

Collecting and Conserving Software-based Art (Outside the Institution)

Tom Ensom
King's College London
Department of Digital Humanities,
King's College London, Drury Lane,
London, WC2E 9NH, UK
thomas.ensom@kcl.ac.uk

Software-based art (i.e. that in which software is the primary medium) presents considerable challenges for artists, collectors, and conservators, who are hoping to preserve artworks in the long-term. It is essential then that these parties consider the implications of technological change, and take action where appropriate. However as with artistic practise itself, software-based artworks vary considerably in their components and behaviour, and typically require a bespoke approach to conservation which can be both time consuming and expensive. There has up until recently been little consideration of the role of the private collections in the context of software-based art, with collections of such works being a relative rarity. Despite this lack of representation, the medium presents great opportunities for the democratisation (via open, internet distribution) of collecting, an activity which has traditionally been something of a rarefied hobby. Drawing on research into the preservation of complex digital objects, I discuss some of the major challenges for those collecting and conserving software-based art, and outline some approaches to these challenges achievable without the resources of a major institution.

Software-based art. Digital preservation. Art conservation. Personal digital archiving. Documentation.

1. INTRODUCTION

Software-based art can be defined as that in which software is the primary medium. Within software-based art we see considerable variation in components (of which there can be many) and characteristics; ranging from born digital internet-distributed works, to those requiring installation in physical space. They may also exhibit a wide range of behaviours essential to the essence of the artwork, such as responding to interaction, or fetching data over the internet. A useful characterisation is that at the heart of all software-based art is code, although this code might not necessarily have been touched by the artist's hand. As a result of their complex nature, software-based artworks pose a considerable challenge to artists, collectors and conservators hoping to preserve them in the long-term. It is essential then, that those creating and collecting software-based artworks consider these challenges (see Section 2 for a more detailed introduction to these) and take action where appropriate. While there remains a level of complacency in some quarters, considerable progress has been made toward stable strategies for the conservation of software-based art in the past decade, some which I will discuss in this paper.

Perhaps in part due to the problems outlined above, there has up until recently been little consideration of the role of the private collection in the context of software-based art, with collections of such works remaining a relative rarity. Despite this lack of representation, the medium presents great opportunities for the democratisation (via open, internet distribution) of collecting, an activity which has traditionally been something of a rarefied hobby. The private collection has had a notable role in art history, with the significance of artworks in the care of such collections perhaps only realised long after their creation. Major cultural heritage institutions may later acquire these works (be that by purchase, gift or loan) and so enable public display and a wider reach. It is important then, that the sustainable, small-scale collecting of software-based art grows, and I hope this paper might provide an entry point to that world. We now find the artist, institution and collector disconnected because of their differing goals, but at a point of converging interest in conservation issues. All three groups find themselves faced with a monumental challenge, which will surely involve making compromises as to achievable preservation goals. The 'outsider collector' and amateur enthusiast are players of unrealised potential in this dynamic, who

might not only possess unique skills for addressing a particular kind of technological preservation challenge, but more importantly have the inclination to spend their time caring for a collection.

2. DIGITAL DOES NOT MEAN FOREVER

On first encountering the notion of preservation for software-based art (and indeed, digital objects in general), I am often greeted with a certain bemusement over where exactly the problem lies. After all, such media are not subject to the physical ravages of time in the same way as say a printed photograph or oil-on-canvas painting might be. Perhaps due a perception of digital files as relatively immutable, this is an unfortunate underestimation of the threat to digital files posed by technological change. A considerable amount has been written on the subject of digital decay – two introductions I like are Fino-Radin's (2011) and Rinehart & Ippolito's (2014) – so I shall not go into great detail in this paper. However, some of the key threats, with particular reference to software-based art, are summarised below.

