ScieCom info
Svenskt Resurscentrum
för vetenskaplig kommunikation
www.sciecom.org

Tomorrow's File Endings: On Archiving Principles and Archiving Formats

Marcus Uneson, PhD Student at
Dept. of Linguistics, Lund University
Marcus.Uneson@ling.lu.se

Artikeln i pdf-format för utskrift

Sammanfattning på svenska

Artikeln diskuterar utan tekniska detaljer några viktiga aspekter på långtidsarkivering, särskilt av vetenskapliga publikationer. Efter en översiktlig framställning av fysisk datasäkerhet läggs tonvikten på sunda principer för dokumentarkivering i allmänhet, och särskilt på hur dessa kan vägas in vid val av filformat. Ur den gyllene regeln ”undvik varje val som i onödan begränsar framtida möjligheter” härleds tre grundsatser för arkivering av publikationer: a) använd enkla, textbaserade format; b) använd öppna standarder; c) arkivera en representation, inte en presentation. Några vanliga arkivformat för text idag kommenteras utifrån dessa ståndpunkter, särskilt pdf, pdf-a, tex/latex, xml.

Abstract

The article discusses without technical detail a few important aspects of long-term archiving, in particular of scientific and scholarly publications. After a brief overview of physical data survival, emphasis is laid on general, sound principles for document archiving, and on how such principles might be considered in the choice of archive file formats. From the Golden Rule of Archiving, “do not unnecessarily restrict future options”, three general principles for publication archiving are derived: a)

use simple, text-based formats; b) use open standards; c) archive a representation, not a presentation. With these as point of departure, some common current document formats for archiving are commented, in particular pdf, pdf-a, tex/latex, xml.

Part One: Introduction, Archive sustainability, Physical data survival

Introduction

An increasingly large part of mankind's records is being made available through electronic archives. For the important special case of scholarly and scientific communication, physicists and computer scientists, in particular, have taken leading roles in this process. For many disciplines, most publications can be accessed online; additionally, to a growing extent, this is true also for the many different types of primary data underlying the research results. On the whole, the neural system of scientific and scholarly communication is already electronic.

There certainly remains much to be done to facilitate resource discovery and access. Nevertheless, many researchers have already found that the ease with which they can get at relevant publications within their field has revolutionized the way the do research, and it has done so in a matter of a few years. The results of many millions of research hours are accessible with just a few mouse clicks, or can at least potentially be made so (1).

We have hitherto unparalleled possibilities to climb the shoulders of giants, in all shapes and sizes. Well within the horizon, we might envision a searchable and browsable domain of automatically interlinked publications (for instance, through citation or field similarity). The domain also accommodates the underlying research data, which has been produced and annotated through a world-wide collaborative effort. This domain is an abstraction, for sure; physically, the resources may exist in several copies on many different servers. However, a researcher needn't know, or care: after having identified herself to the system (with a unique, world-wide valid user ID), she may happily romp around much like she

would in her own, local file system.

Such a level of accessibility would itself be enough to baffle researchers of past generations, were they to pay us a visit. Still, we have so far only sketched this virtual domain as a habitat for human researchers. More stirring to the imagination, it will increasingly also be populated by agents – autonomous computer programs which for instance may exploit natural language processing techniques and inference engines to work on text, metadata, and markup. Agents may be employed to classify, index, and automatically ingest new documents into the domain; to summarize documents and provide semantically relevant links between them; to combine information and draw conclusions; and to do many other things we haven't even thought of yet.

Before we get there, there are many problems to tackle, spanning many areas. For instance, on a social level: how do we best encourage and help young and old researchers to learn whatever they will have to learn and to do whatever they will have to do, in order to contribute most efficiently to their respective community? Or economical: what business models for the publishing industry will benefit the research community most, and how can we promote them? Or legal: how do we handle legal matters (e.g., intellectual property issues; ethical use of research data) in a fast changing world-wide distributed environment, where legislation differs between countries and, furthermore, constantly lags behind the technical development?

