Project Number	IST-2006-033789
Project Title	Planets
Title of Deliverable	Preservation of Dynamic & Interactive Content by use of Binary Translation and Virtualisation – a Methodology for Experimentation
Deliverable Number	TB/6-D1F (PA/6-D12)
Contributing Sub-project and Work-package	TB/6, PA/6
Deliverable Dissemination Level	External
Deliverable Nature	Final Report
Contractual Delivery Date	31st March 2010
Actual Delivery Date	6th April 2010
Author(s)	HATII

Abstract

This report presents a study that moves beyond migration and emulation, by exploring the possibilities of binary translation and virtualisation to preserve dynamic and interactive content. The Planets community has generated a range of information on planning, characterisation and strategies that can be used to address the challenges that these types of material pose to long-term digital preservation. However, experimentation with preservation solutions specifically for software artefacts has either not been extensively conducted or its focus has been narrow.
The objective of this study has been to design, conduct and analyse results from a series of experiments that utilise existing Planets infrastructure in order to investigate the preservation potential of binary translation and virtualisation. The results from the experiments contribute towards determining the viability of the proposed preservation action, by (a) measuring the quality of the results generated by the Planets Testbed; and (b) gauging the impact of binary translation and virtualisation on digital preservation in general.

Keyword list

Emulation, binary translation, virtualisation, preservation of video games, software art, experimentation, Testbed, GRATE

Contributors

Person	Role	Partner	Contribution
Leo Konstantelos	Author	HATII	

Document Approval

Person	Role	Partner

Distribution

Person	Role	Partner

Revision History

Issue	Author	Date	Description
0.1	HATII	31/03/2010	Initial draft version for internal review
0.2	HATII	25/05/2010	PA/6 part added

References

Ref	Document	Date	Details and Version
1	McHugh, A.; Konstantelos, L. & Barr, M. *Emerging digital art characterisation techniques.*	2010	Planets report PC/5-D5, available from: http://www.planets-project.eu/docs/reports/Digital_Art_Characterisation_Techniques.pdf (Retrieved January 2010)
2	Becker, C.; Kolar, G.; Kung, J. & Rauber, A. *Preserving Interactive Multimedia Art: a Case Study in Preservation Planning.*	2007	Paper presented at the Tenth International Conference on Asian Digital Libraries, ICADL 2007, 10-13 December 2007, Hanoi, Vietnam. Available from: http://www.planets-project.eu/docs/papers/becker_icadl07.pdf (Retrieved October 2009)
3	Guttenbrunner, M.; Becker, C.; Rauber, A. & Kehrberg, C. *Evaluating Strategies for the Preservation of Console Video Games*	2008	Proceedings of the Fifth international Conference on Preservation of Digital Objects (iPRES 2008). London, UK, September 29-30, 2008. Available from: http://www.ifs.tuwien.ac.at/~becker/pubs/guttenbrunner_ipres08.pdf (Retrieved September 2009)
4	Sites, R. L., Chernoff, A., Kirk, M. B., *et al.* Binary Translation.	1992	Digital Technical Journal, 4(4), pp. 137-153.
5	Binary Translation	n.d.	Wikipedia. Retrieved August 2009 from: http://en.wikipedia.org/wiki/Binary_translation
6	Altman, E. R., Kaeli, D., & Sheffer, Y. Cover - Welcome to the Opportunities of Binary Translation.	2000	*Computer*, *33*(3), pp. 40-45.
7	Altman, E. R., Ebcioglu, K., Gschwind, M. and Sathaye, S. Advances and Future Challenges in Binary Translation and Optimization.	2001	*Proceedings of the IEEE, 89*(11), pp. 1710-1722.
8	Heinz, T. Preserving temporal behaviour of legacy real-time software across static binary translation.	2008	Proceedings of the 1st workshop on Isolation and integration in embedded systems, 1 April 2008, Glasgow, Scotland (pp. 1-4).
9	Ross, S., & Gow, A. *Digital archaeology: Rescuing neglected and damaged data resources : a JISC/NPO study with the Electronic Libraries (eLib) Programme on the preservation of electronic*	1999	Electronic libraries programme studies, 2. London: Library Information Technology Centre.

	materials.		
10	Kemp, Y. Use of Virtualization Techniques for data preservation and long term analysis	2009	Proceedings of the DESY 2009 Workshop, 26-28 January, Hamburg, Germany. Available from DPHEP: https://www.dphep.org/sites/site_dphep/content/e20/e36625/e43750/dlpta.pdf (Retrieved September 2009)
11	Virtualization	n.d	Wikipedia. Retrieved August 2009 from: http://en.wikipedia.org/wiki/Virtualization
12	Rose, R. *Survey of System Virtualisation techniques.*	2004	Retrieved September 2009, from: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.58.9811&rep=rep1&type=pdf
13	Huang, W; Liu, J.; Abali, B. & Panda, D. K. *A Case for High Performance Computing with Virtual Machines.*	2006	Proceedings of the 20th annual international conference on Supercomputing, 28-30 June, Cairns, Queensland, Australia (pp. 125-134).
14	Application Virtualization.	n.d.	Wikipedia. Retrieved August 2009 from: http://en.wikipedia.org/wiki/Application_virtualization
15	Portable application.	n.d.	Wikipedia. Retrieved August 2009 from: http://en.wikipedia.org/wiki/Portable_application
16	DCC; JISC; CCLRC & BL. *Digital Curation and Preservation: Defining the Research Agenda for the Next Decade.*	2005	Event report for the Warwick Workshop, 7-8 November 2005, Scarman House, University of Warwick. Available from: http://www.dcc.ac.uk/events/warwick_2005/Warwick_Workshop_report.pdf (Retrieved September 2009)
17	Giaretta, D. The CASPAR Approach to Digital Preservation.	2007	*International Journal of Digital Curation, 1*(2), 112-121.
18	Adams, K. & Agesen, O. *A comparison of software and hardware techniques for x86 virtualization.*	2006	Proceedings of the 12th international conference on Architectural support for programming languages and operating systems, 21-25 October, San Jose, California (pp. 2-13).
19	Bogaarts, J.; van den Dobbelsteen, M; Helwig, P. & Puhl, J. *Corpora Report Planets TB.*	2008	Planets Report TB/3-D7 available from: http://www.planets-project.eu/private/pages/wiki/images/4/45/TB3_D7_Corpora_Report_v_3.2.pdf (Retrieved: July 2009)
20	Schlarb, S.; Jackson, A. N.; Kaiser, M. & Lindley, A. *The Planets Testbed: a Collaborative Environment for Experimentation in Digital Preservation.*	2009	Proceedings of the 2009 International Conference of Cultural Heritage Online, 15-16 December, Florence, Italy. Available from: http://rinascimento-digitale.com/eventi/conference2009/proceedings-2009/schlarb.pdf (Retrieved: January 2010)
21	Aitken, B.; Helwig, P.; Jackson, A. *et al.* The Planets Testbed: Science for Digital Preservation.	2008	*code{4}lib Journal, 3,* June 2008. Available from: http://journal.code4lib.org/articles/83 (Retrieved August 2009)
22	Becker, C.; Kulovits, H.; Hofman, H.; Rauber, A. *Plato: a Service Oriented Decision Support System for Preservation Planning.*	2008	Proceedings of JCDL '08, 16-20 June, Pittsburgh, Pennsylvania.
23	Hoyle, D., & Hoyle, D. *SO 9000 quality systems handbook, fifth edition*	2006	Oxford, [U.K.]: Butterworth-Heinemann.
24	Kim, Y. Methodology for Designing and Evaluating Curation and Preservation Experiments.	2008	DCC Publication. Available from: http://www.dcc.ac.uk/docs/publications/TestBedMethodV1.1.pdf (Retrieved: January 2010)
25	Jones, S.; McCann, P. & Kim, Y. A practical guide to implementing the DCC testbed methodology	2008	DCC Publication. Available from: http://www.dcc.ac.uk/docs/publications/TestbedMethodologyManual_1.2.pdf (Retrieved: January 2010)
26	QEmu	n.d	Wikipedia. Available from: http://en.wikipedia.org/wiki/QEMU

27	Welte, T et al. Report on the First Version of GRATE.	2009	Planets Report PA/5-D7 available from: www.planets-project.eu/docs/reports/Planets_PA5-D7_GRATE.pdf
28	VMWare Workstation	n.d.	Wikipedia. Available from: http://en.wikipedia.org/wiki/VMware_Workstation
29	Matthews, B.; McIlwrath, B.; Giaretta, D. & Conway, E. The Significant Properties of Software: A Study	2008	JISC Report. Available from: http://www.jisc.ac.uk/media/documents/programmes/preservation/vector_images.pdf (Retrieved: February 2010)
30	Zabolitzky, J. G. Preserving Software: Why and How.	2002	*Iterations: An Interdisciplinary Journal of Software History, 1*, September 13, 2002, 1-8.
31	Winget, M. & Murray, C. Collecting and preserving videogames and their related materials: A review of current practice, game-related archives and research projects.	2009	*Proceedings of the American Society for Information Science and Technology, 45*(1), 1-9.
32	Software Architecture	n.d.	Wikipedia. Available from: http://en.wikipedia.org/wiki/Software_architecture
33	Van der Hoeven, J. Emulation for digital preservation in practice: the results.	2007	Presentation at the iPress 2007, Beijing, China. Available from: http://www.planets-project.eu/docs/presentations/Planets_iPRES_jrvanderhoeven_emulation.pdf
34	Woodyard, D. Digital Preservation: The Australian Experience.	2000	Proceedings of the 3rd Digital Library Conference on: Positioning the Fountain of Knowledge, Malaysia. Available from: http://www.nla.gov.au/nla/staffpaper/dw001004.html (Accessed February 2010)
35	Russel, K. Digital Preservation and the CEDARS Project Experience.	2000	Proceedings of the International Conference on Preservation and
36	Software Art	n.d.	Wikipedia. Available from: http://en.wikipedia.org/wiki/Software_art
37	Art game	n.d.	Wikipedia. Available from: http://en.wikipedia.org/wiki/Art_game
38	Waters, D. J. *Middle East Digital Libraries: Some Principles for Development.*	2006	Retrieved October 16, 2009, from Middle East Digital Library Workshop. Available from: http://www.sis.pitt.edu/~egyptdlw/papers/Donald_Waters.html
39	Abandonware	n.d.	Wikipedia. Available from: http://en.wikipedia.org/wiki/Abandonware
40	Rothenberg, J. *An experiment in using emulation to preserve digital publications.*	2000	Den Haag : Koninklijke Bibliotheek. Available from: http://nedlib.kb.nl/results/NEDLIBemulation.pdf (Accessed February 2010)
41	Granger, S. Emulation as a Digital Preservation Strategy.	2000	*D-Lib Magazine, 6*(10). Retrieved February 2010 from: http://www.dlib.org/dlib/october00/granger/10granger.html
42	Wheatley, P. Digital Preservation and BBC Domesday.	2004	Paper presented at the Electronic Media Group, Annual Meeting of the American Institute for Conservation of Historic and Artistic Works, Portland, Oregon, June 14, 2004. Available from: http://www.imappreserve.org/pdfs/educ-past_conference/wheatley-emg2004.pdf (Accessed February 2010)
43	Granger, S. Digital Preservation & Emulation: from theory to practice.	2001	Paper presented at the International Cultural Heritage Informatics Meeting (ichim), Milan, Italy, September 3-7 2001. Available from: http://www.archimuse.com/publishing/ichim01_vol2/granger.pdf (Accessed February 2010)
44	Van der Hoeven, J.; Lohman, B. & Verdegem, R. Emulation in Digital Preservation in Practice	2007	*The International Journal of Digital Curation, 2*(2), 123-132.
45	Text-based games	n.d.	Wikipedia. Available from: http://en.wikipedia.org/wiki/Text-based_game
46	Winget M.A., & Murray C. Collecting and preserving videogames and their related materials: A review of current practice, game-related archives and research projects.	2008	*Proceedings of the ASIST Annual Meeting.* 45.

47	DOSBox	n.d.	Wikipedia. Available from: http://en.wikipedia.org/wiki/Dosbox
48	Wine (software)	n.d.	Wikipedia. Available from: http://en.wikipedia.org/wiki/Wine_%28software%29
49	DOSEMU	n.d.	Wikipedia. Available from: http://en.wikipedia.org/wiki/Dosemu
50	Scheib, V. Demos Explained; What are Demos? What is a Demo?	n.d.	Available from: http://www.scheib.net/play/demos/what/index.html (Retrieved February 2010)
51	Demo (computer programming)	n.d.	Wikipedia. Available from: http://en.wikipedia.org/wiki/Demo_%28computer_programming%29

EXECUTIVE SUMMARY

This report presents a study that moves beyond migration and emulation, by exploring the possibilities of binary translation and virtualisation to preserve dynamic and interactive content. The Planets community has generated a range of information on planning, characterisation and strategies that can be used to address the challenges that these types of material pose to long-term digital preservation. However, experimentation with preservation solutions specifically for software artefacts has either not been extensively conducted or its focus has been narrow.

The objective of this study has been to design, conduct and analyse results from a series of experiments that utilise existing Planets infrastructure in order to investigate the preservation potential of binary translation and virtualisation. The results from the experiments contribute towards determining the viability of the proposed preservation action, by (a) measuring the quality of the results generated by the Planets Testbed; and (b) gauging the impact of binary translation and virtualisation on digital preservation in general. In order to situate the use of the Testbed in the wider software preservation arena, we conducted a series of experiments with third-party tools as well.

This study focuses on dynamic and interactive material, whose representations vary significantly and the number of examples is equally vast. It would be therefore impossible to study the entire array of possibilities within the limited scope of this work. Instead, we have decided to focus the experimentation on a specific sample which concentrates on software art and video games. These genres are representative of interactive and dynamic software applications. They are also of increasing interest to a large part of software creators and users, aligning at the same time with previous work conducted within Planets. However, it would be possible to extrapolate our findings to software of similar nature.

In this report we present the experimentation life-cycle, starting with a clarification of terminology and the study background. We then describe the scope and methodology for this study, as well as legal considerations regarding the experimental material. Particular attention is devoted to the design of the experiments and the preparation of the preservation solutions for experimentation. The findings are documented with supporting evidence from screenshots and preservation plans developed in the Planets Plato tool.

The results of the experiments testify that the Testbed is capable of handling complex interactive and dynamic artefacts, providing through GRATE a platform for users to measure the suitability of tools. The experiments have proven that the Testbed offers equivalent results to other alternatives and the advantage of a freely available, integrated, remotely accessed environment. With these in mind, we expect future iterations of the Planets Testbed and GRATE to offer the community a complete solution for preserving dynamic and interactive content.

TABLE OF CONTENTS

LIST OF FIGURES

1. Introduction

Digital preservation (DP) is the widely acknowledged theory and practice of managing digital information over time[1]. The need for digital preservation is created by the realisation that electronic material is not as persistent as analogue, paper-based records. The main reason for this lack of persistency is that the environment to create and use digital material is ever evolving. Computer systems change constantly and every change brings a new environment. Human users of technology have adapted to this change, so much so that it has become a requirement. However, digital material is not necessarily ready to embrace a new computing environment in which it is expected to look and perform as in the system it was originally created for. This is because the usability of digital information depends highly on the usability of the underlying environment that provides the necessary infrastructure to understand and render digital information. When electronic material is no longer supported by contemporary computing environments, it becomes obsolete. Without proper preservation actions, obsolete material is – in most cases – unusable.

Just as our digital documents become obsolete – think of an Ami Pro[2] text document – so does the software that was meant to render them. Software becomes obsolete when the computer architecture and/or Operating System, for which it was developed to operate, are no longer actively supported. To use the previous example, Ami Pro was designed for the Windows 3.x /DOS operating systems (OS). Neither of them is commercially available anymore, nor do they support modern computer systems. In such cases, the digital preservation community suggests *migrating* the digital objects from the older format to a new one. The Planets project has created the Planets Testbed[3] environment to facilitate the conversion process so that it matches organisational needs. For instance, it is possible to migrate a WordPerfect for MS-DOS document into a Microsoft Word 2007 or OpenOffice document. In this way, the original document becomes usable again. But what happens when we are interested in preserving the software itself and not its outputs?

There are a number of cases that this is necessary. One is custom-made software for scientific purposes. An example of this case is software created to process and visualise complex data for research in biology, chemistry or mathematics. Quite commonly, this software is created for a specific project and specific computer architecture. Since its primary purpose is to serve the needs of the project, developers often do not incorporate compatibility elements that will make the software reusable on other computer platforms. However, if its outputs should be reconstructed in the future, it is arguable whether the original software will execute on a modern computer. Similarly, it is arguable whether the data produced by a custom-made application will be readable in future computer systems. Another case for software preservation comes from the Arts and Culture Heritage domains. Examples include video games, interactive and time-based art, as well as dynamic content generated from web resources. In all these cases, the software is not meant to produce some output file – contrary to, say, a word processor. Instead, the purpose is to generate visual and auditory feedback for the user to *experience and interact with.*

This is a particularly challenging area for digital preservation. A successful preservation solution for such interactive and dynamic content must retain both the software *and* the audiovisual experience that it was originally meant to convey. In order to do so, the preservation solution must first identify the audience of the software application, its significant properties, the requirements for hardware and operating system and the level of interaction with the software's users [29]. The next step is to identify suitable tools and methods for software preservation. Migration applied to software means recreating an application to match the specifications of a new / different computing environment[4]. This is a costly and time-consuming process that does not preserve the original (since it needs to be altered) and often some characteristics are lost [29]. Hence, the digital preservation community has turned its attention to using *Emulation* as an alternative for software preservation[5]. With emulation it is possible to recreate "the original operating environment by programming future

[1] http://en.wikipedia.org/wiki/Digital_preservation

[2] Ami Pro was a word processor released in the late 1980s. It was the first full-featured word processor designed to work under the Windows 3.x operating system, pre-dating Microsoft Word. Although Ami Pro is neither actively supported nor marketed nowadays, organisations that were early adopters of digital technologies still hold documents in the Ami Pro (.sam) format. (Source: http://www.ehow.com/how_5894414_open-pro-_sam-word-2007.html)

[3] http://testbed.planets-project.eu/testbed/

[4] This process is also referred to as "porting". For more information, see: http://en.wikipedia.org/wiki/Porting

[5] http://en.wikipedia.org/wiki/Data_migration

platforms and operating systems to [duplicate] the original operating environment, so that software can be preserved in binary and run 'as is'." [29]

Within Planets, the importance of preserving dynamic and interactive digital content has been demonstrated in a number of efforts [e.g. 1, 2, 3]. From new media art to console video games, the Planets community has generated a range of information on planning, characterisation and strategies that can be used to address the challenges that these types of material pose to long-term digital preservation. A point of agreement is that the ephemeral, temporal and experiential facets of dynamic and interactive content present distinct risks to its digital persistence. However, experimentation with preservation solutions specifically for software artefacts has either not been extensively conducted or its focus has been narrow.

This report presents a study that moves beyond migration and emulation, by exploring the possibilities of binary translation and virtualisation to preserve dynamic and interactive content. This is a vast domain, so for this study we are using video games and software art as sources of experimental material. The objective of this task is to design, conduct and analyse results from a series of experiments that utilise existing Planets infrastructure in order to investigate the preservation potential of binary translation and virtualisation. The results from the experiments contribute towards determining the viability of the proposed preservation action, by:

- measuring the quality of the results generated by the Planets software and
- gauging the impact of binary translation and virtualisation on digital preservation in general.

The aims, content and goals of this report present distinct parallels with Planets work undertaken within the PA/6 Workpackage on *Emerging Preservation Action Technologies.* In particular, initial exploratory research in the state-of-the-art of Binary Translation technologies has provided input to this report, by setting the fundamental background on the scope and application of BT in a wider context. At the same time, the objectives of this report converge with the PA/6-D12 deliverable, which is meant to document binary translation experiments encompassing comparative experimentation and coordination of preservation roles, binary translation and virtualisation. Since the latter is implicit within this TB/6 report – and in order to avoid duplicating effort – the presented methodology and experimental outcomes substantially cover the requirements for PA/6-D12 and are presented here as a joint effort between the two Planets Workpackages.

The remaining of this report is structured as follows:

- Section 2 outlines the study background. We discuss software preservation in general and significant properties of software. We also provide brief definitions for Binary Translation and Virtualisation. The purpose of the definitions is to help the unacquainted reader better understand the context of this study. We have therefore refrained from providing exhaustive technical information.
- Section 3 describes the scope and methodology for this study, as well as legal considerations regarding the experimental material.
- Section 4 presents the design of the experimentation, including tools used for experimentation and the sample collection of videogames and software art.
- Section 5 focuses on the preparation of the experiments, documenting the processes for installing and configuring the preservation tools used in this study.
- Section 6 presents the results of the experiments, organised by type of material.
- Section 7 concludes this report with major findings, benefits and limitations, as well as recommendations for future work.
- Appendices A and B provide an explication of the significant properties used to evaluate the preservation potential of the examined tools.
- Appendices C and D include two complete digital preservation plans for video games and software art respectively.

2. Study Background

2.1 Software Preservation

The scope of this study is within the *digital preservation of software* domain. Although the need to preserve software has been identified in a number of occasions, the community has generally produced few studies on the subject. At present, the most prominent case is [29]. This JISC-funded study defines software preservation within four major aspects:

- Persistent storage of a software package, by use of appropriate preservation strategies to ensure safety and authenticity over time.
- Cataloguing and documentation of software packages for retrieval.
- Reconstruction of the computing environment for which the software was originally designed.
- Accurate execution of software applications, which retains the original behaviour and performance (replay).

The first two aspects reflect generic digital preservation methodologies. Reconstruction and replay are specifically related to software preservation. These issues are challenging, because software execution involves a number of *dependencies.* Such dependencies include hardware requirements – which may be obsolete by modern computing standards – demands for specific Operating System(s), as well as special configurations for both hardware and OS [29]. For example, a text document can be easily migrated to newer or alternative formats, so that it is usable on different computer architectures, OS platforms and word processing applications. However, a text processing software application will only run on a distinct OS/hardware environment. There is an expectation that the software will always execute in a consistent, reliable manner. This is further accentuated by the *inherent complexity* of software. Unlike the single text document, the word processing application will consist of many components (files). For the package to be operational, all components must be preserved. These components may include the software's code base, an executable binary file, a configuration / installation file, and documentation to describe the software [29].

There are many reasons why we are interested in tackling the above problems to preserve software. One is to ensure that historically significant software will be available to future generations as a cultural record [30]. Other reasons include: (1) maintenance of software records as output and evidence of research; (2) ability to view and analyse research data created by specialised software; and (3) the possibility to reuse existing software rather than re-develop it [29]. The question is what *kind of information* needs to be preserved so as to ensure continuing access and usability of software artefacts. The answer is in the identification of *Significant Properties* for software. A conceptual model for capturing significant properties has been presented in [29]. The identified properties are based on seven distinct characteristics of software that should be considered for preservation; namely:

- Functionality – what the system does, its inputs and outputs, its operations and the domain of focus.
- The content of components comprising the software artefact and the relationships between them
- The source of origin (provenance) and ownership of the software, which dictates intellectual property conditions and copyright restrictions.
- The prerequisites and methods of user interaction with the software – from necessary peripherals (keyboard, mouse) to user interfaces, look-and-feel and software libraries.
- The computing environment necessary for the software to execute and operate correctly. As mentioned earlier, these range from hardware requirements and peripherals to Operating Systems and configurations.
- The software architecture that defines the structure of the system. This involves software components, the externally visible properties of those components, and the relationships between them [32].

- The manner that the software is meant to perform, given specific computer resources. [29] gives as examples the speed of execution, data storage requirements, response time of input and output devices, and colour resolution capabilities.

2.2 Emulation

One could wonder at this point: Why not just keep legacy computer systems and use them to run obsolete software? This is the *computer museum* approach. Although it seems reasonable, its implications make it impractical. In order to store computers in physical locations, space is required – and space means cost for a hosting organisation. The machines will inevitably be accessible only at the physical location where stored – thus inaccessible to many potential users. Maintaining the legacy computer systems is also not an easy task – obsolete technology is hard to find if, for instance, a component needs replaced. For all these reasons, efforts in Computing Science, Information Technology and Digital Preservation have focused on emulation as a software preservation solution. The main advantage of emulation is the capabilities it offers to adapt any computer environment, so that it can render a software artefact authentically [33]. This means that the original software artefact remains unaltered yet still usable. On these grounds, many have supported the suitability of emulation for digital preservation (e.g. [34], [35]). Rothenberg [40] in particular has illustrated a conceptual view of emulation-based preservation (Figure 1). The diagram shows the relationships between the constituents of the emulation process. Rothenberg's approach separates software applications from the hardware / OS platform to eliminate dependencies (decoupling)[6]. The software can then execute on different computing environments, provided that the emulator is migrated to work with those environments.

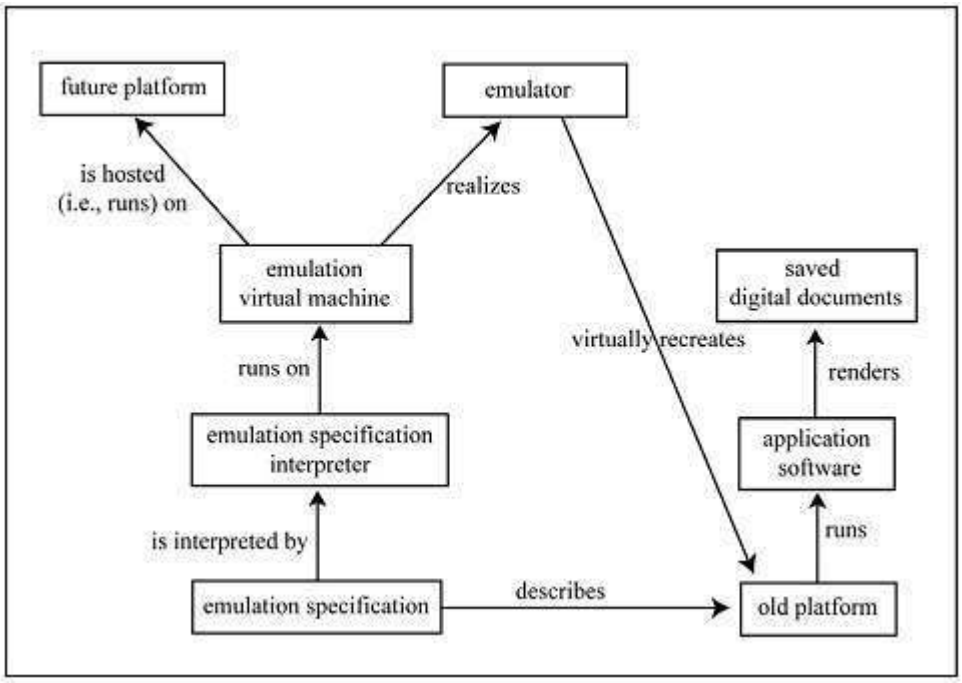


Figure 1: Elements and relationships in emulation-based preservation [40]

[6] For more information, see: http://en.wikipedia.org/wiki/Decoupling#Software_Development

Unfortunately, emulation does not solve the software preservation problems completely. Criticism has centred on the complexity of creating emulator specifications. The process is costly and time-consuming, while the outcome is limited to duplicating one specific computing environment. Furthermore, knowledge of the original environment is required for every software artefact. The limitations of emulation may ultimately hinder information reuse [33]. This is why Granger in [41] suggests that emulation provides partial solutions, but cannot resolve all digital preservation issues.

Emulation as a preservation technique has been used in some projects. A notable case is the CAMiLEON project [42, 43]. This work focused on using emulation to preserve the BBC Domesday software[7] by exploring, developing and evaluating technical strategies. In Planets, emulation has been used in the Dioscuri project[8]. Dioscuri is based on the recognition that for emulation to work as a long-term preservation solution, the emulator needs to be sustainable as well. For this reason, the project team used the Java Virtual Machine (JVM), building on the notion of a Universal Virtual Machine [44]. The advantages of this approach are durability and flexibility. The implementation in Java makes it durable, because it can be run on any computer platform that supports the JVM. Flexibility in Dioscuri comes from modular design: hardware components are emulated by a software surrogate called a module. Combining several modules allows the user to configure any computer system, as long as these modules are compatible[8].

2.3 Binary Translation

Binary translation (BT) is a technique used to convert an executable program from one computer architecture and operating system into an executable program for a different computer architecture and operating system [4]. With BT, one instruction set can be emulated by another through code translation from the *source* to the *target* architecture [5]. There are three types of software-based binary translation methods. One is emulation, where programme instructions are interpreted at runtime but are neither saved nor cached by the system. Another is dynamic translation, which translates between the source and target architecture at runtime, caching the code sequence for future use. Lastly, static translation interprets executable programmes offline so that they can be run on the target architecture. In all three cases, BT is effective as long as the performance of the translated code can resemble – or be competitive with – the performance of the legacy architecture [6].

Binary translation is not a solution without shortcomings. The translators take away memory and resources from the source architecture, causing runtime overheads; execution can be slow, as all code must initially be interpreted to target architecture code. As a field of practice, BT has not yet matured sufficiently to produce optimal results consistently [7]. Nonetheless, BT is a promising approach to solving the legacy software problem because it purports to retain all functional and temporal behaviour of the source architecture to the target one [8]. This includes running code on the new system at least as fast as it used to on the legacy machine, providing transparent execution between different platforms (e.g. code from an MS-DOS application runs on Windows XP exactly as it would on a Unix-based system). As opposed to other translation methods (such as software interpretation, microcoded emulation and native compilers) BT produces code that can be generalised to other platforms and work for all executables – not just the ones for which the source code is available [8].

From a digital preservation perspective, the potential advantages of employing binary translation have been illustrated in [9] by use of three scenarios. The scenarios assume that the source architecture is unusable due to damage or obsolescence. The study offers that binary translation methods can promote cost-effectiveness and automate the conversion of digital material into newer formats.

[7] For more information, see: http://en.wikipedia.org/wiki/BBC_Domesday_Project

[8] http://dioscuri.sourceforge.net/index.html

2.4 Virtualisation

Virtualisation is a broad term that describes a range of aspects in Computing Science, and refers to the abstraction of computer resources. A comprehensive list of these is featured in [11]. For the purposes of this study, virtualisation is defined as a method to partition a computer system in multiple completely separate systems. Every individual system is expressed as a virtual software and hardware environment, which closely assimilates that of a complete physical computer. This environment is called a Virtual Machine (VM)[9].

