ISSN 1747-1524

DCC | Digital Curation Manual

Instalment on

"Preservation Strategies for Digital Libraries"

http://www.dcc.ac.uk/resource/curation-manual/chapters/preservation-strategies-digital-libraries

David Holdsworth

Leeds University

http://www.leeds.ac.uk

October 2007

Version 1.0

Legal Notices

Catalogue Entry

Title	DCC Digital Curation Manual Instalment on File Formats
Creator	David Holdsworth (author)
Subject	Information Technology; Science; Technology--Philosophy; Computer Science; Digital Preservation; Digital Records; Science and the Humanities.
Description	Ensuring that digital data remain accessible and reusable over time requires the implementation of proactive, scalable and sustainable preservation strategies. To be of greatest effect, preservation issues must be considered from the point of creation and throughout the entire life-cycle of the digital resource. This chapter will examine some of the technical issues surrounding digital preservation and also explore some of the philosophical issues that may hinder effective uptake and implementation of digital preservation strategies.
Publisher	HATII, University of Glasgow; University of Edinburgh; UKOLN, University of Bath; Council for the Central Laboratory of the Research Councils.
Contributor	Seamus Ross (editor)
Contributor	Michael Day (editor)
Date	1 November 2007 (creation)
Type	Text
Format	Adobe Portable Document Format v.1.3
Resource Identifier	ISSN 1747-1524
Language	English
Rights	© HATII, University of Glasgow

Citation Guidelines

David Holdsworth, (November 2007), "Preservation Strategies for Digital Libraries", *DCC Digital Curation Manual*, S.Ross, M.Day (eds), Retrieved <date>, from http://www.dcc.ac.uk/resource/curation-manual/chapters/preservation-strategies-digital-libraries

About the DCC

The JISC-funded Digital Curation Centre (DCC) provides a focus on research into digital curation expertise and best practice for the storage, management and preservation of digital information to enable its use and re-use over time. The project represents a collaboration between the University of Edinburgh, the University of Glasgow through HATII, UKOLN at the University of Bath, and the Council of the Central Laboratory of the Research Councils (CCLRC). The DCC relies heavily on active participation and feedback from all stakeholder communities. For more information, please visit www.dcc.ac.uk. The DCC is not itself a data repository, nor does it attempt to impose policies and practices of one branch of scholarship upon another. Rather, based on insight from a vibrant research programme that addresses wider issues of data curation and long-term preservation, it will develop and offer programmes of outreach and practical services to assist those who face digital curation challenges. It also seeks to complement and contribute towards the efforts of related organisations, rather than duplicate services.

DCC - Digital Curation Manual

Editors

Seamus Ross
Director, HATII, University of Glasgow (UK)

Michael Day
Research Officer, UKOLN, University of Bath (UK)

Peer Review Board

Preface

The Digital Curation Centre (DCC) develops and shares expertise in digital curation and makes accessible best practices in the creation, management, and preservation of digital information to enable its use and re-use over time. Among its key objectives is the development and maintenance of a world-class digital curation manual. The *DCC Digital Curation Manual* is a community-driven resource—from the selection of topics for inclusion through to peer review. The Manual is accessible from the DCC web site (http://www.dcc.ac.uk/resource/curation-manual).

Each of the sections of the *DCC Digital Curation Manual* has been designed for use in conjunction with *DCC Briefing Papers*. The briefing papers offer a high-level introduction to a specific topic; they are intended for use by senior managers. The *DCC Digital Curation Manual* instalments provide detailed and practical information aimed at digital curation practitioners. They are designed to assist data creators, curators and re-users to better understand and address the challenges they face and to fulfil the roles they play in creating, managing, and preserving digital information over time. Each instalment will place the topic on which it is focused in the context of digital curation by providing an introduction to the subject, case studies, and guidelines for best practice(s). A full list of areas that the curation manual aims to cover can be found at the DCC web site (http://www.dcc.ac.uk/resource/curation-manual/chapters). To ensure that this manual reflects new developments, discoveries, and emerging practices authors will have a chance to update their contributions annually. Initially, we anticipate that the manual will be composed of forty instalments, but as new topics emerge and older topics require more detailed coverage more might be added to the work.

To ensure that the Manual is of the highest quality, the DCC has assembled a peer review panel including a wide range of international experts in the field of digital curation to review each of its instalments and to identify newer areas that should be covered. The current membership of the Peer Review Panel is provided at the beginning of this document.

The DCC actively seeks suggestions for new topics and suggestions or feedback on completed Curation Manual instalments. Both may be sent to the editors of the *DCC Digital Curation Manual* at curation.manual@dcc.ac.uk.

Seamus Ross & Michael Day.

18 April 2005

Biography of the author

Leeds University was a major participant in three projects looking at digital preservation, viz Cedars (Cedars, 2002) (jointly with the Universities of Oxford and Cambridge), CAMiLEON (CAMiLEON, 2003a) (jointly with the University of Michigan), and the Representation and Rendering Project (Wheatley et al 2003). The author was heavily involved with the first two of these projects, and closely associated with the third. The strategies proposed in this chapter are very much the product of this work at Leeds University. It is developed from a tutorial paper presented at the NASA Goddard / IEEE Storage Conference in April 2004 (NASA/IEEE, 2004), further developed as a chapter in *Digital Preservation* (Deegan, 2006).

Table of Contents

Overview

The purpose of preserving things is to enable access to them at some unspecified date in the future, very probably for purposes not anticipated by the creators. Digital information technology is barely 60 years old, and all of the software from the earliest machines is already lost. It is now clear that this was material of the greatest historical significance, but history was far from the minds of those of us keen to develop the future. We should plan that our digital information will still be safe and accessible in 100 years. It is then likely that developments over that time will render the material safe for millennia. This involves a time span over which all of our existing hardware technology is likely to be obsolete, and also much of the software — a time span often far from the minds of those of us who work in IT.

This chapter is written from the perspective of a digital library, where preservation of material in pristine form has long been the tradition. A distinction is made between preservation and access. In a traditional library an original manuscript would be kept with a view to preserving as well as possible for as long as possible, whereas access might well be by means of microfilm copies. Access to the original would normally only be granted for very special historical research purposes.

In the case of digital information, material preservation is unimportant, because the copies are perfect. In fact it is often impossible to identify something that can be called “the original”. There is still justifiable concern about authenticity (See *Authenticity* below).

It is certain that the technological means for storage of digital information will change over time. The long-term preservation of printed material has focused on the preservation of the media of storage, namely ensuring the longevity of the paper and of the ink. For digital material, one must take an entirely different view. Even long-lived media such as optical disks (e.g. CDs, DVDs) will become unreadable long before they decay because the devices to read them become obsolete and unmaintainable. Choices made early in the life of a project can often have unforseen consequences with regard to its influence on digital posterity — see The section entitled *Before Ingest* (below).

However, if things are done properly, digital information can be preserved indefinitely, and at a cost that is reducing for the foreseeable future — in marked contrast to the preservation of paintings or printing on paper.

