies will contain elements of migration, and both standardization and encapsulation (considered below) are examples of migration. Migration has been recommended by archivists and others [1, 4, 9].

The keys to a successful migration are knowledge of the original data format, and a close match in functionality between the original and replacement formats. Migration cannot be performed if knowledge of the original data format has been lost and so a significant challenge with migration is ensuring that all information is migrated before that knowledge is lost. With many commodity applications (e.g. most desktop applications), the data format is not known by the owner of the information and so must rely on third parties for migration. In this case the information owner cannot judge how fast the knowledge is being lost and so risks leaving the migration too late. Many organizations manage their information poorly and a migration program may miss significant amounts of information. Finally, migration costs money and the temptation in many organizations will be to delay implementing a migration program. Some organizations will delay too long and find their information cannot be converted in a cost effective way.

Another significant challenge is that migration may break the cardinal rule of preservation: minimize harm. Migration explicitly means modifying the data, and this modification will degrade the preserved information if the new format cannot support aspects of the original format. Worse, it may be impossible to subsequently determine what has been lost. Successive migrations may cause the data to be so degraded that it is effectively lost. Demonstrating that the migrated copy of the record is a true and accurate copy of the original may cause problems. Indeed, our experience has been that quality control and testing of migration is a significant cost in any migration.

A final point to make about migration is that it does not support organizational preservation unless both the original and replacement data formats support the necessary information (a specification of the metadata for archival preservation can be found in [11], but this does not necessarily cover other types of organization).

One specific migration strategy adopted by some archival agencies [10, 19] is the ‘post custodial’ model in which historical records are no longer accessioned into an archive, but are maintained in operational systems within an organization. The goal of this strategy is to reduce the cost of preserving the historical records. First, it is not necessary to operate a separate archival system. Second, since an organization must migrate their operational records when upgrading the operational system, the marginal cost of migrating the historical records at the same time using the same software is negligible. In practice, these cost savings may not be achieved. Holding historical records in an operational system will increase the running costs of the operational system; particularly as the proportion of historical records rises. Requiring the software and systems developed for migrating the operational records to accurately migrate historical records is likely to increase the complexity of the migration software and hence increase the cost of writing the software and testing the migration. Providing public access to historical records held in operational systems is likely to require significant re-engineering of the operational system to provide the necessary security. Finally, the cost benefits disappear when all the records become historical and there is no operational need to migrate them to a new system. From this point, the historical records must bear all of the costs of any future migration.

Standardization

Standardization involves migrating digital information once to a standard data format. Information could be migrated to a standard format upon creation, when it becomes inactive, or when it is accessioned by a preservation agency. The use of a standard means that knowledge of the data format should always be available (this allows re-implementation of software to handle the records even if the standard falls from use). This addresses two of the challenges of simple migration: loss of the records due to loss of knowledge of the data format; and fatal degradation of the records due to successive migrations (as there is no need to perform successive migrations).

Standardization cannot be performed if there is no suitable standard preservation format. Even the single migration to the standard format may cause severe degradation if the standard cannot support all aspects of the original data. Standards are unlikely to support the information necessary for organizational preservation. Finally, how does a future user determine what standard was actually used to preserve the information? The answer to the last is metadata, which brings us to the final option: encapsulation.

Encapsulation

Encapsulation involves wrapping the record to be preserved within a human readable wrapper. The wrapper contains information that supports organizational preservation, and documents the preserved information to allow it be decoded in the future. Part of the process of encapsulation may be to migrate the record to a more easily documented data format (which is likely to be a standard format), but this is not an essential component of encapsulation. Although superficially similar to standardization, encapsulation is different as there is no requirement that there be a single encapsulation format, or that that this format be standardized (although a single, standardized, encapsulation format is an advantage when implementing an preservation system). Encapsulation is discussed, to a limited extent, in [3, 17, 23]. UPF [23] is particularly interesting encapsulation proposal.

The documentation in an encapsulated record means that the data format of the preserved information can always be determined, and knowledge about the format obtained. Unlike standardization or simple migration, encapsulation can support the information required for organizational preservation.