Hardware forms an integral operational layer on which software depends. Hardware components might also be integral to the function of an artwork (for example, a games console controller or a video capture device). Of course, such hardware (and the drivers and operating system that sit on top of them) tend to have distinct lifespans. Even since the rise of the personal computer, we have seen significant changes in the nature and power of hardware, and in the event of failure (which is nothing if not inevitable) replacing vintage hardware can be extremely challenging. Simply upgrading hardware is not the solution, as software is itself inextricably tied to this hardware and the associated set of platform specific protocols. When the software itself ceases to function in a current technological environment, updates to combat this are not guaranteed if the product is no longer actively developed. While total failure of software is perhaps the worst case scenario, the effect of technological change on an artwork may be more subtle than just un-executable code. Due to the subtleties of component interaction and rendering, the original qualities of the artwork may be altered as hardware changes.

When any of the aforementioned factors are found to have affected an artwork, other limiting factors can make the problem difficult to address. The use of locked-in systems of production (such as proprietary software) often result in the essence of the artwork, the source code, being inaccessible. Furthermore, there may not be sufficient descriptive information available to interpret the object if this risk has not been assessed and documentation

compiled beforehand. The cumulative effect of the issues I have outlined above is that changes in the performance, behaviour and appearance (if not a total cease of function) of an artwork become increasingly likely with time. The extent of these challenges has resulted in an (ongoing) rethink of the scope and nature of conservation practise for these works, elements of which I will explore in the next section.

3. A PRAGMATIC APPROACH TO THE CONSERVATION OF SOFTWARE-BASED ART

The acquisition aspect of collecting software-based art is a nebulous topic and tangential to the focus of this paper. It does provide some important context however, in that a great deal of software-based art (particularly that which is disseminated via the internet) has bypassed commodification. This is perhaps due in some part to difficulties in creating saleable objects, and in some part due to a rejection of traditional market values by artists. As such, acquiring such artworks is often simply a case of downloading freely available files from an artist's website. Barriers to starting a collection then, are low.

I will now explore some practical approaches to the conservation challenges I outlined in the previous section. As a starting point, the paradigms of 'personal digital archiving' (adapted from those of the Library of Congress 2013) are worth considering:

- (i) Identify the extent of the materials you have, and decide which of them you wish (or are practically able) to preserve
- (ii) Organise and describe your materials
- (iii) Make preservation-ready copies of materials, store them in different places, and manage them over time

This three step process presents the fundamentals of any effective strategy. One important underlying assumption is a consideration of redundant backup strategy; that is, one which ensures data safety and integrity. Fortunately, solutions for achieving this are now available on a relatively low budget, even for large volumes of material. Large hard drives are now relatively cheap (bearing in mind that it is good practise to migrate to new drives around the three years mark), and a multi-copy local backup can be set up easily. Complementary to this is the use of cloud (i.e. online) storage, which – with the advent of very low cost services such as Amazon Glacier – can be suitable for even very large volumes of material. Beyond storage and backup, a number of other more general concepts transfer from other applications of digital preservation, and have been well covered online; the website of the Digital Preservation Coalition (<http://www.dpconline.org>) is

a good place to start. I shall discuss two of the most important aspects of these core activities in detail below, and elaborate more on approaches to some of the more challenging preservation and documentation problems posed by software-based art.

3.1 Documentation and technical description

Perhaps the most important word in the three step strategy above is 'describe'. This indicates the creation of documentation and technical description materials to accompany an artwork. The creation of these ensures that not only will the artwork remain interpretable into the future, but that it can (in theory) be rebuilt from these specifications (for a discussion of restaging see Section 3.2). Accepted documentation models (i.e. standards or structures) specific to software-based art have yet to emerge, but more general models for media art can be repurposed to provide some coverage. The Variable Media Approach (Depocas et al. 2003) is has been particularly influential in the evolution of media art documentation, and the Variable Media Questionnaire (a browser-based version of which is available at: <http://variablemediaquestionnaire.net>) the project developed provides a useful basic template. More closely related to software-based art is the 'Checklist for recording and documentation of digital artworks' (Lurk 2011), which presents a minimal list of suggested fields for documenting artworks involving computers. My own research, as part of an AHRC-funded PhD project involving King's College London and Tate, is looking at how we might augment existing models to better cover software-based art. I will be looking particularly to other domains for inspiration, particularly performance art and software studies. Projects such as Preserving Virtual Worlds (see McDonough et al. 2010) have proposed in-depth strategies for video game and virtual world documentation, artefacts which have much in common with software-based art. Involving some of the same individuals, a team at the Center for Games and Playable Media at University of California Santa Cruz recently undertook an in-depth study focused on an academic game developed at the university, and exhaustively documented the software, and processes involved in its creation (Kaltman et al. 2014).

