Preserving Interactive Content: Strategies, Significant Properties and Automatic Testing

Mark Guttenbrunner

Vienna University of Technology, Vienna, Austria
http://www.ifs.tuwien.ac.at/dp
guttenbrunner@ifs.tuwien.ac.at

Abstract. Preserving digital data is becoming a more important issue with the growing amount of data that only exists in digital form. But not only static documents that can have representations in paper or analog tape have to be kept accessible, also interactive forms of data have to be preserved. Interactive fiction, also known as video games, and digital art are just two forms of interactive content that have to be kept alive for historic reasons. Business and scientific applications have to be preserved as well to have data in databases or in documents which can't be altered for authenticity reasons accessible over a long term. But also documents can contain input fields and interaction ranging from simple formulas to full-fledged interactive 3D-animations and beyond. Emulation is one of the strategies to keep applications for old hardware usable on modern systems. As it is not the only strategy to deal with the digital preservation of complex content, we have to be able to evaluate the different preservation strategies.
This paper identifies significant properties of the different types of interactive objects that are analyzed in order to determine optimal preservation solutions. Then various migration and emulation strategies for preserving complex objects over a long term are shown. We discuss methods to automatically test alternatives also by utilizing characterization languages. Then we outline what further steps have to be taken to verify the significant properties and define features that have to be extractable from environments to support automatic testing.

1 Introduction

Digital preservation is a pressing issue for all kinds of documents. But not only static documents have to be considered, complex interactive content has to be preserved for future use as well. Interactive content comes in various forms, some of which are obvious like video games or interactive art. But also application software, dynamic documents and web-pages present some form of interactive content.

For evaluating the suitability of a digital preservation alternative for complex interactive objects the significant properties of an object have to be identified

and analyzed first. The significant properties of static documents usually differ from those of dynamic and interactive content. While the appearance of the first makes it often possible to migrate the contents to other formats, the task is more complex for interactive content. Potential loss has to be investigated very closely, as for example loss of interaction can render a digital art object completely useless. Visual and audible properties as well as interaction with the object have to be preserved. While emulation might look as an obvious choice, other strategies have to be considered as well. Even with the same significant properties for different types of complex content the weighting of importance of these properties for preservation can be different depending on the type and the designated user community. To support automatic testing of alternatives in the process of preservation planning the significant properties have to be stored in a standardized form.

This article is structured as follows. First an overview of related work is given in Section 2. Then we categorize the different types of interactive content and discuss the challenges that are associated with them in Section 3. A discussion on significant properties of the various categories is done in Section 4. In Section 5 we discuss some possible strategies to preserve the identified properties. Section 6 presents some ideas on the automatic testing of alternatives and finally in Section 7 we discuss the results and give an outlook on future work to be done.

2 Related Work

Most digital preservation projects in the past concentrated on the preservation of static documents. One approach to preserve complex multimedia art was done by the Guggenheim museum with the *Variable Media Initiative*. One outcome was the *Variable Media Questionnaire*, a questionnaire for artists and collectors of art which included descriptive elements needed for recreating the artwork. The research concluded in the *Variable Media Network*[1]. The variable media paradigm lets the artists choose between different strategies for preserving their art. The available options are storage, emulation, migration and reinterpretation. Hunter et. al. describe in [6] an approach to use a combination of emulation or migration and the use of metadata for describing the digital object. With the PANIC[2] project a prototype of a web service based digital preservation tool for semi-automatic preservation of complex multimedia objects is presented in [7]. The PANIC project concentrates on objects composed from different content and does not focus on interactive objects.

The term *Emulation* refers to the capability of a device or software to replicate the behavior of a different device or software. It is possible to use hardware to emulate hardware or to use software to emulate software. In this article the meaning of *Emulator* is used as defined in [11] for a program that virtually recreates a different system then the one it is running on.

[1] http://variablemedia.net/
[2] http://www.metadata.net/panic/

The concept of using emulation for digital preservation is to keep the data in its original, unaltered form and keep using the software originally used to display the data. This software has to be run on the operating system and the operating system on the hardware it was developed for. To keep this chain alive, an emulator for the original hardware is produced. Methods to establish emulation as a long term strategy for digital preservation using chaining, rehosting and Virtual Machines are discussed in [11].

An approach for a Universal Virtual Machine is the *Universal Virtual Computer (UVC)* by IBM ([5]). It introduces a virtual machine which is simple enough that it can be easily implemented but still sufficient for preserving digital data. The digital data is stored together with a program that can be run on the UVC but that is written when the data is archived. On a future system where the data will be restored an implementation of the UVC is hosted and the program which was stored with the data is used to extract the data from the record. A proof of concept for this approach has been done on the archiving of PDF-documents.