Encapsulation has three challenges. The first is the requirement for applications to generate encapsulated records. Since current applications do not do this, a system must be developed with hooks into the various applications. The VERS report [14] considered the costs of this requirement, and the third stage of the VERS project is testing this costing. The second challenge is the potential storage overhead of including documentation about the format within each record. This overhead can be significantly reduced by using published data formats as the encapsulation then only need to include a reference to a published standard. But this leads to a further problem: how long will the published standard be available for? (It is probably likely to be available for far longer than any other form of documentation about non-standard data formats.) The third challenge is information stored in unpublished data formats. The most the encapsulation can document in this case is the identity of the application in the hope that software will still be available to support that data format in the future. This is still better than standardization or simple migration, as future users will at least be able to identify with certainty the application that generated the information.

Migration and encapsulation are, in some sense, duals. Encapsulation does not eliminate the possibility that the information will eventually need to be migrated, although careful selection of data formats and documentation may put off the need for migration for a very long time. As a preservation strategy, migration will fail if it is not possible to identify the data format; this is exactly the information contained within an encapsulation.

Of the five approaches to preservation, we believe that migration is the simplest over the short and medium term for digital information that is actively managed. However, migration is an active, systems based, approach. Migration constantly alters the digital information, potentially leading to information degradation. The long term costs of migration are significant; particularly once the supporting system

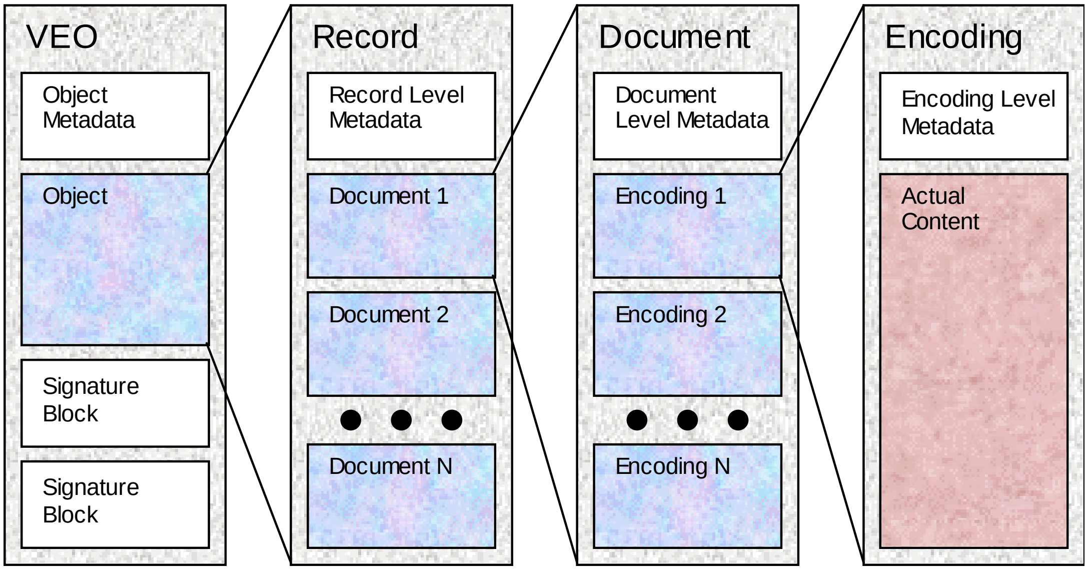

Figure 1: The structure of a VERS Encapsulated Object

ceases to be operational. Encapsulation is the best basis for long term preservation, but requires a system for capturing the digital information. The remainder of this paper discusses encapsulation in more detail.

THE VERS PRESERVATION FORMAT

The basic function of an encapsulation is to wrap the information to be preserved within metadata that describe aspects of this information. Metadata is provided for both functional and organizational preservation. The metadata for required for functional preservation describes the data formats used, including the format of the encapsulation itself. The metadata required for organization preservation is more varied, but broadly describes what the preserved information is, its history, and its relationship to other preserved information. Organizational preservation metadata also covers authentication information that demonstrates that the information has not been modified since encapsulation.