When future access is made to a preserved object, whether it be a painting by Canaletto or a database of climatic measurements, that access has to provide meaningful access to the intellectual content of the original material. In the case of the painting, there is a certain self-evidence in the visual image, but more data about the image adds markedly to the meaning of the image. Such data is, of course, meta-data. In the case of our database of climatic information there is a need for meta-data about who collected it, and why. In order to make any sense of the data, future users will also need to know in what format the data is held, so that they can use appropriate software to access the information. Such meta-data (variously called *technical meta-data* and *representation* *information*) might seem to be a special requirement of digital information, but information about the techniques used in the production of the painting, can help in ensuring its material preservation.

The key to doing things properly is to take a view of digital data as an abstract quantity, divorced from the medium upon which it is stored, but associated with information (technical meta-data – often including software) that permits ready access to its intellectual content. In this chapter we shall be concerned with this technical meta-data. The meta-data to do with provenance and other issues rightly of concern to preservation are covered in the chapter by Michael Day.

There is always the question of which information will be accessed and which information will never be used again. As our current technologies do not encompass digital clairvoyance, the best that can be done today is to make the storage costs so cheap that there is little reluctance to keep things that have only a small probability of being accessed in the future.

Accountant: People tell me that 90% of this archive will never be looked at again. It seems that there is scope for a major cost saving here.

Digital Curator: You tell me which 90% we are talking about and I will delete it.

I have personal experience of having decided to discard material as useless, and now wish that I had kept it — and to keep it would not have been very difficult. I suspect that the contents of Canaletto’s waste-bin would now be very valuable were they available today.

This chapter continues with the introduction of important concepts and standards, and then goes on to discuss the practical matters that flow from these concepts. An appendix gives 3 examples of the way in which changing digital technology impacts specific cases, and how our preservation strategies ensure continued access to the data over time.

Abstraction is Vital

Over the few decades since computers were invented, there have been many changes in the representation of data. The binary digit has survived as an abstraction, and in today’s world the byte is a world-wide standard, although in some circles it is called an *octet*. (See the note on *bits versus bytes* below.)

All that is certain for the long-term future is that there will be further change. However, even though the technology used for representing such bits and bytes has changed enormously over time, the abstract concept of data in digital form lives on. Nonetheless, the uses to which those bits and bytes can be put have grown massively over the years. Society can be confident that the concept of digital information will survive the passage of time. There is a need to bridge the longevity of the information concept to the certain mortality of the media on which the digital data lives.

The fundamental approach is to ensure that everything is represented as a sequence of bytes. I believe that it is reasonable to have confidence that the ability to store a sequence of bytes will survive for many decades, and probably several centuries. Current technology usually does this by calling this sequence a file, and storing it in a file system. There are many files in today's computer systems that had their origins in previous systems.

A new data storage paradigm may emerge in due course (and probably will), but it is unlikely to replace the digital paradigm, unless it can totally subsume it. In short, mankind will retain the ability to store bytes of data for a century or two. Conversely, it is certain that the medium upon which any digital information is recorded will be out-of-date in a decade or two. Developments of content-addressed storage (EMC, 2004) are completely consistent with this view. The "blobs" stored within the system are still byte-streams and perfectly fit this model of viewing storage as abstract.

The challenge that remains is to maintain the ability to extract the information content of such stored byte-streams. The knowledge of the formats of such preserved data is itself information, and is amenable to being represented digitally. It is thus amenable to preservation by the same means as are used for the data itself.

By taking this focus on the storage of a stream of bytes, the problem is divided into two.

1. Providing media for storage, and copying byte-streams from older technology to newer technology.
2. Maintaining knowledge of the data formats, and retaining the ability to process these data formats in a cost-effective manner.

Our work in the CEDARS project came to the conclusion that in most types of collection of digital information, one should copy the digital information from obsolete media onto current media from time to time, and also update knowledge of data formats to reflect changes in current IT practices. There is also the possibility of repeatedly converting the data formats along the way to eliminate obsolete formats. The section on *Format Conversion — when?* below treats this subject in some detail.

Much of the rest of this chapter is concerned with illustrating the viability of this strategy of abstraction and media-independence, by looking at some techniques for achieving preservation in practice. The IT industry is constantly bringing new products to market, and has a certain interest in rendering its previous products obsolete. Although the copyright laws give protection to their intellectual property, the producers of digital material do not always take seriously the obligation that copyright expiry will eventually put their products in the public domain. Digital preservation technology is poised to protect future generations from the ephemeral nature of the products of the IT industry.

Above all, the strategic planning must be truly strategic and always keep in mind the long term. This means that there will always be change ahead that cannot be foreseen — but must not be forgotten. There are those who feel that they can only face this inevitable uncertainty by printing out their digital data onto paper. If such a policy makes for peace of mind, then perhaps it is worth doing, but only as a fall back. Do the digital preservation as the master archival copy. That will be true long-term survivor. In my personal activities in the preservation of historic software, I frequently encounter offers of lineprinter listings of source text that I would dearly love to have in machine-readable form.

However, even paper copies are not always immortal. Back in 1976 or 1977 lineprinter listings of Leeds-developed software were deposited in the Leeds University Library for long-term preservation. We can no longer find them, and suspicion is that they have be discarded. Paper preservation is very demanding of space, and as such is easily seen as a target for deaccessioning. Long-term preservation of bulky material needs a constant stream of decisions to retain. This constant stream cannot be guaranteed. Retention of paper material is not as straightforward as at first appears.

I stick by my thesis that you keep the digital original and copy it from time to time, thus keeping the per item cost as low as possible with the consequent demotivation of the deaccessioners.

Standards

Standards are a good thing. We are fortunate to have so many of them.

There are various standard formats for the contents of files (e.g. GIF, JPEG, XML, MS Word), some of them more standard than others. The section on *Format Conversion* goes into more detail on this issue. The standard of prime importance in the field of digital preservation is the Open Archival Information System, known to its friends as OAIS, and to officialdom as ISO 14721 (Reich, Sawyer 2002).

Standards for meta-data have evolved over the years and have sometimes been widely adopted. The use of MARC records (Furrie, 2003) has been standard practice for many years. This success bears testament to the widely felt need for digital meta-data. However, we must not underestimate the difficulty of agreeing the format and content, as perhaps illustrated by the exercise to develop Dublin Core (Dublin, 2006), which managed to agree only 15 fields. Meta-data is covered in depth in a separate chapter.

Not only do we need care in choice of which standards to use, but we have to avoid being overly prescriptive in their application. Systems for digital preservation have to be designed to adapt to changing ideas and practices.

OAIS

This is a generic standard; it lays down the principles and style of operation of digital preservation, without specifying the detail of data formats or of hardware technology. Section 2 of the standard gives a very good resumé of the subject area.

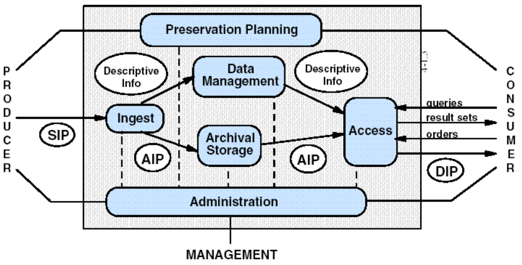


Figure 1

The overall picture is shown in Fig 1, which is taken from (Reich, Sawyer 2002). Digital material is accepted by the process of *ingest*, and is then cared for indefinitely in the *archival storage*, thus allowing *access* by people in the future. There is also a *data management* function that "contains the services and functions for populating, maintaining, and accessing a wide variety of information."