The notion of 'significant properties' is a particularly noteworthy paradigm for the preservation of complex digital objects. I outlined in the introduction some of the many properties which might characterise a software-based artwork. It is important to capture these if one is to make an informed judgement regarding the components and behaviours that should be recorded and/or preserved to ensure the artwork remains

accessible and meaningful (Knight 2009). A number of suggested groups or categorisations of significant properties exist (Matthews et al. 2008, Dipple et al. 2010) but may not necessarily be suitable for every scenario. While this correctly implies that the assessment of certain properties requires human judgement, particularly those regarding user interaction or aesthetic qualities, there exist on the technical side a number of tools from other domains which could be of considerable use in the semi-automated identification of these properties. Tools of particular interest are those that can identify the properties of the software without needing to manually analyse code. For example, network traffic analysis tools could be used to monitor a programs network activity. You could thus identify when and how the software is communicating with the internet. File profiling tools can generate a huge amount of different kinds of metadata and other information through the monitoring of a program and its environment. PERICLES, a large scale EU-funded digital preservation project has developed a tool that brings together a number of open source and free libraries available for runtime and snapshot analysis of software (see Fig. 1). Currently known as the PERICLES Extraction Tool (or 'PET'), this piece of software has potential uses as a pragmatic approach to identifying and gathering information about significant properties (Corubolo 2014).

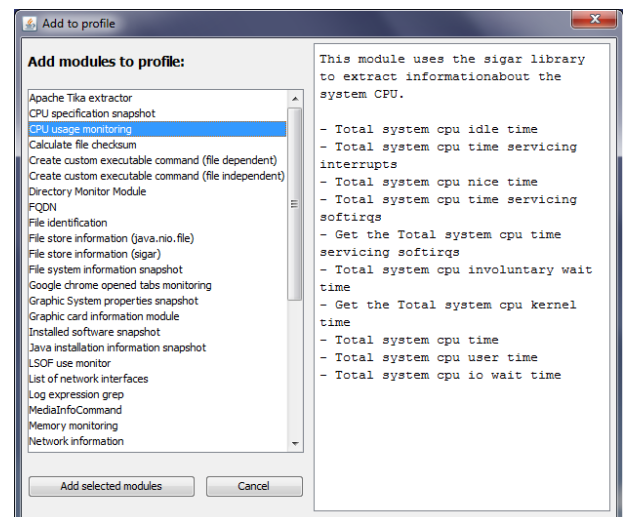


Figure 1: Screenshot of module selection screen in the PERICLES Extraction Tool (PET)

Through isolating a set of significance properties, the task of identifying focal areas for preservation action becomes more manageable, and a meaningful assessment of risk can be made and acted upon. With that in mind, I shall now examine some of these potential actions and their value in the context of software-based art.

3.2. Preparing for and responding to change

When it comes to preparing for the long-term, or responding to a changing environment, there are three core digital preservation strategies to consider: emulation, virtualisation (or encapsulation), and restaging. I shall explore each of these further below, focusing on the extent to which they are feasible to apply on a small scale. It is worth stating here that preservation with a long term goal is not a one-time activity, and even with appropriate preparation, ongoing monitoring is essential for long-term security. Ideally a collection should be manually checked as often as every six months, testing each artwork and responding to technological change (or failure) as appropriate.

