

The Changing Face of the History of Computing: The Role of Emulation in Protecting Our Digital Heritage

David Anderson, Janet Delve, and Vaughan Powell

Future Proof Computing Group, School of Creative Technologies,
University of Portsmouth, UK
cdpa@btinternet.com,
{Janet.Delve,vaughan.powell}@port.ac.uk

Abstract. It is becoming increasingly common for some source material to arrive on our desks after having been transferred to digital format, but little of the material on which we work was actually born digital. Anyone whose work is being done today is likely to leave behind very little that is not digital. Being digital changes everything. This article discusses the issues involved in the protection of digital objects.

Keywords: digital preservation.

Erasmus summarised well the outlook of many historians of computing when he remarked “*When I get a little money, I buy books; and if any is left I buy food and clothes*”! Most of us love second-hand bookshops, libraries and archives, their smell, their restful atmosphere, the ever-present promise of discovery, and the deep intoxication produced by having the accumulated knowledge of the world literally at our fingertips. Our research obliges us to spend many happy hours uncovering and studying old letters, notebooks, and other paper-based records. It is becoming increasingly common for some source material to arrive on our desks after having been transferred to digital format, but little of the material on which we work was actually born digital. Future historians of computing will have a very different experience. Doubtless they, like us, will continue to privilege primary sources over secondary, and perhaps written sources will still be preferred to other forms of historical record, but for the first time since the emergence of writing systems some 4,000 years ago, scholars will be increasingly unable to access directly historical material. During the late 20th and early 21st century, letter writing has given way to email, SMS messages and tweets, diaries have been superseded by blogs (private and public) and where paper once prevailed, digital forms are making inroads, and the trend is set to continue. Personal archiving is increasingly outsourced – typically taking the form of placing material on some web-based location in the erroneous belief that merely being online assures preservation. Someone whose work is being done today, is likely to leave behind very little that is not digital, and being digital changes everything.

Digital ‘objects’ are, in many ways, a curious product of the 20th Century and have attributes that make them quite unlike any of the objects that preceded them. Their immaterial nature gives rise to diverse concerns for those tasked with preservation

activity, requiring substantial changes to be made to preservation practice as well as demanding a significant alteration in the way we think about the nature of objecthood in the digital context.

“We depend on documents to carry messages through space and time. In many cases, this reliability is achieved through fixity: letterforms inked on paper can survive for long periods of time. But with newer media, such as video, this reliability is achieved not by fixity but by repeatability. The moving images on a video screen are by their very nature transient. I will never be able to see those very images again. But I can play the tape repeatedly, each time seeing a performance that, for all practical purposes, is ‘the same as’ the one I saw the first time.”
(Levy 2000)

Digital objects, unlike their physical counterparts, are not capable directly of human creation or subsequent access but require one or more intermediate layers of facilitating technology. In part, this technology comprises further digital objects; software such as a BIOS, an operating system or a word processing package, and in part it is mechanical; a computer. Even a relatively simple digital object such as a text file (ASCII format) has a series of complex relationships with other digital and physical objects from which it is difficult to isolate it completely. This complexity and necessity for technological mediation exists not only at the time when a digital object is created but is present on each occasion when the digital object is edited, viewed, preserved or interacted with in any way. Furthermore the situation is far from static as each interaction with a digital object may bring it into contact with new digital objects (a different editor for example) or new physical technology.

The preservation of, and subsequent access to, digital material involves a great deal more than the safe storage of bits. The need for accompanying metadata, without which the bits make no sense, is understood well in principle and the tools we have developed are reasonably reliable in the short term, at least for simple digital objects, but have not kept pace with the increasingly complex nature of interactive and distributed artefacts. The full impact of the lacunae will not be completely apparent until the hardware platforms on which digital material was originally produced and rendered, become obsolete, leaving no direct way back to the content.

Within the digital preservation community the main approaches usually espoused are migration, and emulation. The very great extent to which opinions appear to be polarised between these two approaches (Bearman 1999; Rothenberg 1999) – neither of which can claim to be a complete solution – is perhaps indicative of the extent to which the fundamental issues of long term digital preservation have not yet been addressed (Stawowczyk Long 2009). The arguments and ‘evidence’ offered on both sides of the debate are frequently far from convincing.

The focus of migration is the digital object itself, and the process of migration involves changing the format of old files so that they can be accessed on new hardware (or software) platforms. Thus, armed with a suitable file-conversion program it is relatively trivial (or so the argument goes) to read a WordPerfect document originally produced on a Data General mini-computer some thirty years ago, on a brand new iPad3. The story is, however, a little more complicated in practice. There is something in excess of 6,000 known computer file formats, with more being produced all the time, so the introduction of each new hardware platform

creates a potential need to develop afresh thousands of individual file-format converters in order to get access to old digital material. Many of these will not be produced for lack of interest among those with the technical knowledge to develop them, and not all of the tools which are created will work perfectly. It is fiendishly difficult to render with complete fidelity every aspect of a digital object on a new hardware platform. Common errors include variations in colour mapping, fonts, and precise pagination. Over a relatively short time, in a digital version of 'whisper down the lane' errors accumulate and erode significantly our ability to access old digital material or to form reliable historical judgements based on the material we can access. The cost of storing multiple versions of files (at least in a corporate environment) means that we cannot always rely on being able to retrieve a copy of the original bits.

The challenges represented by converting a WordPerfect document are as nothing compared to those of format-shifting a digital object as complex as a modern computer game, or the special effects files produced for a Hollywood blockbuster. This fundamental task is well beyond the technical capability or financial wherewithal of any library or archive. While it is by no means apparent from much of the literature in the field, it is nevertheless true that in an ever-increasing number of cases, migration is no longer a viable preservation approach.

Emulation substantially disregards the digital object, and concentrates its attention on the environment. The idea here is to produce a program which when run on one environment, mimics another. There are distinct advantages to this approach: it avoids altogether the problems of file format inflation, and complexity. Thus, if we have at our disposal, for example, a perfectly functioning IBM 360 emulator, all of the files which ran on the original hardware should run without modification on the emulator. Emulate the PS3, and all of the complex games which run on it should be available without modification – the bits need only be preserved intact, and that is something which we know perfectly well how to accomplish. Unfortunately, producing perfect, or nearly perfect emulators, even for relatively unsophisticated hardware platforms is not trivial. Doing so, involves not only implementing the documented characteristics of a platform but also its undocumented features. This requires a level of knowledge well beyond the average and, ideally, ongoing access at least one instance of a working original against which performance can be measured. Over and above all of this, it is critically important to document for each digital object being preserved for future access, the complete set of hardware and software dependencies it has and which must be present (or emulated) in order for it to run¹. Even if all of this can be accomplished, the fact remains that emulators are themselves software objects written to run on particular hardware platforms, and when those platforms are no longer available they must either be migrated or written anew. The EC funded KEEP project² has recently investigated the possibility of squaring that particular circle by developing a highly portable virtual machine onto which emulators can be placed and which aims to permit rapid emulator migration when required. It is too soon to say how effective this approach will prove, but KEEP is a project which runs against the general trend of funded research in preservation in concentrating on emulation as a preservation approach and complex digital objects as its domain.

¹ See TOTEM <http://www.keep-totem.co.uk/>

² <http://www.keep-project.eu>

Even in a best-case-scenario, future historians, whether of computing or anything else, working on the period in which we now live will require a set of technical skills and tools quite unlike anything they have hitherto possessed. The vast majority of source material available to them will no longer be in a technologically independent form but will be digital. Even if they are fortunate enough to have a substantial number of apparently well-preserved files, it is entirely possible that the material will have suffered significant damage to its intellectual coherence and meaning as the result of having been migrated from one hardware platform to another. Worse still, digital objects might very be left completely inaccessible in virtue of either not having a suitable available hardware platform on which to render them, or rich enough accompanying metadata to make it possible negotiate the complex hardware and software dependencies required.