The structure of a VERS Encapsulated Object (VEO) is shown in the figure on this page (details can be found in [6, 14, 15]). An example VEO is shown in Appendix 1 at the end of this paper.

A Record consists of one or more documents (physically separate parts of the record). Each document is represented by one or more physical encodings (concrete data representations). Structuring the preserved information in this way provides significant flexibility. Multiple documents allows several independent components (with different formats) to be associated together and managed as one object. Several different encodings of each document can be preserved.

Content formats

An encoding is a representation of the content in a particular data format. The data format may have been processed in some way (e.g. by compression). A description of the data format and how it has been processed is included in the encoding level metadata.

A VEO may contain digital information in any format, and so a VEO may encapsulate audio files, images, and video as well as conventional documents.

The success of the preservation depends on the choice of encoding; some encodings are more likely to survive over the long term than others. In implementing VERS we came up with the following heuristics for selecting suitable preservation encodings.

The worst case preservation scenario is that no software exists to interpret and display the content. A good preservation encoding is one which can be re-implemented in the future. The ideal encoding format is sufficiently simple to completely describe within the encoding level metadata. An example would be a scientific dataset which is simply a large table. For types of content that are too complex to describe within the encoding, we recommend selecting an encoding that has been formally published and implemented by several vendors. Examples include PDF, TIFF, JPEG, and MPEG. The published specification can be obtained from legal deposit libraries. Implementation by several vendors assists in ensuring that the products actually do implement the published specification. Published encodings include standards, but published proprietary standards are an alternative.

Both of the previous options depend on the information about the encoding being publicly available. If there is no suitable publicly known specification, the next best option is to choose a widely used encoding such as Microsoft Word. Very widely adopted encodings are unlikely to disappear in the short or medium term as the large installed base means that new products will provide forward migration paths.

If there is no widely used encoding, the best that can be achieved is to use the native encoding generated by the software that produced the content. At least, in this case, the metadata surrounding the content describes exactly what software produced the content.

The choice of data format may depend on the purpose for which the information is preserved. An image, for example, may be preserved using lossless or lossy compression and either may be appropriate.

Finally, many modern objects are complex. Documents, for example, may include embedded OLE objects, or may consist of hypertext. Within VERS, research is still being undertaken on the best way to deal with such objects. One option is to 'flatten' such an object to make it into a simple object. A second option is to choice a data format that supports complex object and linking.

Metadata

Each layer (record, document, encoding, and object) contains metadata. The highest level may contain one or more signature blocks.

The Record and Document level metadata is primarily concerned with organizational preservation. The Record level metadata describes the record as a whole. Topics covered by the Record metadata include: what the record is, its history, and its relationship with other records. The record level metadata is identical to the National Archives of Australia (NAA) Recordkeeping Metadata [11]. This is derived from the Australian Government Locator Service (AGLS) metadata [2], which in turn is derived from the Dublin Core metadata [7]. The Document level metadata describes the document within the record. The descriptive component of the Document level metadata is composed of selected elements from the NAA Recordkeeping Metadata. Apart from describing the document itself, the Document level metadata describes the system from which the document was obtained.

The signature blocks contain a digital signature that applies over the entire object and are one method of ensuring authenticity of the information. The signatures are applied when the VEO is created and allow any modification to the object (record) since creation to be detected. They also provide evidence as to who was involved in creating the record. Multiple signatures are supported as this makes forgery more difficult as multiple parties need to collude to forge a record. There are a number of interesting security issues in applying digital signatures to objects which are vulnerable for long periods of time. Space precludes a discussion, and the interested reader is referred to [6, 14].

The signature blocks provide a technical method of determining authenticity. The record level metadata itself can form an alternative, more traditional, method of demonstrating authenticity. In a traditional archive, authenticity is determined by the provenance of the record. Provenance is based on the system that held and preserved the records and is demonstrated by the documentation that accompanies them. This documentation is supported by the NAA Recordkeeping Metadata [11], and hence can be represented in a VERS object.