Emulation can take place on different levels (software, operating system or hardware) ([10]). The most accurate approach to the emulation of console video games is most probably the emulation on a hardware level. Especially video games released later in the life cycle of a video game console are using system specific features on a very low hardware level to produce the best results with the then very well known system behavior. Video game consoles do not differ in the used hardware components like personal computers, so programmers can use very time restrained code and optimize the results to the specific system. Emulating a video game console on a different level than hardware would result in low compatibility for the game software. This approach uses software to reproduce the characteristics of hardware components and is not to be confused with emulating hardware using different hardware, which would not solve the digital preservation problems outlined before.

An approach to developing an emulator on a hardware level is discussed as a conceptual model in [13] as *modular emulation*. It suggests the use of a modular emulator which uses a component library and an emulator configuration document to bind the different components for a specific system to an emulated system. A component for which an emulator once has been written can be used for other systems using the same component. The ideal case is to write a new emulator by creating a configuration file and using all the existing components. The modular emulator is run on a Universal Virtual Machine (UVM) as suggested before. A controller program is used to start the UVM and loads the required components and the emulator.

An emulator which uses the modular emulation approach named *Dioscuri*[3] is currently under development. It is currently able to emulate a machine based on the Intel 80x86 processor with basic input/output facilities. Dioscuri is written in Java and runs on the Java Virtual Machine. The goal of the project is a version

[3] http://dioscuri.sourceforge.net/

of Dioscuri that is able to replace a reference workstation running Microsoft Windows 2000 ([9]).

A practical experiment on how to use emulation to recreate interactive art is presented in [8]. The original piece of art called *The Erl King* (1982-85) by Grahame Weinbren and Roberta Friedman consisted of obsolete and generic hardware and software. It presented itself as an ideal candidate for an emulation project as the original software was written by the artist, so it was a very high priority to preserve the original code.

Becker et. al. present in [2] case studies on sample objects of interactive multimedia art from the collection of the Ars Electronica[4]. The PLANETS preservation planning approach is used for evaluating applicable digital preservation strategies for interactive multimedia objects. In [4] a case study on preserving console video games was done to evaluate existing emulators for their suitability as digital preservation alternatives. Both case studies identify significant properties of complex content.

A generic language for characterizing objects and describing their properties can be found in [3] as *XCDL* (extensible characterization definition language). Another language developed by IBM is called DFDL (data format description language) [1].

3 Types of Interactive Content

This section describes four types of interactive content. The boundaries are not fixed, as it is sometimes difficult to decide what category an object belongs to (e.g. video games vs. games produced as digital art, application vs. interactive document which themselves can be seen as applications again).

3.1 Application Software

Preserving application software can be necessary for various reasons: Using original documents in their unaltered form for authenticity, running scientific software to reproduce experimental results, historical reasons, access to data in databases. Most of these reasons require interactivity. Another issue are applications distributed over the network, as all components and the interaction between them have to be preserved.

3.2 Dynamic Documents

Documents not only contain data, but logic in some form as well. This can range from simple format qualifiers to complex programs with input-fields for users and calculated graphs. Examples of dynamic documents include web pages using Java-script, spreadsheet documents using formulas, PDF-documents with interactive 3D Animations. For some applications it might be possible to convert the document to a static format comparable to a print-out while for others the interactive elements have to be preserved.

[4] http://www.aec.at

3.3 Video Games

From simple games played on early gaming hardware to virtual on-line worlds the requirements to preserve video games differ essentially. Legal issues, unavailable source code and proprietary user interface hardware are only some of the challenges when dealing with interactive fiction. Preserving the non-technical requirements like the feel-aspect or the environment of a playing-experience (e.g. arcade-games in a bar) have to be considered as well. They are a big part of experiencing a game the way it was supposed to be. A great variety of platforms (PC, game consoles, arcade games, mobile telephones, on-line-games) have to be considered when trying to preserve video games.

3.4 Interactive Digital Art

Digital art can contain interactive components just like video games. Additionally it can be implemented on proprietary hardware or even have hardware as part of the art. Preserving interactive digital art can normally only be done on a per piece basis. Art is usually done in a specific context, so this has to be kept in mind when trying to preserve art as well.

4 Significant Properties of Interactive Content

To evaluate a digital preservation alternative it is necessary to know the properties of an object which are significant and which have to be preserved. These properties can be technical as well as social properties. Depending on the type of object and the designated use the weighting of the importance of meeting specific requirements can be different.

Significant properties of all kinds of software include visual and audible properties. All kinds of interactive input possibilities have to be considered. In case of application software and dynamic documents these are e.g. form fields, icons, menus and mouse and keyboard for input. For video games and digital art this can be menus, icons on the user interface, the response and support of hardware like gaming hardware, video cameras, sensors, motion detectors and mouse and keyboard again.