It is commonplace to observe ruefully on the quantity of digital information currently being produced. Unless we begin seriously to address the issue of future accessibility of stored digital objects, and take the appropriate steps to safeguard meaningfully our digital heritage, future generations may have a much more significant cause for complaint. In addressing this challenge, it appears not unreasonable to suppose that Computer Museums might have a productive role to play:

“To avoid the dual problems of corruption via translation and abandonment at paradigm shifts, some have suggested that computer museums be established, where old machines would run original software to access obsolete documents (Swade 1998). While this approach exudes a certain technological bravado, it is flawed in a number of fundamental ways.” (Rothenberg 1999)

“... preserving obsolete hardware and software (a solution which supposes that complete museums of obsolete equipment could be maintained so that any configuration of once used hardware and software could be replicated. Rothenberg reiterates pragmatic arguments against this, many of which I published in a 1987 report. They were not novel then. Fortunately, as far as I know, no serious investment in this ‘solution’ has ever been attempted.)” (Bearman 1999)

So it seems that one of the things about which Bearman agrees with Rothenberg, is that reliance on Computer Museums is not the answer to the problem of digital preservation. It is probably worth mentioning here that approaching digital preservation as a problem that has a single ‘correct’ answer, if only we could discover it, is a caricature of what is, in reality, a complex and multi-faceted series of challenges. Generally, it should be observed, Rothenberg and Bearman write in somewhat polemical terms, and in very large measure this detracts from their cases.

Rothenberg’s concerns about computer (hardware) museums essentially come down to a single point: hardware (including media) will deteriorate over time to the stage where old machines will not be able to access the software written for them (Rothenberg 1999). The long-term inevitability of physical deterioration of systems will not come as a surprise to anyone familiar with the 2nd Law of Thermodynamics³.

³ The 2nd Law of Thermodynamics asserts the universal principle that absolutely everything decays.

On the positive side, Rothenberg concedes two 'limited roles' for computer (hardware) museums: performing "...heroic efforts to retrieve digital information from old storage media." and verifying the behaviour of emulators. Rothenberg's characterisation of these roles as 'limited' notwithstanding, these are, in fact, absolutely essential activities; the first is arguably the only way in which future generations can gain access to some important historical material which would otherwise be lost forever, and the second is vital if we are to verify that the behaviour of emulated computer platforms is faithful to the original. It is particularly important to be able to determine if, for example, some unexpected behaviour exhibited by a digital object running under emulation is the result of a lack of fidelity in the emulator or would have been present on the original platform.

Bearman, writing about computer (software) museums, makes a closely related point.

"... by documenting standards and widespread operating functions, software archives preserve a record of the fundamental structures of the software environment which will contribute to future understanding of more specialized software." (Bearman 1987)

Similar considerations apply equally well to verifying migrated digital objects and computer museums are of equal value independently of the digital preservation approach preferred.

All computers have undocumented features, and preserving original hardware in working condition for as long as reasonably possible is an important aspect of digital preservation. Of course it is not possible to preserve computer systems forever but this only lends urgency to the need to gather as much information as possible from machines while they are available to us. The cost of funding computer museums, particularly when viewed from the national or international perspective, is not high. Rothenberg's assertion that:

"It is unlikely that old machines could be kept running indefinitely at any reasonable cost, and even if they were, this would limit true access to the original forms of old digital documents to a very few sites in the world, thereby again sacrificing many of these documents' core digital attributes." (Rothenberg 1999)

somewhat misstates the position. It is certainly true that preserving old machines in working order is subject to the law of diminishing returns. That is one of the reasons why we should endeavour to make the best use of old machines while it is still feasible for us to do so.

Rothenberg appears to be insensitive to the fact that computer museums would have a continuing role into the future, broadening their collections as more and more machines become obsolete. The custodial remit of computer museums would run further than simply keeping in working order the oldest machines in their care, and it is to be expected that for each machine in a museum's collection, a decision would have to be made, at some point in the future, to preserve the device in a non-working condition. This does nothing to diminish the importance of computer museums, which could and should become repositories of collective knowledge in just the same way as any other memory institution. The 2nd Law of Thermodynamics is unavoidable and if Rothenberg's arguments were sound, they would apply with equal force to every other sort of library or museum; all statues will eventually crumble and every book

ever written will, given sufficient time, turn to dust. Should we therefore conclude that it is folly to suggest a role for memory organisations in preserving our cultural, technological and scientific heritage?

It is fair to say that Rothenberg does not think highly of migration as an approach to digital preservation:

“While it may be better than nothing (better than having no strategy at all or denying that there is a problem), it has little to recommend it. however, to the extent that it provides merely the illusion of a solution, it may in some cases actually be worse than nothing. In the long run, migration promises to be expensive, unscalable, error-prone, at most partially successful, and ultimately infeasible.” (Rothenberg 1999)

Unfortunately many of the arguments that Rothenberg deploys in reaching these damning conclusions seem to apply with similar force when directed at emulation. Rothenberg opines:

“... migration is labor-intensive, time-consuming, expensive, error-prone, and fraught with the danger of losing or corrupting information. Migration requires a unique new solution for each new format or paradigm and each type of document that is to be converted into that new form. Since every paradigm shift entails a new set of problems, there is not necessarily much to be learned from previous migration efforts, making each migration cycle just as difficult, expensive, and problematic as the last. Automatic conversion is rarely possible, and whether conversion is performed automatically, semiautomatically, or by hand, it is very likely to result in at least some loss or corruption, as documents are forced to fit into new forms.” (Rothenberg 1999)

However emulation also requires considerable expenditure of time and effort in order to arrive at a successful outcome. It is true that a great deal of excellent work has been undertaken in the emulation community⁴, which has provided benefits for the digital preservation community without giving rise thereby to any outlay of resources by museums, libraries or archives. But this software windfall should not be allowed to engender complacency. Robust emulation software remains ‘labor-intensive, time-consuming, expensive’ to develop. The emulators currently available to us do not provide either complete coverage of all the required hardware platforms, nor completely reliable and faithful reproduction of the machines that have been emulated. They are, in Rothenberg’s way of speaking, ‘error-prone, and fraught with the danger of losing or corrupting information’ if we were to rely on them for our digital preservation needs. We can have no complaint however, as the available emulators are often the result of the unpaid efforts of enthusiasts, and reflect the interests and obsessions of their authors rather than being driven by digital preservation requirements.

The digital preservation community has made one attempt to develop an emulator. The Dioscuri project was conceived in 2004 and has been under continuous development since 2006. The main development has been undertaken by Nationaal

⁴ For example with the Multiple Arcade Machine Emulator (M.A.M.E.) [see <http://mamedev.org/>] and the Multiple Emulator Super System (M.E.S.S.) [see <http://www.mess.org/>]

Archief of the Netherlands, the Koninklijke Bibliotheek, and Tessella plc, and has benefitted from the efforts of a number of others including Jeff Rothenberg. Financial support for the building of the emulator has been provided by the European Commission and has taken place both within the Planets project and the KEEP project.