The Object and the Encoding level metadata are primarily concerned with functional preservation. The primary function of the Encoding level metadata is to describe the data formats and transformations used to produce that particular encoding of the document. The Object level metadata describes the overall format and structure of the VERS Encapsulated Object.

A VEO is physically expressed as an XML [xml] object. XML is a simplified version of SGML. The DTD can be found in [6, 15].

VERS design principles

In designing the VEO, we followed a number of principles to maximize the longevity of the information.

Self Documentation. The VEO is, itself, a data format. We cannot assume that a future user will have access to the VERS documentation that will allow them to extract information from the VEO. Consequently, VEOs must be self documenting. By this, we mean that a user should be able to directly view the contents of a VEO using the most primitive text editing tools and understand the structure and contents of the VEO. In essence, if the VERS software has been lost, a human must be able to re-implement it using the VEO itself.

Self documentation has four aspects:

1. Textual markup. The structure and content is represented as text. VERS uses XML [20] with the Unicode encoded in UTF-8 [16] which is based on ASCII.
2. Simple structure. Complex XML structures are avoided as they are too difficult to understand. In particular, we decided not to use the Resource Description Framework (RDF) [21] as it produced markup that was too difficult for a human to read.
3. Meaningful tag names. The XML tag names were carefully chosen to be meaningful and, in particular, abbreviations were avoided.
4. Hints. These are short textual descriptions designed to guide a future VERS implementor. For, example, a digital signature is prefixed by a short description that states what signature standard has been used, what options or arguments were chosen, and identifying precisely which parts of the XML object have been signed.

It is impossible to make a VEO completely self documenting as the required information would dwarf the actual preserved content. As discussed in the previous section, however, advantage can be taken of formally published specifications to minimize the amount of information required in the VEO.

Self sufficiency. A VEO should be designed to minimize the dependencies on systems, data, or documentation. The self documentation discussed in the previous subsection is an example of self sufficiency. The self documentation makes the VEO independent of a VERS system or external VERS documentation.

Self sufficiency is the reason why the metadata and content of a record is contained in one VEO instead of being separated and stored in different databases. Splitting the parts of the record up and storing them separately increases the risk of loss because the loss of one of the components will often mean the effective loss of the entire record. For example, if the Record level metadata is lost then it is no longer possible to find the record, understand what the record is, or how it fits into the wider collection of organizational records. The record is consequently effectively lost even though the actual content still exists and may be retrievable. In addition, self sufficiency means that a preservation organization only has one type of object to manage, hence management is simpler and records are less likely to be lost.

A final example of self sufficiency lies in the decision to prohibit encryption of the stored digital information. Access to encrypted information is obviously dependent on the decryption key; lose the key and the information is lost. (Note that there is no objection to encrypting digital information in transmission between servers.)

Content Documentation. It must be possible for future users to clearly identify the data formats (including versions) used to encode the content. The worst possible future scenario is where the computer systems no longer have the software to interpret the content. If the user can clearly identify the data formats, a search for suitable software is aided, and, if none is found, a search for the data format specification can be performed and the content re-implemented.

Organizational Preservation. Most of the preceding principles support functional preservation, and ensure that the VEO has sufficient information for users to be able to decode content. In addition, it is absolutely essential that the VEO contains sufficient information to support organizational preservation.

VERS was designed to support electronic archiving; the metadata supporting organizational preservation was consequently based on archival studies [11, 12]. We believe that most of this metadata will serve for other digital preservation applications. However, we recognize that future work may require additional metadata.

CONCLUSION

No one knows the future. There is no strategy for the long term preservation of digital information that can be guaranteed to work. However, we can guarantee that digital information will be lost if no preservation strategy is adopted. The good news is that two preservation techniques – migration and encapsulation – have been identified that have a high probability of allowing future generations access to information.