The standard talks of the *Designated Community*, "an identified group of potential Consumers who should be able to understand a particular set of information. The Designated Community may be composed of multiple user communities." For some types of archive, such as that held by a public library, the designated community is very large indeed.

In the OAIS model, the meta-data is divided between *Preservation Description Information* (PDI) and *Representation Information* (RI). The PDI is the type of information collected about most types of preserved information, and very little of it is specific to digital data. For proper coverage, see the chapter by Michael Day. On the other hand, representation information describes the digital format of the data — how to get the intellectual content from the stream of bytes.

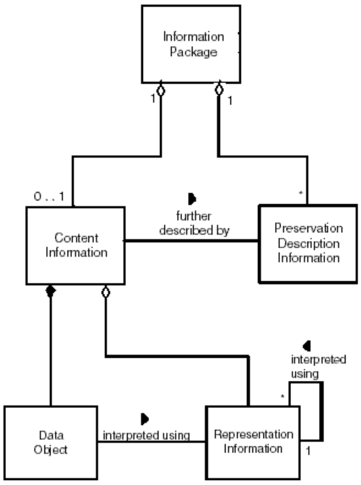


Figure 2

Figure 2 combines two diagrams from the OAIS standard to show how the representation information and the preservation description information are related to the bare digital data. (UML diagrams such as these are described succinctly in Annex C of (Reich, Sawyer 2002)). A preserved digital object is held as an information package, which has two components: the *Content Information* (*CI*) and the *Preservation Description Information*. There is no requirement that these are held together as a single blob of binary data, In many cases it might well be best to hold a large CI object in a low-cost near-line store, and keep the PDI online where it can be used for searching, or even updated as knowledge about the object is gained over time. This approach was learnt as a lesson the hard way by the Museum of Modern Art in New York, where their digitisation project stored meta-data on the same tapes alongside the digital images. They soon found that as people looked at the images, more meta-data came to light, but that the data was already packed in such a way as effectively to render it read-only (Magraeth, 2001).

Representation Information

Further down the diagram we see that the Representation Information is held alongside the data object itself. The Representation Information is that information without which the Data Object is just a sequence of meaningless binary information. Of course, representation information is just more digital data, and there is a need for some description of how to understand it. In other words, representation information can itself have representation information to describe its data formats. this is shown by the link looping round the right-hand side of the box labelled Representation Information.

As a simple example, we might wish to preserve a digital object which is an image file in GIF format. The representation information needs to tell the potential users of today how to access its intellectual content. Ideally we would like it to perform this function for the indefinite future also, but our limited digital clairvoyance circumscribes our ambitions in this area. As GIF is a format in wide use today and understood by many difrerent software packages, the representation information needed for today's user is just the fact of it being a GIF file. The version number might be nice, but that is actually embedded inside the file anyway. Better still would be the URL of the file describing the GIF format (CompuServe, 1990), which is held by the World Wide Web Consortium (W3C, 2006). This file is itself a digital object, and as such should be associated with some representation information to describe how to access its intellectual content. Traditionally documents defining internet formats are written as plain text in a fixed-pitch font. Such documents (known as RFCs) are held in this form (in which they originated) by the Internet Engineering Task Force (IETF, 2006). For documents in this simple format the representation information can take the form of a digital object that is not very large.

In principle, the chain can go on for ever. In fact it is impossible for the system to describe itself completely, so we have to stop at some point where every likely user will already know what the format of the document is. In the Cedars project (Cedars, 2002) we call called such stopping points Gödel ends, after the famous but difficult theorem known as Gödel's Incompleteness Theorem.

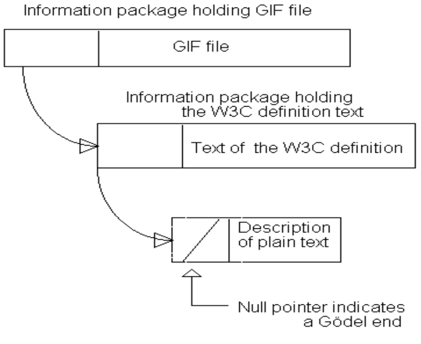

Information package holding GIF file
GIF file
Information package holding the W3C definition text
Text of the W3C definition
Description of plain text
Null pointer indicates a Gödel end

Figure 3

This representation information for our GIF file thus naturally becomes a chain of linked nodes, as shown in Figure 3. In general, the structure can usefully be more complex that a mere chain, and become a directed acyclic graph. In the OAIS model it is called a *representation net*.

Actually the vast majority of people who look at GIF images do so in an environment where the mere mouse click, or even hover, will cause the system to show the image that the file represents. On the other hand very few of them have ever read the above mentioned definition, and many are even unaware of its existence. We actually need our representation information to tell us which software will render the image, and perhaps also to record that it is a format in common use today, and is understood by common desk-top operating systems.

The OAIS representation net is the means by which the knowledge is retained. By treating all data as an abstract byte-stream at the lowest level, we have a common frame of reference in which we can record representation information, independent of any particular data storage technology, and any particular data storing institution.

Keep the Original Data

Our CEDARS work led us to the conclusion that it is folly to have faith in long-lived media (Holdsworth, 1996). Our approach is always to keep the original data as an abstract byte-stream and to regard it as the master.

Why? Because it is the only way to be sure that nothing is lost. Format conversion can lose data through moving to a representation incapable of handling all the properties of the original. It can also lose data through simple software error in the conversion process that goes undetected until it is too late to read the previous data. (I have personal experience of both situations. One in which the data was damaged, and one in which potential damage was avoided by keeping the original and producing a format conversion tool.)

How? It is certainly impossible to preserve the ability to read the medium upon which the data is stored. In Cedars we developed the concept of an *underlying abstract form* which enabled us to convert any digital object into a byte-stream from which we could regenerate the *significant properties* (Holdsworth, Sergeant 2000) of the original. Our approach is to preserve this byte-stream indefinitely, copying it unchanged as storage technology evolves.

The question then remains as to how to achieve continued *access to the intellectual content* (another Cedars phrase) of the data, and not merely a stream of bytes. Our answer to this is that the digital curation exercise should evolve the representation information over time, so that it provides the means to transform the original into a form that can be processed with the tools current at the time of access. We believe that our work in the CAMiLEON project has shown this to be feasible in the case of a very difficult original digital object of great historical importance. Using emulation we successfully preserved the accessibility of the BBC's "Domesday" project, see below and (CAMiLEON, 2003b).

The very essence involves identifying appropriate abstractions, and then using them as the focus of the rendering software. We achieve longevity by arranging that the rendering software is implemented so as remain operational over the decades. The application of our approach to emulation is covered in *Emulation, Preservation and Abstraction* (Holdsworth, Wheatley 2001). We have also investigated the same technique of retention of the original binary data coupled with evolving software tools in the context of format migration (Mellor, Wheatley, Sergeant 2002).

Ingest

In the OAIS model, the word "ingest" is used to describe the process whereby digital information comes into the archive. This probably involves an appropriate form of agreement between the data producer and the archive. Alternatively there may be some legal requirement for digital legal deposit, or even a curator's decision that the material should be preserved for posterity, whatever the producer thinks.