Dioscuri is attempting to emulate a well understood and documented hardware platform (PC x86), numerous copies of which are extant. Conditions for producing an emulator really do not come much better than this. So, if emulator development were simple, even relatively so, then after six years of planning and well-funded development, Dioscuri would be complete and the digital preservation community would be reaping the benefits of its use and perhaps taking forward the lessons learned and applying them to the development of other emulators. However, writing at the end of last year, Stawowczyk Long concluded in his report for the National Library of Australia and the International Internet Preservation Consortium:

“Dioscuri ... has very limited capabilities, It could only be tried with MS DOS 6.2 and MS Windows 3.1 operating systems. Dioscuri is ... rather slow. ... media files could not be rendered sufficiently well to give a useful performance.” (Stawowczyk Long 2009)

Emulators, as we have observed, are not simple to write. It is a time-consuming and highly complex activity. As things stand, neither emulation nor individual emulators have been developed to the point where emulation can seriously challenge migration as a digital preservation approach. Some important first steps towards developing emulation have been taken (e.g. in the EC funded KEEP project), but these are the first steps only, and it is as yet too soon to say where precisely they will lead. We can, however, be reasonably certain that they will not lead to the complete replacement of migration nor, for reasons that we will cover in greater detail below, is it desirable that they should do so.

Rothenberg has a concern that the evolution of formats, encodings and software paradigms defies prediction.

“As has been proven repeatedly during the short history of computer science, formats, encodings, and software paradigms change often and in surprising ways. Of the many dynamic aspects of information science, document paradigms, computing paradigms, and software paradigms are among the most volatile, and their evolution routinely eludes prediction.” (Rothenberg 1999)

It is difficult to see anything in these observations that applies, in any important sense, to migration that does not also apply with equal force to emulation per se. The substance of the point appears to be that the future has often proved unpredictable and is likely to do so again. Each new hardware paradigm is apt to cause problems for all the emulators written to run on the old paradigm and there is not the least justification for believing that, for example, a Commodore 64 emulator written to run on a Mac Powerbook of the early 21st century, will be able to run, without significant conversion, on an as yet unconceived of hardware platform. The KEEP project is directly focussed on just this aspect of emulator portability and it is hoped that progress will be made towards keeping emulator environments portable. However, there is nothing about emulation in and of itself, which makes it immune to the disruption caused by the introduction of new approaches to computing and the inevitable obsolescence of the old.

Even if we were to take seriously Rothenberg's description of the problem, it would remain an open question as to whether an emulation approach to digital preservation were the best, response; it is certainly not the only route open to us. Gladney, for example, has very plausibly suggested a shift towards the routine development of durable digital objects coupled with moving the responsibility for digital object durability away from archival employees to information producers (Gladney 2008).

Another concern to which Rothenberg gives voice is that migration (unlike emulation) involves urgency.

"... there is a degree of urgency involved in migration. If a given document is not converted when a new paradigm first appears, even if the document is saved in its original form (and refreshed by being copied onto new media), the software required to access its now-obsolete form may be lost or become unusable due to the obsolescence of the required hardware, making future conversion difficult or impossible."

To the degree that Rothenberg has a point here it is difficult to see why it does not apply with at least equal force to emulation. Let us leave aside the fact that he has not provided any evidence of previous hardware paradigm shifts causing the sort of data loss or inaccessibility to which he alludes – after all disaster might be waiting to strike at any moment. Let us also leave aside the fact that recent experience within the digital preservation community indicates that the amount of migration intervention that has actually been required was less than expected – the future may be much worse than the past in this respect. Rothenberg does not make it clear why migration involves a peculiar degree of urgency. The introduction of a new hardware paradigm would mean that every emulator written to run on the previous paradigm would no longer function on the new device. This would leave us with two options:

- Write new emulators
- Migrate the old emulators to run on the new platform

The first option is at least as complicated as having to develop new migration tools, while the second is itself a form of migration and must, on that account, be susceptible to any problems which migration faces. The KEEP approach represents a version of option two. KEEP's proposal is to develop a virtual machine as the platform on which emulators will be written (or ported) to run, but which is designed in such a way as to be migratable without difficulty to any conceivable hardware platform of the future. Thus, the plan is to ensure that emulated environments once written can be kept portable. While KEEP's focus is on the portability of emulators, it should not be forgotten that just as emulators can be written to run on the KEEP Virtual Machine, so could other applications such as word processors, spreadsheets, databases etc. Therefore one possible way of ensuring relatively easy migration of files and applications from one hardware platform or paradigm onto another would be to target them for an easily portable virtual machine. Rothenberg is not alone in speaking about emulation and migration as if they were two diametrically opposed approaches. We regard this as a false dichotomy. Migration and emulation are better viewed as complementary approaches. Some digital objects are more amendable to migration than others; ascii files migrate much more easily than complex interactive

digital objects such as games. Even if some digital objects prove, in practice, to be intractable to migration, the emulators on which they run will either have to be themselves migrated or completely replaced at some time in the future. A future in which emulation completely displaces migration is not one that we take seriously. On the whole it would be much better if the migration and emulation ‘camps’ sought common ground.

Migration (unlike emulation) is, on Rothenberg’s account, at a disadvantage because it is an ongoing activity.

“Worse yet, this problem does not occur just once for a given document (when its original form becomes obsolete) but recurs throughout the future, as each form into which the document has migrated becomes obsolete in turn. Furthermore, because the cycles of migration that must be performed are determined by the emergence of new formats or paradigms, which cannot be controlled or predicted, it is essentially impossible to estimate when migration will have to be performed for a given type of document – the only reliable prediction being that any given type of document is very likely to require conversion into some unforeseeable new form within some random (but probably small) number of years.” (Rothenberg 1999)

Rothenberg is quite correct to draw our attention to the fact that our current digital documents are such that it is not possible merely to save their bitstreams and periodically refresh them in order to secure their access for future generations. As indicated above, a shift in hardware platforms or the change of a paradigm requires preservation action to be taken. But this is true whether the preservation strategy is based on migration or emulation. Indeed whatever the preservation strategy employed (other than choosing not to preserve) it would be possible to think of circumstances in which regular preservation action might be required. Rothenberg’s observation is not therefore pertinent to migration alone but is more by way of a statement of how the world works, and as such need not be considered further in the present context. Similar remarks apply to Rothenberg’s concerns about the ‘unpredictability’ of the occurrence of circumstances that will necessitate a preservation intervention. It is to be expected that the mean time between preservation interventions is shorter with some preservation strategies than others. It is difficult to see how this could be proven in advance or determined with any great accuracy but if this information were available it would be valuable (but not decisive) in helping to determine the strategy adopted by individual institutions. Ex hypothesi, nothing except good fortune, can protect us from the unexpected or the unpredictable; so it is reasonable to conclude that however technically brilliant an emulation strategy might be developed, it too will be vulnerable to unforeseen circumstances.

Rothenberg is concerned that migration (unlike emulation) is a piecemeal activity.

“Since different format changes and paradigm shifts affect different (and unpredictable) types of documents, it is likely that some of the documents within a given corpus will require migration before others. ... This implies that any given corpus is likely to require migration on an arbitrarily (and uncontrollably) short cycle, determined by whichever of the component types of any of its documents is the next to be affected by a new format or paradigm shift.” (Rothenberg 1999)

Rothenberg places a great deal of the weight of his argument on the notion that the future is unknowable. He decorates this relatively banal observation with unsupported claims that computing paradigms are particularly susceptible to unforeseen change. Here his variation on the theme of unpredictability concentrates on the notion that some unforeseen changes will impact one type of document more than another. He is quite right of course; indeed he is rather more right than he cares to admit. After all, there are changes likely to occur in the future but about which we presently know nothing that will affect some emulators more than others, or some aspects of some emulators more than others. So what? Those responsible for preserving digital materials for access by future generations will have to respond to whatever circumstances arise. Choosing an emulation approach does not provide a safeguard against the unpredictable and Rothenberg has not offered any grounds for thinking that a migration approach will leave the digital preservation community particularly exposed to an uncertain future.

According to Rothenberg, migration (unlike emulation) does not scale well.