Migration is an active, systems based approach. Migration preserves information by continually moving it between one system and its replacement, modifying the information as necessary. Evidentiary status is preserved by system functionality. We believe that the active nature of migration makes it costly in the long term and opens the door to degradation of the preserved information. Bad migration decisions may lead to irreversible loss. For these reasons we believe that encapsulation is a better option for long term preservation.

Encapsulation is a passive, data driven approach. It involves wrapping the digital information to be preserved within preservation information. This preservation information contains a minimal set of documentation so that a future user can understand the format of the preserved digital information and build, if necessary, a viewer for the information. It also contains, if required, sufficient information to preserve the evidentiary nature of the information. As a data driven approach, encapsulation is not dependent on systems. Since the original information is preserved, encapsulation information loss due to migration is avoided. Encapsulation is a relatively cheap and simple technique.

The key features of an encapsulation format are:

- Simple and self documenting. The encapsulation must be capable of being read and understood by a human using the simplest computer tools. We recommend a textual encoding for the encapsulation.
- Self sufficient. The encapsulation must include all the information required to preserve the digital information. Dependencies on systems or other data mean an increase in the possibility of losing the preserved information.
- Content documentation. The encapsulation must contain sufficient documentation to enable a future user to find or write software to access the preserved information. This documentation may be references to externally published descriptions.
- Organizational Preservation. The encapsulation must support the inclusion of information that addresses the organizational issues involved in continued use of the preserved information.

The proposed VERS format has all these features. More details on the VERS project as a whole, and the VERS format can be found in [6, 15].

ACKNOWLEDGMENTS

The work carried out in this paper was undertaken in conjunction with the Public Record Office Victoria, Ernst & Young, and the Department of Infrastructure. We would like to acknowledge the valuable input staff from these organizations had on the ideas expressed in this paper. The work was funded by the Victorian State Government.

REFERENCES

1. Preserving Digital Information, Report of the Task Force on Archiving of Digital Information, May 1996, ftp://ftp.rlg.org/pub/archtf/final-report.pdf

2. The Australian Government Locator Service (AGLS) Manual for Users, Version 1.1, National Archives of Australia and Office for Government Online, August 1999, http://www.naa.gov.au/govserv/agls/AGLS_User_Manual_1.pdf

3. Bearman, D., Sochats, K., Metadata Requirements for Evidence, University of Pittsburgh, http://www.lis.pitt.edu/~nhprc/BACartic.html (last visited 25.2.00)

4. Bearman, D., Reality and Chimeras in the Preservation of Electronic Records, D-Lib Magazine, Vol 5 No 4, April 1999, http://www.dlib.org/dlib/april99/bearman/04bearman.html

5. Reference Model for an Open Archival Information System (OAIS), Consultative Committee for Space Data Systems, CCSDS 650.0-W-4.0, White Book, September 17, 1998, http://ssdoo.gsfc.nasa.gov/nost/isoas/ref_model.html

6. Waugh, A., Wilkinson, R., Hills, B., & Dell'oro, J., Preserving Digital Information Forever, CSIRO CMIS Technical Report (forthcoming).

7. Weibel, S., Kunze, J., Lagoze, C., Wolfe, M., Dublin Core Metadata for Resource Discovery, RFC 2413, September 1998, ftp://ftp.isi.edu/in-notes/rfc2413.txt

8. Duranti, L., Eastwood, K., The preservation of the Integrity of Electronic Records, http://slais.ubc.ca/users/duranti/into.html (last visited 25.11.99)

9. Hedstrom, M., Migration Strategies (Draft), Prepared for Experts Committee on Software Obsolescence and Migration (1996), May 1997, http://www.sis.pitt.edu/~cerar/ftp-docs/Mig-Stra.doc

10. Keeping Electronic Records (Policy for Electronic Recordkeeping in the Commonwealth Government), National Archives of Australia, http://www.naa.gov/govserv/techpub/elecrecd/KeepingER.html (last visited 25.11.99)