Regarding digital preservation, the usage of virtualisation can be consolidated in two areas – Platform Virtualisation and Application Virtualisation [10]:

- *Platform Virtualisation* abstracts an operating system from the resources of the underlying platform. Although different approaches exist, the most appropriate for the scope of this study is full virtualisation. As the name suggests, Full System Virtualisation creates a virtual facsimile that completely simulates the physical hardware, so that any compatible operating system and software may run on the virtual machine. For instance, a computer running Windows XP on a x86 architecture (e.g. Intel Pentium) can *host* a virtual machine running Linux as a *guest* operating system. In this sense, operating systems and applications developed for different platforms can run on a virtual machine oblivious to the environment in which they are actually running. Digital preservation can benefit from full virtualisation when obsolete operating systems and applications need to run on new systems. The virtualisation tool will ensure that modern system components are mapped to the obsolete ones – for instance, an on-board soundcard of a recent motherboard can be presented to a VM running MS-DOS as a SoundBlaster card. We discuss virtualisation tools further in Section 3.5.
- *Application Virtualisation* encapsulates software applications from the underlying operating system on which they are executed, forcing them to behave as if they were directly interfacing with the original operating system and its resources [14]. Through this type of virtualisation, an application can run on a different computer system than the one it was developed for, while previously incompatible applications can run simultaneously on the same system. Fundamentally, application virtualisation differs from platform virtualisation in that only specific applications are virtualised instead of the whole operating system. By hosting these applications on alien software / hardware environments, application virtualisation technologies can improve portability, manageability and compatibility of applications [14]. An example where application virtualisation has been practised is *Wine*[10] which allows MS Windows applications to run on Linux platforms. Another example are portable applications, where software can run independently without the need to install files to the system it is run upon – commonly used on removable storage media (CD-ROMs, USB Flash drives etc.) [15]

The potential benefits and implementations of virtualisation for preserving digital material have been discussed in the DCC report in [16]. The CASPAR project[11] has also explored the possibilities of virtualisation for digital preservation [17]. Alongside the benefits that these works report, one should also bear in mind that virtualisation is not the golden ticket to preserving dynamic digital content. First, there are limitations regarding the areas and the types of material it can be applied to; long-term preservation might be impeded by obsolescence of the VM software interfaces themselves. Second, as in the case of binary translation, virtualisation causes overheads, particularly in terms of allocating shared resources (memory, I/O operations) between the host computer and the guest VMs.

[9] Virtualisation and binary translation are not independent techniques. Until recently, many VM platforms used binary translation to convert guest kernel code [18].

[10] http://www.winehq.org/

[11] http://www.casparpreserves.eu/

3. Experimental Context

3.1 Scope

As we explained in the Introduction, the software domain covers a huge arena of applications. This study focuses on dynamic and interactive material, whose representations vary significantly and the number of examples is equally vast. It would be therefore impossible to study the entire array of possibilities within the limited scope of this work. Instead, we have decided to focus the experimentation on a specific sample which concentrates on *software art* and *video games*. These genres are representative of interactive and dynamic software applications. They are also of increasing interest to a large part of software creators and users, aligning at the same time with previous work conducted within Planets. However, it would be possible to extrapolate our findings to software of similar nature. The rationale for our decision is based on the following facts:

- Software art and video games are perceived as interrelated genres [36]. The fundamental difference is the degree of interactivity between user and software, with video games being heavily interactive. In cases where the two genres are combined, the result is referred to as *Art (or Arthouse) Gaming* [37]. All aforementioned genres can be dynamic in the manner that they interact with the components of the underlying computing environment.
- Software art and video games primarily address specific user communities who are not necessarily computer experts. This creates additional motivation to provide these communities with a means to protect their work against obsolescence. In the case of video games, software preservation can present users with return on the investment of purchasing a game and being able to play it regardless of changes in computing environments.
- There is an impetus to preserve software art and video games as unique records of modern culture and as study material for future generations. Both genres are fragile not only because of changes in technology, but also because of changes in ownership of copyright.
- Video games have a history that spans more than forty years. Their presence is nowadays more wide-spread than ever. However, it is a valid assumption that modern video games are not in such immediate need for preservation as legacy ones. For this reason, this study concentrates on video games developed for the MS-DOS platform[12]. On the other hand, software art is a relatively new field but the material is at high risk of perishing without preservation action. This is because few institutions have invested in assembling primary collections of such uniquely digital artefacts as cultural records [38].
- Software art and video games have dedicated communities, who have established through the years practices and communications to cater for their material of interest. Some of the preservation alternatives in this study (both internal and external to Planets) have been developed by these communities. This vivid activity offers reasonable reassurance that support for the preservation tools will be ongoing.

Inevitably, the narrow scope of this study means that there are a number of limitations to be considered. First, we discuss systems software (such as Operating Systems and device drivers) only to the degree that its involvement is necessary to satisfy the experimentation. Although virtualisation is literally impossible without systems software, this study does not address their preservation needs. Instead, we report on OS configurations and device driver needs that we encountered while setting up the experimental platforms (see Section 6). Second, the study excludes material whose complete experience and execution relies heavily on Internet access. For instance, we have not experimented with *Massively Multiplayer Online Games* (MMOG)[13]. Similarly, the experimental sample does not include material for which uncommon input devices are required. These cases include special game controllers – such as steering wheels, light guns or dance pads[14].

[12] For more information, see the Wikipedia page on MS-DOS at: http://en.wikipedia.org/wiki/Ms-dos

[13] For more information, see the Wikipedia page on MMOGs at: http://en.wikipedia.org/wiki/MMOG

[14] For more information, see the Wikipedia page on Game Controllers at: http://en.wikipedia.org/wiki/Game_controller

3.2 Overall Methodology

The overall methodology has been divided into eight interrelated steps. These have been designed to inform the objectives of the study, but also situate the experiments and results within a greater context of binary translation and virtualisation usage. The following list elaborates on each step of the methodology.

1. A literature review to explore the state-of-the-art in binary translation and virtualisation technologies. The literature review includes definitions of the techniques, explication of the processes involved, their benefits and shortcomings for digital preservation, as well as case studies where BT and virtualisation have been used or considered for preserving digital material (see Section 2).
2. Requirements capture of the aims of the experiments and experimental design. This step is further discussed in Section 3.3, *Experimental Design*.
3. Definition of requirements for experimental corpora. The set of requirements has been guided by the objectives of the study, as well as the suitability and availability of specific material to run the experiments. Section 3.1 provides the rationale for decision-making to identify software genres. Section 4.2 presents the sources and individual cases used in the experimental sample. The selection mechanism is further influenced by the legal issues addressed in Section 3.4.
4. Collection of the material identified in the previous step. In order to avoid legal implications and promote future usage of the experimental corpora, the sources where the material has been acquired from is explicitly referenced. Any intellectual property issues deriving from usage of the material will be documented.
5. Design of experiments, specifying the digital resources and preservation tools that have been used to evaluate the suitability of BT and Virtualisation for preserving dynamic and interactive digital content. Although the primary tools for experimentation come from the Planets suite, external software will also be used to generate VMs; Section 4.1 offers further information.
6. Clarification of goals and objectives for each experimentation decision. This step aims to disambiguate the role of each set of experiments in the overall study.
7. Formulation – i.e. the combination of methods and processes discussed in the previous steps that have produced the final set of experiments with the identified materials.
8. Execution of the experiments and documentation of the results. These issues are detailed in Sections 5 and 6 of this report.

3.3 Methodology for Experimental Design

The methodology for conducting the experiments has been largely based on the design specifications in [24] and [25]. The tasks that each experiment consists of have been defined in the following steps:

- Identification of digital resource types; in the case of experiments discussed here these will involve software art and video games requiring obsolete hardware/software.
- Usage purpose and objectives for each experiment or set of experiments. The goal here is to not only explain what experimentation is designed to achieve, but also how the results can be meaningfully used to preserve the material in real-world environments.
- Determination of stakeholders and target communities, which influence the extent that experimentation with the specified material is feasible and desirable, but can also provide feedback on the quality of the outcomes.
- Study of related references in order to situate the experimentation within the greater context of software art and video game preservation.
- Selection of input data and preservation tools. Inevitably, the experiments cannot involve the entire range of material under the resource types identified above. Hence, the selection process

has focused on representative cases that highlight differences and varying results among different preservation tools and software art alternatives.

- Execution of the experiments, by utilising Planets tools and external software. Regarding Planets tools, the most suitable platforms are the Testbed and Plato[15]. However, in order to comprehensively test virtualisation and binary translation as preservation methods, external software has been used as well.
- Evaluation of results involves assessing the outcomes of the experiments and measuring their success according to the pre-defined set of objectives, purpose and expected outcomes. Furthermore, we have evaluated the usability of the results based on their suitability for preservation and the needs of stakeholders and audiences.

This experimental design presents an opportunity to not only test different BT and VM software, but also utilise Planets tools to punctuate their versatility and efficiency in dealing with multifarious data types and genres.

3.4 Legal Considerations

There are a number of legal issues to consider when undertaking experimentation. Most are related to intellectual property rights for video games and operating systems. Specifically, we have decided to exclude video games from this study that have been characterised by the community as "Abandonware". The term is used to describe computer software that is no longer sold or supported, or whose copyright ownership may be unclear for various reasons. However, in most cases distributional rights either explicitly remain with the production company or require a time-consuming process of identifying the legal owner of Intellectual Property [39]. In order to overcome this situation, the video games featured in the experiments are either covered by a free public license or a legal copy is owned by the experimenter. Similar issues derive from the necessity to install different operating systems for the Virtual Machines. With the exception of Linux (which is distributed under a GNU General Public License), all other Operating Systems have been installed from a legally acquired copy. Because of these constraints, experiments in other platforms have not been possible. For instance, the original intention was to test binary translators and VMs for the Apple OS X and the Amiga platform. However, the Amiga OS is still proprietary and difficult to acquire nowadays, while Apple restricts installation of its operating systems to Mac computers only. Regarding external software, all cases used for experimentation are either open source or freeware. The only exception is the VMware Workstation, for which a fully-functional 60-day evaluation copy has been used.

4. Experimental Design

For the purposes of this work, we have identified two sets of experiments. The experiment sets correspond to each of the studied software genres: one for videogames and one for software art. We have used the methodology outlined in Section 3.3 to design the experiments. For the reader's convenience, the experimental design is presented in subsections that summarise resource types, the objectives of each experiment set, target communities, as well as preservation tools and input data. The fundamental principle for composing the experimental corpora in this work is based on the flexibility they provide. As specified in [19], there is not one universally applicable corpus to various testing circumstances; different experiments can require different types of corpus composition. Flexibility means that the material in the corpora can cover a wide range of requirements, so that experimentation needs can be accordingly satisfied by selecting objects as needed. We present preparation and execution of the experiments separately in Section 5. The results of the experiments and critique on the findings are discussed in detail in Section 6.

[15] http://www.ifs.tuwien.ac.at/dp/plato/intro.html

4.1 Video Games

4.1.1 Digital Resources

The experimental collection consists of video games developed for the Disk Operating System (DOS). Sample material covers more than ten years of game development – from 1982 to 1993. The collection includes titles from different eras of computer gaming. Each "era" represents variations in the computing environment and related system requirements to execute the game. The selection criteria for digital resources involve:

- Text-based games whose primary input and output are text characters. These games rely on language parsing to provide interactivity through the command-line driven environment for which they were developed [45].
- ASCII games that use text characters – instead of bitmapped graphics – to create the game interface. The designers use only the 255 symbols in the IBM-extended ASCII character set[16].
- Requirements for different computer display standards. The experimental material features games that require CGA, EGA, VGA and SVGA graphics[17]. Hence, the material covers a wide range of display specifications and related outputs.
- Requirements for audio output. As in the previous point, the experimental material covers varying demands on sound output. These range from simple PC speaker sound to advanced requirements for music, sound effects and speech reproduction.
- Requirements for peripherals (other than a keyboard) and optical drives. The sample collection includes objects that need a mouse as input device and access to a CD-ROM drive.

The above criteria show that different preservation strategies may be required, even for software developed for the same platform. Although in modern computers these are standard provisions, this was not the case in the command-line environment of MS-DOS – which was originally meant to work with floppy disks. Floppy disks and CD-ROMs are not persistent storage media either. In order to increase their preservation potential, we have extended these experiments by converting diskettes to floppy images[18] and CD-ROMs to ISO images[19]. Any original box art and instruction manuals have been digitised and saved in JPEG image files and PDF documents.

4.1.2 Purpose and Objectives

The overall purpose for this set of experiments is to promote long-term access to the material without the necessity to maintain obsolete systems (as in the case of computer museums, see Section 2.2). Instead, we want to test the degree that DOS video games can be rendered and playable on modern systems. The preservation solution should also be sustainable, by offering capabilities for portability in future systems. The objectives of this experimentation are:

- To showcase alternative methods for preserving interactive software artefacts, whose system requirements are considered obsolete.
- To compare the results of different binary translation and virtualisation tools. In this manner, we expect to provide guidance for selecting suitable alternatives to preserve interactive software.
- To evaluate the suitability of the Planets Testbed tools to address the complex requirements of interactive software preservation.

16 For more information, see: http://www.textmodegames.com/

17 For more information on display standards and descriptions for CGA, EGA and (S)VGA, see: http://en.wikipedia.org/wiki/Computer_display_standard

18 For more information, see: http://en.wikipedia.org/wiki/Floppy_image

19 For more information, see: http://en.wikipedia.org/wiki/Iso_image

4.1.3 Target Communities

In the first instance, the designated community for this set of experiments are museums and archives. So far, memory institutions have been conservative in their efforts to collect and preserve video games. We expect that this situation will change; the findings from this experimentation may help these institutions to decide on preservation strategies. However, video gaming addresses a large and varied community of collectors, scholars and gamers. Their interest in video games evidently differs, but there is a shared need to have the material readily accessible and operational on modern computer systems. For these communities, the preparation of the preservation alternatives (see Section 5) may offer a roadmap to rendering their software playable.

4.1.4 Related References

An overview of the challenges in collecting and preserving video games is presented in [46]. The authors discuss current practices, game-related archives and research initiatives in the field. However, their approach is high-level and does not offer concrete recommendations on how to actually preserve videogame software. The paper does offer some useful pointers to other game preservation projects. One is the Internet Archive's *Classic Software Preservation Project* (CLASP)[20]. CLASP aims to permanently archive classic, obsolete retail software from the late 1970s through the early 1990s. According to the project's website, the Internet Archive managed to acquire an exemption to copyright legislation in 2003, so that obsolete software can be made accessible. However, at the time of writing this report the CLASP collection was not populated with digital objects. Another effort is the Software Preservation Society (SPS)[21]. The project has produced a wealth of technical and advisory know-how for the preservation of Commodore Amiga's classic software collection. The web resource also provides a repository of approximately 3000 titles that have been cleared from intellectual property restrictions.

Within the greater domain of video games, a project similar to our work is presented in [3]. The objective of the project is to evaluate strategies for the preservation of console video games. Console video games differ from computer games in a number of ways. They are generated by a gaming console[22] which requires a specific and often bespoke storage medium. In most cases this comes in the form of a ROM cartridge[23]. There are numerous examples, from the Atari 2600 to the Nintendo Entertainment System (NES) and the Sony PlayStation Portable (PSP). The gaming console and ROM cartridges are technologies external to PC games. This means that a game developed for a console cannot be straightforwardly executed by standard computer hardware [3]. There are two options to preserve console games. One is to maintain the original gaming console. The other is to re-compile the source code and migrate it to a different platform (often a personal computer) [3]. Although this type of preservation is beyond the scope of this work, there are similarities in the characteristics by which preservation strategies can be evaluated. [3] presents a Requirements Tree that captures those characteristics. We have used an amended version of this structure in our experiments, which better reflects the needs of video games for the DOS platform. This shows that preservation strategies are reusable and can be tailored to match different requirements within a domain of focus.

4.1.5 Input Data

We have selected seven video game titles, which match the criteria identified in Section 4.1.1. Experimental material is exclusively drawn from either free software or copies of games that have been legally purchased. Other alternatives – such as using Abandonware software (see Section 3.4) – have not been considered as they might expose the experimentation to legal implications. The sample collection is comprised of boxed video games; each game consists of one or more diskettes or CD-ROMs and manuals[24]. The experimental data are presented here with a short description and specifications of the original environment, if this is explicitly stated in the game's documentation.

[20] http://www.archive.org/details/clasp

[21] http://www.softpres.org/

[22] For more information, see: http://en.wikipedia.org/wiki/Video_game_console

[23] http://en.wikipedia.org/wiki/Cartridge_%28electronics%29#Use

[24] In cases of freeware games, boxes and/or manuals are not available.

Zork I - The Great Underground Empire	
Type	Text-based adventure
Description	Zork I is the first game made commercially available to the general public. It was published by Infocom in 1982 when personal computers were a brand new thing, with no memory and of course no sound or capability for graphics at all. It was originally published on 5.25-inch floppies.
System Specifications	MS-DOS, IBM PC or compatible with a 5.25inch disk drive.
Media	1 x 5.25'' floppy disk
Copyright Notes	The copy of the game used in the experiments was re-released as freeware for a limited time, but cannot be redistributed[25].

Rogue - The Adventure Game	
Type	ASCII
Description	Originally developed in 1980 for Unix mainframe systems, Rogue eventually found its way to personal computers, including the IBM PC in 1984. An ASCII-character-cell classic, Rogue not only spawned a genre known as roguelikes, but also mothered the action role-playing game (RPG)[26].
System Specifications	MS-DOS, IBM PC or compatible.
Media	1 x 5.25'' floppy disk
Copyright Notes	--

Boulder Dash	
Type	CGA Graphics with PC Speaker sound
Description	Boulder Dash, originally released in 1984, is a classic series of computer games for the Atari 400/800, Apple II, MSX, ZX Spectrum, Commodore 64, and ColecoVision home computers, and later ported to the NES, Acorn Electron, PC, Amstrad CPC, Amiga and many other platforms. It was created by Peter Liepa and Chris Gray, and on October 28, 1983, acquired and later published by First Star Software, which still owns the rights to the game[27].
System Specifications	MS-DOS, IBM PC or compatible.
Media	1 x 5.25'' floppy disk
Copyright Notes	--

[25] For more information, see: http://en.wikipedia.org/wiki/List_of_commercial_video_games_released_as_freeware

[26] Source: http://www.pcadvisor.co.uk/news/index.cfm?newsid=110612&pn=3

[27] Source: http://www.freebase.com/view/en/boulder_dash

TETRIS	
Type	ASCII
Description	Tetris is a video game that requires little description. Having sold more than 70m copies for different platforms, Tetris has been voted in second place of the IGN "100 Greatest Video Games of All Time".
System Specifications	MS-DOS, IBM PC or compatible.
Media	1 x 5.25'' floppy disk
Copyright Notes	The version used in the experiments is a freeware port to IBM PC running DOS.

Prince of Persia	
Type	VGA graphics with PC Speaker sound
Description	Developed by Brøderbund and released in 1989, Prince of Persia impressed audiences by its innovative fluent character movement and well-designed surroundings. The game has been characterised by many as one of the greatest platform games of all times.
System Specifications	MS-DOS, IBM PC or compatible. Minimum Requirements: 512Kb RAM (640 Kb for VGA graphics); IBM/Tandy or 100% Compatibles; Floppy Disk Drive.
Media	1 x 3.5'' inch floppy disk
Copyright Notes	--

Alone in the Dark	
Type	VGA / SVGA Graphics with music and speech
Description	Alone in the Dark is a 1992 survival horror game developed by Infogrames. The game set the standard for later rival popular survival horror games such as Resident Evil and Silent Hill. Items and characters in Alone in the Dark are three-dimensional, rendered upon a two-dimensional fixed background. Mixing polygons and 2D pre-rendered background images required a fixed camera angle, which designers used to their advantage to create dramatic scene setups appropriate for a horror-themed game[28]. The sample record used here is the CD-ROM version, which features speech throughout the game plus sound effects and music.
System Specifications	MS-DOS, IBM PC or compatible. Minimum Requirements: PC AT 16Mhz; DOS 5.0 or Higher; 6MB RAM; VGA Graphics Card; 8MB Hard Disk Space; CD-ROM drive with sound output; MSCDEX 2.2 or Higher.
Media	1 x CD-ROM
Copyright Notes	--

[28] Source: http://en.wikipedia.org/wiki/Alone_in_the_Dark_%28video_game%29

The Legend of Kyrandia Book Two: The Hand of Fate	
Type	VGA / SVGA Graphics with music and speech
Description	Developed by Westwood Studios Inc. and released in 1993 by Virgin Games, Hand of fate is the second instalment in the Legend of Kyrandia trilogy. The CD-ROM version used here features full speech throughout the game, music and sound effects.
System Specifications	MS-DOS, IBM PC or compatible. Minimum Requirements: 386 20 MHz; DOS 5.0 or Higher; 4 MB RAM; CD ROM Drive; VGA or MCGA Graphics Card; MSCDEX 2.2 or Higher; Sound Cards For Voice - SoundBlaster or SoundBlaster Pro and 100% Compatibles, AdLib Gold, and AdLib Gold Standard; Sound Cards For Music - SoundBlaster or SoundBlaster Pro and 100% Compatibles, AdLib, AdLib Gold, AdLib Gold Standard, Roland MT-32/LAPC-1/Sound Canvas and General Midi; 5 MB of Free Hard Drive Space; Microsoft Compatible Mouse and Keyboard
Media	1 x CD-ROM
Copyright Notes	--

4.1.6 Preservation Tools

As mentioned previously, the main focus of this experiment set is to test the tools developed by Planets on their suitability to help preserve interactive software (such as video games). The results from these tools are compared with other alternatives – such as binary translators and software to generate virtual machines. In this section we outline both the Planets-specific and external software used for experimentation.

Planets Testbed and GRATE

GRATE (Global Remote Access to Emulation-Services)[29] is a demonstration service developed within the Planets project. It aims to show the accessibility of 'aged' digital objects through emulated systems using the Internet. GRATE is developed as a component system to allow easy user access to emulation services. It provides a number of virtual machines (DOS/Windows 3.11, Windows 98), emulators and binary translators (eg. DosBox, QEMU, Hatari, Dioscuri). GRATE has been integrated into the Planets Interoperability framework for viewing objects within an emulator (Create View feature). The advantage of GRATE is that it aggregates many virtualisation / emulation alternatives into one environment. As part of the Testbed, it offers greater flexibility in accessing emulation services and uploading files remotely, rather than having to install the VM software locally. Alongside with GRATE, we have used the Floppy Image Helper service of the Testbed. This tool is able to create Floppy disk images containing files of the user's choice, and extract files from a floppy disk image. This migration functionality is particularly useful for video games stored in one or multiple floppy disks, which need to be preserved for future access. This tool is used in combination with GRATE and the external software described later in this Section.

Planets Preservation Planning Tool (Plato)

The planning tool Plato is "a decision support tool that implements a solid preservation planning process and integrates services for content characterisation, preservation action and automatic object comparison in a service-oriented architecture to provide maximum support for preservation planning endeavours"[30]. Plato has been used to create preservation plans for each set of experiments (see Appendix A and B). The advantages of using Plato include seamless integration with the Testbed experimental results, incorporation of the evaluation parameters specified in Section 4.1.7, and an automated process for measuring the suitability of each tool for preserving interactive and dynamic content. Although the use of Plato is complementary to the

[29] GRATE can be accessed from the Testbed or as a standalone tool at: http://planets.ruf.uni-freiburg.de/~randy/Oindex.php

[30] Source: http://www.ifs.tuwien.ac.at/dp/plato/intro.html

experimentation with the Testbed and the external software, its role is fundamental in presenting experimentation results in a comprehensive, organised structure.

DOSBox 0.73 (Windows XP version)

DOSBox is a binary translator working as a command-line program. It is configured either by a set of command-line arguments or by editing a plain text configuration file. For ease of use, several graphical front-ends have been developed by the user community. DOSBox is a full CPU emulator, capable of running DOS applications. Since DOSBox can emulate its CPU by interpretation, it is independent of its host CPU [47]. According to the project's website[31], DOSBox provides an environment with the following system properties:

- Intel 80286 / 80386 CPU or compatible (both real- and protected mode)
- Support for Expanded and Extended Memory Specification (EMS / XMS) which is often required for DOS applications
- Support for Tandy, Hercules, CGA, EGA, VGA and VESA graphics
- Support for Gravis Ultra Sound / SoundBlaster sound cards

DOSBox is free and open source software that has been developed by members of the community. According to [47], a number of commercial video game development companies have used DOSBox to re-release some of their older titles. The advantage of DOSBox is that contains its own internal DOS-like shell, rather than being a fully virtual PC emulator. Hence, it can be used without the necessity to own a licensed copy of MS-DOS. Furthermore, DOSBox supports both physical CD-ROM drives and virtual CD-ROM drives mounted through ISO image files. The main disadvantage of DOSBox is that it does not support post-80486 processor features. Also, the built-in DOS shell does not contain many of the advanced commands included in the latest releases of MS-DOS.

VMWare Workstation for Windows

VMware Workstation[32] is a virtual machine software suite for x86 and x86-64 computers. This software suite allows users to set up multiple x86 and x86-64 virtual computers and to use one or more of these virtual machines simultaneously with the hosting operating system. Each virtual machine instance can execute its own guest operating system, such as Windows, Linux, BSD variants, or others. In simple terms, VMware Workstation allows one physical machine to run multiple operating systems simultaneously. Other VMware products help manage or migrate VMware virtual machines across multiple host machines [28]. VMware is a well established platform for creating virtual machines. Although the software is proprietary, the resulting VMs can be loaded in free alternatives, such as Oracle's Virtual Box. In our experiments with video games, we have used an evaluation copy of VMWare Workstation for Windows XP and Linux.

Wine for Linux

Wine[33] is a free binary translator that aims to allow Unix-like computer operating systems to execute programs written for Microsoft Windows. Although Wine implements some aspects of the Windows kernel, it is not possible to use native Windows drivers with it, due to Wine's underlying architecture. This prevents certain applications from working, such as some copy-protected titles [48]. As of 2009, Wine runs some software packages with good stability and many others with minor issues. The goal in including Wine as an experimentation alternative is to test its suitability to execute Windows-based preservation solutions – such as DOSBox.

DOSEMU for Linux

DOSEMU[34] is a compatibility layer software package that enables DOS software to run under Linux on x86-based PCs (IBM PC compatible computers). It uses a combination of hardware virtualization features and strategic emulation. It is thus able to achieve nearly native speed for 8086-compatible DOS operating systems and applications [49]. Currently it is only available for x86 Linux systems. The goal here is to primarily test the suitability of DOSEMU for running DOS-based video games on a Linux system.

31 http://www.dosbox.com/information.php?page=0

32 http://www.vmware.com/products/workstation/

33 http://www.winehq.org/

34 http://www.dosemu.org/

4.1.7 Significant Properties

As mentioned in Section 4.1.4, the evaluation of preservation requirements for DOS video games is based on the Objective Tree presented in [3]. We have amended the original requirements, so that console gaming-specific elements were removed and others relevant to our study added. The amendments have been informed by feedback from mailing lists and relevant forums. Specifically, the identification of significant properties for DOS video games was discussed in:

- the game preservation mailing list[35] led by the International Game Developers Association (IGDA)[36].
- the message board of the DOS Games Archive Network[37].
- The forum at Abandonia[38] – a web resource dedicated exclusively to DOS video games with a large community of registered users.

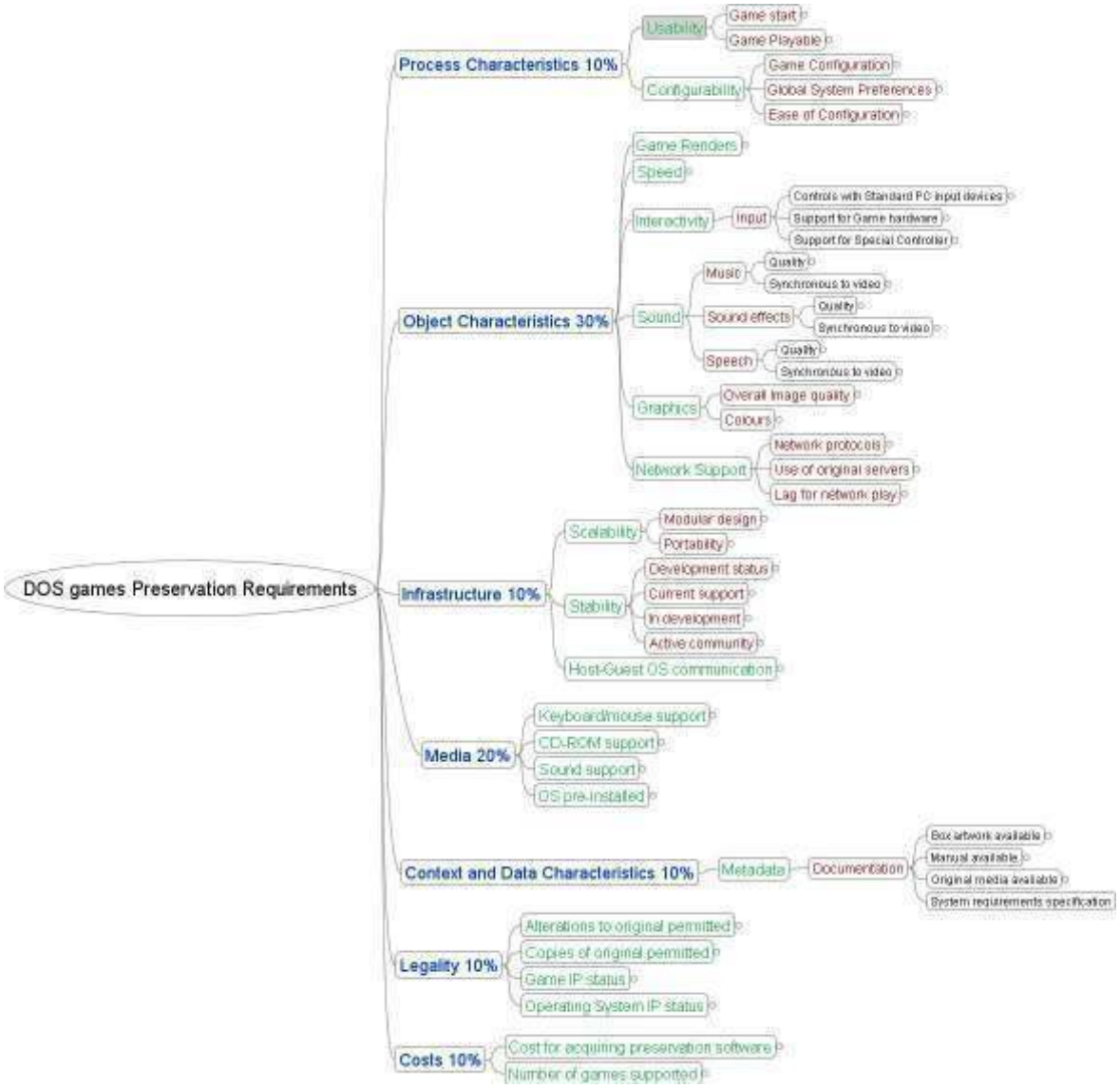


Figure 2: Objective Tree of preservation requirements for DOS video games

35 http://six.pairlist.net/mailman/listinfo/game_preservation

36 http://www.igda.org/preservation

37 http://www.dosgamesarchive.com/

38 http://www.abandonia.com/index.php

The complete set of requirements on which the experimentation results were evaluated is depicted in Figure 2. The requirements are grouped in the following seven main categories:

- *Process characteristics* evaluate usability and configurability of the preservation tools. Usability is measured by the capacity of the tool to support both games that require installation on a hard disk and games that execute without installation. Usability also measures whether a game is playable as executed through the preservation tool. On the other hand, Configurability measures the ease of configuring the VM / binary translator, the manner that global system preferences are assigned (manually or pre-configured) and the configuration of individual games (manual, pre-configured or not necessary).
- *Costs* assess the impact of potential charges for acquiring the tools on the overall preservation process. We consider free tools as generally more appropriate, as their use for preservation is promoted by the alleviation of charges. Costs are further affected by the number of games supported by each preservation tool. The ideal tools should support as many games as possible. It should be noted that costs reflect only potential charges for acquiring the preservation tool, but not for system software. It is assumed that a legally purchased copy is available for installation on the virtual machines.
- *Object characteristics* identify whether a game successfully renders within the virtual system and reflect the level that the tools successfully reproduce the original audiovisual and interactive characteristics of a game. Such characteristics include speed of gameplay, resemblance with original controls, reproduction of sound and graphics, as well as support for networked gameplay.
- *Context and Data Characteristics* identify such elements as the existence of metadata accompanying the game and the availability of other documentation (box artwork, manuals, original media, and specification of system requirements). Although these characteristics do not affect the performance of the preservation tools, they do affect the preservation potential of obsolete video games.
- *Infrastructure* assess the sustainability and preservation potential of the tools themselves. The measurements for these characteristics are identical for all items in the sample collection. We have identified scalability, portability (whether software is OS dependent or not), stability (to ensure that the solution itself is adequately preserved in the future) and Host/Guest OS communication (native, via network, via virtual media or none) as aspects of infrastructure that need to be evaluated.
- *Legality* defines those parameters related to legal issues and restrictions. These include permissions to alter the original game and/or make copies, the IP status of the game and the IP status of the operating system (Freeware/Protected).
- *Media* characteristics assess the support provided by the preservation tools for keyboards, mice, CD-ROMs, sound cards and other peripheral devices. The existence of a pre-installed OS is also evaluated.