“Finally, migration does not scale well. Because it is labor-intensive and highly dependent on the particular characteristics of individual document formats and paradigms, migration will derive little benefit from increased computing power. It is unlikely that general purpose automated or semiautomated migration techniques will emerge, and if they do, they should be regarded with great suspicion because of their potential for silently corrupting entire corpora of digital documents by performing inadvertently destructive conversions on them. As the volume of our digital holdings increases over time, each migration cycle will face a greater challenge than the last, making the essentially manual methods available for performing migration increasingly inadequate to the task.” (Rothenberg 1999)

The effort expended in producing an emulator sufficiently robust and reliable to play a significant role in the context of managed digital preservation is, as we have seen, substantial. The further away in time we move from having access to the original hardware which is being emulated, the less confidence we have in the accuracy of each new emulator. Once the original hardware is no longer available for inspection and comparison we are (at best) left with comparing the behaviour of the newly developed emulators with the performance of the most trusted previous emulator, and this is liable, over time, to result in silent degradation of emulator performance. The question of whether migration involves a greater or lesser amount of human resource than emulation is entirely empirical. Rothenberg is quite mistaken in thinking that the matter can be settled in advance as a matter of principle. Furthermore, it is likely that at some points in time migration will involve less human effort than emulation while at others the situation will be reversed.

We can be reasonably confident that the performance of emulators will be better on faster machines having greater computing resources available to them, but similar expectations must surely be reasonable concerning migration as well. Is it not likely that decision support systems and artificial intelligence driven programs running on future machines will increasingly automate the process of migration? Rothenberg's pessimism is not well supported by argument and seems more axiomatic than reasoned.

Rothenberg is correct to point out that each migration cycle presents new challenges but we simply cannot know in advance if these will always be more taxing than the cycles that have gone before. Each emulation cycle also presents new challenges but in the absence of further information it is idle to speculate concerning the extent to which we might be facing an upward spiral of complexity or some other pattern.

Rothenberg's judgement is that migration is essentially an approach based on wishful thinking but this harsh assessment is not supported by persuasive argument. Such worthwhile points as he has made apply with more or less equal force to his preferred approach of using emulation for digital preservation. He seems either unaware of, or unconcerned about, the fact that emulators are themselves digital objects which, if they are to continue to be of use, must either be migrated on to future hardware platforms or run under another layer of emulation. The effect of this simple fact is to undermine a great deal of what Rothenberg says. Curiously, Rothenberg fails to voice a great deal of what might be said in support of emulation. For example, he does not place enough weight on the notion that migration is better suited to simple digital objects than to the more complex challenges increasingly faced by those responsible for preservation within an institutional context. Too much subtlety is sacrificed in order to draw clear 'battle lines' with the advocates of migration. The KEEP project recognises that value of emulation in dealing with complex digital objects but is also aware of both the practical necessity of migration and its continuing value to end users. Even if it were possible to arrive at a situation in which an emulation could be deployed for all digital objects it would still remain the case that migration would have a role to play. There are numerous situations in which end users would prefer access to files via migration rather than emulation. For example, in making use of old research material in the context of a new report it is usually more desirable to migrate the old work for incorporation in the new than to ask for the material to be made available under emulation. Context is everything and there will continue to be demand for both migration and emulation for as far into the future as we can see.

The depiction of digital preservation as being faced with a choice between migration and emulation is not only a false dichotomy; it is also highly counter-productive. It encourages practitioners to take sides when they should be working together to devise common and pluralistic approaches to a complex set of preservation problems.

Migration purports to concentrate on the information content of the digital object itself and attempts to preserve for future generations the ability to access that content in the face of constantly changing technology. Key to this approach is the very Socratic notion of 'significant properties' that was developed in the CEDARS project and has recently received useful coverage within the InSPECT project (Knight and Pennock 2009) and from the British Library (Dappert and Farquhar 2009). Significant properties are held to refer to the very essence of a digital object; its intellectual content. The argument goes that so long as the significant properties of a digital object are retained, the object's intellectual content will have been preserved in spite of any 'superficial' changes in form or appearance. A property of a digital object that is not held to be significant can, so the reasoning goes, simply be ignored in preservation actions. This does not however represent the ideal state of affairs. As Hedstrom & Lee

put it: “In an ideal world, free from technical and economic constraints, libraries and archives would preserve their physical and digital collections in their original form with all significant properties intact.” (Hedstrom and Lee 2002). The Planets project also engaged with the idea of significant properties (Montague et al., 2010b).

Migration, as a preservation approach, presupposes that the information content of digital objects is both fixed and discernable, and can survive intact any changes to form and structure which are necessitated by a succession of migration processes. However there is little reason to suppose that any of this is true in general. It is not difficult to provide examples which show that the meaning and significance of a digital (or any other) object changes over time and place. Objects which had little perceived significance at the time of their creation often come to have much greater importance for future generations. For example, the Rosetta Stone which at the time of its production in 196 BCE served to record a decree issued at Memphis on behalf of Ptolemy V, came in the 19th Century to have a different (and much greater) significance in unlocking our understanding of Egyptian hieroglyphics. Not only does the Rosetta Stone serve to show that the significance of objects changes over time, but it also shows that the information content of objects is a derived property rather than something which is intrinsic. The creators of the Rosetta Stone would presumably have had no intention of providing a means by which to understand hieroglyphics. They would, one must suppose, have been much more intent on ensuring that Ptolemy’s decree was widely understood. The stone only came to have the importance we attribute to it because knowledge of hieroglyphics was lost from the world and may well have remained lost had not the stone existed. Suppose the creators of the Rosetta Stone had been asked to identify its ‘significant properties’ and been charged with developing a migration-based preservation strategy for preserving its information content into the future. It is entirely possible that they would have been satisfied to preserve a single version of the text in (say) modern Egyptian so long as it captured reasonably well the sense of Ptolemy’s original decree. If the original Stone had not also been preserved, the information content it was originally understood to have contained would, today, be of only passing interest and might perhaps not be worth retaining at all.

Writing in a slightly different context, Dappert & Farquhar come to much the same conclusion; “... *it is not possible to determine out of context which properties reflect content and which reflect circumstance.*” (Dappert and Farquhar 2009).

For digital counterparts of the Rosetta Stone, the situation is made even more complex because preserving an original bitstream in addition to any modified versions will not, of itself, assure future generations access to the original digital object, unless further steps are taken to protect some route back to the original technology platform.

In an interesting and generally stimulating paper (Hedstrom and Lampe 2001) the question of the extent to which users discriminate between or much care about whether the digital objects are preserved using a migration strategy or an emulation approach is explored. Hedstrom and Lampe’s methodology was to train test subjects for one hour on a game called Chuckie Egg, originally devised for the BBC Micro. The subjects first played the game on the original hardware platform using an unmodified copy of the game. Next, the participants were divided into two groups the first of which of which played on a migrated version of the game while the other played on an emulator. Participants’ responses were gathered under four headings:

satisfaction, perceived ease of use, performance, perceived differences between the original game and the game in the test condition.

Many of the differences that users reported concerned the hardware environment and Hedstrom & Lampe were struck by the sensitivity of users to small changes in the digital object. They concluded that: "Although some of these attributes may be unique to interactive games, this user test suggests that archivists and librarians need a much more refined definition of the characteristics of digital objects that may warrant preservation, regardless of whether emulation or migration is the preferred technical strategy. Users in our study identified attributes such as motion, speed, and sound quality, which are present in many contemporary interactive digital objects but have received scant attention in discussions of digital preservation.'(Hedstrom and Lampe 2001).