11. Recordkeeping Metadata Standard for Commonwealth Agencies, National Archives of Australia, Version 1.0, May 1999, http://www.naa.gov.au/govserv/TECHPUB/rkms/intro.htm

12. Functional Requirements for Evidence in Recordkeeping, University of Pittsburgh, School of Information Sciences, http://www.lis.pitt.edu/~nhprc/ (last visited 25.2.00)

13. Keeping Electronic Records Forever, Records Management Vision Development, prepared by Ernst & Young Public for Record Office Victoria, 1996, http://home.vicnet.net.au/~provic/vers/kerf.htm

14. Victorian Electronic Record Strategy, Final Report, Public Record Office Victoria, 1998, ISBN 0-7311-5520-3, http://home.vicnet.net.au/~provic/vers/final.htm

15. Management of Electronic Records, Public Record Office Standard (PROS) 99/007, http://www.prov.vic.gov.au/vers

16. Yergeau F., UTF-8, a transformation format of ISO 10646, RFC 2279, January 1998, ftp://ftp.isi.edu/in-notes/rfc2279.txt

17. Rothenberg, J., Ensuring the Longevity of Digital Documents, Scientific American, January 1995, p24-29

18. Rothenberg, J., Avoiding Technological Quicksand: Finding a Viable Technical Foundation for Digital Preservation, January 1999, Council on Library and Information Resources, ISBN 1-887334-63-7, http://www.clir.org/pubs/reports

19. Documenting the Future (Policy and Strategies for Electronic Recordkeeping in the New South Wales Public Sector), State Records New South Wales, 1995, ISBN 07310 5038 X, http://www.records.nsw.gov.au/publicsector/erk/dtf/tofcont.htm

20. Extensible Markup Language (XML) 1.0, W3C, 1998, http://www.w3.org/TR/REC-xml

21. Resource Description Framework (RDF) Model and Syntax Specification, W3C, 1999, http://www.w3.org/TR/REC-rdf-syntax/

22. Levy, D., Heroic Measures: Reflections on the Possibility and Purpose of Digital Preservation, Proceedings of the third ACM conference on Digital Libraries, Pittsburgh, 1998, p152-161

23. The Universal Preservation Format: Background and Fundanmentals, Sixth DELOS Workshop: Preservation of Digital Information, Tomar, 1998, http:/www.ercim.org/publication/ws.proceedings/DELOS6/upf.pdf (last visited 1.3.00)