Building on the Plato approach in preservation planning, we have assigned importance factors to weigh each of the above characteristics. Object characteristics have been considered as the most significant in preserving video games, as they encapsulate the experiential facets of interactive software discussed in the Introduction to this report. Media characteristics are also very important, because they assess the existence of necessary material to execute a video game. The remaining categories of characteristics have been judged of same importance, as they equally contribute to the preservability of a video game. Appendix A provides a full explication of the preservation requirements for DOS video games.

4.2 Software Art

4.2.1 Digital Resources

The sample collection consists of software art works created in the period 1995-2004. The material represents different genres of software art, each incorporating varying degrees of interactivity with the user. System requirements to execute the selected works are also variable, so as to meaningfully test the preservation suitability of the experimentation tools. In more detail, the sample collection includes four genres of software art:

- Viral art, which aims to draw audience attention by spreading its content through unconventional means. In the Web 2.0 era, these means are no longer as controversial as they used to be, but the term is still used to describe a practice of disseminating software artworks through Internet sharing and "word of mouth" (perceived here in the broader sense of online communication channels)[39].
- Data manipulation software art, which uses computer data to perform operations that are atypical to the original purpose that this data was created. In most cases, the artwork will produce audiovisual outputs of information stored either locally or on the Internet. Unlike data manipulation tools, the artistic counterparts are not necessarily meant to generate a usable, practical outcome. Instead, they use existing information to create a performance which re-invents the purpose of digital data. In this sense, parallels can be drawn to art intervention[40].
- Demos – non-interactive applications "which produce, in real time, engaging computer graphics and music" [50]. Demos are produced to demonstrate combined skills in visual arts, 3D modelling, music composition and programming. The main challenge for demos is to produce code that dynamically interacts with computing resources and stretches their capabilities to the maximum degree possible. As opposed to videos and animations, demos are computed in real-time, making system power considerations a major concern [51]. Demos are the products of a software art subculture known as the demoscene[41]. Demogroups primarily present their work at dedicated events (called Demo Parties). As an artistic practice, demos appeared in the early 80's. Demos have been developed for many platforms, including Commodore 64, Amiga, and later personal computers with DOS / Windows operating systems.
- Art games, which are essentially video games "designed in such a way as to emphasize art or whose structure is intended to produce some kind of reaction in its audience" [37]. Art games aim to produce compelling artefacts that often challenge traditional gaming norms. This is achieved by intriguing visuals, alternative interaction methods or unconventional subjects (such as politics, environmentalism and social structures).

Digital resources in the sample collection consist mainly of single executable files developed for the DOS / Windows / Linux platforms. Art games are the only exception, as they commonly require an installation procedure on a local machine. The objects are freely distributed online as freeware, open source software or under some form of the General Public License (GPL)[42]. Sources of the material are provided in the Plato Preservation Plan for Software Art (see Appendix B).

4.2.2 Purpose and Objectives

The purpose and objectives of this experimentation set are similar to those specified for video games (see Section 4.1.2). The main difference is that the sample collection of software art includes material developed for different operating systems (as opposed to just software for the MS-DOS). Obsolescence should not be overlooked, even for more recent computing environments, particularly when the software is in its very nature revolutionary. Software art is pushing the boundaries of technology. Quite often, this process produces bespoke code that will only execute on the specific environment for which it was developed. For instance, many demos created for Windows 95 / 98 will not execute at all on Windows XP / Vista – even in the compatibility mode that

[39] For more information, see: http://en.wikipedia.org/wiki/Viral_art

[40] For more information, see: http://en.wikipedia.org/wiki/Art_intervention

[41] For more information, see: http://en.wikipedia.org/wiki/Demoscene

[42] For more information, see: http://www.gnu.org/licenses/licenses.html

these OS offer[43]. With this in mind, the purpose of software art experiments has been extended from 4.1.2 to test the degree that software art for various platforms can be executed on modern systems.

4.2.3 Target Communities

Similarly to Section 4.1.3, the target communities come from the memory institutions' domain. Individual roles within these communities that may benefit from this experimentation are archivists, curators, preservation specialists and collection managers. However, the experimentation findings can be equally useful to scholars, digital artists and the wider area of the Arts.

4.2.4 Input Data

We have selected five software art works, which match the criteria identified in Section 4.2.1. Experimental material has been exclusively drawn from free software. The sample collection is comprised of five executable files; each is accompanied by documentation and – where applicable – instructions on installation. The experimental data are presented here with a short description and specifications of the original environment.

[phage] by Mary Flanagan	
Type	Viral art
Description	Created in 1998, [phage] is a computer application which is viral– an artificial life form. [phage] filters through all available material on a specified workstation and places it in an alternate context-a visible and audible moving 3D world. By mapping a user's unique experiences – through images, downloads, web sites visited, emails – the computer program creates spatial memory maps that not only reflect the computer and technoculture in content, but the user's artefacts from his or her interactions[44].
System Specifications	Windows 98, IBM PC or compatible Approx. 3MB hard disk space
Media	1 x executable file (.exe)
Download page:	http://www.maryflanagan.com/phage/

DataDada by August Black	
Type	Data manipulation art
Description	DataDada, version <aldkjallkdjsfa>, is an application that converts the stored data on a hard drive into a movie complete with sound, image, and subtitles. Essentially, it reads all the data on the disk (or, optionally, only specific directories), and writes the data to the computer's sound card and video display. Additionally, it displays the name of the file being read as a human-understandable subtitle. DataDada was created in 2003[45].
System Specifications	Windows 2000 / Linux, IBM PC or compatible
Media	1 executable file (.exe), 1x Dynamic-link library (.dll), 1x text document
Download page:	http://aug.ment.org/datadada/download.php

[43] For more information on compatibility modes, see: http://www.microsoft.com/windowsxp/using/helpandsupport/learnmore/appcompat.mspx

[44] Source: http://www.maryflanagan.com/phage/

[45] http://aug.ment.org/datadada/index.php

Prophecy - Project Nemesis by the Conspiracy Group	
Type	64k Intro (demo)
Description	This is an award winning 64k intro (demo) created for the Assembly 2004 demo party in Finland[46]. The demo is available from the project's website in both executable and video format. Furthermore, video footage is available online in YouTube. Here, we are using the original executable file. Conspiracy's *Prophecy* manages to fit a whole believable world and narrative inside sixty-four kilobytes. The technical excellence displayed here enables pushing even more content, with great if somewhat inconsistent looking scenes, evoking a good sensation of scale[47].
System Specifications	Windows 98 Minimum requirements: MMX compatible processor 64MB RAM Multi-texturing graphics card (TNT2 / ATI Rage) Recommended: Processor (2 GHz+, AMD preferred), 512MB memory, GF4Ti/R9600-class graphics card (nVidia preferred), 128MB video memory, DirectSound compatible sound card
Media	1 executable file (.exe), 1x Dynamic-link library (.dll), 1x text document
Download page:	http://conspiracy.hu/release/64k/prophecy/

a.Shooter - Sonic Invaders by a.Game	
Type	Art game
Description	Released in 2004, a.Shooter is the first prototype release of a.Game production: a solely acoustical ego-shooter game. a.Game is a production and research group examining acoustical mechanism of orientation and interaction by means of computer games. It produces audio games that focus and experiment with sound-driven interaction and acoustical navigation in virtual sonic environments without the support of visual information. The player's task is to shoot up sonic invaders conquering a virtual room as defined by panorama, pitch and volume[48].
System Specifications	Windows 98 DirectSound compatible sound card
Media	1 executable setup file (.exe), 1x text document with instructions
Download page:	http://www.agame.org/en/aShooter/

[46] For more information, see: http://en.wikipedia.org/wiki/Assembly_%28demo_party%29

[47] Source: http://awards.scene.org/archive.php?cat=9&year=2004

[48] Source: http://www.agame.org/en/aShooter/

CAERO by Plant & Electromotive Force (EMF)	
Type	Demo
Description	Caero was created by the demoscene group Plant & Electromotive Force (EMF). The demo won the first place prize in the 1995 Demo Party, an annual demoscene event held from 1991 to 2002 in Denmark. It was one of the first events of its kind and set the trend for many other demoscene parties in Europe. The scenes in the demo are complex (more than 7000 polygons in the butterfly scene for example) and some of them have transparent objects and motion blur with colours. Caero was one of the first demos to use such complexity and bespoke libraries for the DOS platform. This work became popular beyond the demoscene community because it features music composed by Brothomstates - a well-known Finish musician of IDM (Intelligent Dance Music). The version used in the experiments was acquired by the scene.org repository.
System Specifications	MS-DOS, IBM PC or compatible Minimum System Requirements: PC 486/33; DOS 5.0 or Higher; 4MB RAM; VGA Graphics Card; 4MB Hard Disk Space; SoundBlaster or SoundBlaster Pro and 100% Compatibles.
Media	1 executable setup file (.exe), 1x text document with acknowledgments
Download page:	http://www.scene.org/file.php?id=93811

4.2.5 Preservation Tools

The experimentation with software art uses the same preservation tools identified for DOS video games (see Section 4.1.6), albeit with minor alterations. Specifically, we have tested the suitability of the Planets Testbed and Plato tools for preserving software art, alongside with DOSBox for Windows XP and Wine for Linux. However, instead of using the proprietary VMWare Workstation, we have decided to experiment with the free open source alternative, Virtual Box. By employing Virtual Box in the experimentation, we highlight the preservation benefits of free software as well as test its compatibility with virtual machines created in the VMWare Workstation. Following is a brief description of Virtual Box.

Virtual Box for Windows

Virtual Box is a virtualisation product for x86 and AMD64/Intel 64 architectures. Currently it is he only professional-quality virtualization solution that is also Open Source Software[49]. Virtual Box offers many of the benefits of proprietary alternatives, with support for a range of guest operating systems including: Linux, Mac OS X, Windows (98SE, XP, Vista, 7) DOS and Solaris. Similarly to VMWare Worstation, Virtual Box VMs can be customised to virtualise different system architectures and peripheral devices. In the experiments, we have used the Windows XP version, which features the full virtualisation package. This version comes under the VirtualBox Personal Use and Evaluation License (PUEL)[50] which allows free of charge use for personal, educational and evaluation purposes.

[49] Source: http://www.virtualbox.org/wiki/VirtualBox

[50] For more information, see: http://www.virtualbox.org/wiki/VirtualBox_PUEL

4.2.6 Significant Properties

As explained in Section 3.1, software art and video games are genres of software with many similarities in their characteristics. For this reason, we have used the Objective Tree presented in Section 4.17 as a foundation. We have retained the common characteristics between the two genres, but have altered the structure slightly so that it better reflects software art preservation requirements. Figure 3 diagrammatically depicts the content and structure of the significant properties for preserving software art.

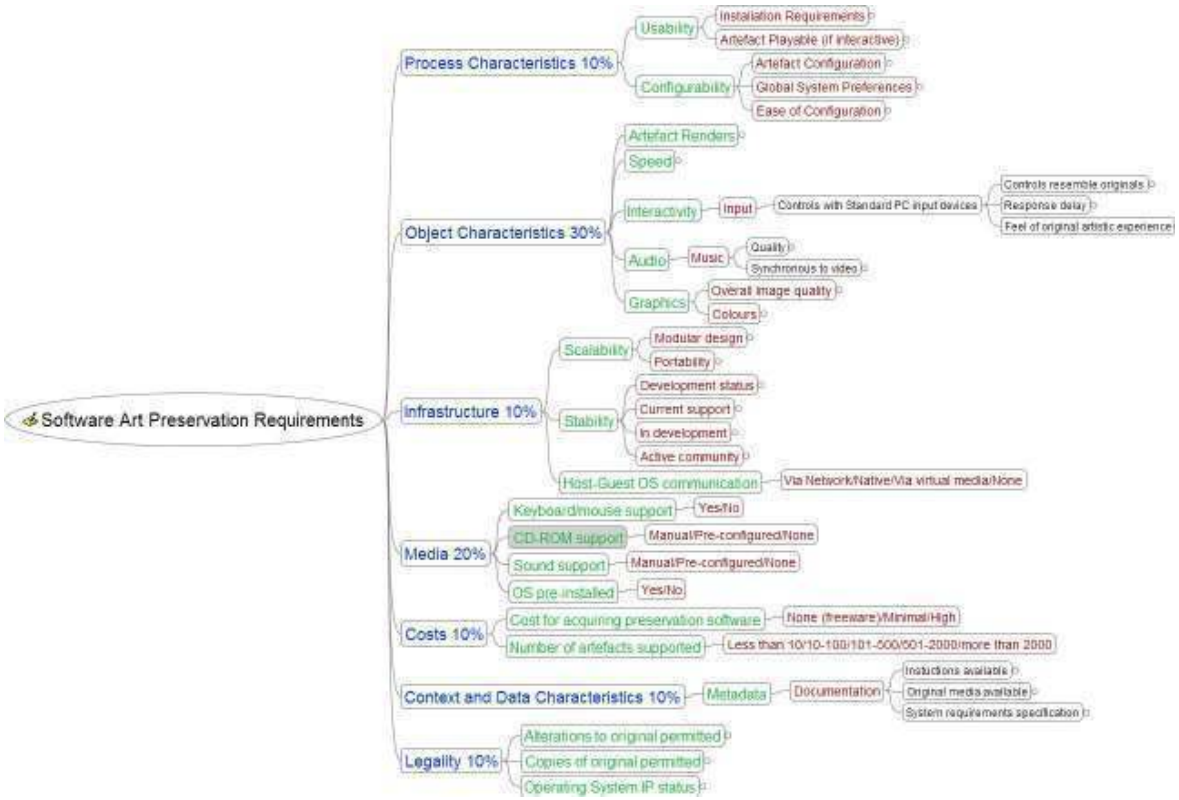


Figure 3: Preservation Requirements for Software art

The main categories and weights have remained identical. We have removed measurements for network support – as our sample does not include cases with such requirements (see Section 3.1). Based on this rationale, measurements for speech, box artwork and artefact Intellectual Property status have been excluded from the significant properties for software art. A compete account and description of the preservation requirements for software art can be found in Appendix B.

5. Preparation of the Experiments

This section presents the preparation of the preservation tools to make them readily available for experimentation. In total, we used seven distinct alternatives, each with varying degrees of customisation needed. In addition to the tools described in Sections 4.1.6 and 4.2.5, a number of auxiliary applications were employed to facilitate experimentation. These tools are not available within the Testbed or the external preservation software. They provide functionality to create and mount CD-ROM and floppy disk images. Specifically, we used Virtual Floppy Drive (VFD) Version 2.1[51] to mount the floppy images created by the Testbed Floppy Images Helper tool (see Section 4.1.6). In this way, it became possible to directly access the contents of the floppy images through a virtual floppy disk drive. This was necessary because the computing environment on which the experiments were conducted did not include a floppy drive. VFD is a suitable alternative, as it provides much functionality free of charge. Given that most modern computers do not have a floppy drive, this application may be of use to organisations and individuals interested in using virtualisation and binary translation for software preservation. Furthermore, we used and evaluation copy Magic ISO Maker[52] to create virtual CD-ROM images. This application offers a wide range of features to both generate and edit image files in the ISO format from either a CD-ROM or from selected files residing in a storage medium. Most of the preservation tools used in the experiments communicate with the host Operating System via virtual media (see Appendices C and D). By using Magic ISO Maker, we were able to import the sample collection material into the virtual environment. The only restriction of the evaluation (free) copy is that it cannot save ISO images larger than 300MB.

All experiments were carried out on a DELL Optiplex 745 Desktop PC[53] with the following system specifications:

- Intel Core 2 Duo 6300 CPU at 1.86GHz (2MB cache)
- 1GB RAM
- 60GB Hard Disk Drive
- 1 x CD/DVD-RAM drive
- 1 x Virtual CD-ROM drive
- 1 x Virtual Floppy Disk Drive
- 256MB ATI Radeon X1300 Pro Graphics Card
- Integrated ADI 1983 High Definition Audio Chipset
- 5 x USB ports
- Windows XP Professional SP 3 Operating System

The above specifications represent a typical modern computing environment, which is not overly powerful. By using this system, we aimed at measuring the performance of the preservation tools in a manner that realistically captures the computing resources available to most institutions. At the same time, the system is contemporary enough to avoid becoming imminently obsolete. Inevitably, the experiments will become outdated as technology evolves, since they depend so heavily on the underlying system where performance of the preservation tools has been measured.

In the remainder of this Section, we explain the necessary setting up the preservation tools for the experiments. We describe the configuration details for binary translators and virtual machines, as well as the issues we encountered in terms of system software, device drivers, customisation of guest operating systems and limitations. The review is organised in distinct sections, analysing each alternative independently.

[51] http://chitchat.at.infoseek.co.jp/vmware/vfd.html

[52] http://www.magiciso.com/

[53] For product details, see: http://www.dell.com/us/en/dfb/desktops/optix_745/pd.aspx?refid=optix_745&cs=28&s=dfb (Retrieved October 2009)

5.1 GRATE

The GRATE emulator in the Planets Testbed has been the most straightforward tool to configure in the experiments. GRATE is an integrated environment that features pre-installed emulators and virtual machines for a variety of computing environments (see Figure 5). In the experiments, we used DOSBox and QEMU for Windows 3.11 / Windows 98 within GRATE.

Figure 4: The GRATE interface

The only action required from the user is to upload the file or set of files that need to be executed within the virtual environment. GRATE automatically converts the input file(s) into a floppy image and then mounts the image as floppy drive A:\ (see Figure 5). Although functionality is offered to examine files using DROID[54], we did not need to use it during the experiments.

Figure 5: File upload in GRATE

[54] DROID (Digital Record Object Identification) is a software tool developed by The National Archives to perform automated batch identification of file formats. For more information, see: http://www.dcc.ac.uk/resources/external/droid-digital-record-object-identification

5.1 DOSBox

Alongside with GRATE, DOSBox is the second easiest tools to install and configure. For the Windows version used here, the installation file can be directly downloaded from the project's website[55]. The installation is completed within a few seconds and the installed software occupies less than 5MB of disk space. DOSBox features a command-line interface similar to that of MS-DOS (Figure 4). The developers have incorporated additional features, such as a comprehensive help tool and straightforward support for mounting CD-ROMs and ISO images. Support for sound reproduction is pre-configured in the form of a SoundBlaster card. Further features include the ability to capture screenshots and record footage of applications.

Figure 6: The DOSBox interface

DOSBox communicates with the host environment by mounting a location of a physical hard disk or CD/DVD drive as a virtual drive. The user can choose both the location and the drive letter which will be used to access the mounted content. For instance, we stored all experimental material in a folder named "dosgames". In order to access the folder within DOSBox, the following command needs to be issued:

```
mount d c:\dosgames
```

In this example, the folder "dosgames" on the local hard disk C:\ has been mounted as a virtual drive D:\ in DOSBox, which is now considered as the root folder. Any video games (or other types of software) installed in DOSBox will appear on the host's file structure within the "dosgames" folder. Virtual drives, folders and files can be accessed by using commands identical to those of MS-DOS (see Figure 4). If the user is not acquainted with the DOS environment, DOSBox offers detailed help. Once the folder containing the material has been mounted and accessed, the games are readily available for execution without further configuration. All device drivers (e.g. for mice and keyboards) are pre-installed. Although DOSBox offers functionality to increase or decrease the speed of execution, we did not find this necessary in our experiments.

[55] http://www.dosbox.com/download.php?main=1

5.2 MS-DOS Virtual Machine

In order to set up a virtual machine for MS-DOS, we initially used the VMWare Workstation. An evaluation copy can be downloaded from the company's website[56]. The computing environment was defined by following the *New Virtual Machine Wizard*. A typical configuration was selected. Although VMWare Workstation provides the option to install a new operating system from a CD-ROM, this is not possible with a MS-DOS installation which only comes in floppy disks. Next, we chose MS-DOS as the guest OS with the specifications below (see Figure 7):

- Single processor (386DX)
- 32MB RAM
- 1 x CD-ROM drive
- 1 x floppy drive (mapping to the virtual device created with VFD – see beginning of Section 5)
- Default host soundcard
- Default host settings for display adaptor / monitor
- Disabled Internet Adapter

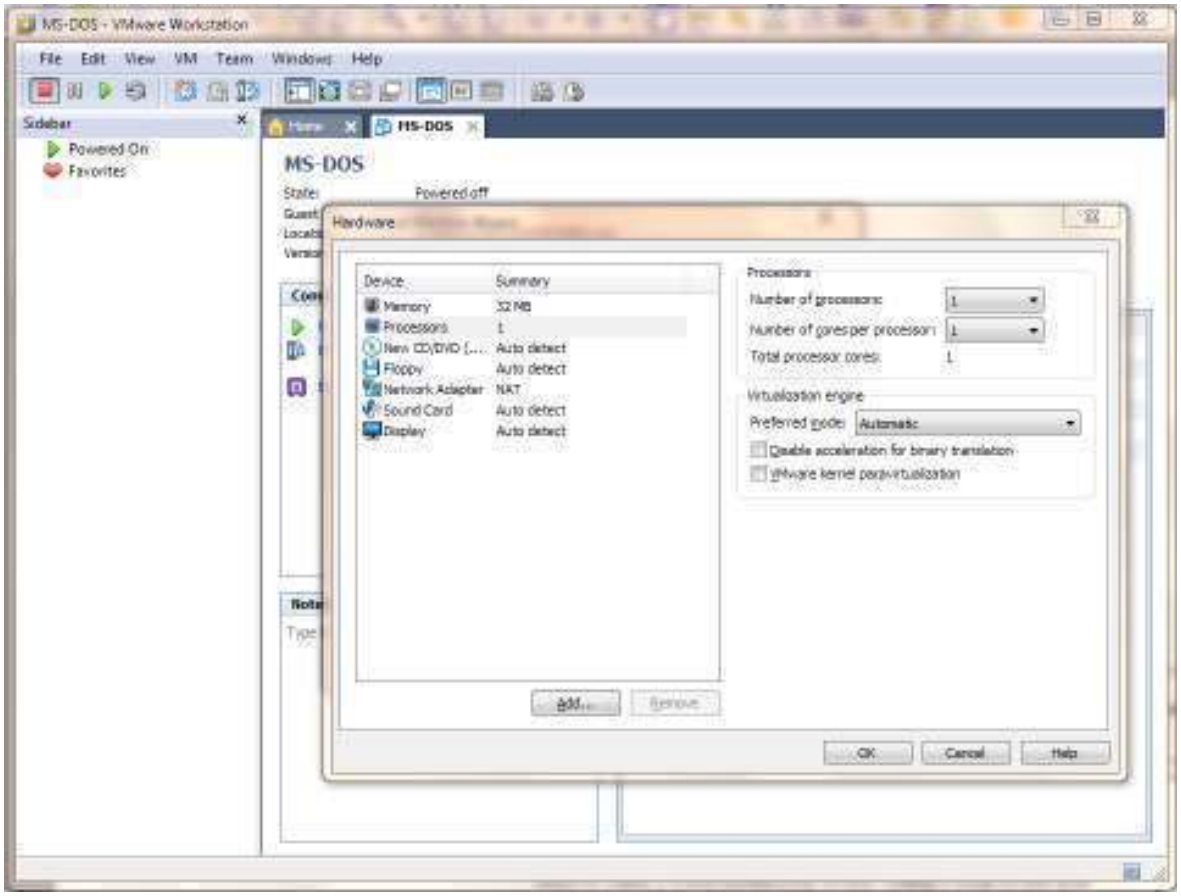


Figure 7: Setting up a DOS VM in VMWare Workstation

[56] Registration is required to activate the evaluation copy, accessible at: http://downloads.vmware.com/d/info/desktop_downloads/vmware_workstation/7_0

Before installing the operating system on the new VM, we converted the five MS-DOS 6.20 floppy disks into image files (.ima) using the Testbed Helper tool (see Section 4.1.6). We mounted the first floppy disk into the virtual drive by using VFD. The VM was then powered on, configured to start from the floppy drive instead of the hard disk. Once the OS installation was complete, we created an ISO image via Magic ISO Maker with the experimental material. The material was copied into a folder on the virtual hard drive of the newly created MS-DOS VM. The installation of the two CD-ROM video games (Alone in the Dark and Hand of Fate – see Section 4.1.5) was not as straightforward. The first issue we encountered was insufficient memory. Even though 32MB RAM is more than enough, the problem derives from the manner that MS-DOS and DOS applications handle memory configuration. In order to solve the problem, we used the `memmaker` command[57] available with this version of the OS. Unlike the pre-configured environment of GRATE or DOSBox, the user is expected to know how to work with the operating system. This is not a fault of the virtualisation software, which in all respects has performed as expected. However, virtualisation requires technical expertise and the memory issue shows the implications of this requirement.

Issues of similar nature were encountered for configuring CD-ROM access and sound card. A MS-DOS virtual machine does not provide configurations for these devices by default. It is again the role of the user to exercise technical know-how in order to render them "visible" to the operating system. VMWare comes with an additional module called VMWare Tools, which includes a number of built-in drivers and configurations for peripheral devices. However, VMWare Tools only supports Microsoft Windows, Linux, Sun Solaris, FreeBSD, and Novell NetWare guest systems[58]. The manual process of configuring CD-ROM and soundcard devices is not easy without previous experience. For CD-ROM access, a device driver first needs to be located. In this case, we used a driver developed by Oak Technology (called `aokcdrom.sys`)[59]. Regarding the sound card, consultation with forums and Internet resources showed that the best choice is to configure the on-board soundcard of the host computer as a SoundBlaster 16 card[60]. Drivers for this legacy model can be downloaded from the manufacturer's website[61]. The driver pack also includes software to set up the sound card on the MS-DOS virtual machine. Although the process is successfully completed (see Figure 8), we still encountered problems with production of sound and music through the general midi. Research in online resources revealed that other users have experienced similar problems with no solution readily available. It should be noted here that VMWare does not offer support neither for MS-DOS nor for evaluation copies of their products.

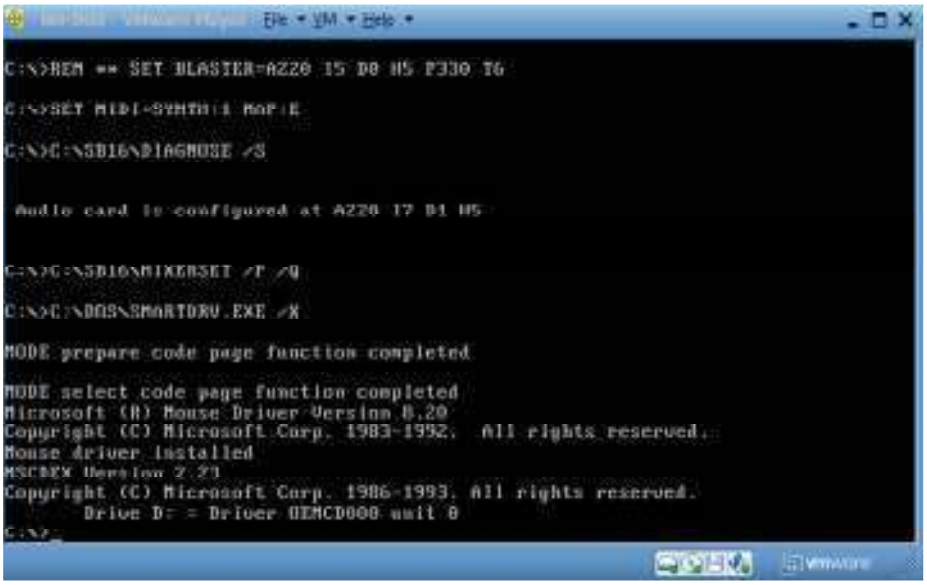


Figure 8: MS-DOS VM with sound support in VMWare Player

57 For more information, see: http://en.wikipedia.org/wiki/Memmaker#memmaker

58 Source: http://en.wikipedia.org/wiki/VMware_Tools#Other_products

59 Installation instructions available at: http://www.computerhope.com/cdromd.htm

60 For more information, see: http://en.wikipedia.org/wiki/Sound_Blaster_16

61 SoundBlaster 16 was manufactured by Creative. The support website can be accessed at: http://support.creative.com/

The stable version of the MS-DOS VM was used with the VMWare version for Linux. No further alterations were necessary. In order to use the VM for the software art experiments with Virtual Box, we utilised the virtual disk file, which stores the VM's hard disk drive. Although it is not possible to import a VM created by VMWare Workstation into Virtual Box directly, the virtual disk can be used as part of a new virtual machine. All its contents (including OS and configurations) will be made available to the new VM (see Figure 9).