In some ways, this experiment provides the most favourable conditions under which to assess migration. In order to draw conclusions about how migration might compare to emulation as a preservation approach over the long term it would have been better to provide the subjects with a migrated digital object that had undergone a succession of migrations, preferably each carried out by a different team. During each migration, one might expect that a digital object will exhibit minor behavioural deviations from the previous migrated version. Over a reasonable number of iterations we might expect this to result in significant and quite noticeable changes. Had the participants in Hedstrom & Lampe's experiment been exposed to a digital object that was repeatedly migrated rather than a 'first generation' migration object, the difference in the user experience offered between accessing preserved digital objects via emulation and via migration might have been more apparent. To illustrate the point, a sentence (in English) was entered into the online translation service Babel Fish⁵ the output (in Italian) was fed back into Babel Fish and the process was repeated a number of times with the following results:

1. (English) This might be the result of successive migration interventions.
2. (Italian) Ciò ha potuto essere il risultato degli interventi successivi di espansione.
3. (French) Cela a pu être le résultat des interventions suivantes d'expansion.
4. (Portugeese) Aquilo pôde ser o resultado das intervenções seguintes d'expansão.
5. (English) That could be the result of the following interventions d' expansion.
6. (Korean) 그것 은뒤 에 오는 내 정 간섭 d' 의 결 과 일 수 있 었 다; 확 장.
7. (English) It the domestic intervention d' which comes after; Justice resultant one possibility was; Expansion.

For properties of a digital object that are not treated as significant there is no reason to assume that their 'fidelity' would be any better preserved across successive migrations than was our test sentence. The effects of this little test appear very stark. However, even under ideal conditions, translation from one natural language to another is a complex and 'lossy' activity. There are no strong grounds for believing that all the 'significant' properties of a digital object are capable of being retained across machine language migrations any better than with those of text survive natural

⁵ http://babelfish.yahoo.com/translate_txt

language migrations. It should also be borne in mind that each migration presents its own challenges and the amount of 'noise' picked up during the move from one hardware platform to another is not a constant. The degree to which a digital object will be damaged during migration will be dependent on the characteristics of the digital object itself and the peculiarities of the platforms concerned.

The potential loss of digital objects and their underpinning hardware presents challenges that traditional archaeology would be hard pressed to resolve as form and function have become separated by layers of abstraction and interrelated complexity. Yet technological obsolescence, collective cultural neglect, lack of familiarity in the cultural conscience coupled with a lack of intuitive affordance of the carrier or hardware exterior and mechanics, means that hardware and software often face the same contextless consignment to obscurity and interpretation that can beset some ancient artefacts and yet the process is occurring in timescales measured in decades rather than centuries.

The awareness of the need for interpretation and intermediated imitation is not new it has merely become more daunting in its scope so that sustaining backward compatible systems on each new generation of hardware is simply neither tenable nor, perhaps more importantly, profitable for developers. Indeed supporting emulation of hardware platforms by commercial companies potentially incurs some losses when marketing new hardware and software and obsolescence can be viewed by some companies as a welcome benefit of systems and software with limited shelf lives as this promotes perpetual renewal of perceived needs in the consumer market.

This in and of itself need not be detrimental in the context of digital preservation after all it is this consumer driven development that has produced competition and subsequent advances in technology which in turn have created the significant impact on all aspects of society that demands preservation but also the ability to envisage plan and technologically implement preserving data on a scale that has never been previously undertaken with other information media. However whilst the capability may exist in theory, the practical implementation and adoption of any given strategy, is beset with problems of phenomenal complexity.

Identifying strategies, technological concerns, stakeholders, investment, obligations legalities and risk are the purpose of this paper but where resources are identified and evaluated they need to be viewed in their wider context and should not in any way be considered stand alone solutions.

Portability of emulation across current and future platforms is obviously a cornerstone in KEEP but in order for this to have current and long term application in digital preservation, a wide array of effective emulators need to be identified classified archived and preserved to ensure that incorporation into an emulation framework is broad enough and robust enough to meet user group needs. Furthermore such a framework and its accessible inventory must ensure adequate, sustainable and adaptable growth as new historical systems are identified, emulated, and added, and new future developments integrated.

Possibly only a virtual system could support such an approach with current levels of understanding but the virtual technology and sourcing emulators are not the only issues. Currently emulation could be viewed as independent yet cooperative, altruistic in motive, yet opportunistic in priorities, and user driven by the desires of enthusiast groups that are disparate in their intentions and often operate in isolated niches.

For long term preservation, utilising an emulation strategy, to operate with any significant breadth and depth and any hope of longevity these groups and resources of similar talents and expertise need to be de-marginalised and inform or contribute directly to a centralised systematic approach to emulating.

Furthermore whilst emulation is sometimes referred to as a strategy for data that gets left behind, interim strategies are variable in effectiveness and in implementation and adherence. To facilitate the recovery of the data left behind standards need to be identified for preservation of carrier mediums and the hardware to read and interpret them. It is likely that as things are progressing data that gets left behind will become more commonplace unless awareness and education of the issues are disseminated widely and furnished with a low burden solution for developers and commercial users.

Currently our cultural approach to the needs and scale of the issues involved in digital preservation have been slow in realisation and much of society remains unaware of the problems and risks that are being faced. Some elements are responding quicker than others but without a clearly laid out unified approach with appropriate sharing of the burden, costs and responsibilities the results will inevitably leave significant digital gaps as a legacy not just for future generations but also for the future of our current generation.

With regard to digital information and objects, memory institutions have established a number of clear guidelines with regard to the selection criteria for digital information and digital objects themselves and digital preservation standards are clear and well established e.g. OAIS etc.

Curation and Archiving is established practice and within such institutions clear policy guidelines and standards are available with a wide range of supporting resources including systematic approaches to migration and many are implemented with good systematic approach to adherence and review at significant institutions. However implementation and adherence is not universal to other appropriate organisations and companies and tackling this potential chasm requires a technological safety net and a creative approach to raising awareness and supporting those companies in engaging in preservation without creating a load that cannot be maintained. In the interim period such information may at best be consigned to abandoned media or permanently and irretrievably lost. Nevertheless standards in this discipline are well reasoned, clearer, better structured and more widely achievable than most but are dependant on other disciplines keeping pace in order for their data to remain accessible.

Hardware preservation on the other hand, with some exceptions, lacks clear guidelines or coherent aims that can be employed as a universal approach. Researching to find policies and standards for the preservation strategies for computing hardware and peripherals should provide an extensive and specific understanding of the different technologies and all the risks prevalent to the collective elements of any given hardware. However enquiries and research yielded no apparent identifiable policy guidelines or standards on preservation (other than for shipment or short term working use). Furthermore individual sources of advice are often generic and sometimes conflicting on critical points even to the extent as to whether machines should be retained. Machine preservation and maintenance approaches often are tackled ad hoc, and although regulated standards might not provide a definitive answer, guidelines and best practice, informed by research and experience, would surely be beneficial.

Working within the context of the KEEP project, the University of Portsmouth Future Proof Computing Group have started identifying emulators that are currently available or in development (a small selection of which are presented in appendix A to this paper). Naturally such a list reflects the diverse intentions of the authors and outcomes and information can be lacking, inconsistent and at face value defy simple categorisation.

Although other projects have attempted to identify emulator sources as part of their approach to digital preservation (Camileon, Creative Archiving at Michigan and Leeds) the reliance on disparate individuals whose personal endeavours are prone to the vagaries of individual circumstances and the fortunes of the host sites through which they are accessed, leave such projects vulnerable to unpredictable loss. As such the emulator landscape can shift significantly between projects with many emulator sites, including those indexed with a range of emulators, being lost, often with little or no warning, in the last few years (emulation9, emula zone, emulation.cc, system16 etc.) and ironically even the findings and reports of such projects as Camileon are also often no longer available online on their sites.