Whatever the politics, there is a need to take in digital information held on some medium, and to preserve it so that it can be accessed after technology had moved on to the use of different media, and probably also different software systems.

We recommend a process of ingest which contains the steps:

1. First step is separation of the data from the medium
2. Second step is to map to a byte-stream (i.e. make the *data-object* in Figure 2), and to produce representation information that allows access to the intellectual content of the original.

This byte-stream is then preserved indefinitely within the archive store, in such a way that it contains a reference to the appropriate representation information, of which, more later.

The form of the data between steps 1 and 2 is the *underlying abstract form* (UAF - see below). These two steps are the parts of *ingest* process that involve the data itself. Of course, there are other steps involving the meta-data, dealt with in Michael Day's chapter, and which the OAIS model sees as construction of the PDI. We find that the UAF concept enables us to identify the *significant properties* of the original that must be preserved if subsequent meaningful access to the preserved digital object is to be achieved. This includes such processes as recreating the experience of viewing (or even interacting with) the original, or analysing a scientific data set in order to test a new theory. For particularly complex objects, emulation might well need to be involved.

We are concerned here with generating the *representation information* (RI) that goes alongside the data object within the content information of Figure 2. For various good reasons explained below, the representation information is best included by reference, rather than as a copy alongside the data.

At its most basic, the RI enables a reversal of the ingest process to deliver a viable copy of the original. This observation provides a useful minimum criterion for testing the acceptability of any scheme for RI and also for the RI of any particular information object.

The representation information must allow the recreation of the significant properties of the original digital object, if one assumes that appropriate hardware technology is available. *Verbatim quote from* (Holdsworth, Sergeant 2000).

Underlying Abstract Form (UAF)

In the Cedars project we used the term *underlying abstract form* (UAF) to encapsulate the recognition that the data has an existence and a content separate from the medium upon which it is written. It is not a term used in the OAIS model. The concept will apply to any attempt to preserve digital data in a medium-independent form.

This underlying abstract form contains all the significant properties of the data, and is independent of the medium upon which the data is written. Any given digital object is likely to have a number of possible UAFs. Choice of the UAF for preservation is part of the ingest process (either in the *receive submission* box or the *Quality Assurance* box in OAIS fig 4-2). Once this choice is made the data can be copied from its submission medium into the data storage facilities of the archive.

Some examples:

- Many a CD actually contains a file system, and successful operation only relies on that file system. Copying such a file system onto a partition on a hard disk delivers an equivalent working representation. File placement is unimportant. Thus the file system is a viable underlying abstract form.
- In some cases it is only important to have a file tree, and the CD contents can be copied into a directory within an existing file system.
- In the case of a complex game CD or DVD, the underlying track structure of the disk may be

important for some of the special effects. This track structure then becomes a significant property, and the picture of the disk as a file system would not be a valid UAF, as it would not encompass all the significant properties.

- Data held in a relational data-base can equally well reside in a variety of data-base engines, and still deliver its original content. Comma-separated files holding the contents or each table can be used as a system-independent representation of that content. There would need to be discussion as to whether the schema were to be part of the representation information, or part of the digital object.
- A plain text document consisting of lines of characters drawn from the ASCII character set is meaningful in a variety of environments. Internet RFCs are typical of such documents.

Access involves realising the UAF on the technology appropriate to the time of access in such a way that the desired form of access (which may not necessarily be viewing) can be achieved. If we take the simple example of the Internet RFC, the same UAF is stored slightly differently on a UNIX system from a PC file system, because of the different conventions for line termination. However, the UAF of lines of text is the same in each case, and the UNIX `cat` command clearly displays the same information as the PC's `NOTEPAD`. The same lines of text represented in an EBCDIC system would be represented very differently in terms of binary digits. The same data would be displayed in the same form. The underlying abstraction of lines of text is the same, but the different platforms represent the information differently internally.

The actual process of future access will involve the use of some kind of computing platform, at whose characteristics we can only guess. It seems likely that the idea of software will continue to exist. Certainly in the early 21st century it shows not the slightest sign of going away in the near future. As well as assisting the identification of significant properties, the UAF can be used to partition the process of access:

1. Recreate the data in its underlying abstract form on technology commonly available at the time of access.
2. The intellectual content is then accessed (e.g. viewed – a process sometimes called rendering) using software.

The software for each of these steps will be identified from information in the representation net. It is important to choose a UAF that is likely to last for a long time, such as a file tree. In some cases the UAF may itself be just the preserved byte-stream, and all the work of viewing is done by the rendering software, after a trivial copying process in step 1.

It is very likely that the rendering will be done by some widely available software, after some conversion of the data format from that in which it was submitted to the archive (i.e. “ingested”). In the next section we discuss the future-proofing of this process.

A digital object may be configured for a variety of platforms (e.g. many a CD will work with both Mac and PC), and the chosen UAF may well encapsulate this. It is up to collection managers to decide whether it is a significant property of the original digital object. The technology should give them that option.

Before Ingest

It is clear that it is easier to decide an underlying abstract form for some data formats than for others. Where proprietary data formats are involved this decision may be hampered by lack of detailed information. The key question concerns the extent to which access to the significant properties of the information is dependent upon proprietary software's processing of data formats which are not publicly known. The longevity of the data could then be dependent on the longevity of the proprietary software and perhaps also on the longevity of the platform upon which it operated. However, for a widely-used proprietary format, there is the prospect that the format will be reverse-engineered in the open source community, such as has happened with Microsoft Word (*viz* Open Office(Sun Microsystems 2006)) and Acrobat PDF (*viz* Xpdf (Foo Labs 2004)).

A legal deposit library usually has little influence over the format of material that it must accept, and must be prepared to face the difficulties of preservation of data whose format is not what it would have chosen. On the other hand any project setting out on a data collection exercise should consider the issues around long-term preservation of its raw data as part of its choice of IT systems for data management. The story of the BBC's Domesday project (see below) shows how these matters were not properly thought through in 1986. It was a case of operating at the (b)leading edge of the technology of time, without giving proper credance to the probability that future technology might develop in a different direction; "abstraction is vital". In the early stages of project implementation, it is probably wise to treat with caution the blandishments of computer salesmen. Conversely, some science projects can only be undertaken by operating at the leading edge of data-handling technology, and long-term preservation activities may have to wait for further technological evolution. Even then, they should be kept in mind.

Format Conversion — when?

It is obvious that when data is to be accessed some time after its initial collection, the technology involved in this access will differ markedly from that in use when data collection took place. There is also the real possibility that other technologies have been and gone in the interim. Thus, format conversion at some point is inevitable.

The opinions expressed here have a resonance with the traditional library view, as expressed in the *Overview*, where historical accuracy has a high priority. Where the emphasis is more on re-use and reworking of digital information, as happens in much e-science work, the preferences for digital preservation strategy might be different.

The controversial choice is between periodic conversions to ensure that master copies of data are held in current formats, and retention of the original byte-stream with format conversion only undertaken when access is required. The author's personal preference is for the latter option, but this is not a universally held view.