Functionality is an important part of software preservation. In case of applications or documents it means accessing the data while with video games and digital art the playing experience, response to input and audible/visual characteristics are important.

4.1 Application Software

For application software there usually is more weighting on the functionality than on audible or visible characteristics. The original look and feel is less important then being able to access the data. One exception to this rule is the preservation of application software for historical reasons.

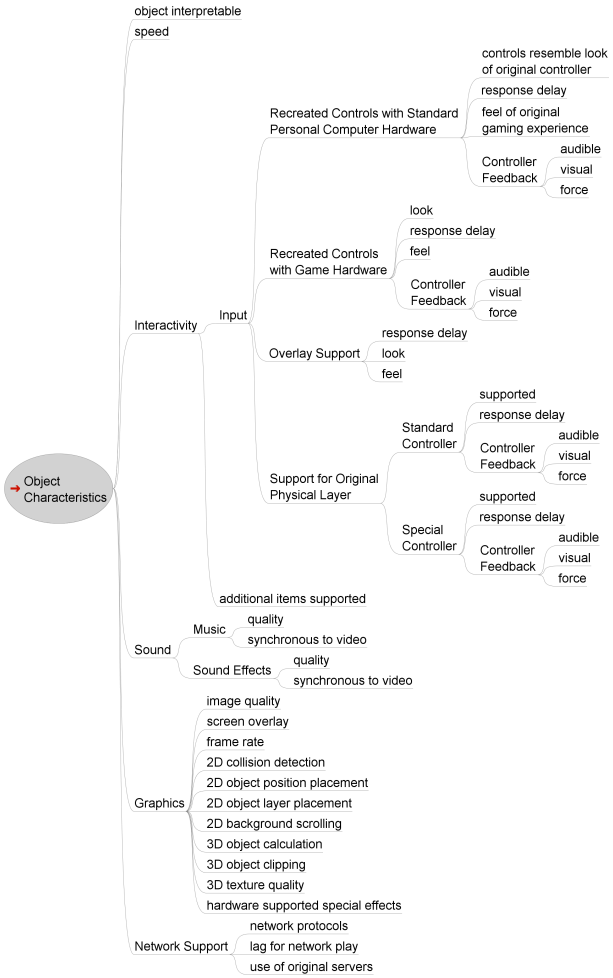

Fig. 1. Significant object properties for console video games [4]

4.2 Dynamic Documents

In dynamic documents the contents and appearance can be important properties with functionality on the original data as well. A good example would be a spreadsheet with a graph thats calculated by using data in the spreadsheet taking current date into account. Of course it also depends on the application of the spreadsheet, if the functionality should always use the current date or if the date should be frozen on ingest reflecting data at that point in time.

4.3 Video Games

Besides visual and audible characteristics the original speed of the game and response to input is very important to re-create the original feel of video games.

But also social aspects have to be considered. Playing a game feels a lot different using keyboard and LCD-screen in the office than playing it in a smoke-filled bar on an arcade machine with controls where one can forcefully push buttons. Another thing to be considered for video games is the fact that the gaming experience not necessarily relies on the original graphics and sounds. Updated versions of games for modern hardware might still bring back the original feeling of playing a game without the feel that the game looks dated. A subtree of significant object properties for console video games as developed in a case study in [4] is shown in Figure 1

4.4 Digital Art

For digital art visual and audible characteristics are important. If a piece of art uses analog and digital material and sensors, then the synchronous interaction between this parts has to be preserved. Methods of interaction with the object have to be preserved as well.

5 Strategies

The two main strategies of digital preservation are migration and emulation. With complex content various forms of these strategies are possible:

Source Ports Migrating the digital object to a different system by re-compiling the source code on that platform. This preservation action has to be performed for every object and can only be done when the source code is available. Complexity and effort are very high. This is a potential strategy for all kinds of complex content.

Simulation If source code is not available software can be reimplemented either by using design documents or by re-engineering the original behavior. Like source ports this is a time consuming alternative and a difficult task for complex software. One possible application of this strategy is interactive digital art.

Video Approach Migrating the visual and audible characteristics to video by filming a digital object. While all the interactivity to the object is lost, this strategy might gives a very good idea about the digital object. As it is very inexpensive it is a good alternative for applications with no importance on interaction to the object.

Database Preservation If the interactivity of software does not have to be preserved and data and application logic are strictly separated, the database content can be stored in a standardized format over a long term for either simple querying or re-import to a database.

Low-Level Emulation Emulating a system on a hardware level makes all applications for this system usable by developing only one piece of software. This strategy preserves all the interactivity. It can be a long-term strategy by making sure the emulator can be preserved as well for example by developing it for an emulation virtual machine. A big disadvantage of emulation is the required knowledge about using the original system.