This first is emulation, which involves the use of software to mimic the original environment of the artwork. While the close approximation of original behaviour that this allows is ideal for software-based art, the creation of the software to achieve it can be very challenging, and brings with it its own set of future maintenance problems. While it is easy to dismiss on those grounds, emulation does benefit from a considerable community of talented hobbyists, largely focused around the development of emulators for vanished gaming systems and consoles. There is then, the possibility that the outdated platform required might be available for a current technological environment in an emulated form. A successful example of emulation in action can be seen in JODI's *Jet Set Willy Variation's*, originally created in 1984 for the Sinclair ZX Spectrum and now running in any modern web browser via Java (see Fig. 2). In this case, the artists repurposed a community developed ZX emulator. Adang (2013) presents another interesting emulation case study, particularly how the technique can be used to develop our historical understanding of artworks.

Virtualisation, although appearing somewhat similar to emulation in result, involves a slightly different paradigm. Rather than imitating the target system, virtualisation involves encapsulating the target system in a closed, virtual container, which is accessed as if it were real hardware. This container uses its own independent resources, allocated through the software. A virtualised artwork then, is a much simplified, portable object which carries with it some of its dependencies. Unlike emulation, the tools to virtualise almost any widely used computer environment are very much accessible to 'home users' (a prominent free example is Oracle's VirtualBox). Unfortunately there are limitations to performance and utilisation of real hardware. For example, graphics cards are currently inaccessible in popular virtualisation software, instead requiring an allocation of RAM which would likely prove insufficient to run a real-time 3D engine. These

limitations aside, virtualisation is a powerful tool in the right situation, and very much feasible without the resources of a large institution. For a more detailed discussion of virtualisations potential as an art conservation measure, see Lurk (2008).

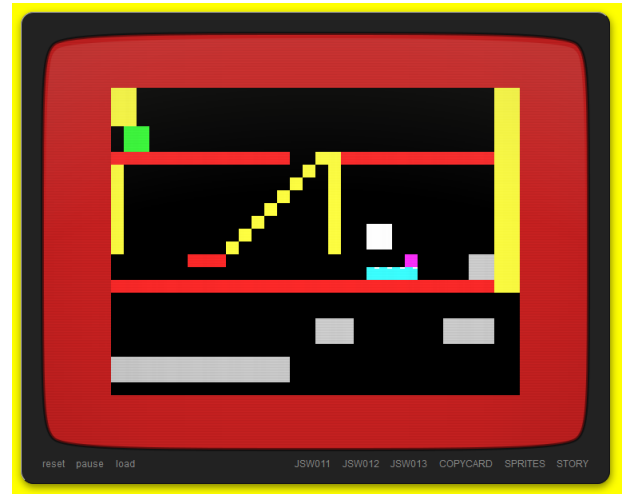


Figure 2: Screen capture of JODI's *Jet Set Willy Variation's* (1984) (<http://jetsetwilly.jodi.org>) running in Firefox 32.03

Restaging (or recreation) is the final, and perhaps most radical, preservation strategy I will discuss. Formalised by the Variable Media Initiative (Depocas et al. 2003), this strategy relies not on the integrity of the original components of the artwork, but on having the required documentation and technical information in place to enable the recreation of the artwork according to the artists original vision at a time when the original is no longer salvageable. Restaging then, bears some relation to process of migration (the move from one file format to another) which is more common among collections of relatively simple digital objects (e.g. digitised document archives). It also has very obvious parallels with other varieties of performed creative work such as performance art, theatrical productions and music. However it is radically different from traditional notions of collecting static, shelve-and-forget objects, and the resulting questions of authenticity mean that the notion of restaging remains controversial (Laurenson 2006). Leaving those aside, the success of this strategy is largely reliant on the depth of documentation generated for the artwork (see Section 3.1), particularly the availability of source code. Additionally, the availability of tools to carry out reverse engineering of software (e.g. decompilers, which can return a piece of software to its constituent components) means that a media archaeology approach has great potential value too. While restaging is in theory the most labour intensive strategy of those discussed here, it is worth bearing in mind that this approach – in the present – is rooted in the creation of

documentation. The more practical process of restaging is a longer-term prospect, and may well be considered in an environment of differing priorities and resources.