In order to make a sustainable strategy for expanding the range of emulators available to institutions and archivists for a coherent digital preservation plan there needs to be a more systematic approach to establishing an inventory resource of emulators themselves. In addition there needs to be a coordinated approach by institutions and archives to identify emulation needs i.e. systems that have a high risk and high impact that have yet to be emulated, and standards required of an emulator, that will allow data to be effectively presented. This will then allow institutions to establish targeted working groups with users, developers and emulator enthusiasts to resolve those emulation needs.

Whilst some projects have attempted to tap into the resources of enthusiasts and emulator groups there is as yet no established, mainstream, institutionally backed, project, to our knowledge, that has attempted a systematic production of key emulator types within a strategy for long term preservation, with identified outcomes and standards. While emulation has a significant role to play in digital preservation, the lack of standardisation criteria leaves our digital legacy in the hands of niche interest groups that are not adequately supported or endorsed by the institutions that need them, and lack legitimacy and recognition in legal terms.

There is a real need to foster closer association with emulation authors, industry and institutions to provide legitimacy and mutual trust within an established protocol which would enable digital preservation approaches such as the KEEP emulation framework to meet the varied needs of diverse research, curatorial and interest groups by incorporating additional emulation developments as they arise into a system that will continue to port the emulation inventory to new hardware systems.

Locating emulator sources however is not the whole solution as indicated by an extract from the NISO document: Initiatives Principle 4: A good digital initiative has an evaluation component.

Of course the veracity of any such system requires the incorporated emulators to meet a defined standard and the only viable means of establishing and checking this is to compare it to known outcomes from the original machines ideally with confirmation from a user or developer that the performance is a faithful representation of the common experience. Given the nature of rapid obsolescence in the computer industry,

such quality checking, needs to be established with some imperative, whilst the limited number of extant, original, machines can be sourced before extinction.

To this end some resources remain available to us, with a number of established museums and some private collections that provide a means to inform emulator development as well as an opportunity to test the behaviour of emulators against the performance of the machines they purport to emulate. In addition they provide an invaluable safeguard by maintaining such machines so that vital data may be extracted and accessed. However such museums are not ideally equipped to work in an appropriately large scale without additional support.

Often such museums are staffed by volunteers and often comprise retired experts and developers along with enthusiasts and key figures in computing history and it is often they, as much as their exhibits, that bring history alive with their unique insights, profound knowledge and personal accounts. Furthermore their knowledge is often extensive and detailed with practical experience of the working machine, its idiosyncrasies and common practice that are not always readily elicited from the literature. These individuals are experts in a very real sense whose knowledge and accounts are also rare and sadly in a pragmatic sense potentially vulnerable to irreplaceable loss. There is a very real need here to make a record of that applied expertise while the opportunity of the dialogue between the individual expert and the operational machines is still available to us. Our reliance on these individual enthusiasts and private groups to maintain our heritage somehow seems to miss the point that they themselves are part of our heritage and their legacy is not yet secure. As part of the effort to see that this is done, the UK National Museum of Computing (TNMOC) at Bletchley Park, working with the University of Portsmouth have carried out a series of interviews with TNMOC staff, volunteers, enthusiasts, experts, historians, engineers restorers, developers preservationists and retired industry users amongst others. The outputs from these events will be made available in due course through the University of Portsmouth website.

References

- Bearman, D.: *Collecting Software: A New Challenge for Archives & Museums*. Archival Informatics Technical Reports 1, 80 (1987)
- Bearman, D.: *Reality and Chimeras in the Preservation of Electronic Records*. D-Lib Magazine 5(4) (1999)
- Dappert, A., Farquhar, A.: *Significance Is in the Eye of the Stakeholder*. In: Agosti, M., Borbinha, J., Kapidakis, S., Papatheodorou, C., Tsakonias, G. (eds.) ECDL 2009. LNCS, vol. 5714, pp. 297–308. Springer, Heidelberg (2009)
- Gladney, H.M.: *Durable Digital Objects Rather Than Digital Preservation*. ErpaePrints (2008)
- Hedstrom, M., Lampe, C.: *Emulation vs. Migration: Do Users Care?* RLG Diginews 5 (2001)
- Hedstrom, M., Lee, C.A.: *Significant properties of digital objects: definitions, applications, implications*. DLM-Forum, Barcelona (2002)
- Knight, G., Pennock, M.: *Data Without Meaning: Establishing the Significant Properties of Digital Research*. International Journal of Digital Curation 4(1), 159–174 (2009)
- Levy, D.M.: *Where's Waldo? Reflections on Copies and Authenticity in a Digital Environment*. *Authenticity in a Digital Environment*, pp. 24–31. Council on Library and Information Resources, Washington D.C (2000)

- Rothenberg, J.: Avoiding Technological Quicksand: Finding a Viable Technical Foundation for Digital Preservation. Council on Library and Information Resources (1999)
- Stawowczyk Long, A.: Long-term preservation of web archives – experimenting with emulation and migration methodologies. Internation Internet Preservation Consortium 54 (2009)
- Swade, D.: Preserving Software in an Object-Centred Culture. In: Higgs, E. (ed.) History and Electronic Artefacts, pp. 195–206. Clarendon Press, Oxford (1998)

Appendix A: Computer Emulators/ Multiple Emulators/ Atypical Emulation

Computer Emulators list [1] (updated March 2012). This list is not comprehensive. Please note that this list does not cover specific emulated games nor games consoles except where covered under multiple emulation systems or under the auspices of arcade emulations.

| | | | |
|-----------------|---|-------------|----------|
| Neko Project II | http://www.emulator-zone.com/doc.php/computer/kekoproject2.html | Windows | Freeware |
| CPCE | http://www.emulator-zone.com/doc.php/computer/cpce.html | Windows/Dos | Freeware |
| CCS64 | http://www.emulator-zone.com/doc.php/computer/ccs64.html | Windows | Freeware |
| Hoxs64 | http://www.emulator-zone.com/doc.php/computer/hoxs64.html | Windows | Freeware |
| Mini vMac | http://www.emulator-zone.com/doc.php/computer/minivmac.html | Many | Free |
| PearPC | http://www.emulator-zone.com/doc.php/computer/pearpc.html | Windows | Freeware |
| YAPE | http://www.emulator-zone.com/doc.php/computer/yape.html | Windows | Freeware |

Intelivision Console Emulator

| | | | |
|--------------|---|---------------|----------|
| Nostalgia | http://www.emulator-zone.com/doc.php/misc/nostalgia.html | Windows | Freeware |
| WinArcadia | http://www.emulator-zone.com/doc.php/misc/winarcadia.htm | Windows | Freeware |
| RetroCopy | http://www.emulator-zone.com/doc.php/misc/retrocopy.html | Windows/Linux | Freeware |
| Turbo Engine | http://www.emulator-zone.com/doc.php/misc/engine.html | Windows/Linux | Freeware |

Arcade Emulators: (<http://www.emulator-zone.com/doc.php/arcade/>)

| | | | |
|-----------------|---|---------|----------|
| MAME | http://www.emulator-zone.com/doc.php/arcade/mame.html | All(?) | Freeware |
| Kawaks | http://www.emulator-zone.com/doc.php/arcade/kawaks.html | Windows | Freeware |
| FinalBurn Alpha | http://www.emulator-zone.com/doc.php/arcade/finalburnalpha.html | Windows | Freeware |
| Zinc | http://www.emulator-zone.com/doc.php/arcade/zinc.html | Windows | Freeware |

| | | | |
|-----------|---|---------------|----------|
| Nebula | http://www.emulator-zone.com/doc.php/arcade/nebula.html | Windows | Freeware |
| Calice | http://www.emulator-zone.com/doc.php/arcade/calice.html | Windows | Freeware |
| VivaNonno | http://www.emulator-zone.com/doc.php/arcade/vivanonno.html | Windows | Freeware |
| Daphne | http://www.emulator-zone.com/doc.php/arcade/daphne.html | Many | Free |
| Raine | http://www.emulator-zone.com/doc.php/arcade/raine.html | Windows | Freeware |
| RetroCopy | http://www.emulator-zone.com/doc.php/misc/retrocopy.html | Windows/Linux | Free |