24. Australian Standard on Records Management, AS4390-1996, Standards Australia, ISBN 0-7337-0306-2

APPENDIX 1 – VERS RECORD

The following is a VERS record. It has been slightly edited to remove most of the Base64 encoded binary data (this affects the Signer's certificate and the actual content). Deleted text is shown by '[…]' The DTD can be found on the VERS Web site: http://www.prov.vic.gov.au/vers/

```
<?xml version="1.0" encoding="ISO-8859-1"
standalone="no" ?>
<?namespace
name="http://www.prov.vic.gov.au//standards//pro
s99007.htm" as="vers"?>
<?namespace name="http://www.naa.gov.au//RKM-
1.0.htm" as="naa"?>
<!DOCTYPE vers:VERSEncapsulatedObject SYSTEM
"file:///h:\src\prismEd\schemas\vers\vers.dtd">
<vers:VERSEncapsulatedObject>
 <vers:VEOFormatDescription>
 <vers:Text>
Produced according to the Victorian Electronic
Strategy, Version 1.2 of 1 July 1999. The
structure of this record is represented using
Extensible Markup Lanugage (XML) 1.0, W3C, 1998
 </vers:Text>
 </vers:VEOFormatDescription>
 <vers:Version>1.2</vers:Version>
 <vers:SignatureBlock>
 <vers:SignatureFormatDescription>
The contents of this archivable object are
signed using NIST FIPS-186 (Digital Signature
Algorithm) with a 1024 bit key. All of the text
from starting with the 'less than' symbol of
the vers:SignedObject start tag up to and
including the 'greater than' symbol of the
vers:SignedObject end are included in the
signature. The resulting signature is encoded
used BASE64 and can be found in the
vers:Signature tag. The signer's public key can
be found in the vers:SignersCertificate tag,
also encoded in BASE64. The software used to
calculate the digital signature is the Java
security package, version 1.1.
 </vers:SignatureFormatDescription>
 <vers:SignatureDate>
03 Feb 2000 04:06:16 GMT</vers:SignatureDate>
 <vers:Signer>DOI</vers:Signer>
 <vers:Signature>
MCwCFBlemkxkhgIAe/V1TTVHL92lBXz/AhR5I8XnxaCxxO0K
rPn/Sof8ObMnAg==
 </vers:Signature>
 <vers:CertificateBlock>
 <vers:SignersCertificate>
MIICtDCCAnSgAwIBAgIBETAJBgcqhkjOOAQDMDcxCzAJBgNV
BAYTAkFVMQwwCgYDVQQKEwNHT1Yx
[…]
uHd5HdOcvO7mMAkGByqGSM44BAMDLwAwLAIUUwEpJ6HYXkbJ
6FOub8567nUt5DoCFC09L7n42oRs

jlAgue83VZ4o83ON
 </vers:SignersCertificate>
 </vers:CertificateBlock>
 </vers:SignatureBlock>
 <vers:SignedObject>
 <vers:ObjectMetadata>
 <vers:ObjectType>Record</vers:ObjectType>
 <vers:ObjectTypeDescription>
This object contains a record; that is a
collection of information that must be preserved
for a period
 </vers:ObjectTypeDescription>
 <vers:ObjectCreationDate>
03 Feb 2000 04:05:29 GMT
 </vers:ObjectCreationDate>
 </vers:ObjectMetadata>
 <vers:ObjectContent>
 <vers:Record>
 <vers:RecordMetadata>
  <naa:Agent>
  <naa:AgentType>Publisher</naa:AgentType>
  <naa:Jurisdiction> Victoria
  </naa:Jurisdiction>
  <naa:CorporateId>VA 527</naa:CorporateId>
  <naa:CorporateName>
Public Record Offic Victoria</naa:CorporateName>
  </naa:Agent>
  <naa:RightsManagement>
  <naa:SecurityClassification>
Unclassified</naa:SecurityClassification>
  <naa:UsageCondition>
Copyright State of Victoria 2000
  </naa:UsageCondition>
  </naa:RightsManagement>
  <naa:Title>
  <naa:SchemeType>Free text</naa:SchemeType>
  <naa:SchemeName>None</naa:SchemeName>
  <naa:TitleWords>
Victorian Electronic Records Strategy Final
Report
  </naa:TitleWords>
  <naa:Alternative>
VERS Final Report</naa:Alternative>
  </naa:Title>
  <vers:Subject>
  <vers:KeywordLevel>1</vers:KeywordLevel>
  <vers:Keyword>Archiving</vers:Keyword>
  <vers:Keyword>
Electronic Records</vers:Keyword>
  </vers:Subject>
  <naa:Description>
This report describes the Victorian Electronic