Certainly, there are technological bites, as well, to chew for quite a while yet (for instance citability, presupposing among other things robust version handling with unique document and version IDs as well as a reliable addressing system; and better searchability, perhaps through more efficient metadata and more mature ontologies). Still, leaving now the more futuristic applications aside, it seems fair to say that the main obstacles for providing, maintaining, and enlarging the basic infrastructure for scientific communication are no longer predominantly technological.

Most technical aspects of this budding research infrastructure are of concern mainly for IT and computer professionals, and there is no need to change that. Nevertheless, there is at least one area where some general knowledge could be useful for most researchers (the 'content providers', as the Newspeak goes), in order to grasp the risks and tradeoffs connected to a certain choice: archive sustainability.

The present article tries to give a bit of non-technical background knowledge of the long-term perspective on document archiving, particularly focusing on the question of archiving file formats for text-based documents, such as typical scientific and scholarly

publications. (The somewhat sloppy term “documents” will here refer to such publications, with no attempt made to find a more precise definition.) However, it doesn’t hurt to have a notion about general sustainability issues either, so we will start one floor up.

Archive sustainability

We can read and with varying degrees of success also interpret ancient media, such as clay tablets, parchment scrolls, and runic stones, thousands of years old. By contrast, mankind hasn’t shown much concern for preserving its more recent, digital heritage for posterity. It is salutary, for instance, to consider that data from the first moon landings are irrevocably lost – even if we could dig up machines that could read the since long obsolete tape format, we would not know how to interpret the undocumented, unstructured bitstream we would find.

We can only strive to do better today. Of course, not all long-term aspects of digital archiving are under the control of the archivist (most archives are funded, set up, and maintained through political decisions, and political situations are known, sub specie aeternitatis, to be transitory). However, where we do have a choice, we should consider the long-term consequences duly.

Admittedly, it is hard to make any detailed predictions about future archiving technologies, in view of the mind-boggling pace of development. Here and now, we cannot think of all problems we might meet in the future, much less solve them. Likewise, we know little of tomorrow’s research methodologies, or their particular requirements. Google will perish, and Citeseer will cease; but the archiving standards we set up should serve their successors, and ours, as well.

In the absence of a crystal ball, we will have to rely on general, sound principles for archiving sustainability. They may provide guidance through uncertainties; they may help us to identify current weak points, and to recognize a better solution when we see it.

A principal challenge of an archive is to cater for the physical survival, interpretability, and usability of the data it holds. Of course, this task is as old as archiving itself; but the dangers and expectations are different in the digital era.

Physical data survival

Storage media deteriorate, at an appalling rate which we have just begun to realize. Indeed, when comparing expected life times of

current data carrier such as CDs (perhaps 20 years), DVDs (possibly significantly less, given the higher information density), or hard discs (typically 5 years or less) to the practically imperishable clay tablets of Sumer, we have little reason to boast about technical progress.

Unfortunately, most research efforts on new media formats goes into increasing access speed and storage space, not longevity. As a result, we get ever faster and more capacious physical formats, with ever shorter life cycles. From a marketing point of view, this is certainly nothing unexpected --- in fact, regularly introducing new formats is a central strategy for customer recycling. Not even ten years ago, small, portable DAT players were state-of-the-art equipment for audio recordings in the field; nowadays, it's practically impossible to find spare parts even if you would dare open one to try to repair it.

Planning for physical survival of digital data under such conditions is not a pleasant task. The only sustainable and scalable solution is to arrange for continuous, automated mass migration: to build an archiving system that is capable of automatically identifying and replacing individual data carriers at risk, and automatically moving the entire archive to a new physical format when the day comes.

This sounds expensive, and it is; but it is also an area where centralization is a very efficient measure – the cost per archived terabyte falls drastically with the size of the archive. National computing centres (or, even better, international ones, thus giving some security against political uncertainties) may offer archiving facilities on. The Australian Partnership for Advanced Computing (2) is an early and interesting step in this direction, with several successors world-wide.