MAME (Multiple Arcade Machine Emulator)

Official MAME Emulators

| File | Link | Platform | License |
|--|---|----------|----------|
| MAME (regular) 0.145 Win32 command line version | http://www.emulator-zone.com/download.php/emulators/arcade/mame/official_version/winmame/mame0145b.exe | Windows | Freeware |
| MAME (i686) 0.145 Win32 command line version (I686 optimized) | http://www.emulator-zone.com/download.php/emulators/arcade/mame/official_version/winmame/mame0145b_i686.exe | Windows | Freeware |
| MAME (64 bit) 0.145 64-bit Windows command-line binaries | http://www.emulator-zone.com/download.php/emulators/arcade/mame/official_version/winmame/mame0145b_64bit.exe | Windows | Freeware |
| MAME 0.100 DOS command line version | http://www.emulator-zone.com/download.php/emulators/arcade/mame/official_version/dosmame/mame0100b_dos.zip | Windows | Freeware |

MAME Ports

| File | Link | Platform | License |
|--|---|----------|----------|
| MAMEUI (32bit) 0.145 32 bit version | http://www.emulator-zone.com/download.php/emulators/arcade/mame/mameui/MameUI32_0.145.7z | Windows | Freeware |
| MAMEUI (64bit) 0.145 64 bit version | http://www.emulator-zone.com/download.php/emulators/arcade/mame/mameui/MameUI64_0.145.7z | Windows | Freeware |
| MAMEUI FX 32 0.145 MAMEUI | http://www.emulator-zone.com/download.php/emulators/arcade/mame/MAMEUIFX32/mameuifx32_0145.exe | Windows | Freeware |

MAME Frontends

| File | | Platform | License |
|---|---|----------|----------|
| Emuloader 5.8.2 Command line MAME frontend | http://www.emulator-zone.com/download.php/emulators/archives/mame/frontend/emuloader/e1582-bin.rar | Windows | Freeware |
| MAME Classic 5.5.0 Command line MAME frontend | http://www.emulator-zone.com/download.php/emulators/archives/mame/frontend/mame_classic/F1550.zip | Windows | Freeware |

MAME Support Files

| File | | Platform | License |
|---|---|----------|----------|
| mameinfo.dat 0.136 Information DAT file | http://www.emulator-zone.com/download.php/emulators/archives/mame/support_files/mameinfo-dat/Mameinfo0136.zip | N/A | Freeware |
| Cheat.dat 0.132 Cheat codes DAT file | http://www.emulator-zone.com/download.php/emulators/archives/mame/support_files/cheat-dat/cheat0132.zip | N/A | Freeware |
| History.dat 0.136 History DAT file | http://www.emulator-zone.com/download.php/emulators/archives/mame/support_files/history-dat/mamehistory136.zip | Windows | Freeware |
| Highscore.dat 0.132 High score DAT file | http://www.emulator-zone.com/download.php/emulators/archives/mame/support_files/highscore-dat/mame0132.zip | Windows | Freeware |

Multiple Systems Emulators (M.E.S.S.)

| File | | Platform | License |
|--|---|-----------|----------|
| M.E.S.S. | http://www.emulator-zone.com/doc.php/misc/mess.html | Windows | Freeware |
| M.E.S.S. (32bit) 0.145 | http://www.emulator-zone.com/download.php/emulators/misc/mess/mess0145b.zip | Windows | Freeware |
| M.E.S.S. (64bit) 0.145 | http://www.emulator-zone.com/download.php/emulators/misc/mess/mess0145b_x64.zip | Windows | Freeware |
| MESS 0.101b Multi Emulator Super System | http://www.emulator-zone.com/download.php/emulators/mac/mess/macmess0101b.zip | Macintosh | Freeware |

Multiple Emulator Super System Claims to Emulate the Following:

- Adventurevision
- Amstrad 464plus
- Amstrad 6128plus
- Amstrad PC1512 (version 1)
- Amstrad PC1512 (version 2)
- Amstrad PC1640 / PC6400 (US)
- Amstrad PC20
- APF M-1000
- Apple II
- Apple II+
- Apple IIc
- Apple IIc Plus
- Apple IIc (UniDisk 3.5)
- Apple IIe
- Apple IIe (enhanced)

- Amstrad/Schneider CPC464
- Amstrad/Schneider CPC6128
- Amstrad/Schneider CPC664
- APEXC (as described in 1957)
- APF Imagination Machine
- Atom with Eprom Box
- Bally Pro Arcade/Astrocade
- Bally Pro Arcade/Astrocade (white case)
- BBC Micro Model A
- BBC Micro Model B
- BBC Micro Model B+ 128k
- BBC Micro Model B+ 64K
- BBC Micro Model B with WD1770 disc controller
- C64GS (PAL)
- C65 / C64DX (Prototype, German PAL, 910429)
- C65 / C64DX (Prototype, NTSC, 910111)
- C65 / C64DX (Prototype, NTSC, 910523)
- C65 / C64DX (Prototype, NTSC, 910626)
- C65 / C64DX (Prototype, NTSC, 910828)
- C65 / C64DX (Prototype, NTSC, 911001)
- CBM4064/PET64/Educator64 (NTSC)
- Channel F
- Chess Champion MK II
- Colecovision
- Colecovision (Thick Characters)
- Color Computer
- Color Computer 2
- Color Computer 2B
- Color Computer 3 (NTSC)
- Color Computer 3 (NTSC; HD6309)
- Color Computer 3 (PAL)
- Color Computer (Extended BASIC 1.0)
- Colour Genie EG2000
- Commodore 128 French (PAL)
- Commodore 128 German (PAL)
- Commodore 128 Italian (PAL)
- Commodore 128 NTSC
- Commodore 128 Swedish (PAL)
- Commodore 16/116/232/264 (PAL)
- Commodore 16/116/232/264 (PAL), 1551
- Commodore 16 Novotrade (PAL, Hungarian Character Set)
- Commodore 30xx (Basic 2)
- Commodore 30xx (Basic 2) (business keyboard)
- Commodore 364 (Prototype)
- Commodore 40xx FAT (CRTC) 50Hz
- Commodore 40xx FAT (CRTC) 60Hz
- Commodore 40xx THIN (business keyboard)
- Commodore +4 (NTSC)
- Commodore +4 (NTSC), 1551
- Commodore 64 (NTSC)
- Commodore 64 Swedish (PAL)
- Commodore 64/VC64/VIC64 (PAL)
- Commodore 80xx 50Hz
- Commodore 80xx 60Hz
- Commodore 80xx German (50Hz)
- Commodore 80xx Swedish (50Hz)
- Commodore B128-40/Pet-II/P500 60Hz
- Commodore B128-80HP/710
- Commodore B128-80LP/610 60Hz
- Commodore B256-80HP/720
- Commodore B256-80HP/720 Swedish/Finnish
- Commodore B256-80LP/620 50Hz
- Commodore B256-80LP/620 Hungarian 50Hz
- Apple //e (Platinum)
- Apple I
- Aquarius
- Arcadia 2001
- Atom
- Dragon 64
- Enterprise 128
- Enterprise 128 (EXOS 2.1)
- EURO PC
- Famicom
- Galaksija
- Gamboy - (PAL) Japanese SMS BIOS v2.1
- GameBoy
- GameBoy Color
- GameBoy Pocket
- Game Gear - European/American
- Game Gear - European/American Majesco Game Gear BIOS
- Game Gear - Japanese
- Game Gear - Japanese Majesco Game Gear BIOS
- Geneve 9640
- HB-8000 Hotbit 1.1
- HB-8000 Hotbit 1.2
- IBM PC 08/16/82
- IBM PC 10/27/82
- IBM PC/XT (CGA)
- Intellivision
- Intellivision Keyboard Component (Unreleased)
- Intellivision (Sears)
- Inves Spectrum 48K+
- Jupiter Ace
- Kaypro 2x
- KC 85/3
- KC 85/4
- KC Compact
- KIM-1
- Laser 110
- Laser 200
- Laser 210
- Laser 310
- Laser 350
- Laser 500
- Laser 700
- Lisa2
- Lisa2/10
- LNW-80
- Lynx
- Lynx (alternate rom save!)
- Lynx II
- Macintosh 512ke
- Macintosh Plus
- Macintosh XL
- Mark III - (PAL) Japanese SMS BIOS v2.1
- Master System III Compact (Brazil) - (PAL) European BIOS with Sonic The Hedgehog
- Master System II - (NTSC) US/European BIOS with Alex Kidd in Miracle World
- Master System II - (PAL) European BIOS with Sonic The Hedgehog
- Master System II - (PAL) US/European BIOS with Alex Kidd in Miracle World
- Master System - (NTSC)
- Master System - (NTSC) Hacked US/European BIOS v1.3