If the *Cedars* policy of copying the unchanged original digital information from obsolete media onto current media from time to time is adopted, this requires us to be able to perform format conversion when access is required. To do this, it is necessary to update knowledge of data formats to reflect changes in current IT practices. The question as to who might undertake this open-ended commitment is dealt with in the section on *Share and Cross-Reference Representation Nets*.

There is also the possibility of repeatedly converting the data formats along the way to eliminate data held in obsolete formats. This is superficially attractive, in that access is always easy, but the convenient division of the problem is undermined. The only time at which format conversion of the whole data is likely to be cost-effective is when copying from one medium to another, thus changing a basically straightforward job into a much more complex one. It is vital that the copying process does not lose any information (i.e. preserves all the significant properties), because the data input to the copying process is about to be destroyed. Just as one of the periodic reviews of archived paper at Leeds University's Library took a bad and irrevocable decision, there is always the danger of a format conversion that (irrevocably) loses a significant property. Any mistakes in this complex process will lead to erosion of the intellectual content over time. On the other hand, errors in the representation net relating to past data formats will usually be amenable to correction.

Repeated format conversion may be suitable for an archive addressing a particular restricted field (say the CAD data for nuclear power stations), where the number of formats is small, and the whole regime includes specification of allowed formats at the outset. Archives such as libraries do not have this luxury, and need to able to absorb all appropriate material, no matter in what format it is held.

For data held in currently common formats, the amount of representation information needed is trivial. Meaningful access to the data normally happens at the click of a mouse. A current computer platform will render a PDF file merely by being told that the format is PDF. Conversely, faced with an EBCDIC file of IBM SCRIPT mark-up, the same current platform might well render something with little resemblance to the original, whereas back in 1975, the file could be rendered as formatted text with minimal formality. (Actually, the user had to type the word SCRIPT followed by the name of the file.)

However, if the representation information for IBM SCRIPT files is updated over time so that it points at appropriate software for rendering the file contents on current platforms, the historic data becomes accessible to today’s users. Alternatively, all the world’s IBM SCRIPT files could have been converted into Word-for-Windows, or L^{A}T$_{E}$X, or .... The argument about the choice could continue until all the current formats become obsolete, and it could well be that the chosen format quickly became obsolete. Of course, there would be the temptation to convert from EBCDIC to ASCII, but that could have lost information because EBCDIC contains a few more characters than ASCII.

In my own experience my colleagues and I at Leeds University debated in 1990 whether to convert the EBCDIC text files in our VM/CMS archive to ASCII characters as we shifted to a basically UNIX and PC platform. Instead we kept the EBCDIC data in its original format and wrote a program to convert the files to ASCII on demand. A small enhancement to this program produces a program that converts to UNICODE instead, preserving all the characters that exist in EBCDIC and not in ASCII. Back in 1990, UNICODE had a very low profile.

This chapter argues for a policy in which the format of preserved data is converted only when access is required to the data, i.e. on creation of what the OAIS model calls *the Dissemination Information Package* (DIP). For a popular item, it would obviously make sense to cache the data in a format that is in current use, but not to allow the reformatted data to replace the original as master. This means that the tracking of developments in storage technology involves only the copying of byte-streams. Moreover, when the format conversion has to be done, there will be improved computational technology with which to do it (Wheatley, 2001).

Authenticity

A further concern regarding format conversion concerns authenticity. The current technique for ensuring the authenticity of any digital document is to use a digital signature. This involves computing a *digest* of the document, and then encrypting that digest with the private key of the authenticating authority (e.g. author). The corresponding public key can then be used to confirm the authenticity of the document. A converted document would not pass this test, and re-signing would involve access to a private key that may well not be available. Clifford Lynch (Lynch, n.d.) has studied this problem in the contect of format conversion.

A further issue arises owing to the long time-scales envisaged. It is reasonable to expect our current digital cryptographic technology to be broken in due course. Say, in 20 years time we may be able to deduce the private key from the public key in the signature systems in use today. However, if the digital archive re-signs the digital object and its existing signature with a new technology signature before the existing signature is compromised, this will confirm there were no previous modifications, and will enable the detection of subsequent modifications. The process can be repeated indefinitely, and should perhaps form part of the routine process of copying from old media to current media, becuase it will be necessary to read the entire byte-stream in order to confirm the old signature and to compute a new one.

Indirection is Vital

There isn't a problem in computer science that cannot be solved by an extra level of indirection. *Anon*

The essence of our approach involves keeping the preserved data unchanged, and ensuring that we always have representation information that tells us how to access it, rather than repeatedly converting to a format in current use. Without doubt, it is very difficult (impossible?) to provide representation information that will be adequate for ever. This is likely to be true even if the format of the master is periodically changed. Our work at Leeds and elsewhere has led us to propose that representation information evolve over time to reflect changes in IT practice, but that the preserved digital objects be kept in their original abstract form. This clearly implies a structure in which each stored object contains a pointer to its representation information. This is easily said, but begs the question as to the nature of the pointer. This must be a pointer that will remain valid over the long-term (i.e. 100 years). The world needs to be wary of depending on institutions whose continued existence cannot be guaranteed. What we need is not so much a pointer as a reference ID for each preserved object. This needs to be distinct from the location of the object, but there needs to be a service that translates a reference ID into a current location. The reference ID has thus become a pointer to a pointer. This is the essence of the Cedars architecture (Holdsworth, 2002). Reference IDs could be managed locally within an archive store. Such IDs could then be made global, by naming each archive store, and prefixing each local name with that of the archive store.

There are various global naming schemes, ISBN, DNS, Java packages, URL, URI, URN, DOI, etc. It may even be necessary to introduce another one, just because there is no clear candidate for long-term survival. What is certain is that there have to be authorities that give out reference IDs and take responsibility for providing resolver facilities which translate these IDs into facilities for access to the referenced stored objects. Currently the DOI handle system (DOI, 2005) has a resolver service at `http://dx.doi.org/`.

If the digital storage community grasps the nettle of a global name space for reference IDs of stored objects and keeps the representation information in the same name space; there is even the prospect of sharing the evolving representation information on a world-wide basis. Some discipline will be needed if dangling pointers are to be avoided.

Enhance Representation Nets over time

In the Cedars Project we produced a prototype schema for a representation net following the OAIS model, and populated it with some examples. After this experience, we had some new ideas on the schema of the representation net. We believe that it is inevitable that this area is allowed to develop further, and that operational archives are built so that evolution in this area is encouraged to take place. It is important to accept that there is likely to be revision in the OAIS model itself over the 100 year time-frame.

Also, we could see that to require a fully specified representation net before ingest is allowed could act as a disincentive to preservation of digital objects whose value is not in doubt. In many cases, representation information exists as textual documentation. An operational archive needs to be capable of holding representation information in this purely textual form, although with an ambition to refine it later. Such information would not actually violate the OAIS model, but there is a danger of being over-prescriptive in implementing the model. For instance the NISO technical metadata standard for still images (NISO, 2004) has over 100 elements, at least half of which are compulsory.