Notes and references

(1) The information compression rate in such a collection is stunning. The world's collected publications in mathematics, for instance, have been calculated to fit on around 100 GB, very soon to be standard on any laptop. Of course, it might be that local copies of that kind will be little needed in an even more networked future, if fast, reliable, and omnipresent connection points will offer access to constantly updated archives.
(2) APAC: **http://www.apac.edu.au/**

Part Two:
Archiving principles and archiving formats

Interpretability and usability -- Archiving principles and formats

How do we choose a good encoding for archiving, i.e., what file format should we use? Obviously, as an absolute minimum, we will want one that we can be reasonably sure we can interpret in the future, even in a future when the software tools used to produce it may be long extinct. There is nothing particularly pessimistic about such a scenario. Tools are dependent on the technology of the day; their development time and their lifetime (a few years, 10-20 at most) are both a twinkling of an eye on archival timescales. The data collections the tools produce or process, on the other hand, have generally required much more resources. Consequently, we will want them to be useful for a much longer time – not seldom indefinitely. This requirement already excludes any format that is not well-documented, open standard, and vendor-independent.

Furthermore, our chosen format should be useful, in the sense that it should support whatever operations we might want to subject it to in the future. This is rather speculative – how could we possibly predict what future generations might want to do with the data, or through what devices they will want to have it presented? We can't, of course; but when choosing digital formats, we can try to observe the very general Golden Rule of Archiving (3): *Do not unnecessarily restrict future options*.

Admittedly, such a wording is too abstract to be of much use in a particular case. However, several other, more specific archival principles can be derived from it. For the special case of archiving text-based scientific and scholarly communication, the following ones are suggested; they are meant to be thought-provoking, rather than exhaustive.

Keep it simple

The World Wide Web has thrived much due to its use of simple, text-oriented network protocols. One computer sends a plain-text request to another one and receives an answer in plain-text. Any decent programmer on any platform can quickly understand the specification well enough to exploit this basic framework, in simple or complex applications. By contrast, employing sophisticated formats means encapsulating data in a shell. If we do, we will need more complex tools to process it; tools that no longer can be written by anyone, tools that will need more maintenance in order to work in changed hardware and software environments; tools that are more likely to contain bugs.

To archive text, nothing is better than text. If we need tagged text, we should use relevant markup, and define new if necessary. Simplicity does not mean lack of expressivity; at most, it might mean a bit of verbosity.

In passing, we might note that text files are easily manipulated,

and that issues around document integrity must be taken seriously. We should not, however, plan for security through obscurity.

Use Open Standards

People who come from the proprietary world and try open source software are often surprised by the fact that programs actually may exchange data through shared file formats – thus, programs may complement each other, rather than compete. (One might find it saddening that there should be something remarkable about this, but such is reality.) You may prefer one tool and I may prefer another, perhaps due to differences in natural disposition or in task at hand; but if we decide on an open file format, our programs may exchange data anyway. The actual programs we use are of no interest and they need not know anything about each other: as long as they both fulfil their part of the bargain, they will exhibit interoperability (as the technical term for this most treasured property goes).

The question of interoperability in an archiving scenario is in principle not very different, only more pressing. A program may need to read a file many years after the file was originally created: by then, there might be no trace left of the creating program, the operating system this program was built for, the hardware it used to run on, or their respective authors (indeed, perhaps not even of the country they used to live in).

Interoperability can only be achieved by strict adherence to a public, non-proprietary, well-documented open standard. Such standards should be designed by truly independent bodies, such as The World Wide Web Consortium (4) – any “de facto standard” may sooner or later turn into a marketing weapon.

Incompatible, secret file formats remain an efficient strategy to forcing clients to do all worshipping at a single altar. However, clients provide the money, and therefore they have a strong bargaining position. There is a growing and gratifying tendency to require from software that it be able to save files into open standard formats -- we can hope that it will be unmarketable otherwise.

