Preface

to

J. Rothenberg: 'Avoiding technological quicksand; Finding a viable technical foundation for digital preservation'.
Published by ECPA (European Commission for Preservation and Access), Amsterdam 1999.

by Edo H. Dooijes

Promises...

The preservation of digital information is a central issue in the management of long-term acces to materials in libraries and archives. In 1996, the report of the Task Force on the Archiving of Digital Information outlined the different aspects of the problem: technical, legal, organizational. The Council on Library and Information Resources is now working on a follow-up of this report, in the form of a series of papers proposing different approaches to the preservation of digital information. This report by Jeff Rothenberg is the first in the series. While other topics - in particular the legal issues - of digital preservation may have a different flavour in different countries, there is no doubt that the subject of this report is as relevant in the European situation as it is in the United States.

To guarantee long-term access to digital resources, various solutions have been proposed over the last years, such as relying on printed copies, the development and promotion of standards, using the expertise and the equipment of computer museums, and periodically migrating data to new formats. Rothenberg shows that all of these solutions are flawed in one way or another and will in the end prove to be unsatisfactory. His proposal is instead to preserve a digital document in its original *logical* format i.e. the collection of bits representing the text, images and possibly other types of data in the document. This logical format is to be distinguished from the document’s physical appearance which may be any of the multitude of media which have been and will be around: magnetic tapes, decks of punched cards, diskettes, DVD’s and so on.

In this model, the original digital document is encapsulated together with the computer program originally used to create and handle it (the parent program), and a detailed description of the machine M on which the parent program was designed to run. The description of M will enable a future user of the document to build an M-emulator, executable on any computer system of his or her choice. Executing the M-emulator will create a virtual machine, M, which ideally has the 'touch and feel' of the original M. On M, the parent program can then be made to run, and in this way it can handle the document in almost the same (or even exactly the same) fashion as in the original environment. Once M has been created, all software written for M can be run on it, and consequently all documents created with this software can be read with it.

Rothenberg offers strong arguments for this approach, which promises to lead to interesting solutions when it is further developed and explored. At the present time, emulation techniques are already routinely used for running applications designed for Macintosh computers using an 68000-series processor on Macintoshes equipped with a PowerPC processor. Also, emulators for many of the big machines of the early years of computing are available, often running on a standard PC. The experience gained in this area may help to realize the kind of model we need to keep digital information accessible not only for just a few decades, but for centuries.

...and problems

Rothenberg's thought-provoking paper clearly outlines the direction into which further work on preservation through emulation should go; some of the comments which suggest themselves are the following.

1. In order to be preserved, a digital document must be read at least once in its present physical form. Likewise, the original parent program must be run on the original machine, supposed that both are still available. However, more often than not, it is only after years that a document is judged to be interesting enough to be preserved. Hence, 'computer museums' lagging say ten years

behind the actual developments will be indispensible, as can indeed be observed already today.
2. It is not obvious how one should define the stream of bits (as refered to by Rothenberg) from the digital document to its parent program. The bit pattern 'seen' by a disk drive's read head is quite different from what appears at the interface of the drive controller and the computer's internal data bus. Should we store the punched hole patterns on Hollerith cards as they are, or according to the symbols they represent? This problem is different for each kind of medium.
3. For retrieving a preserved data set from its encapsulation, a 'bootstrap' is necessary in order to explain to the future computer how to read the data from the medium, and how to interpret the encapsulated formal description of the emulator, in order to build an emulator running on the host machine. (The former piece of information cannot be a physical part of the encapsulation). Rothenberg proposes to provide a human-readable bootstrap. For the description of a bootstrap – by necessity a rather complex thing - a user's manual-style text will not be satisfactory. Instead, some formal language - a computer language – should be used. To make this work, there should be a world-wide agreement to 'freeze' the definition of some general-purpose, simple computer language, and to require of future computer vendors that they offer a compiler for this language with their machines. A standard – in general rightly dismissed by Rothenberg – is indispensible here. Notice in passing that the Fortran language has in a way played this role for almost half a century already!
These and other issues identified in Rothenberg's report (notably the necessity to identify a physical medium suitable for the purpose) make the preservation of digital data an intriguing and challenging subject for further study, in academic as well as in industrial context.

*Dr E.H. Dooijes is an associate professor at the Department of Computer Science, University of Amsterdam, and curator of the University's Computer Museum.