For some formats the most useful representation information is in the form of viewing software. It is desirable for representation nets to enable the discovery of such software (see below). Many current objects need only be introduced to a typical desktop computer in order for them to be rendered. On the other hand, at Leeds we experimented with obsolete digital objects (from 1970s and 1980s) in order to see some of the issues likely to arise when our grandchildren wish to gain access to today’s material. We even tried to imagine how we would have gone about preserving for the long-term future using the technology of the 1970s. It was abundantly clear that ideas are very different now from those of 30 or 40 years ago. Designers of systems must expect that today’s ideas could well be superseded over the long-term.

In order to accommodate this, systems must allow the content of objects in the representation net to be changed over time, in sharp contrast to the original preserved objects where the recommendation is for retention of original byte-streams. It is vital that the reference ID that is originally used for representation information be re-used for newer representation information which gets produced as a result of development of new tools and ideas. That way, old data gets to benefit from new techniques available for processing it. The representation information that is being replaced should of course be retained, but with a new ID, which should then be referenced by the replacement.

Representation Nets should link to software

Our representation nets in Cedars very deliberately contained software, or in some cases references to it. We have no regrets on this issue. Ideally software should be included in source form, in a programming language for which implementations are widely available. However, it would have been be churlish to refuse to reference the Acrobat viewer as a way of rendering PDF files, just because we did not have the source, but see example 1 below.

A format conversion program that is known to work correctly on many different data objects is clearly a valuable resource for access to the stored data, and should be available via the representation network.

As regards the issue of longevity of such software, we argued earlier for the longevity of abstract concepts such as bits, bytes and byte-streams. Programming languages are also abstract concepts, and they too can live for a very long time. Current implementations of C or FORTRAN will run programs from long ago. Other languages which have been less widely used also have current implementations that function correctly.

The source text of a format conversion program which is written in a language for which no implementation is available is still a valuable specification of the format, and has the benefit of previously proven accuracy. We address the issue of evolving emulator programs in *C-ing Ahead for Digital Longevity* (Holdsworth, 2001), which proposes using a subset of C as the programming language for writing portable emulators.

Share and Cross-Reference Representation Nets

An earlier section argued for the impossibility of producing an adequate standard for representation information which would retain its relevance over the decades. To attempt to do so would stifle research and development. It is therefore to be expected that different data storage organisations may develop different forms of Representation Information. Initiatives such as the PRONOM (National Archives 2004) file format database and the proposed Global File Format Registry (Harvard, 2002-2005) will also produce valuable resources that should be linked from representation information. It would seem that collaboration should be the watchword here. For instance, the emerging solutions for IBM SCRIPT files in example 2 are likely to be applicable to any institution holding such data. With our proposed global namespace, they can all reference the same representation net, and benefit from advancing knowledge on the rendering of such files.

Global Considerations

The NASA Goddard / IEEE Storage Conference in 2004 (NASA/IEEE, 2004) had the theme Long-Term Stewardship of Globally-Distributed Storage, and this chapter draws heavily from the author's paper at that conference (Holdsworth, Wheatley 2004)).

The implementation of preservation on a global basis means that there will be no overall command, and co-operation will have to be by agreement rather than by diktat. This situation has some aspects that resemble the problems of achieving true long-term preservation. We cannot predict the future accurately, nor can we control it to any great extent, so the ambition to operate on a global scale despite being unable to control activities everywhere in the world sits well with the need for future-proofing. The future is another country whose customs and practices we cannot know.

Referential Integrity

On the World Wide Web, links that lead to pages that no longer exist (known as dangling pointers by computer scientists) are perhaps inevitable (though annoying) in such a dynamic and anarchic network. Before criticising the anarchy, it is important to acknowledge that it is the source of much of the dynamism.

Digital storage archives referencing representation information on a global scale also have potential for the generation of dangling pointers. Furthermore, the whole *raison d'être* of the archive makes it inevitable (and desirable) that there will be lots of pointers into the archive using the archive's own reference IDs for its stored digital objects.

I would contend that once a reference ID has been allocated, it should exist for ever in the resolver services. Ideally, the object to which it points should have the same longevity, but if it should be lost or deleted, that fact should be revealed by the resolver. In the case of representation information, it may be modified, but never deleted. Thus, anyone may use a reference to an object in the OAIS digital storage world confident that it will never become a dangling pointer.

A vital part of the management of such an archive will involve keeping an inventory of the external references in the representation nets (so called Gödel ends), and maintaining a process of review of the inventory in the search for things that are no longer generally understood or refer to information that is no longer available. The remedy in such cases is to update the referring nodes to reflect the new realities. Clearly it is in the interests of good management to try to keep the number of such nodes to a minimum.

For example, a store would have a single node that describes the current version of Microsoft Word to which the representation information for any ingested Word file would refer. When this version becomes obsolete, this one node is updated with information on how to access data in the old format, or to convert to a newer format.

The two level naming proposed earlier helps greatly in implementation of such a policy.

Cost-Effectiveness

We earlier talked of keeping down the cost per item stored so that there was less pressure to discard things that might well be useful. Many things that survive to be exhibits in museums were initially seen as current fashion, then went downhill in the scale of importance, often descending to the level of rubbish, before scarcity and antiquity made them interesting. Much of what survives had a period when its survival was purely by accident. Archeologists spend much of their time sifting through the rubbish bins (and worse) of ancient settlements. Things which survive tend to do so because there is little to be saved by destroying them. If it costs very little to keep digital data, we might resist the temptation to discard those items of little interest to us, but which later researchers might find valuable.

Digital data has a very special and convenient property: the cost of keeping it falls over time. Reagan Moore's (Moore, 2004) group at UCSD has a policy of copying data when the medium upon which it is stored costs twice as much as would current technology. Thus each copying involves half the expenditure of the previous copying. So the total media cost of keeping the information for ever is only twice the cost of original storage media, including that original expenditure.

The policy of keeping the original byte-stream, unmodified after ingest, means that the incremental cost of keeping something which shares its representation information with other items is little more than that of the media cost.

Summary

I argue strongly for retention of the original in the form of a byte-stream derived as simply as possible from the original data, and for the use of representation information to enable continued access to the intellectual content.

I take the view that for much material it is impossible to have perfect representation information at the time of ingest, but that we must preserve the data and develop its representation information over time.

Ideas on the nature of representation information will evolve over time. We must have systems capable of taking on board changing schemas of representation information.

A two-level naming system, separating reference ID from location (and translating between them) should be the practice for implementing pointers in an OAIS archive, as a prerequisite for our proposed policy of evolving representation information over time, and sharing it on a global scale.

A Footnote on Bits versus Bytes

This chapter talks of preserving byte-streams rather than bit-streams, even though the OAIS model uses the bit as the lowest level.

$$1+\frac{1}{2}+\frac{1}{4}+\frac{1}{8}+\frac{1}{16}+\frac{1}{32}+\frac{1}{64}+\ldots\ldots\ldots=2$$

However, the byte is the ubiquitous unit of data storage. In today's systems one cannot see how the bits are packed into bytes. When a file is copied from one medium to another we know that whether we read the original or the copy, we shall see the same sequence of bytes, but we know nothing of the ordering of bits within the byte, and these may be different on the two media types. On some media (e.g. 9-track tape) the bits are stored side-by-side.

Pragmatically, we regard the byte as the indivisible unit of storage. If the OAIS model requires us to use bits, then we shall have a single definition of the assembly of bits into a byte. This would enable us unambiguously to refer to the millionth bit in a file, but not constrain us to hold it immediately after the nine-hundred-and-ninety-nineth bit.