The main trap to look out for in this process is that the standards simultaneously are somewhat “improved” (i.e., extended with some arbitrary and redundant features, just enough to make them awkward to use in competing programs). There is a name to this strategy: “embrace, extend, extinguish”. It would not be a serious threat on a balanced market, but given the current situation, where a very small number of vendors are responsible for billions of installations, the danger is real.

Archive a representation, not a presentation

Human inertia is a strong moderator of change rate. Not too long ago, the ‘paperless office’ or even the ‘paperless society’ were envisioned by some writers. As prophecies, both have so far failed miserably; in tech journalist Dick Pountain’s words, IT has rendered paper superfluous in much the same way that the car has made legs unnecessary.

A bit of inertia might be salutary at times. In our child age of digital archiving, our not-so-impressive records of lost data would probably have been even worse, had we been more eager to replace paper by discs. We won’t argue here about the pros and cons of paper. Disregarding environmental aspects, paper is fine, and some future technology might be fine, too (perhaps large, cheap, soft, thin, foldable screens). The point to be made, however, is that there is an unfortunate human tendency to equate a work with its physical presentation; for textual works, this usually means ‘as it is printed on paper’. In the terms of library science, we tend to confuse a work itself with a specific manifestation of a specific expression of that work. For instance, most citation techniques are built around page numbers of a particular edition of a written work, rather than internal references of the text itself.

This view of a work as tied to a particular version with a particular layout printed on a particular page size is problematic, for several reasons. First, it is difficult to sustain in a world where text documents can be presented to the user through a number of different (most of which are yet to be invented), and neither text nor devices are necessarily page-oriented.

Second, and more importantly, such a view is an obstacle whenever we want to use technology for something more than just facilitating paper reading – when we want to go beyond just mimicking current practice. Several of the more visionary applications we have hinted at (and countless others we have not) will be carried out by computer programs, written by language technologists, knowledge engineers, artificial intelligence researchers, and others. Computer programs do not benefit from having to deal with presentation formats – they are not, primarily, paper readers.

A chief hallmark of a good archive document format is that it holds a *representation* rather than a *presentation* (5). Logical, structural, or semantic markup form part of the representation. A specific layout, by contrast, does not; instead, it is generated for a particular presentation in a particular set of circumstances. For instance, in a representation, references are internal to the text, expressed in some dialect of computerese (6). In a particular representation which also happens to be page-oriented, they might instead be converted to page numbers. Taking a slightly more

imaginative example, we can think of texts on history, in which all named entities (persons, places, organizations, etc) have been tagged as such in the representation (manually, automatically, or a combination). Humans usually need no help in identifying named entities, and so the markup need not be seen in a presentation format meant for reading. To information-processing agents, however, such a tagging is of great help.

In fact, it is fruitful to think of a specific presentation format as just another export option from a representation, where all choices can be made according to the needs of that moment: technology available, task at hand, user preferences, presentation device, restrictions of bandwidth, storage space, etc. For a long time to come, one such export option will undoubtedly be printing on paper, but there is no reason to believe that it will be the only one for all future and, above all, there is no reason to choose this particular form for archiving.

Admittedly, it might sometimes be difficult to identify the borderline between content and presentation. For some disciplines, such as legal science, the ability to faithfully reproduce a certain layout may be crucial (see more about pdf-a below). For general publications, however, exact copies are usually not needed (and when they are, a structured representation could be linked to a page image).

Notes and references

(3) This will ring familiar to archivists in general, but perhaps in particular to people working with digitization of cultural heritage. See for instance the Ninch guide,
http://www.nyu.edu/its/humanities/ninchguide/
(4) **http://www.w3c.org/**
(5) These terms can be seen as generalizations of the well-known distinction of “content” versus “style”.
(6) The current tool to do so would be the XML substandard XPath. (It is interesting to consider one of the most widespread and quoted books in existence: the Bible. It has been translated, reprinted, and orally transmitted to the point that nobody would confuse the work with a particular representation. A typical reference may read “Cor. 1:13” – not very different from an XPath expression.)