4. CONCLUSION

In a shift away from conservation efforts focused on the integrity of the original object, over the past decade we have seen the media art conservation community move toward an acceptance of ephemerality and a focus on documentation as the crux of good conservation practise. This trend is likely to continue, and we can hope to see more concrete standards of best practise emerge for the creation of such documentation. As large scale, institutionally focused digital preservation projects forge ahead, it is important that an exchange between these and the enthusiast community is encouraged. Indeed, some recent research (Rechert et al. 2013) explicitly proposes community driven approaches to preserving complex digital objects, which combined with growing interest in open sourcing artwork (Dekker 2014) could see a dramatic change in the way software-based art is collected and conserved. The current time presents great opportunities alongside the challenges I have discussed in this paper, and the promise of a world in which anyone with an internet connection and some time on their hands can start collecting art is undeniably exciting. That – with the right care – such collections might be able to live on into the future and contribute to our shared cultural heritage, is more exciting still.

5. REFERENCES

- Adang, L. (2013) *Untitled Project: A Cross Disciplinary Investigation of JODI's Untitled Game*. Rhizome.org, New York, USA.
- Corubolo, F., Eggers, A.G., Hasan, A., Hedges, M., Waddington, S., Ludwig, J. (2014) *A pragmatic approach to significant environment information collection to support object reuse*. Presented at iPRES 2014, Melbourne, Australia.
- Dekker, A. (2014) *Enabling the Future, or How to Survive FOREVER1: A study of networks, processes and ambiguity in net art and the need for an expanded practice of conservation* (PhD thesis).
- Depocas, A., Ippolito, J., & Jones, C. (2003) *Permanence through change: The variable media approach*. Guggenheim Museum Publ., New York.
- Dipple, K., Laurenson, P., & Fadenza-Rodrigues, F. (2010) *Describing Networked Art for the Purpose of Documentation and Conservation*. Tate, London, UK.
- Fino-Radin, B. (2011) *Digital Preservation Practices and the Rhizome Artbase*. Rhizome.org, New York, USA.
- Kaltman, E., Wardrip-Fruin, N., Lowood, H., & Caldwell, C. (2014) *A Unified Approach to Preserving Cultural Software Objects and their Development Histories*. Center for Games and Playable Media, University of California, Santa Cruz, USA.
- Knight, G. (2009) *InSPECT – Framework Report – Investigating Significant Properties of Electronic Content*.
<http://www.significantproperties.org.uk/inspect-finalreport.pdf> (Retrieved 18 March 2015).
- Laurenson, P. (2006) Authenticity, change and loss in the conservation of time-based media installations. *Tate Papers*, Issue 6.
- Library of Congress (2013). *Perspectives on Personal Digital Archiving*.
http://www.digitalpreservation.gov/documents/ebookpdf_march18.pdf (retrieved 19 March 2015).
- Lurk, T. (2011) *Checklist for recording and documentation of digital artworks. Computer and software based artworks or installations with computer controlled elements*. Bern University of the Arts, Bern, Switzerland.
- Lurk, T. (2008) Virtualisation as conservation measure. *Archiving Conference 2008(1)*. Society for Imaging Science and Technology.
- Matthews, B., McIlwrath, B., Giaretta, D., & Conway, E. (2008) *The Significant Properties of Software: A Study*. JISC report, UK.
- McDonough, J.P., Olendorf, R., Kirschenbaum, M., Kraus, K., Reside, D., Donahue, R., Phelps, A., Egert, C., Lowood, H., & Rojo, S. (2010) *Preserving Virtual Worlds Final Report*.
<http://hdl.handle.net/2142/17097> (retrieved 19 March 2015).
- Rechert, K., Espenschied, D., Valizada, I., Liebetraut, T., Russler, N., & von Suchodoletz, D. (2013). An Architecture for Community-Based Curation and Presentation of Complex Digital Objects. *Digital Libraries: Social Media and Community Networks*. pp. 103–112. Springer International Publishing.
- Rinehart, R. & Ippolito, J. (2014) *Re-collection: Art, new media, and social memory*. MIT Press, Cambridge, Massachusetts.