Appendix 1

Examples

We illustrate the way in which we see representation information evolving over time, by reference to three examples drawn from rather different computational environments.

Example 1: Acrobat files

In today’s IT world it is very common to use Adobe Acrobat® portable document format (PDF) for holding and transmitting electronic forms of what are thought of as printed documents. The only representation information needed by today’s computer user is the URL for downloading the Acrobat® Reader™. The representation net for PDF files is basically this single node, detailing how to gain access to the software for rendering the data. In reality, it should be an array of nodes with elements for different platforms. All preserved PDF files would reference this one piece of representation information. The recent appearance of the GNU open-source Xpdf (Foo Labs 2004) would be reflected by adding it to this array.

Example 2: IBM SCRIPT files

Once upon a time, the representation information for a preserved IBM SCRIPT file would point to the IBM SCRIPT program for the IBM/360 platform. Unfortunately, there was no OAIS model in the 1970s, but if there had been an OAIS archive for storage of our VM/CMS data, this is the only representation information that would have been needed. (Actually the CMS file-type of SCRIPT performed the role of representation information, much as file extensions do today on a PC.)

As the 30+ years elapsed, our putative OAIS archive would have expanded the representation information for SCRIPT by information suitable for more current platforms — including the human readable documentation for a live-ware platform. There would probably also be reference to the Hercules project (Hercules, 2001) which allows emulation of IBM/360/370 systems of yesteryear. This need to keep up-to-date was highlighted in the InterPARES project (InterPARES, 1999-2001), which continues as InterPARES2 (InterPARES, 2002-2006).

Example 3: The BBC Domesday Project

In 1986, to commemorate the 900th anniversary of the Domesday Book, the BBC ran a project to collect a picture of Britain in 1986, to do so using modern technology, and to preserve the information so as to withstand the ravages of time. This was done using a micro computer coupled to a Philips LaserVision player, with the data stored on two 12" video disks. Software was included with the package, some on ROM and some held on the disks, which then gave an interactive interface to this data. The disks themselves are robust enough to last a long time, but the device to read them is much more fragile, and has long since been superseded as a commercial product.

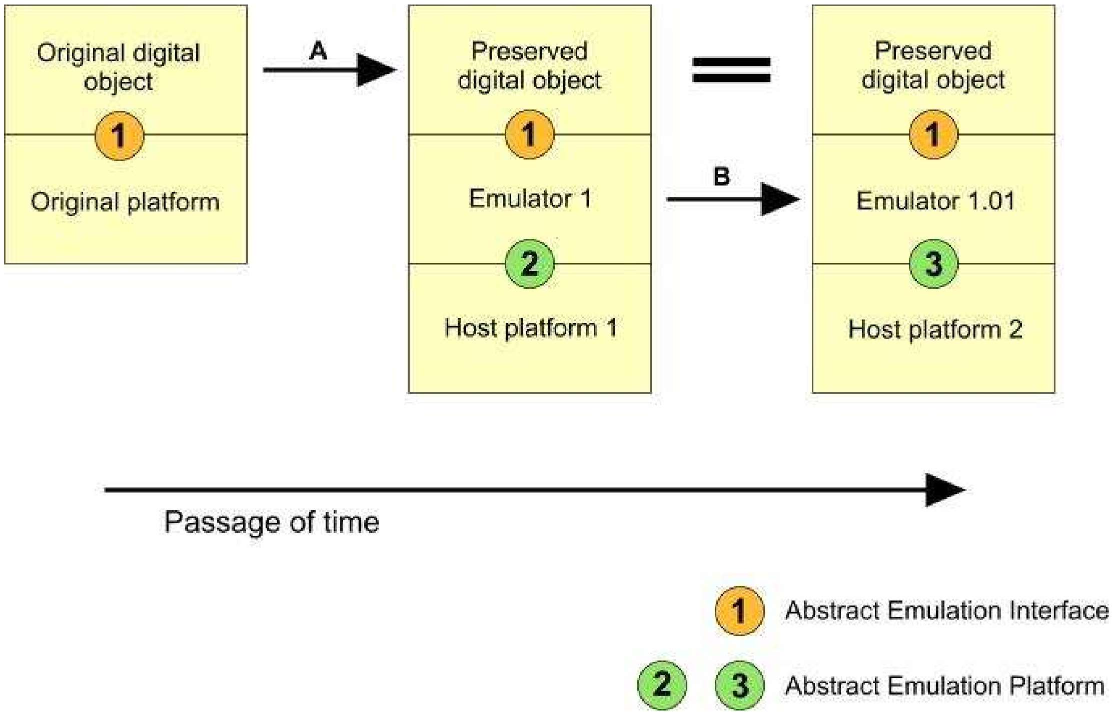
Here is a clear example where the preservation decisions placed (mis-placed) faith in the media technology of the day, and more crucially in the survival of the information technology practices of the time.

The CAMiLEON project used this example as a test case to show the effectiveness of emulation as a preservation technique. A detailed treatment is to be found on the CAMiLEON web site (CAMiLEON, 2003b).

We can look at this example with particular reference to its long-term viability, both with regard to the original efforts in 1986, and to the emulation work of 2002. We shall use it to illustrate our ideas about the appropriateness of emulation software as part of the representation information.

Firstly, a bit of background to the work.

In the project, we took our own advice and preserved the data from the original disks as abstract byte-streams. We can represent this step as the process marked A in the diagram (taken from reference (Holdsworth, Wheatley 2001)):



The technique was to show that we could use emulation to bridge from the Original platform to a different host platform, labelled Host platform 1 in the diagram. The ingest step (marked A in the diagram) involves identifying the *significant properties* of the original. The data consisted of the four disk surfaces, each with 3330 tracks, and some software in ROM held inside the BBC micro computer. Some tracks were video images and some held digital data which was often textual. We preserved the

ROM contents straightforwardly as binary files, and made each track of the disk into a binary file of pixels for the video images, and a straightforward binary file for each of the digital data tracks. This we claim preserves the significant properties of the software and data necessary for it to run on the BBC computer with its attached video disk player. An example representation network describing the capture process was constructed as part of the Representation and Rendering Project (Wheatley et al 2003)

To demonstrate the validity of this claim, we produced the emulator shown as Emulator 1 on the diagram. The original software relied on an order code and an API (applications program interface) labelled 1 in the diagram. In order to achieve successful preservation of this digital object, we need to reproduce this API with software that operates with a more modern API, labelled 2 in the diagram.

The emulation of the BBC micro-computer was obtained from an open-source emulation written by an enthusiast (Richard Gellman) and available on the internet (Gellman, Gilbert 1994). Although the achievements of enthusiasts are not always ideally structured for use in digital preservation work, they can often provide a useful starting point for further development. At the very least the source code can act as a handy reference point for new work.