Part Three:
Current archive format practice, Conclusion

Current archive format practice

The most common formats for text archiving today are native

formats (mostly MS Word), pdf, pdf-a, tex/latex, and different xml applications. These are briefly presented below. Other current formats include sgml (7) (still around for legacy reasons, but unnecessarily complex, little supported and superseded by xml for all practical purposes), html (far too restricted for general use; can easily and better be expressed in xml if needed, as xhtml (8), plain text (rather restricted, but reliable; when nothing more sophisticated is needed, archiving plain text is the closest we can get to digital clay tablets).

Native formats

Software companies are profit-driven (and they can hardly be criticized for being so). Their main responsibility lies with their share-holders; if they choose to publish specifications, or lock them in a cellar, or to continue or discontinue development or support, or to double or halve or setting to zero the price of their products, they do so on approval of their market analysts – to do profit in the short-term perspective (short-term at least from an archivist's point of view).

Native application formats (e.g., MS Word, WordPerfect) are fine for something you know you will never share with anyone, including yourself a few years from now. For any wider or longer perspectives, they are very unsuited. It does not take much fantasy to grasp the risk implicit in locking important data encoded in some binary, closed, proprietary format. There is no guarantee that the data can be recovered at all; if it can, it may cost practically anything. Just to mention a few scenarios: the company behind your program may be put out of business, or discontinue support for your platform or version, or abandon backwards compatibility, or charge ten times more than you expected for the next upgrade, or refuse to fix a bug which happens to be crucial for you.

From an archival point of view, it is important not to be short-sighted (and among other chief virtues of an electronic archivist, we might in particular note parsimony, distrust, and paranoia).

Pdf

The most common choice of archiving format is the portable document format (pdf), created by Adobe. In contrast to most native applications, the pdf specifications up to and including the current has been made public (9), allowing third-party software including some open source projects to create and read pdf. Since the format works well on most current platforms, looks nice on screen, and in particular gives good quality printing for our paper-oriented minds, one might be tempted to think that the question of archiving format is solved.

This is exactly wrong. Adobe controls the pdf format and it may be

changed at any time, with no specification made public. Most pdfs are created by software from Adobe; if the company see fit, it could for instance introduce a new pdf version, “improved” but unfortunately closed and only readable with Adobe software.

More general drawbacks from an archiving point of view is that pdf allows encryption and scripting, both of which should be banned from archiving formats. It also permits embedding of audio and video. (While we certainly should be able to link media files to publications, embedding is not the mechanism.)

Even more generally, the relatively good support for metadata does not change the fact that pdf is strongly presentation-oriented and so less useful for automatic processing. It prints nicely and thereby lets humans go on the way they use to, but it mixes up content and layout and it does not necessarily hold any representation of logical structure.

Pdf-a

Pdf has been criticized for archiving purposes, and rightly so. As a reaction to that, representatives from Adobe and several communities and (US) governmental bodies, especially representing the legal sector, have presented a new, slimmed-down version of the format, known as pdf-a. The format was proposed to ISO as a text archiving standard and recently also accepted.

Pdf-a (a for ‘archiving’) is basically a subset of the pdf 1.4 specification; in particular, it prescribes that all fonts of a document must be included in a pdf-a and that no commercial fonts be used; furthermore, that no encryption, scripts, or embedded media be used. Pdf-a is not owned by Adobe, and several of the objections to pdf are thus bypassed.

Still, pdf-a is just as presentation- (and paper-) oriented as pdf; in fact, it is proposed as a “preferred format for page-oriented textual (or primarily textual) documents when layout and visual characteristics are more significant than logical structure.”(10) When layout is crucial, for instance in court (11), it may form a good complement to other, more structured formats. However, in scientific and scholarly communication, this shouldn’t be too often, once we learn not to confuse work and representation.

TeX

Research communities oriented towards science, mathematics, or computation mostly use the TeX system for communication (12). It is free, stable, extensible, reliable, and does a remarkably good job of typesetting demanding texts, such as mathematical formulae or multilingual works.