- Commodore Max (Ultimax/VC10)
- Commodore SP9000/MMF9000 (50Hz)
- CP400
- CPS Changer (Street Fighter ZERO)
- Dragon 32
- Master System - (PAL) Hacked US/European BIOS v1.3
- Master System - (PAL) Japanese SMS BIOS v2.1
- Master System - (PAL) US/European BIOS v1.3
- Master System - (PAL) US/European BIOS v3.4 with Hang On
- Master System Plus - (NTSC) US/European BIOS v2.4 with Hang On and Safari Hunt
- Master System Plus - (PAL) US/European BIOS v2.4 with Hang On and Safari Hunt
- MC-10
- Megadrive / Genesis
- Microbee 32 IC
- Microbee 32 PC
- Microbee 32 PC85
- Microbee 56
- Microtan 65
- MSX 1
- MSX 1 (Japan)
- MSX 1 (Korea)
- MSX 1 (UK)
- MSX 2
- MSX 2 (BASIC 2.1)
- MSX 2 (Japan)
- MTX 512
- MZ-700
- MZ-700 (Japan)
- Nascom 1 (NasBug T1)
- Nascom 1 (NasBug T2)
- Nascom 1 (NasBug T4)
- Nascom 2 (NasSys 1)
- Nascom 2 (NasSys 3)
- Nintendo Entertainment System (NTSC)
- Nintendo Entertainment System (PAL)
- Odyssey 2
- Oric 1
- Oric Atmos
- Oric Telestrat
- PC200 Professional Series
- PC-8801 MKIISR (Hires display, VSYNC 24KHz)
- PC-8801 MKIISR (Lores display, VSYNC 15KHz)
- PC/AT (CGA, MF2 Keyboard)
- PC/AT (VGA, MF2 Keyboard)
- PC (CGA)
- PC Engine/TurboGrafx 16
- PC (MDA)
- PC/XT (VGA, MF2 Keyboard)
- PDP-1
- PET2001/CBM20xx Series (Basic 1)
- Philips P2000M
- Philips P2000T
- PK-01 Lviv
- PK-01 Lviv (alternate)
- PK-01 Lviv (prototype)
- Pocket Computer 1251
- Pocket Computer 1350
- Pocket Computer 1401
- Pocket Computer 1402
- Pocket Computer 1403
- Master System - (NTSC) US/European BIOS v1.3
- Master System - (NTSC) US/European BIOS v3.4 with Hang On
- Master System - (PAL)
- Sam Coupe
- Sanyo / Dick Smith VZ200
- Sanyo / Dick Smith VZ300
- Sorcerer
- Sord M5
- Spectrum I+
- Superboard II
- Super GameBoy
- Super Nintendo Entertainment System (NTSC)
- Super Nintendo Entertainment System (PAL)
- Super Vision
- SVI-318
- SVI-328
- SVI-328 (BASIC 1.11)
- System-80
- Tandy 1000HX
- Tatung Einstein TC-01
- TC-2048
- Textet TX8000
- TI-81 Ver. 1.8
- TI-85 ver. 10.0
- TI-85 ver. 3.0a
- TI-85 ver. 4.0
- TI-85 ver. 5.0
- TI-85 ver. 6.0
- TI-85 ver. 8.0
- TI-85 ver. 9.0
- TI-86 homebrew rom by Daniel Foesch
- TI-86 ver. 1.2
- TI-86 ver. 1.3
- TI-86 ver. 1.4
- TI-86 ver. 1.6
- TI99/4A Home Computer (Europe)
- TI99/4A Home Computer (US)
- TI99/4A Home Computer with EVPC
- TI99/4 Home Computer (Europe)
- TI99/4 Home Computer (US)
- TI Avigo 100 PDA
- TI Model 990/10 Minicomputer System
- TK-90x Color Computer
- TK-95 Color Computer
- TRS-80 Model I (Level I Basic)
- TRS-80 Model I (Radio Shack Level II Basic)
- TRS-80 Model I (R/S L2 Basic)
- TS-2068
- UK101
- UK-2086 ver. 1.2
- Vectrex
- VIC1001 (NTSC)
- VIC20 (NTSC)
- VIC20 (NTSC), IEEE488 Interface (SYS45065)
- VIC20 PAL, Swedish Expansion Kit
- VIC20/VC20(German) PAL
- XP-800 Expert 1.0
- XP-800 Expert 1.1
- Z88
- ZX Spectrum
- ZX Spectrum 128
- ZX Spectrum 128 (Spain)
- ZX Spectrum +2

- Pocket Computer 1403H
- Pravetz 8D
- Pravetz 8D (Disk ROM)
- Pravetz 8D (Disk ROM, RadoSoft)
- Salora Fellow
- ZX Spectrum +3
- ZX Spectrum +3e
- ZX Spectrum +3e (Spain)
- ZX Spectrum +3 (Spain)
- ZX Spectrum +4
- ZX Spectrum +2a
- ZX Spectrum +2 (France)
- ZX Spectrum +2 (Spain)
- ZX Spectrum (BusySoft Upgrade v1.18)
- ZX Spectrum (Collier's Upgrade)
- ZX Spectrum (De Groot's Upgrade)
- ZX Spectrum (LEC Upgrade)
- ZX Spectrum (Maly's Psycho Upgrade)

Multiple Systems Emulators (Kawaks)

| File | Platform | License |
|---|----------|----------|
| Kawaks 1.62 | Windows | Freeware |
| http://www.emulator-zone.com/download.php/emulators/arca/de/kawaks/winkawaks162.zip | | |

Multiple Systems Emulators (Nebula)

| File | Platform | License |
|---|----------|----------|
| Nebula 2.25b CPS2, Neogeo, PGM and Konami emulator | Windows | Freeware |
| http://www.emulator-zone.com/download.php/emulators/arca/de/nebula/nebula225b.zip | | |

| File | Platform | License |
|---|----------|-------------|
| ScummVM | Many | Open Source |
| http://www.emulator-zone.com/doc.php/computer/scummvm.html | | |
| ScummVM (32bit) 1.1.1 | Windows | Freeware |
| http://www.emulator-zone.com/download.php/emulators/computer/ScummVM/scummvm-1.1.1-win32.zip | | |
| ScummVM (64bit) 1.1.1 | Windows | Freeware |
| http://www.emulator-zone.com/download.php/emulators/computer/ScummVM/scummvm-1.1.1-win64.zip | | |