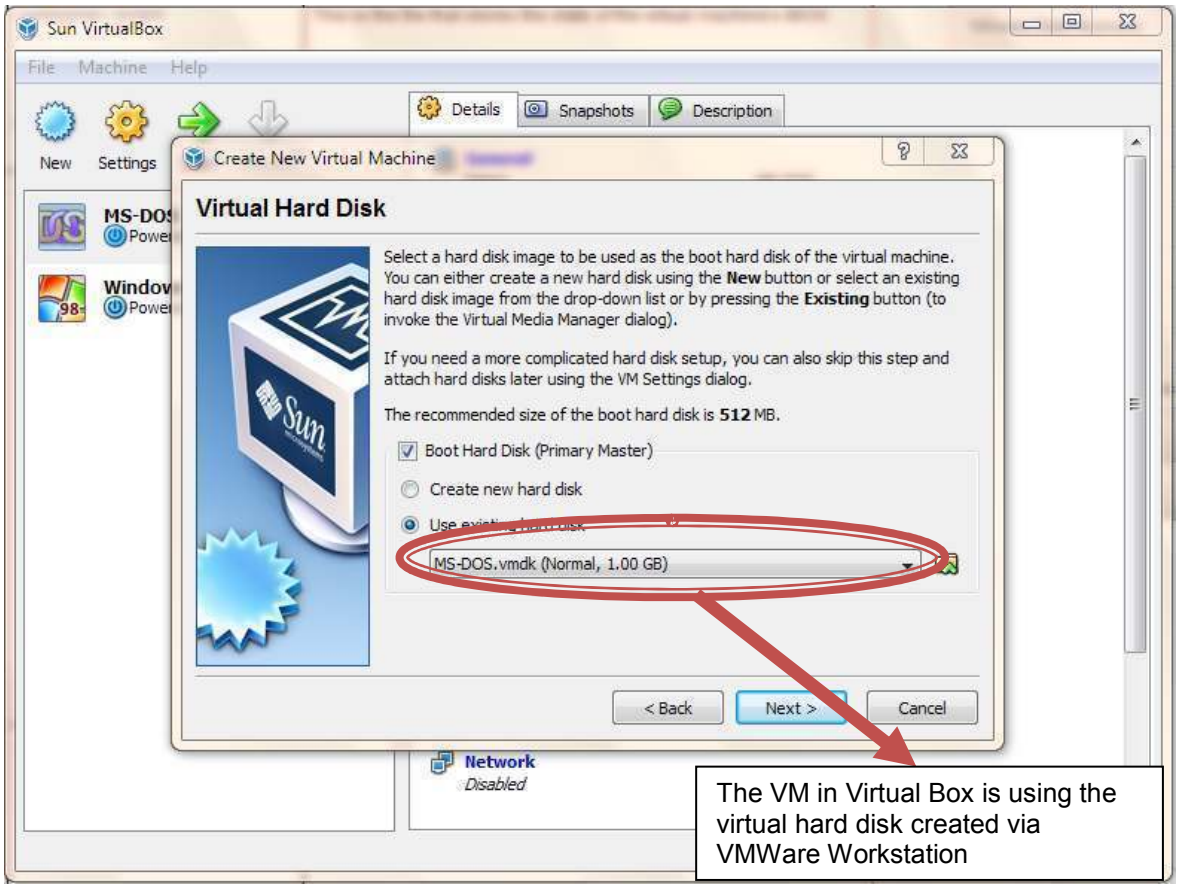


Figure 9: Importing an existing virtual hard disk to a Virtual Box VM

Virtual Box provides a similar wizard to create new virtual machines (see Figure 10), albeit without the provision of the VMWare Tools. From our experiments, we derived that the two virtualisation platforms can interoperate at this "virtual hard disk" level only, but the results are accurate and satisfactory.

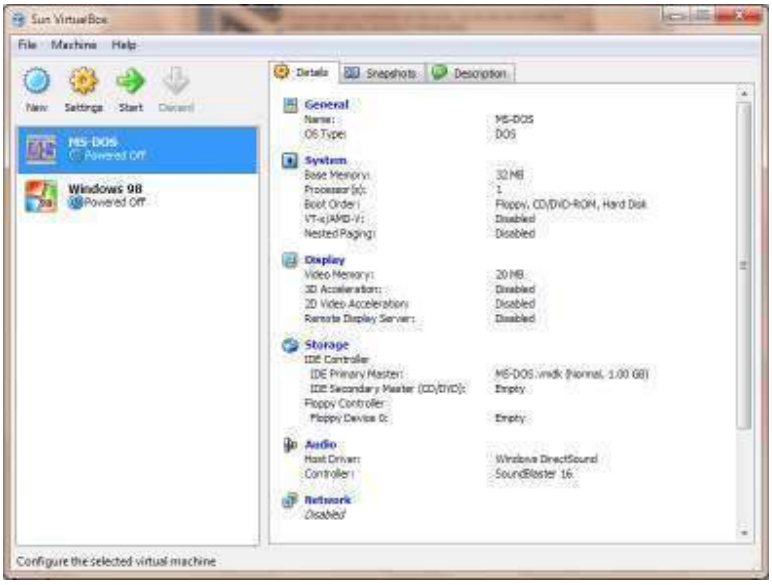


Figure 10: The MS-DOS VM set up for the Virtual Box environment

5.3 Windows 98 VM

The creation of a virtual machine running Windows 98 (for the software art experiments) followed the same process as that described in Section 5.2. In the first instance, we used VMWare Workstation to create a VM with the following characteristics (Figure 11):

- Single processor (586DX2)
- 256MB RAM
- 1 x CD-ROM drive
- 1 x floppy drive (mapping to the virtual device created with VFD – see beginning of Section 5)
- Support for USB input devices (mass storage media)
- Default host soundcard
- Default host settings for display adaptor / monitor
- Disabled Internet Adapter

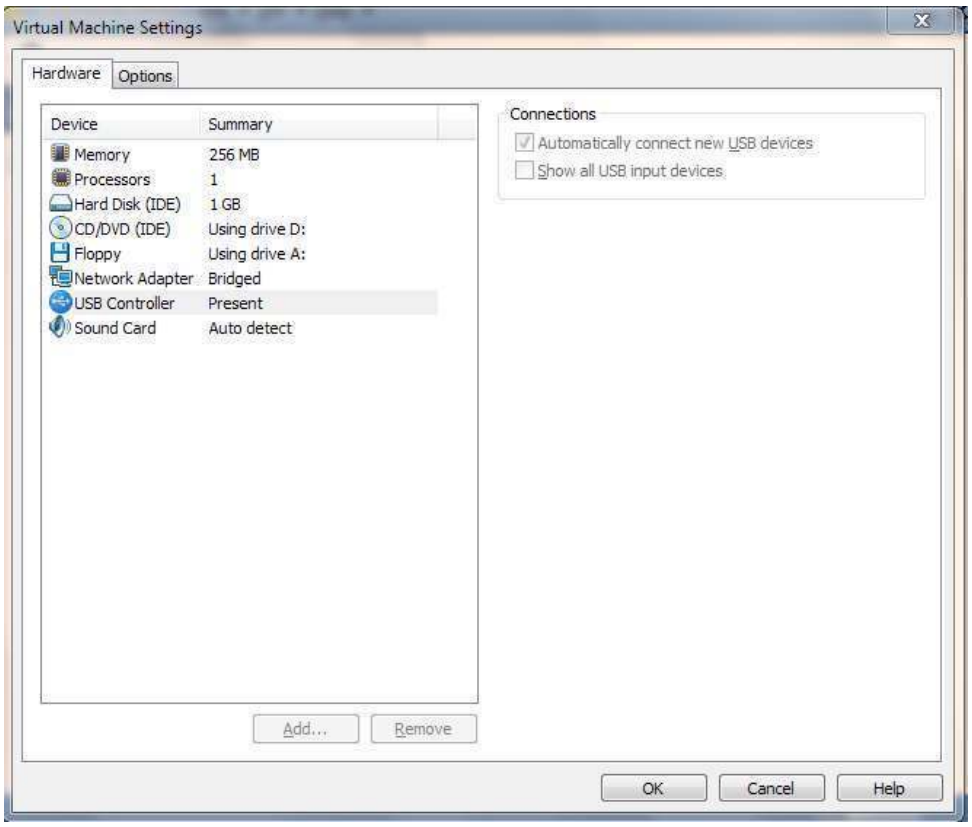


Figure 11: Configuration Windows 98 VM in VMWare Workstation

Since we owned a CD-ROM for this operating system, we used VMWare Workstation's option to install Windows 98 at the time of creation (and not retrospectively, as in the case of the MS-DOS VM). The virtualisation software offers additional support for installing more recent operating systems and configuring system software. In this way, the tedious process experienced in the MS-DOS VM for configuring device drivers was eliminated. VMWare Tools efficiently managed the process, rendering the VM immediately available (Figure 12).

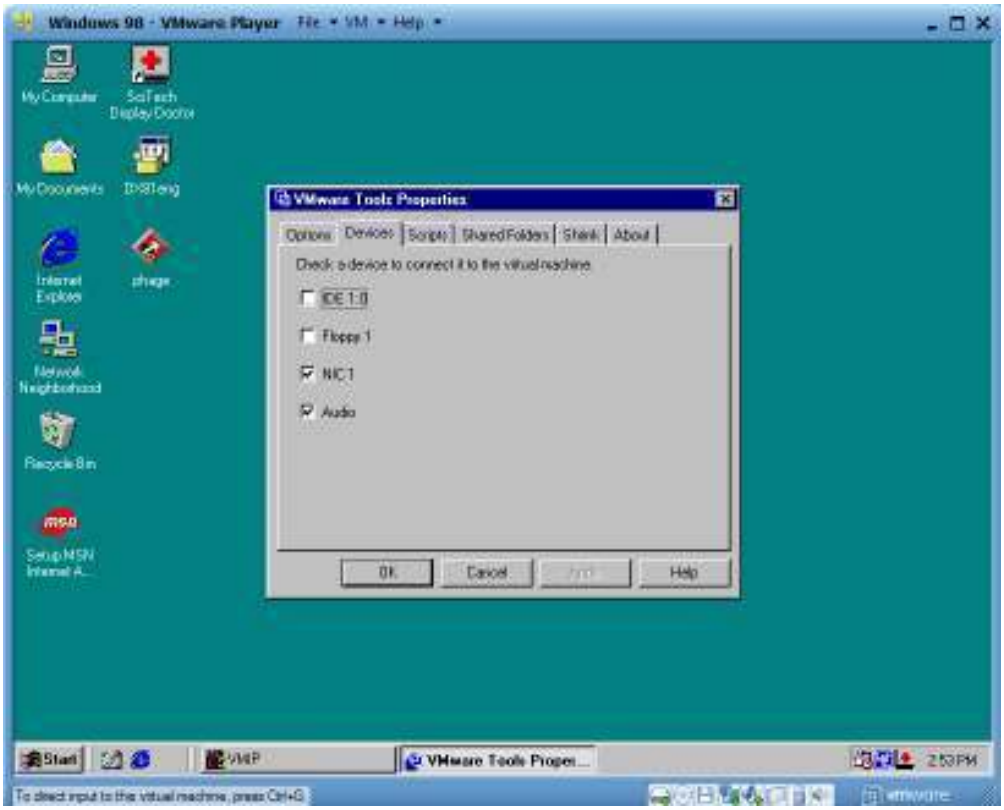


Figure 12: The Windows 98 VM with VMWare Tools running

However, the configuration process was not as straightforward in the case of Virtual Box. Understandably, support from VMWare Tools is only available when the VM is running within VMWare products. It does not extend to other alternatives, such as Virtual Box. Although Windows 98 found and installed most of the device drivers (the original CD-ROM is essential here), problems persisted with the identification of the sound card. Again, much advice and guidance is provided in online resources dedicated to virtualisation. Most resources agreed that for audio hardware to work in Virtual Box, the device should be declared and installed as a SoundBlaster 16 (Figure 13). This solution worked in this case with drivers obtained by Creative's online support.

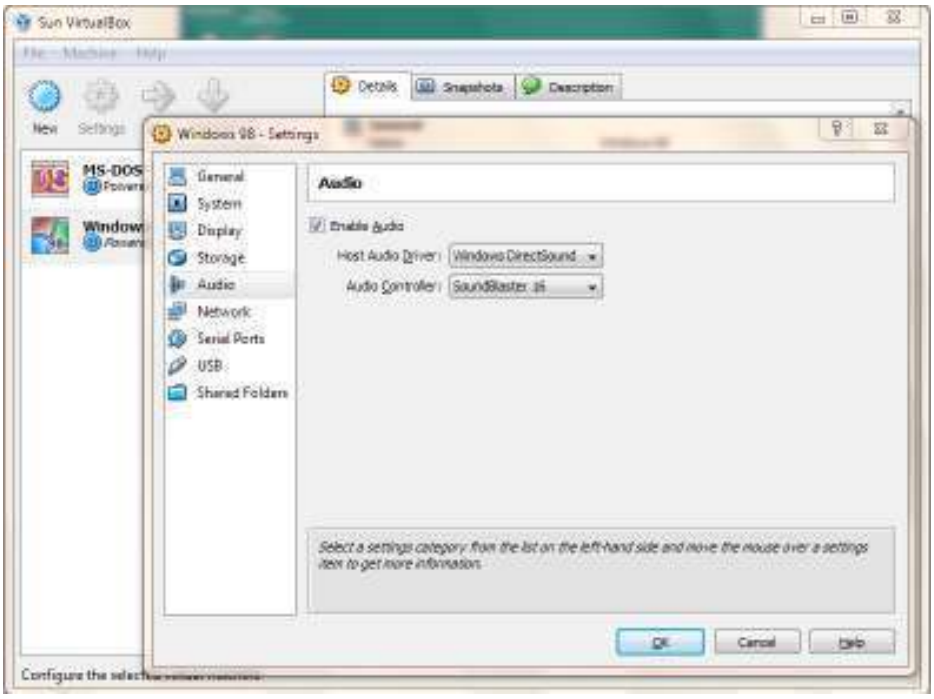


Figure 13: Setting the audio controller to SoundBlaster 16 in Virtual Box

5.4 Wine and DOSEMU for Linux

As described in Section 4.1.6, we used two Linux virtualisation tools for Linux: Wine and DOSEMU. Their purpose is fundamentally different and therefore their role in the experimentation addressed different needs. Wine is meant to allow Windows applications to run on a Linux system. For this reason, testing whether Wine would render DOS video games would yield similar results to running the software within Windows. In fact, we explored this possibility for all the items in the sample collections. The results were extremely varied, proving that this is not a sustainable solution for digital preservation. However, Wine presents a suitable alternative for executing preservation tools developed for Windows platforms within Linux. In order to gauge this suitability, we used Wine to run a Windows version of DOSBox in Ubuntu Linux 9.10. Ideally, we expected both versions to perform in a consistent manner. On the other hand, DOSEMU is a virtual machine combined with emulation, which theoretically provides similar functionality to DOSBox.

The installation of Wine in the Karmic Koala distribution is very straightforward through the Graphic User Interface (GUI). The process can be performed through the *Ubuntu Software Center* under the *Applications* menu. Once Wine has been located by the search tool, it can be downloaded and installed automatically on the local machine. Wine offers an extensive library of drivers for audio and display adapters. For the experiments, we used the EsounD driver, as it provided greater stability with the computing environment. Screenshots of the process are shown in Figure 14.

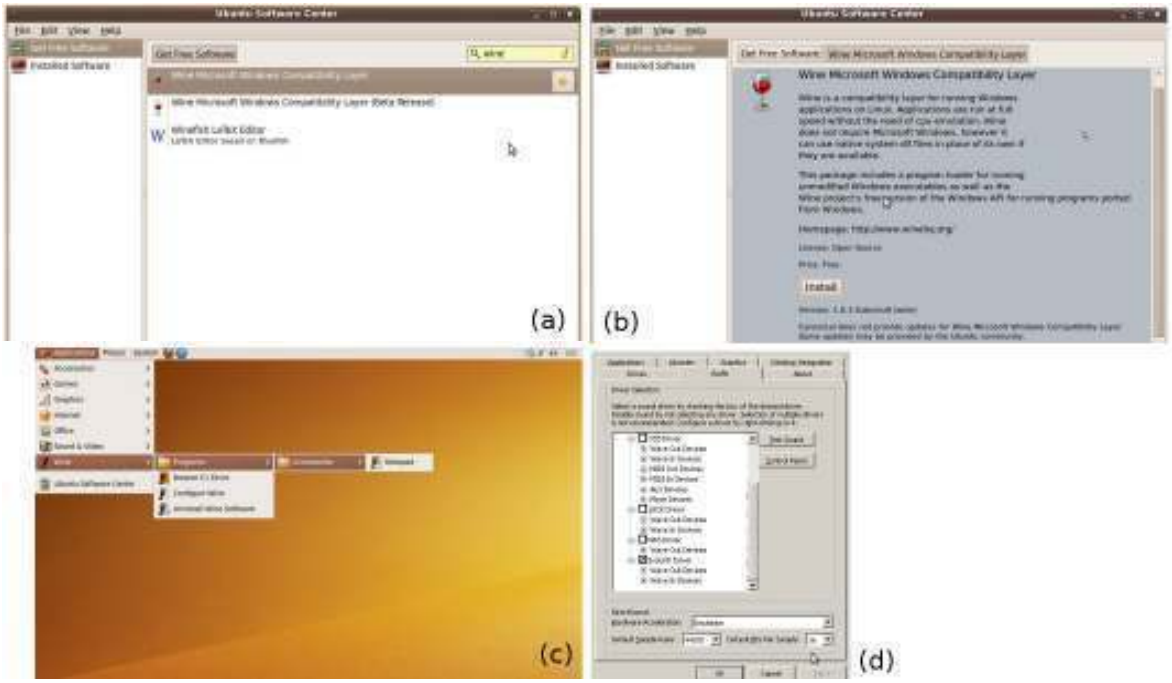


Figure 14: (a) Searching for Wine in the Ubuntu Software Center; (b) Installing Wine; (c) Locating installed application; (d) Configuring device drivers

DOSEMU is equally easy to install, as long as the instructions accompanying the software are followed accurately. Much like DOSBox, DOSEMU comes with a pre-configured sound card, as well as automatically mounted virtual drives for the Ubuntu *Home* folder and the CD-ROM. The environment is also based on the DOS command-line prompt approach (Figure 15), which means that the user should be familiar to some extent with this operating system.

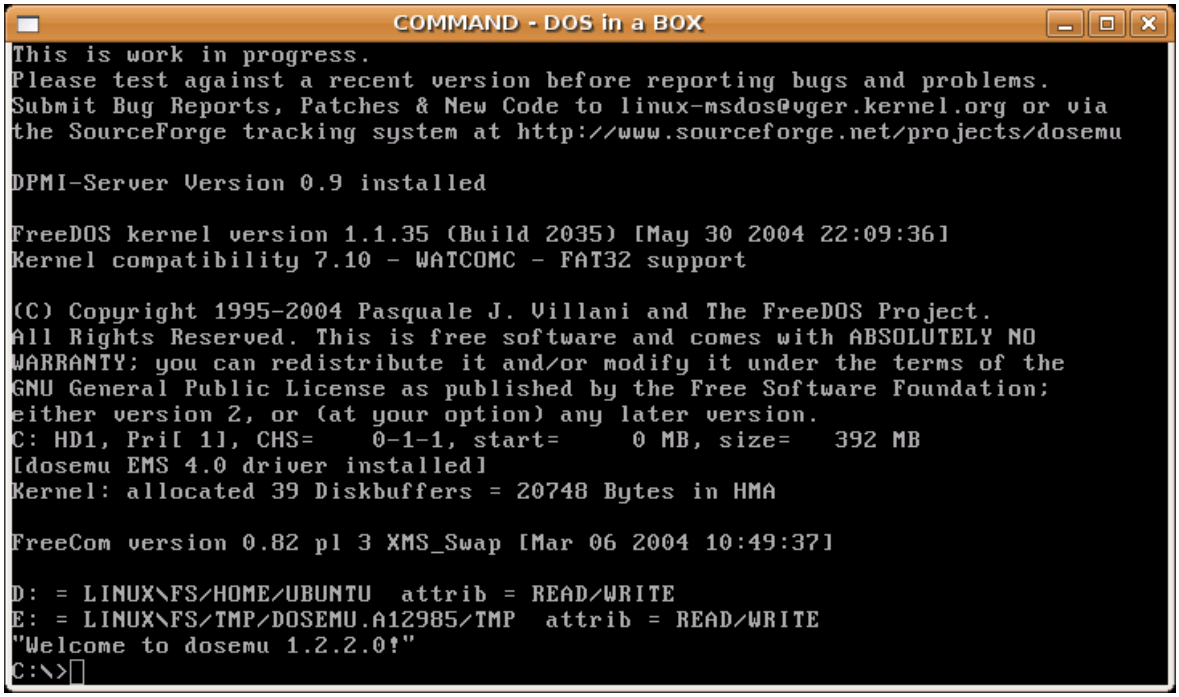


Figure 15: The DOSEMU Interface

6. Results

In this Section the findings of the experiments are presented. For each preservation tool, we discuss the produced outputs based on our observations and – where applicable – comparisons between tools. The observed behaviour has been recorded in preservation plans created in the Planets Preservation Planning Tool (Plato) and are included in this report in Appendices C and D. The Plato plans offer an aggregated view of the results and a direct means to compare outputs among the experimental alternatives. The analysis of the findings in Plato is based on the significant properties identified in Sections 4.1.7 and 4.2.6. Although an extensive description of the Plato representation is beyond the scope of this report, we will explicate the quantified results produced by the planning tool.

6.1 Video Games

As an overall remark, the experimentation corroborated the complex requirements of preserving interactive software in practical terms. There is a large number of parameters that need to be considered in order to produce the optimum results. Optimum results are understood in this case as fully operational games that retain all of their original characteristics. The look-and-feel of the original environment should also be preserved to an extent. Furthermore, the preservation solution should be easy to manage and use by non-experts in Computer systems. Only in this manner we can claim that virtualisation and binary translation are successful preservation solutions. In the preparation for experimentation (Section 5), we reported on a number of considerations towards preserving the original as accurately as possible. These include installation of device drivers, support for optical media and configuration of audio adapters. Each preservation tool responded to these requirements with variable levels of success. The ease of creation and usage also varied significantly. In the next sections we will review these issues for each preservation tool employed in the experimentation.

6.1.1 Testbed and GRATE

The most straightforward solution in our experiments was GRATE created by the Planets project. As mentioned previously, GRATE offers the distinct advantage of allowing users to test different environments under a common interface. In this sense, GRATE is a selection of existing solutions rather than a tool in its own right. Nevertheless, this approach presents distinct benefits for the user. All necessary parameters have been pre-configured, which alleviates the need for technical expertise. The tools integrated within GRATE can be accessed and used through the common interface without the need to learn the particular procedures of the individual tools. For instance, digital files for use in Windows 98 can be mounted automatically in GRATE through a simple visual environment that assimilates a website. In contrast, a virtual machine running Windows 98 requires the user to manually specify the existence of a CD-ROM drive. Similarly, virtual drives need to be mounted in DOSBox using a command-line interface that may be unfamiliar to many parties trying to preserve their software artefacts. One of the biggest benefits of GRATE is the possibility to remotely access the preservation tools. In this way, local hosting is not necessary. Virtual machines are particularly demanding in terms of resources, but they also necessitate the existence of a legal copy of an Operating System. GRATE offers a significant advancement towards overcoming these problems. Being part of the Planets Testbed further provides users with additional tools to support their preservation needs – such as the Floppy Image Helper service we utilised in these experiments.

In terms of outputs, GRATE was able to render all of the video games not requiring installation. This is due to inevitable restrictions on the size of uploads to a remote environment. Since both the Testbed and GRATE are experimental platforms and the results of ongoing research, security measures must be taken to prevent malicious usage of the offered services. The true strength of GRATE is in its role as a test environment, where users can assess the suitability of preservation alternatives for software. However, it is fair to say that the Testbed / GRATE approach also provides the necessary means to straightforwardly preserve obsolete video games. The performance of the games was comparable to that of the stand-alone tools. Image quality and colours were accurately reproduced, albeit with minor response delays due to the remote nature of the tools. Figures 16 and 17 show screenshots of games execution in GRATE.

Figure 16: Rogue running in GRATE

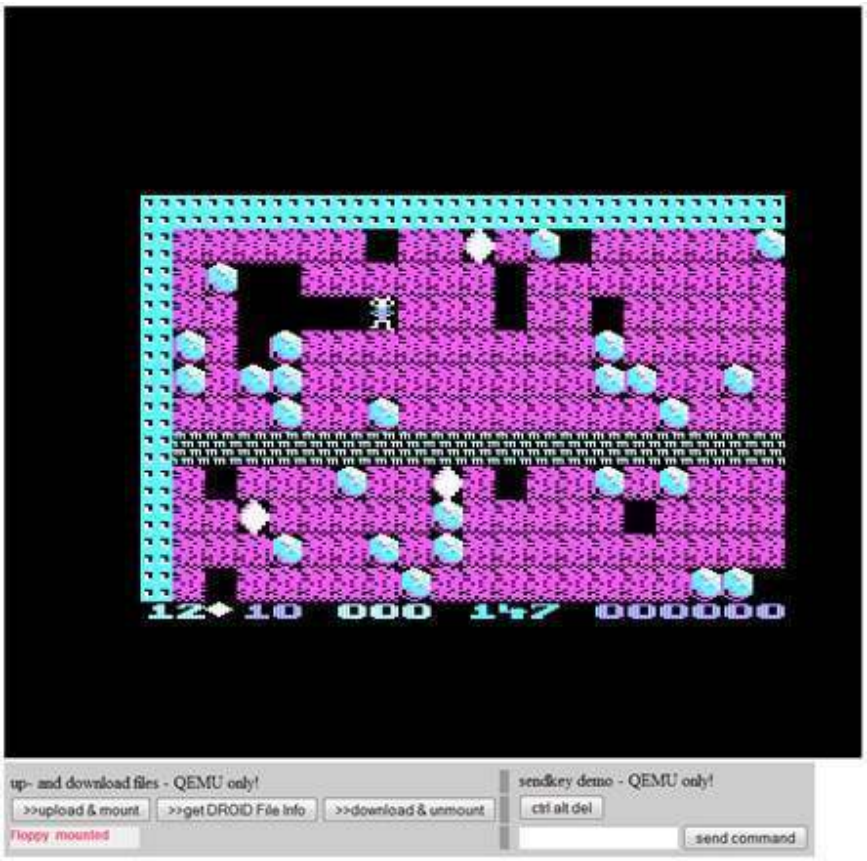


Figure 17: Boulder Dash running in GRATE

The look-and-feel is further preserved through authentic reproduction of speed and resemblance to original controls. The only notable issue we observed was the lack of sound. Some cases in the experimental objects perform an inspection for the existence of an audio adapter. Although the games managed to identify a sound card, audio output was not produced. Interactive material often relies on sound cues to guide the user or reveal further actions. In this sense, correct reproduction of audio elements should be incorporated in GRATE to preserve the overall gaming experience. It should be noted, however, that this issue might have arisen from configuration problems, rather than from GRATE itself. The remote environment has many benefits, but it also limits the extent that the user can investigate such issues and find their source. In any case, this makes a valid consideration for future work.

A final comment from experimentation with GRATE is the flexibility it provides through platform independency. GRATE only requires the existence of a Java Virtual Machine (JVM)[62] to be present on the host system. As part of the Testbed suite, GRATE inherits all the benefits that stem from the Planets Interoperability Framework. In our tests, GRATE was the only preservation tool that performed consistently both across Windows and Linux platforms. The Planets Testbed and GRATE further incorporate the expertise of the Planets community specifically for digital preservation. The scope of the remaining tools we examined is different: they focus primarily on business processes or simply “playing an old game on a new computer”[63]. Instead, the Testbed and GRATE strive to provide the community of digital users with not only efficient tools, but with the technical support that is necessary to successfully complete preservation actions.

[62] JVM is freely available for download from: http://www.java.com/en/download/manual.jsp

[63] Quote from discussion on an DOS gaming forum. When participants were asked to identify the suitability of DOSBox and other tools for digital preservation, it became evident that the community had not considered these issues thoroughly.

6.1.2 DOSBox and Wine with DOSBox

The results from DOSBox for Windows and from DOSBox running within Wine on Linux were identical. For the reader's convenience, we are presenting both cases under one section. References to DOSBox from here onwards describe both implementations of the tool collectively.

DOSBox has been a strong candidate in our experiments, because its focus is specifically on running obsolete MS-DOS video games. The community has invested a lot of effort in making this experience as effortless as possible. After GRATE, DOSBox was the second easiest tool to configure. It successfully rendered all cases in the experimental material with some problems observed. Support for game hardware (mouse, keyboard, soundcard, graphics adapter) was available with no parameterisation. DOSBox automatically mapped the hardware on the host machine to the DOS virtual environment. The speed of gameplay was also acceptable. All tests at the default 3,000 CPU circles, frame skip 0, apart from Alone in The Dark (CPU at 12,000 circles) and Hand of Fate (CPU at 10,000). These games were developed for faster processors and therefore performed slower at the default CPU speed. The only exception was Tetris, which executed considerably faster than the original – to the extent that the game was hardly playable. We used the built-in speed regulator to adjust this inconsistency, but this reduced the accuracy and performance of the graphics.

Figure 18: Prince of Persia in DOSBox

Figure 19: Hand of Fate introduction with speech in DOSBox

Problems were also identified in terms of response delay in input devices. Boulder Dash and Tetris were the most prominent cases, for which we were not able to find a solution. This had an immediate effect on the feel of the original gaming experience, because the games were playable but not without serious compromises. Possibly the strongest attribute of DOSBox is its ability to precisely emulate audio production, even for games with high demands and speech elements. In all cases examined, the audio was also well synchronised with video. Small errors were observed in Alone in the Dark, but with no serious implications on gameplay. Similarly, DOSBox offered authentic reproduction of image quality and colours (Figures 18-19).