6 Testing

With various possible strategies for the preservation of interactive content it is necessary to evaluate which alternative should be used for a specific digital preservation situation. One way to compare alternatives is the Planets preservation planning approach described in [12]. It uses requirements trees to identify how well significant properties are met by a specific alternative.

To allow tools that automate parts of the preservation planning process to determine these figures, characterization languages can be used. While tools are able to examine migrated documents, this is more difficult for interactive objects.

Below are two methods to test interactive environments:

- The original visual and audible features are recorded in a digital video format. Interaction (e.g. mouse movement and clicks) are recorded and replayed using a preservation alternative. If no random elements appear (like e.g. in video games) the resulting visual and audible features can be compared. Frame count differences and image resolution can easily be extracted.
- Another way to test environments is having the environment extract features. By defining the significant properties and identifying the properties that can be extracted, this feature could be implemented in e.g. emulators. Possible properties are CPU-cycles per second, frames per second, resolution, information about recorded interaction. To simulate the interaction of a user a recorded user input macro has to be provided and executed by the tool.

7 Discussion and Outlook

In this article the types of interactive complex content were described along with challenges for their preservation for a long term. We defined some significant properties and showed the difference between the types. It showed that common significant properties do exist and provided possible strategies for preserving them.

Describing the significant properties in a generic language allows the use of comparator tools to automatically compare different alternatives. While most technical significant properties can probably extracted either from the tools used for emulation or from running the migrated software on a system, social significant properties have to be manually evaluated. By defining technical properties now, support for automatic extraction can be implemented.

Future work in refining and verifying the significant properties in case studies with a number of alternatives has to be done. These properties then have to be represented in one or more languages for characterization. For this purpose it could be necessary to extend an existing characterization language. Tools for automatic testing of emulators have to be written. Features that have to be extracted from emulation environments have to be defined, so that emulators can provide support for extraction of those to aid automatic comparison.

Acknowledgements

Part of this work was supported by the European Union in the 6th Framework Program, IST, through the PLANETS project, contract 033789.

References

1. Beardsmore, A. Schema description for arbitrary data formats with the data format description language. *Enterprise Interoperability II* (2007), 829–840.
2. Becker, C., Kolar, G., Küng, J., and Rauber, A. Preserving interactive multimedia art: A case study in preservation planning. In *Proceedings of the Tenth International Conference on Asian Digital Libraries* (2007), p. (accepted for publication).
3. Becker, C., Rauber, A., Heydegger, V., Schnasse, J., and Thaller, M. A generic XML language for characterising objects to support digital preservation. In *Proc. 23rd Annual ACM Symposium on Applied Computing (SAC'08)* (Fortaleza, Brazil, March 16-20 2008), vol. 1, ACM, pp. 402–406.
4. Guttenbrunner, M., Becker, C., and Rauber, A. Evaluating strategies for the preservation of console video games. In *Proceedings of the Fifth international Conference on Preservation of Digital Objects (iPRES 2008)* (London, UK, September 2008).
5. Hoeven, J., Van Der Diessen, R., and Van En Meer, K. Development of a universal virtual computer (UVC) for long-term preservation of digital objects. *Journal of Information Science Vol. 31 (3)* (2005), 196–208.
6. Hunter, J., and Choudhury, S. Implementing preservation strategies for complex multimedia objects. In *The Seventh European Conference on Research and Advanced Technology for Digital Libraries* (2003), pp. 473–486.
7. Hunter, J., and Choudhury, S. PANIC: an integrated approach to the preservation of composite digital objects using semantic web services. *International Journal on Digital Libraries 6*, 2 (April 2006), 174–183.
8. Jones, C. Seeing double: Emulation in theory and practice. the erl king case study. *Electronic Media Group, Annual Meeting of the American Institute for Conservation of Historic and Artistic Works* (Variable Media Network, Solomon R.Guggenheim Museum, 2004).
9. KB nationaal archief. Dioscure - digital preservation. Online in Internet, 2007. `http://dioscuri.sourceforge.net/preservation.html` (accessed at September 24, 2008).
10. Rothenberg, J. *Using Emulation to Preserve Digital Documents.* Koninklijke Bibliotheek, 2000.
11. Slats, J. Emulation: Context and current status. Online in Internet, 2003. `http://www.digitaleduurzaamheid.nl/bibliotheek/docs/white_paper_emulatie_EN.pdf` (accessed at September 24, 2008).
12. Strodl, S., Becker, C., Neumayer, R., and Rauber, A. How to choose a digital preservation strategy: Evaluating a preservation planning procedure. In *Proceedings of the 7th ACM IEEE Joint Conference on Digital Libraries (JCDL'07)* (June 2007), pp. 29–38.
13. van der Hoeven, J., and van Wijngaarden, H. Modular emulation as a long-term preservation strategy for digital objects. In *5th International Web Archiving Workshop (IWAW05)* (2005).