The emulation of the video disk player was done by our own project staff. This emulation software then becomes the major component of the representation information for this data. Its longevity depends crucially on the longevity of the interface labelled 2. Here we have used code that is written in C, and makes use of only a few Win32-specific API calls. In other words our interface labelled 2, is not the whole API of Host platform 1, but only the facilities that we have chosen to use. The move to another platform is made easier by choosing to use as few as possible of the proprietary features of Host platform 1. We may need to recode a few bits of the screen driving routines, but by and large we can expect to find on Host platform 2 an API (shown as 3) that has most of the features needed on the new platform. We expect that a slightly revised emulator called Emulator 1.01 will readily be generated (step B) to run on Host platform 2. Meanwhile, the preserved digital object will be completely unchanged, as indicated by the large equals sign.

Example 3: The BBC Domesday Project — Evolution of Representation Information

At the outset, the storage media consisted of two 12" video disks. The representation information (a booklet supplied with the disks) advised the purchase of the appropriate hardware including the two E-PROM chips holding software that is used in accessing the video disk player. In addition, the BBC microcomputer had a well documented API for applications programs. This API (or preferably the subset of this that happened to be used) provides the interface labelled 1 in the diagram.

Our preservation of the data from its original preservation medium created byte-streams that closely mirrored the actual physical data addressing. This maximised the validity of the existing representation information, *viz.* the documentation of the API mentioned above.

The emulator then implements this API, opening up the question of the API upon which it itself runs. Thus we add to the representation information the emulator, and the information concerning the API needed to run it. This is not yet stored in a real OAIS archive, but we do have the materials necessary to achieve this, and the data from the disks is stored in our LEEDS archive (Holdsworth, 1992).

Our care in producing an emulation system that is not tied too closely to the platform upon which it runs illustrates our desire to produce representation information that will indeed stand the test of time by being easily revised to accommodate newly emerging technologies. This revised emulator becomes an addition to the representation information, extending the easy availability of the original data to a new platform. InterPARES (InterPARES, 1999-2001) identified clearly the desire of users to access the material on the technology of their own time.

So why emulate in this case? The interactive nature of the digital object is really a part of it. There is no readily available current product that reproduces that interaction, so we treat the interaction software as part of the data to be preserved. On powerful implementations of current desk-top hardware, it runs faster than the original.

Since the CAMiLEON work, the National Archives in London have outsourced a modern web implementation of the access to the original information(Pearce, n.d.). This information should now be added to our representation information for the original data.

References

CAMiLEON (2003a) *CAMiLEON project* www.si.umich.edu/CAMILEON/

CAMiLEON (2003b) *Domesday* www.si.umich.edu/CAMILEON/domesday/domesday.html

Cedars (2002) *Cedars project* www.leeds.ac.uk/cedars/

CompuServe (1990) *GIF89a file format* www.w3.org/Graphics/GIF/spec-gif89a.txt

DOI (2005) *The Digital Object Identifier System* www.doi.org/

Deegan,M. and Tanner,S. (2006) Digital Preservation, *Facet Publishing*, ISBN 978-1-85604-458-1

Dublin Core (2006) *Dublin Core Metadata Element Set, Version 1.1: Reference Description* http://dublincore.org/documents/dces/

EMC (2004) *EMC^2 Centera* www.emc.com/products/systems/centera.jsp

Foo Labs (2004) *Xpdf Acrobat® renderer* www.foolabs.com/xpdf/about.html

Furrie,B, Library of Congress (2003) *What is a MARC record, and why is it important?* www.loc.gov/marc/umb/um01to06.html

Gellman,R. and Gilbert,D. (1994) *Richard Gellman and David Gilbert, BBC Emulator* www.mikebuk.dsl.pipex.com/beebem/

Harvard (2002-2005) *Global File Format Registry* http://hul.harvard.edu/gdfr/

Hercules (2001) *Hercules IBM Emulator* www.schaefernet.de/hercules/index.html

Holdsworth,D. (1992) *LEEDS archive* www.leeds.ac.uk/iss/systems/archive/

Holdsworth,D. (1996) *The Medium is NOT the message.* Fifth NASA Goddard Conference on Mass Storage Systems and Technologies, NASA publication 3340, September 1996. http://esdis-it.gsfc.nasa.gov/MSST/conf1996/A6_07Holdsworth.html

Holdsworth,D. (2001) *C-ing Ahead for Digital Longevity* www.si.umich.edu/CAMILEON/reports/cingahd.html

Holdsworth,D. (2002) *Cedars architecture* www.leeds.ac.uk/cedars/archive/architecture.html

Holdsworth,D. and Sergeant,D.M. (2000) *A blueprint for Representation Information in the OAIS model* http://romulus.gsfc.nasa.gov/msst/conf2000/PAPERS/D02PA.PDF

Holdsworth,D. and Wheatley,P.R. (2001) *Emulation, Preservation and Abstraction,* RLG DigiNews vol5 no4, 2001 www.rlg.org/preserv/diginews/diginews5-4.html#feature2

Holdsworth,D. and Wheatley,P.R. (2004) *Long-term Stewardship of Globally-distributed Representation Information,* NASA/IEEE Storage Conference 2004, NASA/CP-2004-212750 http://romulus.gsfc.nasa.gov/msst/conf2004/Papers/MSST2004-03-Holdsworth-a.pdf

IETF (2006) Internet Engineering Task Force www.ietf.org/

InterPARES (1999-2001) *InterPARES project*, see also next item www.interpares.org/book/index.cfm

InterPARES (2002-2006) *InterPARES2* www.interpares.org/ip2.htm

Lynch,C. (n.d.) *Authenticity and Integrity in the Digital Environment: An Exploratory Analysis of the Central Role of Trust* www.clir.org/pubs/reports/pub92/lynch.html

Magraeth,M (2001) Michael Magraeth, Museum of Modern Art – private communication

Mellor,P., Wheatley,P.R. and Sergeant,D.M. (2002) *Migration on Request, a Practical Technique for Preservation* in Research and Advances Technology for Digital Technology : 6th European Conference, ECDL 2002 pp. 516 - 526 www.springerlink.com/openurl.asp?genre=article&issn=0302-9743&volume=2458&spage=516

Moore,R. et al (2004) *Data Grid Management Systems*, NASA/IEEE Storage Conference 2004, NASA/CP-2004-212750 http://romulus.gsfc.nasa.gov/msst/conf2004/Papers/MSST2004-01-Moore-a.pdf

NASA/IEEE (2004) *NASA / IEEE Conference* NASA/CP-2004-212750 www.storageconference.org/2004/

NISO (2004) *NISO technical metadata standard for still images* www.niso.org/committees/committee_au.html

National Archives (2004) *PRONOM* www.records.pro.gov.uk/pronom/

Pearce,A. (n.d.) *Adrian Pearce Long Life Data's implementaion of Domesday 1986* http://domesday1986.com/

Reich,L and Sawyer,D (2002) *Reference Model for an Open Archival Information System (OAIS)* ISO 14721:2003: http://public.ccsds.org/publications/archive/650x0b1.pdf

Sun Microsystems (2006) *Open Office* www.openoffice.org/

W3C (2006) *World Wide Web Consortium – W3C* www.w3.org/

Wheatley,P.R. (2001) *Migration — A CAMiLEON discussion paper*, Ariadne, Issue 29(September 2001) www.ariadne.ac.uk/issue29/camileon/

Wheatley,P.R. et al (2003) *Representation and Rendering Project* www.leeds.ac.uk/reprend/