6.1.3 DOSEMU for Linux

DOSEMU was one of the preservation tools that performed rather poorly in the experiments. Three out of the seven video games titles failed to render altogether. Boulder Dash froze at the introductory screen with no further response to user controls. Although the CD-ROM drive was recognised and mounted, DOSEMU could not access the files on the CD-ROM disks. As a result, Alone in the Dark and Hand of Fate halted execution at the initial inspection for the presence of the CD-ROM disk. Furthermore, only three of the examined titles were actually playable. The speed of Tetris was again significantly faster than the original. However, DOSEMU does not provide the DOSBox functionality to reduce the CPU circles – thus rendering the game unplayable. The remaining titles performed adequately in terms of speed, although we still observed a small deviation towards faster performance.

Input devices for game controls functioned consistently well for all working material. Unlike previous alternatives, no delays were observed in the response of the devices to user input. Other hardware required by the video games was not as compatible. Apart from the CD-ROM drive, we encountered further issues with audio reproduction. In some cases, the quality of sound (either music or sound effects) was low, with synchronisation problems as well. Overall image and colour quality was nevertheless as expected without deviations from the original execution (Figures 20-21). For these reasons, we concluded that DOSEMU would be more suitable for cases where Linux use is necessary and simple video games need to be preserved.

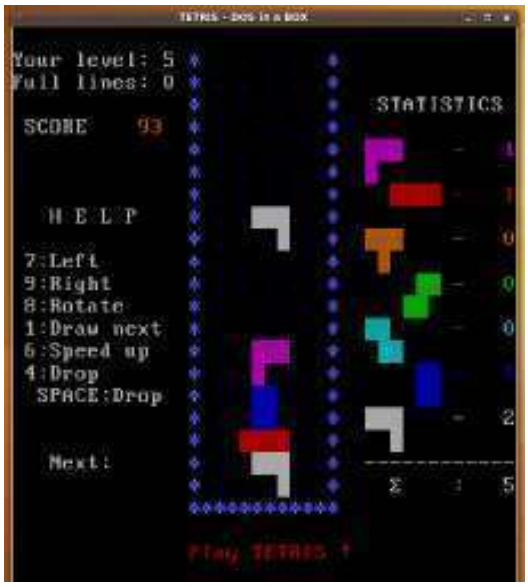


Figure 20: Tetris running in DOSEMU. The fast speed did not allow for gameplay.

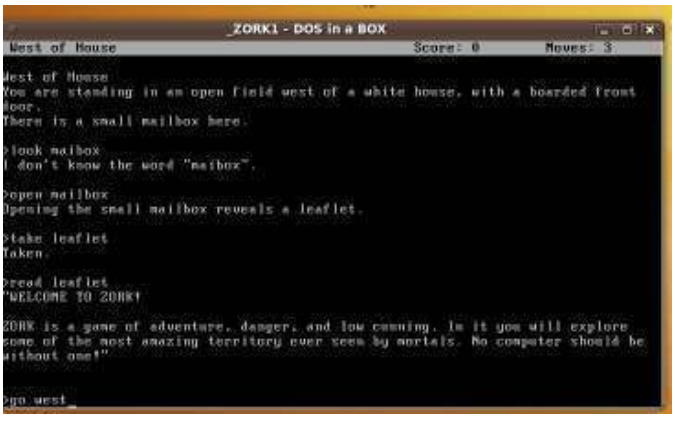


Figure 21: Zork 1 running in DOSEMU

6.1.4 MS-DOS Virtual Machines for Windows and Linux hosts

Similarly to Section 6.1.2 on DOSBox, the MS-DOS virtual machines on both Windows XP and Ubuntu Linux 9.10 returned identical results. We are therefore presenting their analysis collectively, referring henceforth to the DOS VM for both implementations.

The DOS VM yielded the lowest scores among all other preservation alternatives (see Appendix C). Despite the careful and time-consuming preparation (see Section 5.2), the results do not meet the current standards of preservation quality. Only two out of seven experimental cases were truly playable – and both involved simple text-based and ASCII graphics. In more detail, Boulder Dash stopped responding at the start-up screen with no further response to user input. Tetris was again running too fast to permit playability. Prince of Persia generated an error message within two minutes of gameplay. Problems were also experienced with the two CD-ROM games. Alone in the Dark stopped responding at start-up without producing an error message. Hand of Fate continuously reported insufficient memory issues (Figure 22).

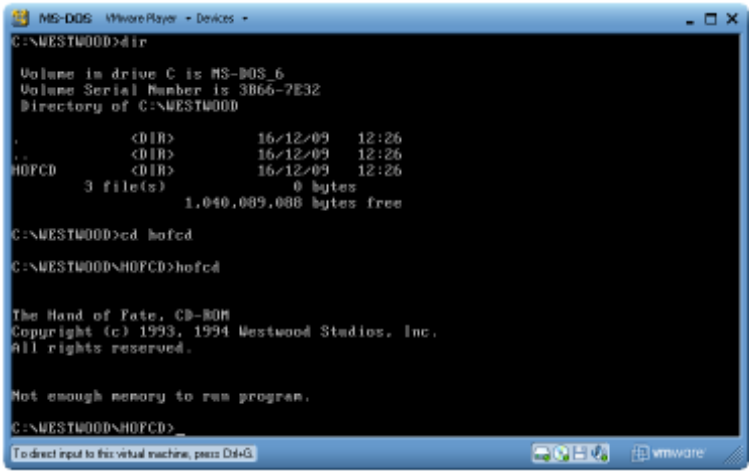


Figure 22: Memory issues with MS-DOS VM and Hand of Fate

In terms of speed, most titles executed faster than the original experience. Virtual Machines do not offer a straightforward manner to reduce the CPU circles, hence no solution was available to this problem. On the other hand, controls for the input devices yielded good response rates and resemblance to the original controls. Nonetheless, the overall feel of the gaming experience is significantly hindered by the aforementioned issues. The majority of problems stemmed from errors generated by the audio adapter configuration. Surprisingly, the diagnostic tool for Sound Blaster 16 reported that the sound card was properly configured, even producing the test audio samples. However, during the experiments most games crashed with error messages concerning sound. In the few cases where the audio adapter was recognised, severe errors were observed in quality and synchronisation to video components. Errors were not recorded for graphics, as the image quality and colours of the working titles was identical to the originals (Figure 23).

Figure 23: Prince of Persia in MS-DOS VM

6.2 Software Art

Experimentation with software art artefacts generally followed a similar pattern with the DOS video games with regards to results. Software art was marginally more complex material to experiment with, as variation occurs in platforms, degrees of interactivity and demands on system requirements. Hence, our observations have focused on the performance of the preservation tools with software art, rather than with specific computing environments (as in the case of MS-DOS in the previous Section). The results are organised on a tool-by-tool basis.

6.2.1 Testbed and GRATE

The many benefits we observed from the Planets Testbed and the GRATE emulation service (Section 6.1.1) have remained unaltered in experimentation with software art. However, a small decrease in speed was noted in comparison to the MS-DOS video games. This is possibly because the selected software art works have higher demands in processing power, memory allocation and capabilities for graphics (such as screen resolution and 3D acceleration). These issues did not severely affect the overall quality of the overall reproductions, but still influenced the feel of the original artistic experience. All examined artefacts rendered successfully on GRATE, apart from the a.shooter art game. Although the requirements for this work stated Windows 98 as the Operating System, it became evident from the experiments that a.shooter needs at least Windows XP to render successfully. Nonetheless, it was still possible to upload the setup executable in GRATE and run the installation application. Precisely because this is an error external to the preservation tool, we decided to consider GRATE as an inapplicable solution. GRATE does not offer virtualisation capabilities for Windows XP guest machines. Audio was not available within GRATE – similarly to the experiments with DOS games. The overall image quality and colours retained their original properties in the virtualised environment (Figures 24-25).

Figure 24: Caero for MS-DOS running in GRATE

The experiments with software art highlighted the advantages stemming from GRATE's integrated environment, which brings together many emulators and virtual machines. Through GRATE, it was possible to test different solutions with the same interface and without the need to install VMs on a local computer. To exhibit this flexibility, Figure 24 shows a screenshot with an artwork developed for MS-DOS while in Figure 25 a Windows 98 application is running in GRATE.

Figure 25: DataDada for Windows 98 running in GRATE

6.2.2 DOSBox and MS-DOS VM for Windows

In order to compare the results from GRATE to other alternatives for preserving DOS software art, we used DOSBox and a Virtual Machine with MS-DOS. The experimental corpus included only one such case of software (Caero), so this formed the basis for our observations. In the first instance, both DOSBox and the DOS VM successfully rendered the artefact. The speed in the VM was slightly slower than that produced by both DOSBox and GRATE. Also, the virtual machine experienced problems with audio and video reproduction. In more detail, the musical score of Caero was not rendered – although the software checked for and located the sound card. More acute errors were observed with colours in the VM. At certain points, the image became "posterised" with erroneous reproduction of the colour palette (Figure 26). On the other hand, DOSBox consistently and accurately reproduced the original artwork.

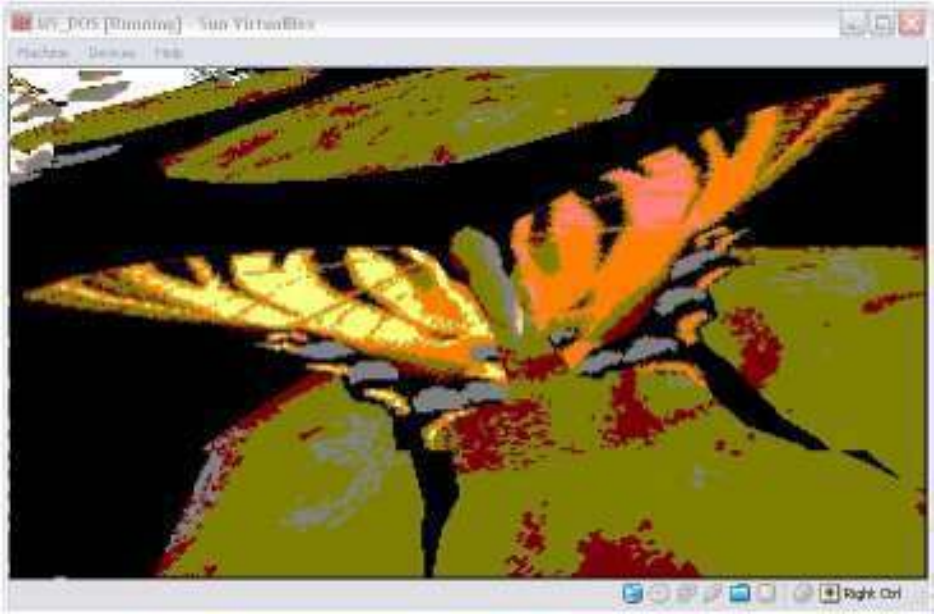


Figure 26: Posterised colours in a MS-DOS VM running Caero (Virtual Box)

6.2.3 Windows 98 VM

A virtual machine running Windows 98 was included in the experimentation, so as to enable comparisons with GRATE. Through our tests we discovered that the VM was able to render all artefacts (apart from a.shooter) including Caero developed for MS-DOS. However, the outputs were of varying quality. The speed of execution in all cases was slower than the original. In the case of [phage] and Prophecy (Figures 27-28) the speed was slower than GRATE, but remained the same for the rest of the cases. In terms of audio capacities, the VM produced disappointing results, with breaking sound quality in DataDada and Prophecy, and only monophonic sound in Caero.

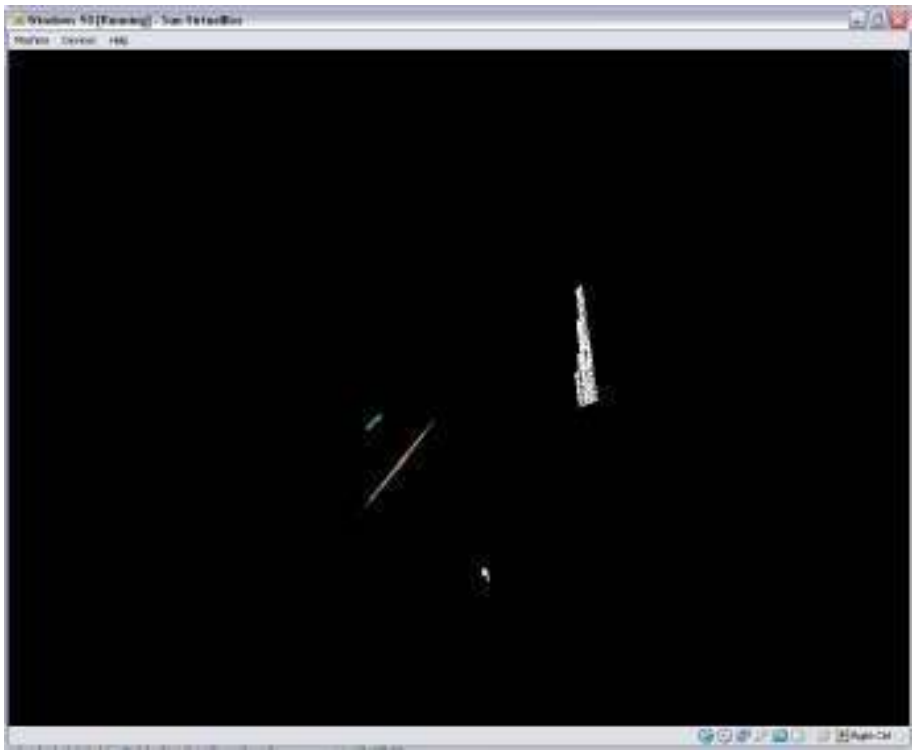


Figure 27: [phage] in Windows 98 VM (Virtual Box)

Figure 28: Prophecy in Windows 98 VM (Virtual Box)

The image quality was generally acceptable, with the exception of Caero that appeared very pixelated in full-screen mode. The colours correctly reproduced the originals for all cases except of Caero, which again experienced problems with posterised colours.

6.2.4 Wine for Linux

Wine was used in this set of experiments as a benchmark for quality of outcomes between the previously examined preservation tools. In the Plato preservation plan (see Appendix D), Wine achieved the highest score, but only because it was able to run the art game for Windows XP (a.shooter). Wine also managed to execute Caero, but with significant problems. In general, speed of execution was kept at acceptable levels, with only three artefacts performing more slowly than the original (phage, Prophecy and Caero). However, the range or the speed discrepancies was limited and the results did not severely affect the feel of the artistic experience.

In terms of sound reproduction, Wine performed in a reliable manner, with only minor breaking in Prophecy and Caero. The audio was consistently synchronised to video, except for Caero where the music score had not finished by the end of the video sequence. The problem was more noticeable in this case with sound effects. For instance, the auditory feedback of an explosion was produced 3 minutes after the visual representation of the same event. Similarly, the overall image quality and colours were reproduced at a high standard (Figure 29), on a par with GRATE. For the art game case in the sample collection, no problems were observed in terms of response delays from input devices; the overall feel of the controls precisely resembled the originals.

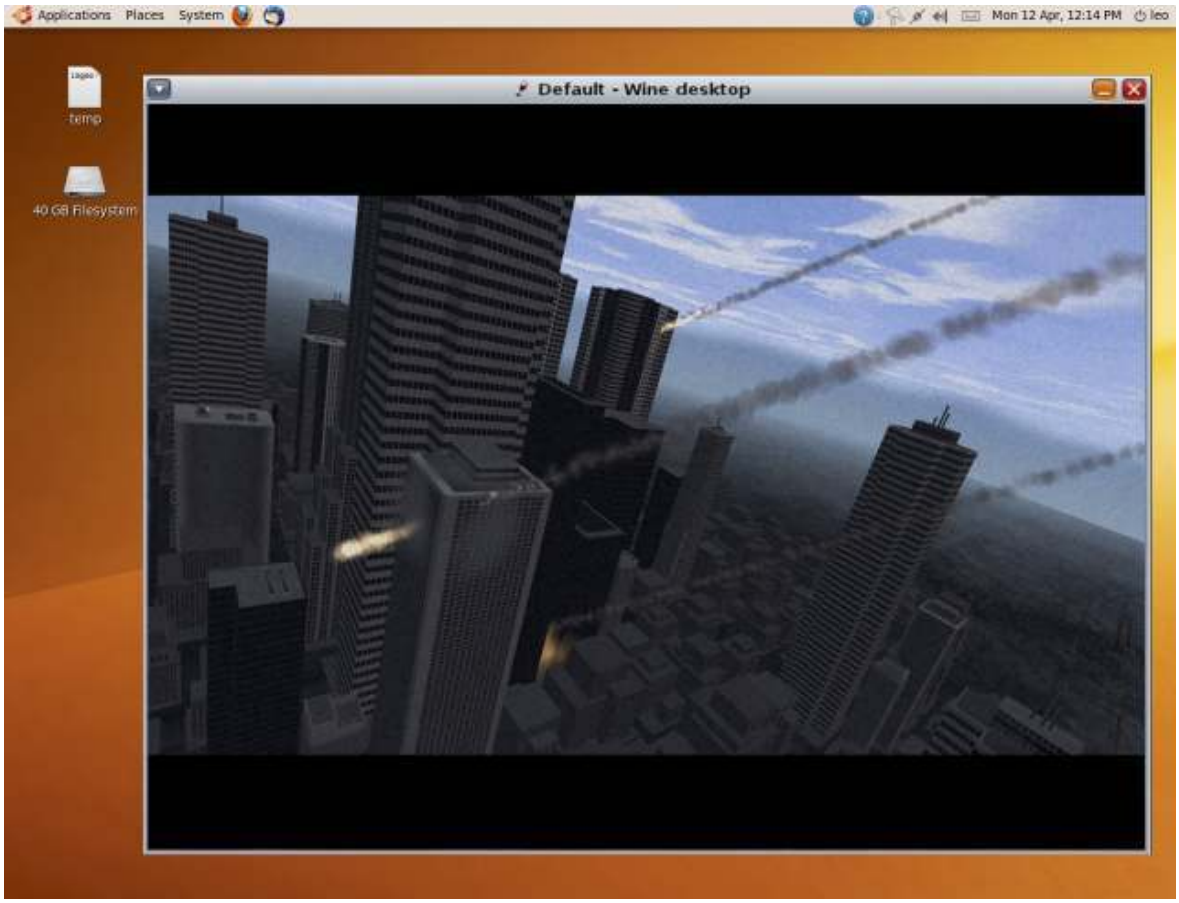


Figure 29: Prophecy running in Wine (Ubuntu Linux 9.10)

7. Conclusions

This report has presented the life-cycle of a set of experiments on the suitability of binary translation and virtualisation for digital preservation. The analysis focused on interactive and dynamic content, which better highlights the complexities of preserving software artefacts. In order to conduct the experimentation, we began with definitions of the techniques used, as well as disambiguation of the meaning of software preservation. From our review, it became evident that software is not just a means to render digital objects, but a computing component worthy of preservation in its own right. We discussed how and why software is as vulnerable to obsolescence as any other type of electronic information.

It has been shown that software preservation is not straightforward. Possibly this has been an inhibiting factor to engage with the subject on a wider scale. Although we recognise that the scope of our experiments is limited to specific genres, the results demonstrate a situation whereby time and effort need to be systematically invested for organisations to effectively preserve their software components. Video games and software art have been historically more prone to obsolescence, due to unclear intellectual property rights and lack of large-scale investments in preserving them systematically. However, our background research identified a strong number of parties – both individuals and organisations – who have taken action to preserve such software artefacts on their own initiative.

This effort represents – more than anything else – a roadmap to preserving software through binary translation and virtualisation. As expected, none of these techniques is the golden ticket to the software preservation problem. However, the experiments have revealed benefits and limitations not only of the overall approach, but of specific tools as well. Within these preservation tools, we examined the case of the Planets Testbed and its emulation / virtualisation service, GRATE. In this manner, it has been possible to situate the Testbed use beyond migration of documents. The results of the experiments testify that the Testbed is capable of handling complex interactive and dynamic artefacts, providing through GRATE a platform for users to measure the suitability of tools.

Throughout this report, we have emphasised on these benefits of the Testbed and GRATE based on our experiences with the other preservation tools examined. It is true that binary translators, such as DOSBox and Wine, performed well during the experiments with little errors. However, each one comes with its own dependencies. For instance, DOSBox operates in a command-line environment, which is not straightforward for the unacquainted user. What is more, the DOSBox environment is an abstraction of MS-DOS, with limited capabilities and therefore a scope limited to rendering DOS video games. On the other hand, Wine operates in the Linux platform, but it is not available or equally efficient on all Linux distributions. Customisations and configurations on Linux can be cumbersome and hence not a suitable choice for long-term preservation.

Virtual machines have emerged as an attractive approach for duplicating the behaviour of computing platforms – promising increased stability with reduced costs. Indeed, cost is an issue with virtualisation, particularly if an organisation is looking at this technology as solely a preservation solution. Virtualisation software is not only expensive to acquire, it is cost- and time-demanding to install and configure. Free virtualisation alternatives alleviate these costs, but they often require further customisation to become fully operational. For digital preservation of interactive and dynamic content, being “fully operational” is possibly more important than other aspects. In this study, the preparation of the virtual machines (Section 5.2 & 5.3) took considerably more time to complete than the experiments themselves. The major issues we encountered were:

- Lack of drivers for modern devices designed to work with obsolete operating systems. Even if drivers exist, they are most commonly not supported any longer by the manufacturer.
- Unexpected and non-standardised behaviours from the virtual machines. There is officially no guarantee that the current configurations on a VM will work in future computing environments.
- Implications stemming from the virtualisation software. Virtual machines are not stand-alone; they will always rely on some rendering platform. However, there are no assurances that (a) current virtualisation software will be available in the long term; and (b) current VMs will be compatible with future virtualisation software.
- Even at present there are compatibility issues between different virtualisation platforms. For instance, VMWare Tools will provide a virtual machine with all the necessary system software to

operate fully. The same virtual machine cannot use this system software when rendered in another platform (e.g. Virtual Box).

From the above it becomes clear that preserving software through virtualisation means preserving the virtualisation platforms themselves. It also presupposes preservation of the systems software - such as operating systems and device drivers. In this study, we have not dealt with these issues. However, their vital role in preservation through virtualisation suggests that further work needs to be conducted in this area.

The Planets Testbed allays these issues, because it is a controlled environment for experimentation. The tools in the Testbed are configured and maintained by technical staff and preservation specialists. Systems software and customisations are pre-configured and ready for the user to employ in research. In our experiments, the Testbed and GRATE produced outputs of high standards that are comparable to those of dedicated, stand-alone tools. Figures 30-31 show the quantified experimentation results from Plato for both sets of experimental material. The results clearly show that the Testbed and GRATE performed significantly better than the stand-alone virtualisation alternatives.

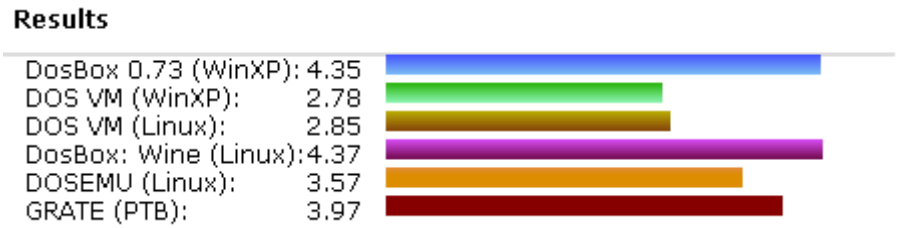


Figure 30: Quantified experimentation results for Video Games (Appendix C)

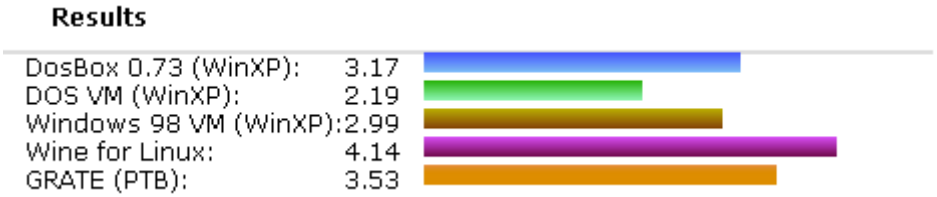


Figure 31: Quantified Experimentation results for Software Art (Appendix D)

In both cases, the binary translators have acquired the highest scores. This is not necessarily because they are more suitable for preservation. There are other – non quantifiable – aspects that need to be considered. Unlike the integrated approach of GRATE, DOSBox is an application dedicated to rendering MS-DOS video games. It is therefore expected that it offers a comprehensive solution for preserving this specific type of software. DOSBox is inevitably limited to MS-DOS applications and cannot be used for experimentation or indeed preservation of other software. This is more obvious in the results from Software art experiments. DOSBox scored lower than Wine and GRATE, because it cannot handle the entire range of requirements that the sample material called for. On the other hand, Wine was able to run all artefacts in the experimental corpus. However, for video games Wine was “assisted” by DOSBox. In other words, it was DOSBox which enabled the execution of the MS-DOS games through Wine. For Software art, Wine managed to execute all samples and scored the highest in the results. This is because one case (the art game a.shooter) was erroneously reported to work with Windows 98 when it was a Windows XP application. Wine was the only preservation tool capable of rendering Windows XP software, thus gaining a comparative advantage over the remaining alternatives.

As a final remark, interpretation of the findings should consider the experimental nature of the Planets Testbed and GRATE. Both are outputs of ongoing research and therefore represent products in development. The remaining tools examined in this report are all “established” in their field. Even more so, some are commercial (and rather costly) solutions. The experiments have proven that the Testbed offers equivalent results to these alternatives and the advantage of a freely available, integrated, remotely accessed environment. With these in mind, we expect future iterations of the Planets Testbed and GRATE to offer the community a complete solution for preserving dynamic and interactive content.

Appendix A Significant Properties for Video Games

Process Characteristics	
Usability	
Game start	With installation / Without installation *Specifies whether a video starts with or without installation.*
Game Playable	Yes / No *Specifies whether a video game is playable. Measurement takes into account controls through input devices, image quality and audio reproduction.*
Configurability	
Game Configuration	Manual/Pre-configured/Not needed *Specifies whether the audio and video configuration of a video game is done by the user (Manual), by the game (Pre-configured) or whether no configuration is needed.*
Global System Preferences	Manual/Pre-configured/Not applicable (single) *Specifies whether configuration of a preservation tool is either done by the user (Manual), by the tool (Pre-configured) or whether no configuration is needed.*
Ease of Configuration	Positive number *Measures the ease of configuration, in terms of time necessary to set up the preservation tool. Expressed in minutes.*
Costs	
Cost for acquiring preservation software	None (freeware)/Minimal/High *Specifies the costs incurred by acquiring a preservation tool. None for freeware, minimal for prices below £100, high otherwise.*
Number of games supported	Less than 10/10-100/101-500/501-2000/more than 2000 *Specifies the number of video games supported by a preservation tool. Evidence is provided by (a) the capabilities of the tool; (b) listings of supported titles by the user community.*
Object Characteristics	
Game Renders	Yes / No *Specifies whether a video game executes within a preservation tool. A game that renders is not necessarily playable.*
Speed	As in original/Faster than original/Slower than original/Not applicable *Specifies the speed of execution of a video game within a preservation tool. Evidence is based on (a) comparison with results in original environment; (b) playability. Property not applicable for games that do not render.*

Interactivity / Input / Controls with Standard PC input devices	
Controls resemble originals	Yes / No / Not applicable *Specifies whether controls for a video game through standard PC input devices (keyboard, mouse) resemble the controls in the original environment. Property not applicable for games that do not render.*
Response delay	Considerable delay/Short delay/No delay/Not applicable *Specifies potential delay between user input through controls and response to the input by the video game. Property not applicable for games that do not render.*
Feel of original gaming experience	Not acceptable/Playable/Perfectly recreated/Not applicable *Specifies the degree to which the gaming experience from the original environment is replicated in the virtual environment. Property not applicable for games that do not render.*
Support for Game hardware	Yes / No *Specifies whether the preservation tool offers adequate support for the video game requirements. Such hardware includes device drivers for audio and video adapters.*
Support for Special Controller	Not applicable/Supported (drivers needed)/Supported (no drivers needed) *Specifies whether special controllers (see Section 3.1) are supported by a preservation tool. Alternatives distinguish between support with need for drivers and without. Property not applicable for games that do not require special controller or games that do not render.*
Sound	
Music quality	No music/Unacceptable/Acceptable/Near Perfect/As in original/Not applicable *Measures the level of quality of the music output of a video game. Property not applicable for games that do not render or with no music scores.*
Music Synchronous to video	Not applicable/Severe errors/Small errors/No errors *Specifies the degree to which music reproduction is synchronised with video output. Property not applicable for games that do not render or with no music scores.*
Sound effects quality	No sound effects/Unacceptable/Acceptable/Near Perfect/As in original/Not applicable *Measures the level of quality of the sound effects' output of a video game. Property not applicable for games that do not render or with no music scores.*
Sound effects Synchronous to video	Not applicable/Severe errors/Small errors/No errors *Specifies the degree to which sound effects reproduction is synchronised with video output. Property not applicable for games that do not render or with no sound effects.*

Speech quality	No speech/Unacceptable/Acceptable/Near Perfect/As in original/Not applicable *Measures the level of quality of the speech output of a video game. Property not applicable for games that do not render or with no music scores.*
Speech Synchronous to video	Not applicable/Severe errors/Small errors/No errors *Specifies the degree to which speech reproduction is synchronised with video output. Property not applicable for games that do not render or with no speech.*
Graphics	
Overall image quality	Nothing displayed/Severe errors that affect gameplay/Errors but not affecting gameplay/Near perfect/As in original *Specifies the quality level of the overall image quality of the executed video game. Evidence include resolution, positioning and composition. Property set to 'Nothing displayed' for games that do not render.*
Colours	Not applicable/Severe errors affecting gameplay/Errors but not affecting gameplay/Near perfect/As in original *Specifies the degree of accurate reproduction of original colours. Evidence include bit depth and colour palette. Property not applicable for games that do not render.*

Context and Data Characteristics	
Metadata / Documentation	
Box artwork available	Yes / No *Specifies whether digital versions of the game's original box are available.*
Manual available	Yes / No *Specifies whether a digitised version of the instructions manual is available for a video game.*
Original media available	Yes / No *Specifies whether the original storage media (floppy disks, CD-ROMs) of a video game are available.*
System requirements specification	Yes / No *Specifies whether a specification of system requirements is provided with a video game.*
Infrastructure	
Scalability	
Modular design	Code not available/Code not modular/Code modular (one system support)/Code modular (multiple system support) *Specifies whether a preservation tool provides support for one or more systems.*