A TeX file is made up of pure text, or rather source code, which is compiled into a specific, typeset presentation format. Thus it is enough to archive the source code.

The drawbacks are again its focus on presentation (and exclusively visual presentation, at that). TeX is meant for typesetting, to produce documents to be read from paper or screen by human. Nevertheless, being compiled, TeX sources are strongly syntactically structured; this makes automatic processing much more feasible.

Another drawback is that, even though TeX can excellently perform most of the tasks which are today done by word processors, it is somewhat demanding to use without a bit of technical knowledge. Many researchers have never written a computer program; if so, the thought of directly manipulating source code may be paralysing. Nice graphical user interfaces, such as LyX (13), can possibly reduce the need to do so.

Xml

Xml is a metalanguage, This means that it is a language to design languages, in this case, markup languages. It is very well apt for representation, transmission, and storage of textual information: it is text-based, readable by humans as well as by computers, self-documenting, portable, expressive, international (all xml is in Unicode). In front of all, it is a free, open standard, defined by the W3C Consortium (14).

Among the drawbacks of xml is that it is rather wordy, that it forces data into a hierarchical structure, and that it handles binary data only with difficulty. None of these are very crucial to scientific communication; while xml might not be what we will have for all future, it is about as far as we can get in not restricting future options today.

The particular languages defined in xml are called xml applications, formally specified through particular computer languages known as schemas. An xml document is said to be validated against its schema, thus ensuring interoperability. Furthermore, to the benefit of agents, .xml markup can be combined with a computer-readable semantic specification of the elements, known as rdf.

There are already many xml applications for very diverse purposes. We might for instance note OpenDocument (15), which is a brand-new open file format for general-purpose office uses – but in contrast to proprietary ones, it is an office format we can expect to be able to read in the future. However, to fully exploit the possibilities, communities themselves need to define new xml applications, according to their specific needs – be it for communicating research in generative syntactics or in exospheric

chemistry.

Nobody wants to write xml directly, but given a schema, a generic xml editor or one specifically written for a certain application could be used. However, much remains to be done in terms of user-friendliness – editing xml is currently no more pleasant than producing TeX.

Conclusion. Now what?

Survival of digital data will be a growing concern in all corners of society, and any single measure will be hopelessly insufficient. Still, a well-known prescription is to offer tools and education, and the world of digital scientific communication is no exception.

Although far from optimal, we will have to live with pdf as archiving format for some time to come (pdf-a perhaps for quite some time). Likewise, the far better TeX format will live long, together with xml (automated conversion between the two is nothing impossible.)

From a text-archival point of view, however, it seems clear that xml currently is the best choice for long-term purposes. However, to make xml useful, we need good tools to help in writing – general xml tools, community-specific xml tools, word processors, any tool is fine, as long as it might be persuaded to produce valid xml according to the community's schema. But first we need standards – xml and rdf schemas specified by the respective communities and suited to their particular wishes.

Education is another important point; on a general level, all researchers should know about the possibilities of digital communication through space and time; but also about the risks involved. More specifically, although most researchers are not too interested in technical details of file formats (and they shouldn't need to be), they do care about the survival and usability of their work. A tiny bit of digital long-term hygiene would not be out of the way in any curriculum (and strongly recommended also for senior researchers).

Notes and references

(7) see for instance **http://www.w3.org/MarkUp/SGML/** or **http://xml.coverpages.org/sgml.html**
(8) **http://www.w3.org/TR/xhtml1/**
(9) **http://partners.adobe.com/public/developer/pdf/index_reference.html**
(10) **http://www.digitalpreservation.gov/formats/fdd/fdd000125.shtml**
(11) see for instance **http://www.scientific-computing.com/**

scwmayjun05archive.html
(12) A good introduction is found on http://www.ctan.org/what_is_tex.html
(13) http://www.lyx.org/
(14) http://www.w3.org/XML/
(15) see for instance http://www.oasis-open.org/news/oasis_news_05_23_05.php