Records Strategy which deals with the problem of
indefinite preservation of digital records. The
report defines electronic records and how to
archive them, canvases possible architectures
for an electronic archive, describes
implementation issues and the theoretical and
legal background to archiving, and finally
provides a cost analysis
  </naa:Description>
  <naa:Language>en</naa:Language>
  <naa:Date>
  <naa:DateTimeCreated>
19990201:1741GMT </naa:DateTimeCreated>
  <naa:DateTimeTransacted> 20000118:124512GMT
</naa:DateTimeTransacted>
  <naa:DateTimeRegistered> 20000118:124512GMT
</naa:DateTimeRegistered>
  </naa:Date>
  <naa:AggregationLevel>
Item</naa:AggregationLevel>
  <naa:ManagementHistory>
  <naa:EventDateTime>
19990330:1700GMT</naa:EventDateTime>
  <naa:EventType>Published</naa:EventType>
  <naa:EventDescription>
Report launched on PROV Website
  </naa:EventDescription>
  </naa:ManagementHistory>
  <naa:Disposal>
  <naa:DisposalAuthorisation>
PROS 96/013 - Function Description Ref no. 8.1.0
  </naa:DisposalAuthorisation>
  <naa:Sentence>
Transfer to PROV after 5 years </naa:Sentence>
  <naa:DisposalActionDue>
```

```
20040110:0900GMT</naa:DisposalActionDue>
  <naa:DisposalStatus>
Permanent</naa:DisposalStatus>
  </naa:Disposal>
  <vers:VEOIdentifier>
  <vers:AgencyIdentifier>
  <vers:Text>VA 527</vers:Text>
  </vers:AgencyIdentifier>
  <vers:SeriesIdentifier>
  <vers:Text>VPRS 14809</vers:Text>
  </vers:SeriesIdentifier>
  <vers:FileIdentifier>
  <vers:Text>97/102</vers:Text>
  </vers:FileIdentifier>
  <vers:VERSRecordIdentifier>
  <vers:Text>HJ82750689</vers:Text>
  </vers:VERSRecordIdentifier>
  </vers:VEOIdentifier>
 </vers:RecordMetadata>
 <vers:Document>
  <vers:DocumentMetadata>
  <vers:DocumentAgent>
  <vers:Text>
Publisher is Public Record Office Victoria (VA
527)
  </vers:Text>
  </vers:DocumentAgent>
  <vers:DocumentTitle>
  <vers:Text>
Victorian Electronic Records Strategy Final
Report
  </vers:Text>
  </vers:DocumentTitle>
  <vers:DocumentSubject>
  <vers:Text>Archiving</vers:Text>
  </vers:DocumentSubject>
  <vers:DocumentDate>
  <vers:Text>19990201:1645GMT</vers:Text>
  </vers:DocumentDate>
  <vers:DocumentType>
  <vers:Text>Report</vers:Text>
  </vers:DocumentType>
  <vers:DocumentSource>
  <vers:Text>
Text laid out using Pagemaker 7.0. PDF produced
using Distiller 3.01.
  </vers:Text>
  </vers:DocumentSource>
  </vers:DocumentMetadata>
  <vers:Encoding>
  <vers:EncodingMetadata>
  <vers:FileEncoding>
  <vers:Text>
See the vers:FileRendering element</vers:Text>
  </vers:FileEncoding>
  <vers:FileRendering>
  <vers:RenderingText>
   <vers:Text>
The original document was in PDF 1.2 (Portable
Document Format) by Adobe Systems Incorporated.
The PDF was encoded into Base64 and the result
can be found in the vers:DocumentData tag.
Details of Base64 can be found in the IETF RFC
2045 "Multipurpose Internet Mail Extensions
(MIME) Part One: Format of Internet Message
Bodies", Section 6.8 "Base64 Content-Transfer-
Encoding".
   </vers:Text>
  </vers:RenderingText>
  <vers:RenderingKeywords>b64
pdf</vers:RenderingKeywords>
  </vers:FileRendering>
  </vers:EncodingMetadata>
  <vers:DocumentData>
JVBERi0xLjINJeLjz9MNCjE0MDAgMCBvYmoNPDwgDS9MaW5l
YXJpemVkIDEgDS9PIDE0MDMgDS9I
IFsgODkzIDE5MzIgXSANL0wgMjE3Nzk4OCANL0UgNjE0MTg0
IA0vTiAxNDIgDS9UIDIxNDk4Njgg
[...]
MThiNjcxMDk5ZmE4NzJmPjw3NDA1NjlkMGI3MjJhZjljMDE4
YjY3MTA5OWZhODcyZj5dDT4+DXN0
YXJ0eHJlZg0xNzMNJSVFT0YN
  </vers:DocumentData>
  </vers:Encoding>
 </vers:Document>
 </vers:Record>
 </vers:ObjectContent>
 </vers:SignedObject>
</vers:VERSEncapsulatedObject>
```