Portability	Not applicable/Platform dependent/Platform independent *Specifies whether a preservation tool can be used in one or multiple platforms.*
Stability	
Development status	Commercial (open source)/Commercial (non-open source)/Free (open source)/Free (non-open source) *Specifies the license under which a preservation tool is provided to the community.*
Current support	Yes / No *Specifies whether support exists for a preservation tool by the developers / marketing organisation.*
In development	Yes / No *Specifies whether a preservation tool is currently and actively developed / supported.*
Active community	Yes / No *Specifies whether there exists and active community of users for a preservation tool, who can offer further support and guidance.*
Host-Guest OS communication	Via Network/Native/Via virtual media/None *Specifies the manner that the virtual environment of a preservation tool communicates with that of the host machine.*
Legality	
Alterations to original permitted	Yes / No *Specifies whether it is allowed to make changes to the original code of a video game. This is useful in preservation for, e.g. reverse engineering.*
Copies of original permitted	Yes / No *Specifies whether a video game can be freely copied for backup, distribution or public viewing.*
Game IP status	Freeware/Protected *Specifies the intellectual property status of a video game.*
Operating System IP status	Freeware/Protected *Specifies the intellectual property status of the Operating System installed on a virtual environment through a preservation tool.*
Media	
Keyboard/mouse support	Yes / No *Specifies whether a preservation tool offers native support for keyboards and pointing devices.*

CD-ROM support	Manual/Pre-configured/None *Specifies whether a preservation tool offers support for CD-Rom drives. Options are support for manual configuration by users, pre-configured in the preservation solution or no support at all.*
Sound support	Manual/Pre-configured/None *Specifies whether a preservation tool offers support for audio adapters. Options are support for manual configuration by users, pre-configured in the preservation solution or no support at all.*
OS pre-installed	Yes / No *Specifies whether a preservation solution comes with a pre-installed Operating System by default.*

Appendix B Significant Properties for Software Art

Process Characteristics	
Usability	
Installation requirements	With installation / Without installation/Solution Inapplicable *Specifies whether an artefact starts with or without installation. Property set to 'Solution inapplicable' for preservation tools that cannot render the artefact.*
Artefact Playable (if interactive)	Yes/No/Not Interactive/Solution inapplicable *Specifies whether an artefact is playable. Measurement takes into account controls through input devices, image quality and audio reproduction. Property set to 'Not interactive' for artefacts without interactivity features. Property set to 'Solution inapplicable' for preservation tools that cannot render the artefact.*
Configurability	
Artefact Configuration	Manual/Pre-configured/Not needed/Solution inapplicable *Specifies whether the audio and video configuration of an artefact is done by the user (Manual), by the artefact (Pre-configured) or whether no configuration is needed. Property set to 'Solution inapplicable' for preservation tools that cannot render the artefact.*
Global System Preferences	Manual/Pre-configured/Not applicable *Specifies whether configuration of a preservation tool is either done by the user (Manual), by the tool (Pre-configured) or whether no configuration is needed.*
Ease of Configuration	Easy (less than 15 mins)/Moderate (16-60 mins)/Difficult (more than 60 mins) *Measures the ease of configuration, in terms of time necessary to set up the preservation tool. Expressed in minutes.*
Costs	
Cost for acquiring preservation software	None (freeware)/Minimal/High *Specifies the costs incurred by acquiring a preservation tool. None for freeware, minimal for prices below £100, high otherwise.*
Number of artefacts supported	Less than 10/10-100/101-500/501-2000/more than 2000 *Specifies the number of artefacts supported by a preservation tool. Evidence is provided by (a) the capabilities of the tool; (b) listings of supported titles by the user community.*

Object Characteristics	
Artefact Renders	Yes/No/Solution Inapplicable *Specifies whether an artefact executes within a preservation tool. An artefact that renders is not necessarily playable. Property set to 'Solution inapplicable' for preservation tools that cannot render the artefact.*
Speed	As in original/Faster than original/Slower than original/Not applicable/Solution inapplicable *Specifies the speed of execution of an artefact within a preservation tool. Evidence is based on (a) comparison with results in original environment; (b) playability. Property not applicable for artefacts that do not render. Property set to 'Solution inapplicable' for preservation tools that cannot render the artefact.*
Interactivity / Input / Controls with Standard PC input devices	
Controls resemble originals	Yes/No/Not applicable/Solution inapplicable *Specifies whether controls for an artefact through standard PC input devices (keyboard, mouse) resemble the controls in the original environment. Property not applicable for artefacts that do not render / are not interactive. Property set to 'Solution inapplicable' for preservation tools that cannot render the artefact.*
Response delay	Considerable delay/Short delay/No delay/Not applicable/Solution inapplicable *Specifies potential delay between user input through controls and response to the input by the artefact. Property not applicable for artefacts that do not render / are not interactive. Property set to 'Solution inapplicable' for preservation tools that cannot render the artefact.*
Feel of original artistic experience	Not acceptable/Playable/Perfectly recreated/Not applicable/Solution inapplicable *Specifies the degree to which the artistic experience from the original environment is replicated in the virtual environment. Property not applicable for artefacts that do not render. Property set to 'Solution inapplicable' for preservation tools that cannot render the artefact.*
Sound	
Music quality	No music/Unacceptable/Acceptable/Near Perfect/As in original/Not applicable/Solution inapplicable *Measures the level of quality of the music output of an artefact. Property not applicable for artefacts that do not render or with no music. Property set to 'Solution inapplicable' for preservation tools that cannot render the artefact.*

Music Synchronous to video	Not applicable/Severe errors/Small errors/No errors/Solution inapplicable *Specifies the degree to which music reproduction is synchronised with video output. Property not applicable for artefacts that do not render or with no music scores. Property set to 'Solution inapplicable' for preservation tools that cannot render the artefact.*
Graphics	
Overall image quality	Nothing displayed/Severe errors that affect experience/Errors but not affecting experience/Near perfect/As in original/Solution inapplicable *Specifies the quality level of the overall image of the executed artefact. Evidence include resolution, positioning and composition. Property set to 'Nothing displayed' for artefacts that do not render. Property set to 'Solution inapplicable' for preservation tools that cannot render the artefact.*
Colours	Not applicable/Severe errors affecting experience/Errors but not affecting experience/Near perfect/As in original/Solution inapplicable *Specifies the degree of accurate reproduction of original colours. Evidence include bit depth and colour palette. Property not applicable for artefacts that do not render. Property set to 'Solution inapplicable' for preservation tools that cannot render the artefact.*
Context and Data Characteristics	
Metadata / Documentation	
Instructions available	Yes / No / Solution inapplicable *Specifies whether instructions available for an artefact.*
Original media available	Yes / No / Solution inapplicable *Specifies whether the original storage media (floppy disks, CD-ROMs) of an artefact are available.*
System requirements specification	Yes / No / Solution inapplicable *Specifies whether a specification of system requirements is provided with an artefact.*
Infrastructure	
Scalability	
Modular design	Code not available/Code not modular/Code modular (one system support)/Code modular (multiple system support) *Specifies whether a preservation tool provides support for one or more systems.*
Portability	Not applicable/Platform dependent/Platform independent *Specifies whether a preservation tool can be used in one or multiple platforms.*

Stability	
Development status	Commercial (open source)/Commercial (non-open source)/Free (open source)/Free (non-open source) *Specifies the license under which a preservation tool is provided to the community.*
Current support	Yes / No *Specifies whether support exists for a preservation tool by the developers / marketing organisation.*
In development	Yes / No *Specifies whether a preservation tool is currently and actively developed / supported.*
Active community	Yes / No *Specifies whether there exists and active community of users for a preservation tool, who can offer further support and guidance.*
Host-Guest OS communication	Via Network/Native/Via virtual media/None *Specifies the manner that the virtual environment of a preservation tool communicates with that of the host machine.*
Legality	
Alterations to original permitted	Yes / No *Specifies whether it is allowed to make changes to the original code of an artefact. This is useful in preservation for, e.g. reverse engineering.*
Copies of original permitted	Yes / No *Specifies whether an artefact can be freely copied for backup, distribution or public viewing.*
Operating System IP status	Freeware/Protected *Specifies the intellectual property status of the Operating System installed on a virtual environment through a preservation tool.*
Media	
Keyboard/mouse support	Yes / No *Specifies whether a preservation tool offers native support for keyboards and pointing devices.*
CD-ROM support	Manual/Pre-configured/None *Specifies whether a preservation tool offers support for CD-Rom drives. Options are support for manual configuration by users, pre-configured in the preservation solution or no support at all.*

Sound support	Manual/Pre-configured/None *Specifies whether a preservation tool offers support for audio adapters. Options are support for manual configuration by users, pre-configured in the preservation solution or no support at all.*
OS pre-installed	Yes / No *Specifies whether a preservation solution comes with a pre-installed Operating System by default.*

Appendix C Plato preservation plan: Video Games

Validate plan for Digital Preservation of Video Games (DOS)

Report creation date: Apr 13, 2010 1:01:36 PM

Display Changelogs

Plan name	Digital Preservation of Video Games (DOS)
Current state	Plan Defined
Plan description	This preservation plan focuses on assessing the suitability of binary translation and virtualisation for preserving DOS-based video games. The assessment uses material for different eras of computer gaming, spanning from 1982 to 1993.
Responsible planners	Leo Konstantelos
Organization	HATII, Univ. of Glasgow

Identification code

Planning purpose This plan is meant to assess the suitability of different binary translation / virtualisation alternatives for preserving video games developed for the DOS platform.

Plan relations This is the organisation's first attempt to create a preservation plan for the DOS-based video games collection.

Triggers

Trigger	Description
New Collection	we are evaluating the preservation potential of alternatives for a newly established collection of video games created for the MS-DOS platform. The organisation has assembled the collection from material previously scattered among other artefacts and a recent donation by a patron.

[↑] Policies

No policies have been defined yet.

Document types Video games created for the Disk Operating System (DOS); each consisting of a number of files; PC CD-ROM games in ISO format.

Mandate For the purposes of this plan, we assume that the games form part of an organisation's collection of vintage video games. The organisation's mission is to ensure long-term access to this collection without the necessity of maintaining obsolete systems (as in the case of a computer museum). Instead, the organisation wants to test how and to what degree the collected video games can be rendered and playable on modern systems. The solution should offer capabilities for encapsulating the virtual system for easy portability in the future.

Designated community The designated community in this instance are the users of the organisation's collection. There is no mandate to make the material available online. The users of the collection range from scholars investigating the field of video gaming, to gamers and possibly the general public.

Applying policies Due to IP / copyright restrictions, the organisation is planning to preserve only games for which it possesses the original or freeware material.

Relevant organisational procedures and workflows Currently the organisation holds the manuals and original boxes of the video games. These specify minimum requirements for running the software, as well as other characteristics such as multilingual support, classification rating (suitability for different audiences, e.g No-one younger than 18 may play the game).

Contracts and agreements specifying preservation rights None; the organisation has the right to preserve the material as specified by the games' copyright statement by virtue of possesing an original version. Access restrictions may apply for public usage of the preserved - or otherwise - game.

Reference to agreements of maintenance and access

Samples description: We have selected 7 video games created for the DOS platform from the collection, which represent different eras of game development. These include: text-based, ASCII graphics, CGA-only graphics, VGA graphics with simple sound effects (pc speaker), games with complex graphics (SVGA support, requirement for CD ROM support, advanced requirements for sound and speech). The latter categories are the hardest to preserve, as the preservation solution needs to map modern computer hardware to the specifications of the obsolete devices that the games need to render successfully.

Collection profile

Collection ID:	
Description:	Physical collection of boxed video games designed for the DOS platform. Each object consists of one or more diskettes or CD-ROMs and manuals.
Type of objects:	Data from diskettes converted to virtual floppy images. Data from CD-ROMs converted to iso and nrg images. Manuals converted to pdf files.
Number of objects:	
Expected growth rate:	Moderate, provided that (a) the designated community exhibits adequate interest in the current collection; (b) copyright issues with some titles are resolved; and (c) further contributions are made to the current collection.

Name	Short name	Description	Original environment	Data	Object-format
Zork I - The Great Underground Empire	ZORK	Zork I is the first game made commercially available to the general public. It was published by Infocom in 1982 when personal home computers were a brand new thing, with no memory and of course no sound or capability for graphics at all. It was originally published on 5.25-inch floppies.	DOS (IBM PC)	No data	**PUID: Name: Version: mime-type:**
Rogue - The Adventure Game	ROGUE	Originally developed in 1980 for Unix mainframe systems, Rogue eventually found its way to personal computers, including the IBM PC in 1984. An ASCII-character-cell classic, Rogue not only spawned a genre known as roguelikes, but also mothered the action role-playing game (RPG).	DOS (IBM PC)	No data	**PUID: Name: Version: mime-type:**
Boulder Dash	BD	Boulder Dash, originally released in 1984, is a classic series of computer games for the Atari 400/800, Apple II, MSX, ZX Spectrum, Commodore 64, and ColecoVision home computers, and later ported to the NES, Acorn Electron, PC, Amstrad CPC, Amiga and many other platforms. It was created by Peter Liepa and Chris Gray, and on October 28, 1983, acquired and later published by First Star Software, which still owns the rights to the game. The game features only CGA graphics.	DOS (IBM PC)	No data	**PUID: Name: Version: mime-type:**
TETRIS	TS	Tetris is a video games that requires little description. Having sold more than 70m copies for different platforms, Tetris has been voted in second place of the IGN "100 Greatest Video Games of All Time". The version used here is the port to IBM PC running DOS.	DOS (IBM PC)	No data	**PUID: Name: Version: mime-type:**
Prince of Persia	PP	Developed by Brøderbund and released in 1989, Prince of Persia impressed audiences by its innovative fluent character movement and well-designed surroundings. The game has been characterised by many as one of the greatest platform games of all times. The game uses VGA graphics and PC speaker sound effects.	DOS (IBM PC) Minimum System requirements: 512Kb RAM (640 Kb for VGA grahics); IBM/Tandy or 100% Compatibles; Floppy Disk Drive	No data	**PUID: Name: Version: mime-type:**
Alone in the Dark	ATD	Alone in the Dark is a 1992 survival horror game developed by Infogrames. The game set the standard for later rival popular survival horror games such as Resident Evil and Silent Hill. Items and characters in Alone in the Dark are three-dimensional, rendered upon a two-dimensional fixed background. Mixing polygons and 2D prerendered background images required a fixed camera angle, which designers used to their advantage to create dramatic scene setups appropriate for a horror-themed game. The sample record used here is the CD-ROM version of the game, which features speech throughout the game plus sound effects and music.	DOS (IBM PC) Minimum System Requirements: PC AT 16Mhz; DOS 5.0 or Higher; 6MB RAM; VGA Graphics Card; 8MB Hard Disk Space; CD-ROM drive with sound output; MSCDEX 2.2 or Higher	No data	**PUID: Name: Version: mime-type:**

Name	Short name	Description	Original environment	Data	Object-format
The Legend of Kyrandia Book Two: The Hand of Fate	HOF	Developed by Westwood Studios Inc. and released in 1993 by Virgin Games, Hand of fate is the second installement in the Legenf of Kyrandia trilogy. The Cd-ROM version used here features full speech throughout the game, music and sound effects.	DOS (IBM PC) Minimum System Requirements: 386 20 MHz; DOS 5.0 or Higher; 4 MB RAM; CD ROM Drive; VGA or MCGA Graphics Card; MSCDEX 2.2 or Higher; Sound Cards For Voice - SoundBlaster or SoundBlaster Pro and 100% Compatibles, AdLib Gold, and AdLib Gold Standard; Sound Cards For Music - SoundBlaster or SoundBlaster Pro and 100% Compatibles, AdLib, AdLib Gold, AdLib Gold Standard, Roland MT-32/LAPC-1/Sound Canvas and General Midi; 5 MB of Free Hard Drive Space; Microsoft Compatible Mouse and Keyboard	No data	**PUID:** **Name:** **Version:** **mime-type:**

Node	Weight	Total weight	Scale and Description
DOS games Preservation Requirements	1	1	
Process Characteristics	0.1	0.1	
Usability	0.6	0.06	
Game start	0.2	0.01	Ordinal With Installation or Without Installation
Game Playable	0.8	0.05	Boolean Yes or No
Configurability	0.4	0.04	
Game Configuration	0.2	0.01	Ordinal Manual, Pre-configured or Not needed
Global System Preferences	0.5	0.02	Ordinal Manual, Pre-configured or Not applicable (single)
Ease of Configuration	0.3	0.01	Positive number [Minutes] (single)
Costs	0.1	0.1	
Cost for acquiring preservation software	0.5	0.05	Ordinal None (freeware), Minimal or High (single)
Number of games supported	0.5	0.05	Ordinal Less than 10, 10-100, 101-500, 501-2000 or more than 2000 (single)
Object Characteristics	0.3	0.3	
Game Renders	0.2	0.06	Boolean Yes or No
Speed	0.1	0.03	Ordinal As in original, Faster than original, Slower than original or Not applicable
Interactivity	0.2	0.06	
Input	1	0.06	
Controls with Standard PC input devices	0.4	0.02	
Controls resemble originals	0.4	0.01	Ordinal Yes, No or Not applicable
Response delay	0.3	0.01	Ordinal Considerable delay, Short delay, No delay or Not applicable
Feel of original gaming experience	0.3	0.01	Ordinal Not acceptable, Playable, Perfectly recreated or Not applicable
Support for Game hardware	0.4	0.02	Boolean Yes or No
Support for Special Controller	0.2	0.01	Ordinal Not applicable, Supported (drivers needed) or Supported (no drivers needed)

Sound	0.2	0.06		
Music	0.33	0.02		
Quality	0.7	0.01		
	Ordinal	No music, Unacceptable, Acceptable, Near Perfect, As in original or Not applicable		
Synchronous to video	0.3	0.01	Ordinal	Not applicable, Severe errors, Small errors or No errors
Sound effects	0.34	0.02		
Quality	0.7	0.01		
	Ordinal	No sound effects, Unacceptable, Acceptable, Near Perfect, As in original or Not applicable		
Synchronous to video	0.3	0.01	Ordinal	Not applicable, Severe errors, Small errors or No errors
Speech	0.33	0.02		
Quality	0.7	0.01		
	Ordinal	No speech, Unacceptable, Acceptable, Near Perfect, As in original or Not applicable		
Synchronous to video	0.3	0.01	Ordinal	Not applicable, Severe errors, Small errors or No errors
Graphics	0.2	0.06		
Overall Image quality	0.5	0.03		
	Ordinal	Nothing displayed, Severe errors that affect gameplay, Errors but not affecting gameplay, Near p		
Colours	0.5	0.03		
	Ordinal	Not applicable, Severe errors affecting gameplay, Errors but not affecting gameplayNear perfect c		
Network Support	0.1	0.03		
Network protocols	0.33	0.01	Ordinal	Not applicable, Supported or Not supported
Use of original servers	0.34	0.01	Ordinal	Not applicable, Supported or Not supported
Lag for network play	0.33	0.01	Ordinal	Not applicable, Major delay, Minor delay or No delay
Context and Data Characteristics	0.1	0.1		
Metadata	1	0.1		
Documentation	1	0.1		
Box artwork available	0.2	0.02	Boolean	Yes or No
Manual available	0.3	0.03	Boolean	Yes or No
Original media available	0.3	0.03	Boolean	Yes or No
System requirements specification	0.2	0.02	Boolean	Yes or No
Infrastructure	0.1	0.1		
Scalability	0.4	0.04		
Modular design	0.4	0.02		
	Ordinal	Code not available, Code not modular, Code modular (one system support) or Code modular (mu		
Portability	0.6	0.02	Ordinal	Not applicable, Platform dependent or Platform independent (single)
Stability	0.4	0.04		
Development status	0.2	0.01		
	Ordinal	Commercial (open source), Commercial (non-open source), Free (open source) or Free (non-oper		

Current support	0.25	0.01	Boolean Yes or No (single)
In development	0.35	0.01	Boolean Yes or No (single)
Active community	0.2	0.01	Boolean Yes or No (single)
Host-Guest OS communication	0.2	0.02	Ordinal Via Network, Native, Via virtual media or None (single)
Legality	0.1	0.1	
Alterations to original permitted	0.25	0.02	Boolean Yes or No
Copies of original permitted	0.25	0.02	Boolean Yes or No
Game IP status	0.25	0.02	Ordinal Freeware, Abandonware or Protected
Operating System IP status	0.25	0.02	Ordinal Freeware, Abandonware or Protected (single)
Media	0.2	0.2	
Keyboard/mouse support	0.3	0.06	Boolean Yes or No (single)
CD-ROM support	0.3	0.06	Ordinal Manual, Pre-configured or None (single)
Sound support	0.3	0.06	Ordinal Manual, Pre-configured or None (single)
OS pre-installed	0.1	0.02	Boolean Yes or No (single)

Importance factors comments:
Description:
Attached files: **File name**
Evaluation comments:

Name	Description	Reason for considering	Config settings	Necessary resources
DosBox 0.73 (WinXP)	DOSBox is a binary translator working as a command-line program, configured either by a set of command-line arguments or by editing a plain text configuration file. For ease of use, several graphical front-ends have been developed by the user community. DOSBox is a full CPU emulator, capable of running DOS programs that require the CPU to be in either real mode or protected mode. Other similar programs, such as dosemu or VDMs for Windows and OS/2, provide compatibility layers and rely on virtualization capabilities of the 386 family processors. Since DOSBox can emulate its CPU by interpretation, it is independent of its host CPU. However, on systems which provide the i386 instruction set, the option to use dynamic instruction translation is available in DOSBox. Though this setting is less accurate and reliable, it is faster than interpretive CPU emulation.	It is capable of emulating many types of graphics and sound hardware. Graphics emulation includes text mode, Hercules, CGA (including composite and 160x100x16 tweaked modes), Tandy, EGA, VGA (including Mode X and other tweaks), VESA, and full S3 Trio 64 emulation.[3] Sound hardware that can be emulated includes the PC speaker, AdLib, Gravis Ultrasound, Tandy, Creative Music System/GameBlaster, Sound Blaster 1.x/2.0/Pro/16, MPU-401, and Disney Sound Source. (MT-32/CM-32L emulation is included in unofficial builds, but not in the official source code repository due to need for copyrighted ROM images.) A component that differentiates DOSBox from other emulators is its ability to simulate peer-to-peer or Internet/Intranet networking. This includes modem simulation over TCP/IP, allowing for DOS modem games to be played over modern LANs or the Internet, and IPX network tunneling, which allows for old IPX DOS multiplayer games to be played as UDP/IP over modern LANs or the Internet. Win32 and Linux specific builds support direct serial port access. DOSBox contains its own internal DOS-like shell, rather than being a fully virtual PC emulator like Bochs. This means that it can be used without owning a license to any real DOS operating system. Most commands that are typically used in	Default configuration, plus mounting a virtual C:\ drive, a virtual CD-ROM or ISO images as virtual CD-ROMs through the imgmount command.	DosBox 0.73 Installation files Windows 9x / XP / Vista 5MB free space

Name	Description	Reason for considering	Config settings	Necessary resources
		installer batch files are supported, but many of the more advanced commands of later MS-DOS versions (e.g. post-Windows 98 DOS shells) are not. In addition to its internal shell, it also supports running image files of games and software originally intended to start without any operating system. DOSBox is capable of timing-compatible implementation of the serial ports, and can run older hardware and software dependent on such; however, some USB devices that are supported by the host OS can act as a replacement for older serial port devices when using the emulator. DosBox can also be used to run many non-game DOS programs, including Windows 3.1. However, the project has a policy of not adding features that are of no use for DOS games. Also, there is no support for the emulation of post-80486 CPU features although some games that require a Pentium or higher run smoothly. Some unofficial CVS versions contain experimental patches that add support for these elements.		
DOS VM (WinXP)	A virtual machine built with the popular VMware virtualisation suite. The VM features a complete installation of MS-DOS 6.2, inclusing drivers for a (virtual) CD-ROM, virtual floppy and drive and SoundBlaster sound card. The VM further includes support for mouse and international keyboard arrangements.	Virtual machines offer a distinct advantage in terms of preserving digital material - particularly in the game of video game when the material is dynamic. VMware is a well established platform for creating virtual machines. Although the software is proprietary (in this experiments we are using an evaluation copy), the resulting VMs can be loaded in free alternatives, such as Sun's VirtualBox.	Processor: 486DX RAM: 32MB Video: VGA card (generic) Sound: Soundblaster 16 Compatible Disk Drives: 1xFloppy, 1xCD-ROM, 1xHDD 60MB Ports: 1xLPT, 2xCOM	VMWare Workstation (or free alternative, such as Sun VirtualBox) Drivers for CD-ROM (generic) Drivers for SB-16 Mouse driver (generic)
DOS VM (Linux)	A virtual machine built with the popular VMware virtualisation suite. The VM features a complete installation of MS-DOS 6.2, inclusing drivers for a (virtual) CD-ROM, virtual floppy and drive and SoundBlaster sound card. The VM further includes support for mouse and international keyboard arrangements.	Virtual machines offer a distinct advantage in terms of preserving digital material - particularly in the game of video game when the material is dynamic. VMware is a well established platform for creating virtual machines. Although the software is proprietary (in this experiments we are using an evaluation copy), the resulting VMs can be loaded in free alternatives, such as Sun's VirtualBox.	Processor: 486DX RAM: 32MB Video: VGA card (generic) Sound: Soundblaster 16 Compatible	VMWare Workstation (or free alternative, such as Sun VirtualBox) Drivers for CD-ROM (generic) Drivers for SB-16 Mouse driver (generic)
DosBox: Wine (Linux)	Wine is a free software application that aims to allow Unix-like computer operating systems to execute programs written for Microsoft Windows. Wine also provides a software library known as Winelib against which developers can compile Windows applications to help port them to Unix-like systems. Wine implements the Windows API entirely in user-space, rather than as a kernel module. Services normally provided by the kernel in Windows are provided by a daemon known as wineserver. Wineserver implements basic Windows functionality, as well as providing extra functions such as	- Test the suitability of Wine as a platform for running Windows software on a Linux system - Assess performance of DosBox running via Wine on a Linux system - For DOS/Win 3.x compatible games, evaluate results from Wine	Default configuration of DosBox (as specified in the alternative for WinXP) Wine running on Ubuntu 9.10 Karmic Koala	As specified in the alternative for WinXP + Wine for Linux Linux Implementation

Name	Description	Reason for considering	Config settings	Necessary resources
	integration with the X Window System and translation of signals into native Windows exceptions. Although Wine implements some aspects of the Windows kernel, it is not possible to use native Windows drivers with it, due to Wine's underlying architecture. This prevents certain applications from working, such as some copy-protected titles. As of 2009, Wine runs some software packages with good stability and many others with minor issues. The developers of the Direct3D portions of Wine have continued to implement new features such as pixel shaders to increase game support. Wine can also use native DLLs directly, thus increasing functionality, but then a license for Windows is needed unless the DLLs were distributed with the application itself.			
DOSEMU (Linux)	DOSEMU is a compatibility layer software package that enables MS-DOS systems, DOS clones such as FreeDOS, and DOS software to run under Linux on x86-based PCs (IBM PC compatible computers). It uses a combination of hardware virtualization features and strategic emulation. It is thus able to achieve nearly native speed for 8086-compatible DOS operating systems and applications on x86 compatible processors, and for 32-bit DPMI applications on x86 compatible processors as well as on x86-64 processors (Virtual 8086 mode is not available in x86-64 long mode, so DOSEMU includes an 8086 processor emulator for use with 16-bit applications.) Currently it is only available for x86 Linux systems. DOSEMU is an option for people who need or want to continue to use legacy DOS software, in some cases virtualisation is good enough to drive external hardware such as EPROM programmers connected to the parallel port. To quote the manual, "dosemu" is a user-level program which uses certain special features of the Linux kernel and the 80386 processor to run MS-DOS, FreeDOS or DR-DOS in what people in the biz call a `DOS box.' The DOS box, a combination of hardware and software trickery, has these capabilities: * Virtualize all input/output and processor control instructions * Supports the word size and addressing modes of the iAPX86 processor family's "real mode," while still running within the full protected mode environment * Trap all DOS and BIOS system calls and emulate such calls as are necessary for proper operation and good performance * Simulate a hardware environment over which DOS programs are accustomed to having control. * Provide	- Test the suitability of DOSEMU for running DOS-based video games ona Linux system - Assess performance of games with this alternative	Default configuration Running on Ubuntu 9.10 Karmic Koala	DOSEMU installation files

Name	Description	Reason for considering	Config settings	Necessary resources
	DOS services through native Linux services; for example, dosemu can provide a virtual hard disk drive which is actually a Linux directory hierarchy. * Packet driver API for 3Com 3C503.			
GRATE (PTB)	GRATE (Global Remote Access to Emulation-Services) is a demontration service developed within the Planets project, which aims to show the accessibility of 'aged' digital objects through emulated systems using the Internet (Webbrowser). GRATE is developed as a component system to allow easy user access to emulation services. GRATE is coded in Java / PHP / Perl / JavaScript, GRATE is based on the TightVNC technology and calls on the DROID / PRONOM services provided by the The National Archives of the United Kingdom. It provides a number of virtual machines (DOS/Windows 3.11, Windows 98), emulators and binary translators (eg. DosBox, QEMU, Hatari, Dioscuri). GRATE has been integrated into the Planets Interoperability framework for viewing objects within an emulator (Create View feature).	GRATE is an environment that aggregates many virtualisation / emulation alternatives. As part of the Testbed, it offers greater flexibility in accessing emulation services and uploading files remotely, rather than having to install the VM software locally. In this sense, it is a more time- and cost-effective solution for testing, executing and viewing obsolete software in its native environment.	Platform independent system loaded remotely.	Standard Webbrowser with enabled Java Virtual Machine (JSE >= 1.4.2) JavaScript enabled.

Decision: GO

Reason: All the above alternatives need to be evaluated as part of the experiment for generating comparative results between approaches.

Action needed: None needed

Alternative	Experiment description	Run description	Experiment data uploaded
DosBox 0.73 (WinXP)	Experiment carried out on a Dell Optilex 745 desktop PC; system specifications: - Intel Core 2 CPU 6300 @ 1.86GHz - Intel Q965 Express Chipset - 1GB RAM - 40GB HDD - 1 x CD/DVD-RAM Drive - 256MB ATI Radeon X1300 Pro Graphics Card, max 1920x1200 @ 75Hz & 16.7M colours - Integrated ADI 1983 High Definition Audio Chipset - 5 x USB 2.0 ports - 1 x Virtual CD-ROM - 1 x Virtual FLoppy Drive - Windows XP Professional SP 3 Software Used: - Virtual Floppy Drive (VFD) 2.1 (http://chitchat.at.infoseek.co.jp/vmware/vfd.html) - WinImage 8.50 (http://www.winimage.com/) - DosBox 0.73 (http://www.dosbox.com/) System Configuration: - Default for PC system (clean install) - Default for DosBox 0.73 plus mounted Virtual CD-ROM - Created 1 virtual CD-ROM drive with WinImage - Created 1 virtual 1.44 Diskette Drive with VFD	- null - null - null - null - null - null - null	No
DOS VM (WinXP)	Experiment carried out on a Dell Optilex 745 desktop PC; system specifications: - Intel Core 2 CPU 6300 @ 1.86GHz - Intel Q965 Express Chipset - 1GB RAM - 40GB HDD - 1 x CD/DVD-RAM Drive - 256MB ATI Radeon X1300 Pro Graphics Card, max 1920x1200 @ 75Hz & 16.7M colours - Integrated ADI 1983 High Definition Audio Chipset - 5 x USB 2.0 ports - 1 x Virtual CD-ROM - 1 x Virtual FLoppy Drive - Windows XP Professional SP 3 Software Used: - VMWare Workstation 6.5.2 (Evaluation copy - full functionality) - 1 x Virtual Machine running MS-DOS 6.20 - Sound Drivers and Configuration Utilities (SB 16) System Configuration: - Default for PC system (clean install) - For DOS VM: Processor: 486DX RAM: 32MB Video: VGA card (generic) Sound: Soundblaster 16 Compatible Disk Drives: 1xFloppy, 1xCD-ROM, 1xHDD 60MB Ports: 1xLPT, 2xCOM	- Game freezes at start-up; System hangs up; sound is produced from pc speaker for approx. 5 seconds then stops - null - null - null - null - null	No
DOS VM (Linux)	Experiment carried out on a Dell Optilex 745 desktop PC; system specifications: - Intel Core 2 CPU 6300 @ 1.86GHz - Intel Q965 Express Chipset - 1GB RAM - 40GB HDD - 1 x CD/DVD-RAM Drive - 256MB ATI Radeon X1300 Pro Graphics Card, max 1920x1200 @ 75Hz & 16.7M colours - Integrated ADI 1983 High Definition Audio Chipset - 5 x USB 2.0 ports - Ubuntu 9.10 Linux distro Software Used: - VMWare Workstation 6.5.2 (Evaluation copy - full functionality) - 1 x	- null - null - null - null - null - null - null	No

Alternative	Experiment description	Run description	Experiment data uploaded
	Virtual Machine running MS-DOS 6.20 - Sound Drivers and Configuration Utilities (SB 16) System Configuration: - Default for PC system (clean install) - For DOS VM: Processor: 486DX RAM: 32MB Video: VGA card (generic) Sound: Soundblaster 16 Compatible Disk Drives: 1xFloppy, 1xCD-ROM, 1xHDD 60MB Ports: 1xLPT, 2xCOM		
DosBox: Wine (Linux)	Experiment carried out on a Dell Optilex 745 desktop PC; system specifications: - Intel Core 2 CPU 6300 @ 1.86GHz - Intel Q965 Express Chipset - 1GB RAM - 40GB HDD - 1 x CD/DVD-RAM Drive - 256MB ATI Radeon X1300 Pro Graphics Card, max 1920x1200 @ 75Hz & 16.7M colours - Integrated ADI 1983 High Definition Audio Chipset - 5 x USB 2.0 ports - Ubuntu 9.10 Linux distro Software Used: - DosBox 0.73 (http://www.dosbox.com/) - Wine for Linux (downloaded via Synaptic Package Manager) System Configuration: - Default for PC system (clean install) - Default for DosBox 0.73 plus mounted CD/DVD-ROM drive	- null - null - null - null - null - null - null	No
DOSEMU (Linux)	Experiment carried out on a Dell Optilex 745 desktop PC; system specifications: - Intel Core 2 CPU 6300 @ 1.86GHz - Intel Q965 Express Chipset - 1GB RAM - 40GB HDD - 1 x CD/DVD-RAM Drive - 256MB ATI Radeon X1300 Pro Graphics Card, max 1920x1200 @ 75Hz & 16.7M colours - Integrated ADI 1983 High Definition Audio Chipset - 5 x USB 2.0 ports - Ubuntu 9.10 Linux distro Software Used: - DOSEMU (http://www.dosemu.org/) System Configuration: - Default for PC system (clean install) - Default for DOSEMU	- null - null - null - null - null - null - null	No
GRATE (PTB)	Experiment carried out on a Dell Optilex 745 desktop PC; system specifications: - Intel Core 2 CPU 6300 @ 1.86GHz - Intel Q965 Express Chipset - 1GB RAM - 40GB HDD - 1 x CD/DVD-RAM Drive - 256MB ATI Radeon X1300 Pro Graphics Card, max 1920x1200 @ 75Hz & 16.7M colours - Integrated ADI 1983 High Definition Audio Chipset - 5 x USB 2.0 ports - 1 x Virtual CD-ROM - 1 x Virtual FLoppy Drive - Windows XP Professional SP 3	- null - null - Game requires installation, which is not possible remotely. - null - null - Game requires installation which is not possible remotely. - null	No

Hide Evaluation & Transformation

Usability > Game start

Results

Alternatives	1	2	3	4	5	6	7
DosBox 0.73 (WinXP)	Without Installation	Without Installation	Without Installation	Without Installation	Without Installation	With Installation	With Installation
DOS VM (WinXP)	Without Installation	Without Installation	Without Installation	Without Installation	Without Installation	With Installation	With Installation
DOS VM (Linux)	Without Installation	Without Installation	Without Installation	Without Installation	Without Installation	With Installation	With Installation
DosBox: Wine (Linux)	Without Installation	Without Installation	Without Installation	Without Installation	Without Installation	With Installation	With Installation
DOSEMU (Linux)	Without Installation	Without Installation	Without Installation	Without Installation	Without Installation	With Installation	With Installation
GRATE (PTB)	Without Installation	Without Installation	Without Installation	Without Installation	Without Installation	With Installation	With Installation

Transformer

Ordinal Value	Target Value
With Installation	-> 2.0
Without Installation	-> 5.0

Transformed Results

Alternatives	1	2	3	4	5	6	7	Aggregated	Comments
DosBox 0.73 (WinXP)	5	5	5	5	5	2	2	4.14	ATD & HOF require CD-ROM
DOS VM (WinXP)	5	5	5	5	5	2	2	4.14	
DOS VM (Linux)	5	5	5	5	5	2	2	4.14	
DosBox: Wine (Linux)	5	5	5	5	5	2	2	4.14	
DOSEMU (Linux)	5	5	5	5	5	2	2	4.14	
GRATE (PTB)	5	5	5	5	5	2	2	4.14	

Aggregation mode: Arithmetic mean

Usability > Game Playable

Results

Alternatives	1	2	3	4	5	6	7
DosBox 0.73 (WinXP)	Yes	Yes	Yes	Yes	Yes	Yes	Yes
DOS VM (WinXP)	Yes	Yes	No	No	No	No	No

Transformer

Ordinal Value	Target Value
Yes	-> 5.0
No	-> 0.0

Transformed Results

Alternatives	1	2	3	4	5	6	7	Aggregated	Comments
DosBox 0.73 (WinXP)	5	5	5	5	5	5	5	5	

Alternatives	1	2	3	4	5	6	7
DOS VM (Linux)	Yes	Yes	No	No	No	No	No
DosBox: Wine (Linux)	Yes	Yes	Yes	Yes	Yes	Yes	Yes
DOSEMU (Linux)	Yes	Yes	No	No	Yes	No	No
GRATE (PTB)	Yes	Yes	Yes	Yes	Yes	No	No

Alternatives	1	2	3	4	5	6	7	Aggregated	Comments
DOS VM (WinXP)	5	5	0	0	0	0	0	1.43	BD freezes at start-up. TS runs too fast. PP throws error after 2mins of gameplay. ATD freezes at start-up. HOF freezes after intro.
DOS VM (Linux)	5	5	0	0	0	0	0	1.43	BD graphics inhibit playability. TS runs too fast. PP throws error after 2mins of gameplay. ATD freezes at start-up. HOF freezes after intro.
DosBox: Wine (Linux)	5	5	5	5	5	5	5	5	
DOSEMU (Linux)	5	5	0	0	5	0	0	2.14	BD freezes at intro screen. TS runs too fast. ATD and HOF: CD ROM not recognised.
GRATE (PTB)	5	5	5	5	5	0	0	3.57	Game installation not possible remotely.

Aggregation mode: Arithmetic mean

Configurability > Game Configuration

Results

Alternatives	1	2	3	4	5	6	7
DosBox 0.73 (WinXP)	Not needed	Not needed	Not needed	Not needed	Pre-configured	Manual	Manual
DOS VM (WinXP)	Not needed	Not needed	Not needed	Not needed	Pre-configured	Manual	Manual
DOS VM (Linux)	Not needed	Not needed	Not needed	Not needed	Pre-configured	Manual	Manual
DosBox: Wine (Linux)	Not needed	Not needed	Not needed	Not needed	Pre-configured	Manual	Manual
DOSEMU (Linux)	Not needed	Not needed	Not needed	Not needed	Pre-configured	Manual	Manual
GRATE (PTB)	Not needed	Not needed	Not needed	Not needed	Not needed	Manual	Manual

Transformer

Ordinal Value	Target Value
Manual	-> 1.0
Pre-configured	-> 3.0
Not needed	-> 5.0

Transformed Results

Alternatives	1	2	3	4	5	6	7	Aggregated	Comments
DosBox 0.73 (WinXP)	5	5	5	5	3	1	1	3.57	
DOS VM (WinXP)	5	5	5	5	3	1	1	3.57	
DOS VM (Linux)	5	5	5	5	3	1	1	3.57	
DosBox: Wine (Linux)	5	5	5	5	3	1	1	3.57	
DOSEMU (Linux)	5	5	5	5	3	1	1	3.57	
GRATE (PTB)	5	5	5	5	5	1	1	3.86	

Aggregation mode: Arithmetic mean

Configurability > Global System Preferences

Results

Alternatives	Single
DosBox 0.73 (WinXP)	Pre-configured
DOS VM (WinXP)	Manual
DOS VM (Linux)	Manual
DosBox: Wine (Linux)	Pre-configured
DOSEMU (Linux)	Pre-configured
GRATE (PTB)	Pre-configured

Transformer

Ordinal Value	Target Value
Manual	-> 1.0
Pre-configured	-> 5.0
Not applicable	-> 5.0

Transformed Results

Alternatives	Single (=Aggregated)	Comments
DosBox 0.73 (WinXP)	5	
DOS VM (WinXP)	1	
DOS VM (Linux)	1	
DosBox: Wine (Linux)	5	
DOSEMU (Linux)	5	
GRATE (PTB)	5	

Configurability > Ease of Configuration

Results

Alternatives	Single
DosBox 0.73 (WinXP)	1.0
DOS VM (WinXP)	60.0
DOS VM (Linux)	60.0
DosBox: Wine (Linux)	5.0

Transformer

Threshold	Target value
60.0 Minutes	-> 1
45.0 Minutes	-> 2
30.0 Minutes	-> 3
15.0 Minutes	-> 4

Transformed Results

Alternatives	Single (=Aggregated)	Comments
DosBox 0.73 (WinXP)	5	Need to only mount directory from hdd, cdrom, floppy or image file
DOS VM (WinXP)	1	Time needed to install OS and drivers. Figure presumes that drivers are readily available (excludes time to locate and test

Alternatives	Single
DOSEMU (Linux)	1.0
GRATE (PTB)	1.0

1.0 Minutes -> 5
Threshold stepping: Steps

Alternatives	Single (=Aggregated)	Comments
		soundcard/cdrom/mouse drivers)
DOS VM (Linux)	1	Time needed to install OS and drivers. Figure presumes that drivers are readily available (excludes time to locate and test soundcard/cdrom/mouse drivers)
DosBox: Wine (Linux)	4	Time needed to install Wine on Linux system
DOSEMU (Linux)	5	Drives (C, Home, CD-ROM) automatically mounted. Soundcard automatically configured.
GRATE (PTB)	5	Time needed to select an OS platform and upload files.

Costs > Cost for acquiring preservation software

Results

Alternatives	Single
DosBox 0.73 (WinXP)	None (freeware)
DOS VM (WinXP)	High
DOS VM (Linux)	High
DosBox: Wine (Linux)	None (freeware)
DOSEMU (Linux)	None (freeware)
GRATE (PTB)	None (freeware)

Transformer

Ordinal Value	Target Value
None (freeware)	-> 5.0
Minimal	-> 3.0
High	-> 1.0

Transformed Results

Alternatives	Single (=Aggregated)	Comments
DosBox 0.73 (WinXP)	5	
DOS VM (WinXP)	1	
DOS VM (Linux)	1	
DosBox: Wine (Linux)	5	
DOSEMU (Linux)	5	
GRATE (PTB)	5	

Costs > Number of games supported

Results

Alternatives	Single
DosBox 0.73 (WinXP)	more than 2000
DOS VM (WinXP)	more than 2000
DOS VM (Linux)	more than 2000
DosBox: Wine (Linux)	more than 2000
DOSEMU (Linux)	more than 2000
GRATE (PTB)	more than 2000

Transformer

Ordinal Value	Target Value
Less than 10	-> 1.0
10-100	-> 2.0
101-500	-> 3.0
501-2000	-> 4.0
more than 2000	-> 5.0

Transformed Results

Alternatives	Single (=Aggregated)	Comments
DosBox 0.73 (WinXP)	5	
DOS VM (WinXP)	5	
DOS VM (Linux)	5	
DosBox: Wine (Linux)	5	
DOSEMU (Linux)	5	
GRATE (PTB)	5	

Object Characteristics > Game Renders

Results

Alternatives	1	2	3	4	5	6	7
DosBox 0.73 (WinXP)	Yes	Yes	Yes	Yes	Yes	Yes	Yes
DOS VM (WinXP)	Yes	Yes	No	Yes	Yes	No	Yes
DOS VM (Linux)	Yes	Yes	Yes	Yes	Yes	No	Yes
DosBox: Wine (Linux)	Yes	Yes	Yes	Yes	Yes	Yes	Yes
DOSEMU (Linux)	Yes	Yes	No	Yes	Yes	No	No
GRATE (PTB)	Yes	Yes	Yes	Yes	Yes	No	No

Transformer

Ordinal Value	Target Value
Yes	-> 5.0
No	-> 0.0

Transformed Results

Alternatives	1	2	3	4	5	6	7	Aggregated	Comments
DosBox 0.73 (WinXP)	5	5	5	5	5	5	5	5	
DOS VM (WinXP)	5	5	0	5	5	0	5	3.57	
DOS VM (Linux)	5	5	5	5	5	0	5	4.29	Game freezes when accessing CD-ROM at start-up.
DosBox: Wine (Linux)	5	5	5	5	5	5	5	5	
DOSEMU (Linux)	5	5	0	5	5	0	0	2.86	
GRATE (PTB)	5	5	5	5	5	0	0	3.57	

Aggregation mode: Arithmetic mean

Object Characteristics > Speed

Results

Alternatives	1	2	3	4	5	6	7
DosBox 0.73 (WinXP)	As in original	As in original	Slower than original	Faster than original	As in original	As in original	As in original
DOS VM (WinXP)	As in original	Faster than original	Not applicable	Faster than original	Faster than original	Not applicable	As in original
DOS VM (Linux)	As in original	Faster than original	Faster than original	Faster than original	Faster than original	Not applicable	Faster than original
DosBox: Wine (Linux)	As in original	As in original	Slower than original	Faster than original	As in original	As in original	As in original
DOSEMU (Linux)	As in original	Faster than original	Not applicable	Faster than original	As in original	Not applicable	Not applicable
GRATE (PTB)	As in original	As in original	As in original	As in original	As in original	Not applicable	Not applicable

Transformer

Ordinal Value	Target Value
As in original	-> 5.0
Faster than original	-> 1.0
Slower than original	-> 1.0
Not applicable	-> 0.0

Transformed Results

Alternatives	1	2	3	4	5	6	7	Aggregated	Comments
DosBox 0.73 (WinXP)	5	5	1	1	5	5	5	3.86	All tests @ 3000 CPU circles, Frame skip 0; Except ATD (CPU@12000) and HOF (CPU@10000)
DOS VM (WinXP)	5	1	0	1	1	0	5	1.86	
DOS VM (Linux)	5	1	1	1	1	0	1	1.43	
DosBox: Wine (Linux)	5	5	1	1	5	5	5	3.86	
DOSEMU (Linux)	5	1	0	1	5	0	0	1.71	
GRATE (PTB)	5	5	5	5	5	0	0	3.57	

Aggregation mode: Arithmetic mean

Controls with Standard PC input devices > Controls resemble originals

Results

Alternatives	1	2	3	4	5	6	7
DosBox 0.73 (WinXP)	Yes	Yes	No	Yes	Yes	Yes	Yes
DOS VM (WinXP)	Yes	Yes	Not applicable	Yes	Yes	Not applicable	Not applicable
DOS VM (Linux)	Yes	Yes	No	Yes	Yes	Not applicable	Not applicable
DosBox: Wine (Linux)	Yes	Yes	No	Yes	Yes	Yes	Yes
DOSEMU (Linux)	Yes	Yes	Not applicable	Yes	Yes	Not applicable	Not applicable
GRATE (PTB)	Yes	Yes	Yes	Yes	Yes	Not applicable	Not applicable

Transformer

Ordinal Value	Target Value
Yes	-> 5.0
No	-> 1.0
Not applicable	-> 0.0

Transformed Results

Alternatives	1	2	3	4	5	6	7	Aggregated	Comments
DosBox 0.73 (WinXP)	5	5	1	5	5	5	5	4.43	
DOS VM (WinXP)	5	5	0	5	5	0	0	2.86	
DOS VM (Linux)	5	5	1	5	5	0	0	3	
DosBox: Wine (Linux)	5	5	1	5	5	5	5	4.43	
DOSEMU (Linux)	5	5	0	5	5	0	0	2.86	
GRATE (PTB)	5	5	5	5	5	0	0	3.57	

Aggregation mode: Arithmetic mean

Controls with Standard PC input devices > Response delay

Results

Alternatives	1	2	3	4	5	6	7
DosBox 0.73 (WinXP)	No delay	No delay	Considerable delay	Considerable delay	No delay	No delay	No delay
DOS VM (WinXP)	No delay	No delay	Not applicable	No delay	Not applicable	Not applicable	Not applicable
DOS VM (Linux)	No delay	No delay	Short delay	Not applicable	Not applicable	Not applicable	Not applicable
DosBox: Wine (Linux)	No delay	No delay	Considerable delay	Considerable delay	No delay	No delay	No delay
DOSEMU (Linux)	No delay	No delay	Not applicable	Not applicable	No delay	Not applicable	Not applicable
GRATE (PTB)	Short delay	Short delay	Short delay	Short delay	Short delay	Not applicable	Not applicable

Transformer

Ordinal Value	Target Value
Considerable delay	-> 1.0
Short delay	-> 3.0
No delay	-> 5.0
Not applicable	-> 0.0

Transformed Results

Alternatives	1	2	3	4	5	6	7	Aggregated	Comments
DosBox 0.73 (WinXP)	5	5	1	1	5	5	5	3.86	
DOS VM (WinXP)	5	5	0	5	0	0	0	2.14	
DOS VM (Linux)	5	5	3	0	0	0	0	1.86	
DosBox: Wine (Linux)	5	5	1	1	5	5	5	3.86	
DOSEMU (Linux)	5	5	0	0	5	0	0	2.14	
GRATE (PTB)	3	3	3	3	3	0	0	2.14	

Aggregation mode: Arithmetic mean

Controls with Standard PC input devices > Feel of original gaming experience

Results

Transformer

Transformed Results

Alternatives	1	2	3	4	5	6	7
DosBox 0.73 (WinXP)	Perfectly recreated	Perfectly recreated	Playable	Not acceptable	Perfectly recreated	Perfectly recreated	Perfectly recreated
DOS VM (WinXP)	Perfectly recreated	Perfectly recreated	Not applicable	Not acceptable	Not applicable	Not applicable	Not applicable
DOS VM (Linux)	Perfectly recreated	Perfectly recreated	Playable	Not applicable	Not applicable	Not applicable	Not applicable
DosBox: Wine (Linux)	Perfectly recreated	Perfectly recreated	Playable	Not acceptable	Perfectly recreated	Perfectly recreated	Perfectly recreated
DOSEMU (Linux)	Perfectly recreated	Perfectly recreated	Not applicable	Not applicable	Playable	Not applicable	Not applicable
GRATE (PTB)	Perfectly recreated	Perfectly recreated	Perfectly recreated	Perfectly recreated	Perfectly recreated	Not applicable	Not applicable

Ordinal Value	Target Value
Not acceptable	-> 1.0
Playable	-> 3.0
Perfectly recreated	-> 5.0
Not applicable	-> 0.0

Alternatives	1	2	3	4	5	6	7	Aggregated	Comments
DosBox 0.73 (WinXP)	5	5	3	1	5	5	5	4.14	
DOS VM (WinXP)	5	5	0	1	0	0	0	1.57	
DOS VM (Linux)	5	5	3	0	0	0	0	1.86	
DosBox: Wine (Linux)	5	5	3	1	5	5	5	4.14	
DOSEMU (Linux)	5	5	0	0	3	0	0	1.86	
GRATE (PTB)	5	5	5	5	5	0	0	3.57	

Aggregation mode: Arithmetic mean

Input > Support for Game hardware

Results

Alternatives	1	2	3	4	5	6	7
DosBox 0.73 (WinXP)	Yes	Yes	Yes	Yes	Yes	Yes	Yes
DOS VM (WinXP)	Yes	Yes	Yes	Yes	No	No	No
DOS VM (Linux)	Yes	Yes	Yes	Yes	No	No	No
DosBox: Wine (Linux)	Yes	Yes	Yes	Yes	Yes	Yes	Yes
DOSEMU (Linux)	Yes	Yes	No	Yes	Yes	No	No
GRATE (PTB)	Yes	Yes	Yes	Yes	Yes	No	No

Transformer

Ordinal Value	Target Value
Yes	-> 5.0
No	-> 1.0

Transformed Results

Alternatives	1	2	3	4	5	6	7	Aggregated	Comments
DosBox 0.73 (WinXP)	5	5	5	5	5	5	5	5	
DOS VM (WinXP)	5	5	5	5	1	1	1	3.29	Soundcard errors cause PP, ATD & HOF to crash.
DOS VM (Linux)	5	5	5	5	1	1	1	3.29	Soundcard errors cause PP, ATD & HOF to crash.
DosBox: Wine (Linux)	5	5	5	5	5	5	5	5	
DOSEMU (Linux)	5	5	1	5	5	1	1	3.29	Soundcard errors cause BD to crash. Errors recognising CD ROM cause ATD & HOF to crash.
GRATE (PTB)	5	5	5	5	5	1	1	3.86	

Aggregation mode: Arithmetic mean

Input > Support for Special Controller

Results

Alternatives	1	2	3	4	5	6	7
DosBox 0.73 (WinXP)	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable
DOS VM (WinXP)	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable
DOS VM (Linux)	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable
DosBox: Wine (Linux)	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable
DOSEMU (Linux)	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable
GRATE (PTB)	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable

Transformer

Ordinal Value	Target Value
Not applicable	-> 5.0
Supported (drivers needed)	-> 1.0
Supported (no drivers needed)	-> 5.0

Transformed Results

Alternatives	1	2	3	4	5	6	7	Aggregated	Comments
DosBox 0.73 (WinXP)	5	5	5	5	5	5	5	5	
DOS VM (WinXP)	5	5	5	5	5	5	5	5	
DOS VM (Linux)	5	5	5	5	5	5	5	5	
DosBox: Wine (Linux)	5	5	5	5	5	5	5	5	
DOSEMU (Linux)	5	5	5	5	5	5	5	5	
GRATE (PTB)	5	5	5	5	5	5	5	5	

Aggregation mode: Arithmetic mean

Music > Quality

Results

Alternatives	1	2	3	4	5	6	7

Transformer

Ordinal	Target

Transformed Results

Alternatives	1	2	3	4	5	6	7	Aggregated	Comments

Alternatives	1	2	3	4	5	6	7
DosBox 0.73 (WinXP)	No music	No music	No music	No music	Near Perfect	As in original	As in original
DOS VM (WinXP)	No music	No music	Not applicable	Not applicable	Unacceptable	Not applicable	Not applicable
DOS VM (Linux)	No music	No music	Not applicable	Not applicable	Unacceptable	Not applicable	Not applicable
DosBox: Wine (Linux)	No music	No music	No music	No music	Near Perfect	As in original	As in original
DOSEMU (Linux)	No music	No music	Not applicable	Not applicable	Acceptable	Not applicable	Not applicable
GRATE (PTB)	No music	No music	Unacceptable	Unacceptable	Unacceptable	Not applicable	Not applicable

Value	Value
No music	-> 5.0
Unacceptable	-> 1.0
Acceptable	-> 3.0
Near Perfect	-> 4.0
As in original	-> 5.0
Not applicable	-> 0.0

Alternatives	1	2	3	4	5	6	7	Aggregated	Comments
DosBox 0.73 (WinXP)	5	5	5	5	4	5	5	4.86	
DOS VM (WinXP)	5	5	0	0	1	0	0	1.57	
DOS VM (Linux)	5	5	0	0	1	0	0	1.57	
DosBox: Wine (Linux)	5	5	5	5	4	5	5	4.86	
DOSEMU (Linux)	5	5	0	0	3	0	0	1.86	
GRATE (PTB)	5	5	1	1	1	0	0	1.86	

Aggregation mode: Arithmetic mean

Music > Synchronous to video

Results

Alternatives	1	2	3	4	5	6	7
DosBox 0.73 (WinXP)	Not applicable	Not applicable	Not applicable	Not applicable	No errors	Small errors	No errors
DOS VM (WinXP)	Not applicable	Not applicable	Not applicable	Not applicable	Severe errors	Not applicable	Not applicable
DOS VM (Linux)	Not applicable	Not applicable	Not applicable	Not applicable	Severe errors	Not applicable	Not applicable
DosBox: Wine (Linux)	Not applicable	Not applicable	Not applicable	Not applicable	No errors	Small errors	No errors
DOSEMU (Linux)	Not applicable	Not applicable	Not applicable	Not applicable	Severe errors	Not applicable	Not applicable
GRATE (PTB)	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable

Transformer

Ordinal Value	Target Value
Not applicable	-> 5.0
Severe errors	-> 1.0
Small errors	-> 3.0
No errors	-> 5.0

Transformed Results

Alternatives	1	2	3	4	5	6	7	Aggregated	Comments
DosBox 0.73 (WinXP)	5	5	5	5	5	3	5	4.71	
DOS VM (WinXP)	5	5	5	5	1	5	5	4.43	
DOS VM (Linux)	5	5	5	5	1	5	5	4.43	
DosBox: Wine (Linux)	5	5	5	5	5	3	5	4.71	
DOSEMU (Linux)	5	5	5	5	1	5	5	4.43	
GRATE (PTB)	5	5	5	5	5	5	5	5	

Aggregation mode: Arithmetic mean

Sound effects > Quality

Results

Alternatives	1	2	3	4	5	6	7
DosBox 0.73 (WinXP)	No sound effects	No sound effects	Acceptable	As in original	As in original	As in original	As in original
DOS VM (WinXP)	No sound effects	No sound effects	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable
DOS VM (Linux)	No sound effects	No sound effects	Unacceptable	Unacceptable	Unacceptable	Not applicable	Not applicable
DosBox: Wine (Linux)	No sound effects	No sound effects	Acceptable	As in original	As in original	As in original	As in original
DOSEMU (Linux)	No sound effects	No sound effects	Not applicable	Not applicable	Acceptable	Not applicable	Not applicable
GRATE (PTB)	No sound effects	No sound effects	Unacceptable	Unacceptable	Unacceptable	Not applicable	Not applicable

Transformer

Ordinal Value	Target Value
No sound effects	-> 5.0
Unacceptable	-> 1.0
Acceptable	-> 3.0
Near Perfect	-> 4.0
As in original	-> 5.0
Not applicable	-> 0.0

Transformed Results

Alternatives	1	2	3	4	5	6	7	Aggregated	Comments
DosBox 0.73 (WinXP)	5	5	3	5	5	5	5	4.71	
DOS VM (WinXP)	5	5	0	0	0	0	0	1.43	
DOS VM (Linux)	5	5	1	1	1	0	0	1.86	
DosBox: Wine (Linux)	5	5	3	5	5	5	5	4.71	
DOSEMU (Linux)	5	5	0	0	3	0	0	1.86	
GRATE (PTB)	5	5	1	1	1	0	0	1.86	

Aggregation mode: Arithmetic mean

Sound effects > Synchronous to video

Results

Alternatives	1	2	3	4	5	6	7
DosBox 0.73 (WinXP)	Not applicable	Not applicable	Small errors	Small errors	No errors	Small errors	No errors
DOS VM (WinXP)	Not applicable	Not applicable	Not applicable	Severe errors	Severe errors	Not applicable	Not applicable
DOS VM (Linux)	Not applicable	Not applicable	Severe errors	Severe errors	Severe errors	Not applicable	Not applicable
DosBox: Wine (Linux)	Not applicable	Not applicable	Small errors	Small errors	No errors	Small errors	No errors
DOSEMU (Linux)	Not applicable	Not applicable	Not applicable	Not applicable	Small errors	Not applicable	Not applicable
GRATE (PTB)	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable

Transformer

Ordinal Value	Target Value
Not applicable	-> 5.0
Severe errors	-> 1.0
Small errors	-> 3.0
No errors	-> 5.0

Transformed Results

Alternatives	1	2	3	4	5	6	7	Aggregated	Comments
DosBox 0.73 (WinXP)	5	5	3	3	5	3	5	4.14	
DOS VM (WinXP)	5	5	5	1	1	5	5	3.86	
DOS VM (Linux)	5	5	1	1	1	5	5	3.29	
DosBox: Wine (Linux)	5	5	3	3	5	3	5	4.14	
DOSEMU (Linux)	5	5	5	5	3	5	5	4.71	
GRATE (PTB)	5	5	5	5	5	5	5	5	

Aggregation mode: Arithmetic mean

Speech > Quality

Results

Alternatives	1	2	3	4	5	6	7
DosBox 0.73 (WinXP)	No speech	No speech	No speech	No speech	No speech	As in original	As in original
DOS VM (WinXP)	No speech	No speech	No speech	No speech	No speech	Not applicable	Not applicable
DOS VM (Linux)	No speech	No speech	No speech	No speech	No speech	Not applicable	Not applicable
DosBox: Wine (Linux)	No speech	No speech	No speech	No speech	No speech	As in original	As in original
DOSEMU (Linux)	No speech	No speech	No speech	No speech	No speech	Not applicable	Not applicable
GRATE (PTB)	No speech	No speech	No speech	No speech	No speech	Not applicable	Not applicable

Transformer

Ordinal Value	Target Value
No speech	-> 5.0
Unacceptable	-> 1.0
Acceptable	-> 3.0
Near Perfect	-> 4.0
As in original	-> 5.0
Not applicable	-> 0.0

Transformed Results

Alternatives	1	2	3	4	5	6	7	Aggregated	Comments
DosBox 0.73 (WinXP)	5	5	5	5	5	5	5	5	
DOS VM (WinXP)	5	5	5	5	5	0	0	3.57	
DOS VM (Linux)	5	5	5	5	5	0	0	3.57	
DosBox: Wine (Linux)	5	5	5	5	5	5	5	5	
DOSEMU (Linux)	5	5	5	5	5	0	0	3.57	
GRATE (PTB)	5	5	5	5	5	0	0	3.57	

Aggregation mode: Arithmetic mean

Speech > Synchronous to video

Results

Alternatives	1	2	3	4	5	6	7
DosBox 0.73 (WinXP)	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	No errors	No errors
DOS VM (WinXP)	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable
DOS VM (Linux)	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable
DosBox: Wine (Linux)	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	No errors	No errors
DOSEMU (Linux)	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable
GRATE (PTB)	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable

Transformer

Ordinal Value	Target Value
Not applicable	-> 5.0
Severe errors	-> 1.0
Small errors	-> 3.0
No errors	-> 5.0

Transformed Results

Alternatives	1	2	3	4	5	6	7	Aggregated	Comments
DosBox 0.73 (WinXP)	5	5	5	5	5	5	5	5	
DOS VM (WinXP)	5	5	5	5	5	5	5	5	
DOS VM (Linux)	5	5	5	5	5	5	5	5	
DosBox: Wine (Linux)	5	5	5	5	5	5	5	5	
DOSEMU (Linux)	5	5	5	5	5	5	5	5	
GRATE (PTB)	5	5	5	5	5	5	5	5	

Aggregation mode: Arithmetic mean

Graphics > Overall Image quality

Results

Alternatives	1	2	3	4	5	6	7
DosBox 0.73 (WinXP)	As in original	As in original	Near perfect	As in original	As in original	As in original	As in original
DOS VM (WinXP)	As in original	As in original	Nothing displayed	As in original	As in original	Nothing displayed	Nothing displayed
DOS VM (Linux)	As in original	As in original	Severe errors that affect gameplay	Near perfect	As in original	Nothing displayed	As in original
DosBox: Wine (Linux)	As in original	As in original	Near perfect	As in original	As in original	As in original	As in original
DOSEMU (Linux)	Near perfect	Near perfect	Nothing displayed	Near perfect	Near perfect	Nothing displayed	Nothing displayed
GRATE (PTB)	Near perfect	Near perfect	Near perfect	Near perfect	Near perfect	Nothing displayed	Nothing displayed

Transformer

Ordinal Value	Target Value
Nothing displayed	-> 0.0
Severe errors that affect gameplay	-> 1.0
Errors but not affecting gameplay	-> 3.0
Near perfect	-> 4.0
As in original	-> 5.0

Transformed Results

Alternatives	1	2	3	4	5	6	7	Aggregated	Comments
DosBox 0.73 (WinXP)	5	5	4	5	5	5	5	4.86	
DOS VM (WinXP)	5	5	0	5	5	0	0	2.86	
DOS VM (Linux)	5	5	1	4	5	0	5	3.57	
DosBox: Wine (Linux)	5	5	4	5	5	5	5	4.86	
DOSEMU (Linux)	4	4	0	4	4	0	0	2.29	
GRATE (PTB)	4	4	4	4	4	0	0	2.86	

Aggregation mode: Arithmetic mean

Graphics > Colours

Results

Alternatives	1	2	3	4	5	6	7
DosBox 0.73 (WinXP)	As in original	As in original	As in original	As in original	As in original	As in original	As in original
DOS VM (WinXP)	As in original	As in original	Not applicable	As in original	As in original	Not applicable	As in original
DOS VM (Linux)	As in original	As in original	As in original	As in original	As in original	Not applicable	As in original
DosBox: Wine (Linux)	As in original	As in original	As in original	As in original	As in original	As in original	As in original
DOSEMU (Linux)	As in original	As in original	Not applicable	As in original	As in original	Not applicable	Not applicable
GRATE (PTB)	As in original	As in original	As in original	As in original	As in original	Not applicable	Not applicable

Transformer

Ordinal Value	Target Value
Not applicable	-> 0.0
Severe errors affecting gameplay	-> 1.0
Errors but not affecting gameplayNear perfect	-> 3.0
As in original	-> 5.0

Transformed Results

Alternatives	1	2	3	4	5	6	7	Aggregated	Comments
DosBox 0.73 (WinXP)	5	5	5	5	5	5	5	5	
DOS VM (WinXP)	5	5	0	5	5	0	5	3.57	
DOS VM (Linux)	5	5	5	5	5	0	5	4.29	
DosBox: Wine (Linux)	5	5	5	5	5	5	5	5	
DOSEMU (Linux)	5	5	0	5	5	0	0	2.86	
GRATE (PTB)	5	5	5	5	5	0	0	3.57	

Aggregation mode: Arithmetic mean

Network Support > Network protocols

Results

Alternatives	1	2	3	4	5	6	7
DosBox 0.73 (WinXP)	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable
DOS VM (WinXP)	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable
DOS VM (Linux)	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable
DosBox: Wine (Linux)	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable
DOSEMU (Linux)	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable
GRATE (PTB)	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable

Transformer

Ordinal Value	Target Value
Not applicable	-> 5.0
Supported	-> 5.0
Not supported	-> 0.0

Transformed Results

Alternatives	1	2	3	4	5	6	7	Aggregated	Comments
DosBox 0.73 (WinXP)	5	5	5	5	5	5	5	5	
DOS VM (WinXP)	5	5	5	5	5	5	5	5	
DOS VM (Linux)	5	5	5	5	5	5	5	5	
DosBox: Wine (Linux)	5	5	5	5	5	5	5	5	
DOSEMU (Linux)	5	5	5	5	5	5	5	5	
GRATE (PTB)	5	5	5	5	5	5	5	5	

Aggregation mode: Arithmetic mean

Network Support > Use of original servers

Results

Transformer

Transformed Results

Alternatives	1	2	3	4	5	6	7
DosBox 0.73 (WinXP)	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable
DOS VM (WinXP)	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable
DOS VM (Linux)	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable
DosBox: Wine (Linux)	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable
DOSEMU (Linux)	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable
GRATE (PTB)	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable

Ordinal Value	Target Value
Not applicable	-> 5.0
Supported	-> 5.0
Not supported	-> 0.0

Alternatives	1	2	3	4	5	6	7	Aggregated	Comments
DosBox 0.73 (WinXP)	5	5	5	5	5	5	5	5	
DOS VM (WinXP)	5	5	5	5	5	5	5	5	
DOS VM (Linux)	5	5	5	5	5	5	5	5	
DosBox: Wine (Linux)	5	5	5	5	5	5	5	5	
DOSEMU (Linux)	5	5	5	5	5	5	5	5	
GRATE (PTB)	5	5	5	5	5	5	5	5	

Aggregation mode: Arithmetic mean

Network Support > Lag for network play

Results

Alternatives	1	2	3	4	5	6	7
DosBox 0.73 (WinXP)	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable
DOS VM (WinXP)	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable
DOS VM (Linux)	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable
DosBox: Wine (Linux)	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable
DOSEMU (Linux)	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable
GRATE (PTB)	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable

Transformer

Ordinal Value	Target Value
Not applicable	-> 5.0
Major delay	-> 1.0
Minor delay	-> 3.0
No delay	-> 5.0

Transformed Results

Alternatives	1	2	3	4	5	6	7	Aggregated	Comments
DosBox 0.73 (WinXP)	5	5	5	5	5	5	5	5	
DOS VM (WinXP)	5	5	5	5	5	5	5	5	
DOS VM (Linux)	5	5	5	5	5	5	5	5	
DosBox: Wine (Linux)	5	5	5	5	5	5	5	5	
DOSEMU (Linux)	5	5	5	5	5	5	5	5	
GRATE (PTB)	5	5	5	5	5	5	5	5	

Aggregation mode: Arithmetic mean

Documentation > Box artwork available

Results

Alternatives	1	2	3	4	5	6	7
DosBox 0.73 (WinXP)	Yes	No	No	No	Yes	Yes	Yes
DOS VM (WinXP)	Yes	No	No	No	Yes	Yes	Yes
DOS VM (Linux)	Yes	No	No	No	Yes	Yes	Yes
DosBox: Wine (Linux)	Yes	No	No	No	Yes	Yes	Yes
DOSEMU (Linux)	Yes	No	No	No	Yes	Yes	Yes
GRATE (PTB)	Yes	No	No	No	Yes	Yes	Yes

Transformer

Ordinal Value	Target Value
Yes	-> 5.0
No	-> 1.0

Transformed Results

Alternatives	1	2	3	4	5	6	7	Aggregated	Comments
DosBox 0.73 (WinXP)	5	1	1	1	5	5	5	3.29	
DOS VM (WinXP)	5	1	1	1	5	5	5	3.29	
DOS VM (Linux)	5	1	1	1	5	5	5	3.29	
DosBox: Wine (Linux)	5	1	1	1	5	5	5	3.29	
DOSEMU (Linux)	5	1	1	1	5	5	5	3.29	
GRATE (PTB)	5	1	1	1	5	5	5	3.29	

Aggregation mode: Arithmetic mean

Documentation > Manual available

Results

Alternatives	1	2	3	4	5	6	7
DosBox 0.73 (WinXP)	Yes	No	No	No	Yes	Yes	Yes
DOS VM (WinXP)	Yes	No	No	No	Yes	Yes	Yes
DOS VM (Linux)	Yes	No	No	No	Yes	Yes	Yes
DosBox: Wine (Linux)	Yes	No	No	No	Yes	Yes	Yes

Transformer

Ordinal Value	Target Value
Yes	-> 5.0
No	-> 1.0

Transformed Results

Alternatives	1	2	3	4	5	6	7	Aggregated	Comments
DosBox 0.73 (WinXP)	5	1	1	1	5	5	5	3.29	
DOS VM (WinXP)	5	1	1	1	5	5	5	3.29	
DOS VM (Linux)	5	1	1	1	5	5	5	3.29	
DosBox: Wine (Linux)	5	1	1	1	5	5	5	3.29	

Alternatives	1	2	3	4	5	6	7
DOSEMU (Linux)	Yes	No	No	No	Yes	Yes	Yes
GRATE (PTB)	Yes	No	No	No	Yes	Yes	Yes

Alternatives	1	2	3	4	5	6	7	Aggregated	Comments
DOSEMU (Linux)	5	1	1	1	5	5	5	3.29	
GRATE (PTB)	5	1	1	1	5	5	5	3.29	

Aggregation mode: Arithmetic mean

Documentation > Original media available

Results

Alternatives	1	2	3	4	5	6	7
DosBox 0.73 (WinXP)	No	No	No	No	No	Yes	Yes
DOS VM (WinXP)	No	No	No	No	No	Yes	Yes
DOS VM (Linux)	No	No	No	No	No	Yes	Yes
DosBox: Wine (Linux)	No	No	No	No	No	Yes	Yes
DOSEMU (Linux)	No	No	No	No	No	Yes	Yes
GRATE (PTB)	No	No	No	No	No	Yes	Yes

Transformer

Ordinal Value	Target Value
Yes	-> 5.0
No	-> 1.0

Transformed Results

Alternatives	1	2	3	4	5	6	7	Aggregated	Comments
DosBox 0.73 (WinXP)	1	1	1	1	1	5	5	2.14	
DOS VM (WinXP)	1	1	1	1	1	5	5	2.14	
DOS VM (Linux)	1	1	1	1	1	5	5	2.14	
DosBox: Wine (Linux)	1	1	1	1	1	5	5	2.14	
DOSEMU (Linux)	1	1	1	1	1	5	5	2.14	
GRATE (PTB)	1	1	1	1	1	5	5	2.14	

Aggregation mode: Arithmetic mean

Documentation > System requirements specification

Results

Alternatives	1	2	3	4	5	6	7
DosBox 0.73 (WinXP)	No	No	No	No	Yes	Yes	Yes
DOS VM (WinXP)	No	No	No	No	Yes	Yes	Yes
DOS VM (Linux)	No	No	No	No	Yes	Yes	Yes
DosBox: Wine (Linux)	No	No	No	No	Yes	Yes	Yes
DOSEMU (Linux)	No	No	No	No	Yes	Yes	Yes
GRATE (PTB)	No	No	No	No	Yes	Yes	Yes

Transformer

Ordinal Value	Target Value
Yes	-> 5.0
No	-> 1.0

Transformed Results

Alternatives	1	2	3	4	5	6	7	Aggregated	Comments
DosBox 0.73 (WinXP)	1	1	1	1	5	5	5	2.71	
DOS VM (WinXP)	1	1	1	1	5	5	5	2.71	
DOS VM (Linux)	1	1	1	1	5	5	5	2.71	
DosBox: Wine (Linux)	1	1	1	1	5	5	5	2.71	
DOSEMU (Linux)	1	1	1	1	5	5	5	2.71	
GRATE (PTB)	1	1	1	1	5	5	5	2.71	

Aggregation mode: Arithmetic mean

Scalability > Modular design

Results

Alternatives	Single
DosBox 0.73 (WinXP)	Code modular (one system support)
DOS VM (WinXP)	Code modular (one system support)
DOS VM (Linux)	Code modular (one system support)
DosBox: Wine (Linux)	Code modular (multiple system support)
DOSEMU (Linux)	Code modular (one system support)
GRATE (PTB)	Code modular (multiple system support)

Transformer

Ordinal Value	Target Value
Code not available	-> 1.0
Code not modular	-> 2.0
Code modular (one system support)	-> 3.0
Code modular (multiple system support)	-> 5.0

Transformed Results

Alternatives	Single (=Aggregated)	Comments
DosBox 0.73 (WinXP)	3	
DOS VM (WinXP)	3	
DOS VM (Linux)	3	
DosBox: Wine (Linux)	5	
DOSEMU (Linux)	3	
GRATE (PTB)	5	

Scalability > Portability

Results

Alternatives	Single
DosBox 0.73 (WinXP)	Platform dependent
DOS VM (WinXP)	Platform dependent
DOS VM (Linux)	Platform dependent
DosBox: Wine (Linux)	Platform dependent
DOSEMU (Linux)	Platform dependent
GRATE (PTB)	Platform independent

Transformer

Ordinal Value	Target Value
Not applicable	-> 0.0
Platform dependent	-> 1.0
Platform independent	-> 5.0

Transformed Results

Alternatives	Single (=Aggregated)	Comments
DosBox 0.73 (WinXP)	1	
DOS VM (WinXP)	1	
DOS VM (Linux)	1	
DosBox: Wine (Linux)	1	
DOSEMU (Linux)	1	
GRATE (PTB)	5	

Stability > Development status

Results

Alternatives	Single
DosBox 0.73 (WinXP)	Free (open source)
DOS VM (WinXP)	Commercial (non-open source)
DOS VM (Linux)	Commercial (non-open source)
DosBox: Wine (Linux)	Free (open source)
DOSEMU (Linux)	Free (open source)
GRATE (PTB)	Free (open source)

Transformer

Ordinal Value	Target Value
Commercial (open source)	-> 3.0
Commercial (non-open source)	-> 2.0
Free (open source)	-> 5.0
Free (non-open source)	-> 4.0

Transformed Results

Alternatives	Single (=Aggregated)	Comments
DosBox 0.73 (WinXP)	5	
DOS VM (WinXP)	2	
DOS VM (Linux)	2	
DosBox: Wine (Linux)	5	
DOSEMU (Linux)	5	
GRATE (PTB)	5	

Stability > Current support

Results

Alternatives	Single
DosBox 0.73 (WinXP)	Yes
DOS VM (WinXP)	Yes
DOS VM (Linux)	Yes
DosBox: Wine (Linux)	Yes
DOSEMU (Linux)	No
GRATE (PTB)	Yes

Transformer

Ordinal Value	Target Value
Yes	-> 5.0
No	-> 1.0

Transformed Results

Alternatives	Single (=Aggregated)	Comments
DosBox 0.73 (WinXP)	5	
DOS VM (WinXP)	5	
DOS VM (Linux)	5	
DosBox: Wine (Linux)	5	
DOSEMU (Linux)	1	
GRATE (PTB)	5	

Stability > In development

Results

Alternatives	Single
DosBox 0.73 (WinXP)	Yes
DOS VM (WinXP)	Yes
DOS VM (Linux)	Yes
DosBox: Wine (Linux)	Yes
DOSEMU (Linux)	No
GRATE (PTB)	Yes

Transformer

Ordinal Value	Target Value
Yes	-> 5.0
No	-> 1.0

Transformed Results

Alternatives	Single (=Aggregated)	Comments
DosBox 0.73 (WinXP)	5	
DOS VM (WinXP)	5	
DOS VM (Linux)	5	
DosBox: Wine (Linux)	5	
DOSEMU (Linux)	1	
GRATE (PTB)	5	

Stability > Active community

Results

Alternatives	Single
DosBox 0.73 (WinXP)	Yes
DOS VM (WinXP)	Yes
DOS VM (Linux)	Yes
DosBox: Wine (Linux)	Yes
DOSEMU (Linux)	No
GRATE (PTB)	Yes

Transformer

Ordinal Value	Target Value
Yes	-> 5.0
No	-> 1.0

Transformed Results

Alternatives	Single (=Aggregated)	Comments
DosBox 0.73 (WinXP)	5	
DOS VM (WinXP)	5	
DOS VM (Linux)	5	
DosBox: Wine (Linux)	5	
DOSEMU (Linux)	1	
GRATE (PTB)	5	

Infrastructure > Host-Guest OS communication

Results

Alternatives	Single
DosBox 0.73 (WinXP)	Via virtual media

Transformer

Ordinal Value	Target Value
Via Network	-> 2.0

Transformed Results

Alternatives	Single (=Aggregated)	Comments
DosBox 0.73 (WinXP)	3	

Alternatives	Single
DOS VM (WinXP)	Via virtual media
DOS VM (Linux)	Via virtual media
DosBox: Wine (Linux)	Via virtual media
DOSEMU (Linux)	Via virtual media
GRATE (PTB)	Via virtual media

Ordinal Value	Target Value
Native	-> 5.0
Via virtual media	-> 3.0
None	-> 1.0

Alternatives	Single (=Aggregated)	Comments
DOS VM (WinXP)	3	
DOS VM (Linux)	3	
DosBox: Wine (Linux)	3	
DOSEMU (Linux)	3	
GRATE (PTB)	3	

Legality > Alterations to original permitted

Results

Alternatives	1	2	3	4	5	6	7
DosBox 0.73 (WinXP)	No	No	No	No	No	No	No
DOS VM (WinXP)	No	No	No	No	No	No	No
DOS VM (Linux)	No	No	No	No	No	No	No
DosBox: Wine (Linux)	No	No	No	No	No	No	No
DOSEMU (Linux)	No	No	No	No	No	No	No
GRATE (PTB)	No	No	No	No	No	No	No

Transformer

Ordinal Value	Target Value
Yes	-> 5.0
No	-> 1.0

Transformed Results

Alternatives	1	2	3	4	5	6	7	Aggregated	Comments
DosBox 0.73 (WinXP)	1	1	1	1	1	1	1	1	
DOS VM (WinXP)	1	1	1	1	1	1	1	1	n
DOS VM (Linux)	1	1	1	1	1	1	1	1	
DosBox: Wine (Linux)	1	1	1	1	1	1	1	1	
DOSEMU (Linux)	1	1	1	1	1	1	1	1	
GRATE (PTB)	1	1	1	1	1	1	1	1	

Aggregation mode: Arithmetic mean

Legality > Copies of original permitted

Results

Alternatives	1	2	3	4	5	6	7
DosBox 0.73 (WinXP)	No	Yes	Yes	Yes	Yes	No	No
DOS VM (WinXP)	No	Yes	Yes	Yes	Yes	No	No
DOS VM (Linux)	No	Yes	Yes	Yes	Yes	No	No
DosBox: Wine (Linux)	No	Yes	Yes	Yes	Yes	No	No
DOSEMU (Linux)	No	Yes	Yes	Yes	Yes	No	No
GRATE (PTB)	No	Yes	Yes	Yes	Yes	No	No

Transformer

Ordinal Value	Target Value
Yes	-> 5.0
No	-> 1.0

Transformed Results

Alternatives	1	2	3	4	5	6	7	Aggregated	Comments
DosBox 0.73 (WinXP)	1	5	5	5	5	1	1	3.29	
DOS VM (WinXP)	1	5	5	5	5	1	1	3.29	
DOS VM (Linux)	1	5	5	5	5	1	1	3.29	
DosBox: Wine (Linux)	1	5	5	5	5	1	1	3.29	
DOSEMU (Linux)	1	5	5	5	5	1	1	3.29	
GRATE (PTB)	1	5	5	5	5	1	1	3.29	

Aggregation mode: Arithmetic mean

Legality > Game IP status

Results

Alternatives	1	2	3	4	5	6	7
DosBox 0.73 (WinXP)	Abandonware	Abandonware	Abandonware	Abandonware	Abandonware	Protected	Protected
DOS VM (WinXP)	Abandonware	Abandonware	Abandonware	Abandonware	Abandonware	Protected	Protected
DOS VM (Linux)	Abandonware	Abandonware	Abandonware	Abandonware	Abandonware	Protected	Protected
DosBox: Wine (Linux)	Abandonware	Abandonware	Abandonware	Abandonware	Abandonware	Protected	Protected
DOSEMU (Linux)	Abandonware	Abandonware	Abandonware	Abandonware	Abandonware	Protected	Protected
GRATE (PTB)	Abandonware	Abandonware	Abandonware	Abandonware	Abandonware	Protected	Protected

Transformer

Ordinal Value	Target Value
Freeware	-> 5.0
Abandonware	-> 5.0
Protected	-> 1.0

Transformed Results

Alternatives	1	2	3	4	5	6	7	Aggregated	Comments
DosBox 0.73 (WinXP)	5	5	5	5	5	1	1	3.86	
DOS VM (WinXP)	5	5	5	5	5	1	1	3.86	
DOS VM (Linux)	5	5	5	5	5	1	1	3.86	
DosBox: Wine (Linux)	5	5	5	5	5	1	1	3.86	
DOSEMU (Linux)	5	5	5	5	5	1	1	3.86	
GRATE (PTB)	5	5	5	5	5	1	1	3.86	

Aggregation mode: Arithmetic mean

Legality > Operating System IP status

Results **Transformer** **Transformed Results**

Alternatives	Single
DosBox 0.73 (WinXP)	Freeware
DOS VM (WinXP)	Protected
DOS VM (Linux)	Protected
DosBox: Wine (Linux)	Freeware
DOSEMU (Linux)	Freeware
GRATE (PTB)	Protected

Ordinal Value	Target Value
Freeware	-> 5.0
Abandonware	-> 5.0
Protected	-> 1.0

Alternatives	Single (=Aggregated)	Comments
DosBox 0.73 (WinXP)	5	
DOS VM (WinXP)	1	
DOS VM (Linux)	1	
DosBox: Wine (Linux)	5	
DOSEMU (Linux)	5	
GRATE (PTB)	1	

Media > Keyboard/mouse support

Results

Alternatives	Single
DosBox 0.73 (WinXP)	Yes
DOS VM (WinXP)	Yes
DOS VM (Linux)	Yes
DosBox: Wine (Linux)	Yes
DOSEMU (Linux)	Yes
GRATE (PTB)	Yes

Transformer

Ordinal Value	Target Value
Yes	-> 5.0
No	-> 1.0

Transformed Results

Alternatives	Single (=Aggregated)	Comments
DosBox 0.73 (WinXP)	5	
DOS VM (WinXP)	5	
DOS VM (Linux)	5	
DosBox: Wine (Linux)	5	
DOSEMU (Linux)	5	
GRATE (PTB)	5	

Media > CD-ROM support

Results

Alternatives	Single
DosBox 0.73 (WinXP)	Pre-configured
DOS VM (WinXP)	Manual
DOS VM (Linux)	Manual
DosBox: Wine (Linux)	Pre-configured
DOSEMU (Linux)	Pre-configured
GRATE (PTB)	Pre-configured

Transformer

Ordinal Value	Target Value
Manual	-> 1.0
Pre-configured	-> 5.0
None	-> 1.0

Transformed Results

Alternatives	Single (=Aggregated)	Comments
DosBox 0.73 (WinXP)	5	
DOS VM (WinXP)	1	
DOS VM (Linux)	1	
DosBox: Wine (Linux)	5	
DOSEMU (Linux)	5	
GRATE (PTB)	5	

Media > Sound support

Results

Alternatives	Single
DosBox 0.73 (WinXP)	Pre-configured
DOS VM (WinXP)	Manual
DOS VM (Linux)	Manual
DosBox: Wine (Linux)	Pre-configured
DOSEMU (Linux)	Pre-configured
GRATE (PTB)	Pre-configured

Transformer

Ordinal Value	Target Value
Manual	-> 2.0
Pre-configured	-> 5.0
None	-> 1.0

Transformed Results

Alternatives	Single (=Aggregated)	Comments
DosBox 0.73 (WinXP)	5	
DOS VM (WinXP)	2	
DOS VM (Linux)	2	
DosBox: Wine (Linux)	5	
DOSEMU (Linux)	5	
GRATE (PTB)	5	

Media > OS pre-installed

Results

Alternatives	Single
DosBox 0.73 (WinXP)	Yes
DOS VM (WinXP)	No
DOS VM (Linux)	No
DosBox: Wine (Linux)	Yes

Transformer

Ordinal Value	Target Value
Yes	-> 5.0
No	-> 1.0

Transformed Results

Alternatives	Single (=Aggregated)	Comments
DosBox 0.73 (WinXP)	5	
DOS VM (WinXP)	1	
DOS VM (Linux)	1	
DosBox: Wine (Linux)	5	

Alternatives	Single
DOSEMU (Linux)	Yes
GRATE (PTB)	Yes

Alternatives	Single (=Aggregated)	Comments
DOSEMU (Linux)	5	
GRATE (PTB)	5	

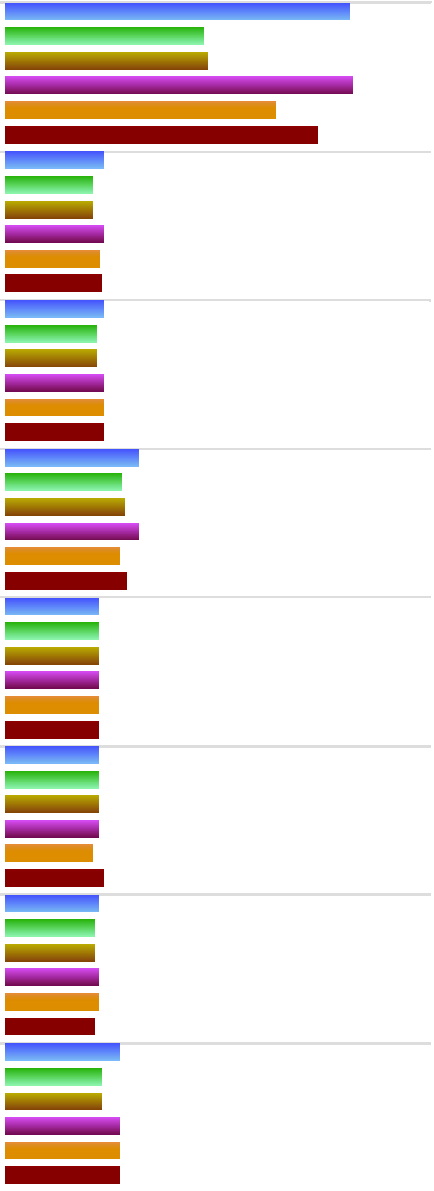
Evaluation comments:

Transformation comments:

How do the aggregation mechanisms work?

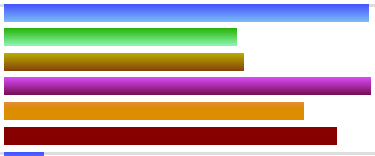
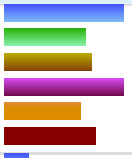
Result-Tree with all Alternatives, Aggregation method: Weighted multiplication

Node	Results
DOS games Preservation Requirements	DosBox 0.73 (WinXP): 4.12 DOS VM (WinXP): 2.36 DOS VM (Linux): 2.41 DosBox: Wine (Linux): 4.14 DOSEMU (Linux): 3.23 GRATE (PTB): 3.72
Process Characteristics	DosBox 0.73 (WinXP): 1.17 DOS VM (WinXP): 1.05 DOS VM (Linux): 1.05 DosBox: Wine (Linux): 1.17 DOSEMU (Linux): 1.12 GRATE (PTB): 1.15
Costs	DosBox 0.73 (WinXP): 1.17 DOS VM (WinXP): 1.08 DOS VM (Linux): 1.08 DosBox: Wine (Linux): 1.17 DOSEMU (Linux): 1.17 GRATE (PTB): 1.17
Object Characteristics	DosBox 0.73 (WinXP): 1.60 DOS VM (WinXP): 1.40 DOS VM (Linux): 1.42 DosBox: Wine (Linux): 1.60 DOSEMU (Linux): 1.37 GRATE (PTB): 1.46
Context and Data Characteristics	DosBox 0.73 (WinXP): 1.11 DOS VM (WinXP): 1.11 DOS VM (Linux): 1.11 DosBox: Wine (Linux): 1.11 DOSEMU (Linux): 1.11 GRATE (PTB): 1.11
Infrastructure	DosBox 0.73 (WinXP): 1.11 DOS VM (WinXP): 1.10 DOS VM (Linux): 1.10 DosBox: Wine (Linux): 1.12 DOSEMU (Linux): 1.05 GRATE (PTB): 1.16
Legality	DosBox 0.73 (WinXP): 1.11 DOS VM (WinXP): 1.07 DOS VM (Linux): 1.07 DosBox: Wine (Linux): 1.11 DOSEMU (Linux): 1.11 GRATE (PTB): 1.07
Media	DosBox 0.73 (WinXP): 1.38 DOS VM (WinXP): 1.15 DOS VM (Linux): 1.15 DosBox: Wine (Linux): 1.38 DOSEMU (Linux): 1.38 GRATE (PTB): 1.38



Result-Tree with all Alternatives, Aggregation method: Weighted sum.

This tree contains only strategies that do not have knock-out evaluation criteria; see above

Node	Results	
DOS games Preservation Requirements	DosBox 0.73 (WinXP): 4.35 DOS VM (WinXP): 2.78 DOS VM (Linux): 2.85 DosBox: Wine (Linux): 4.37 DOSEMU (Linux): 3.57 GRATE (PTB): 3.97	
Process Characteristics	DosBox 0.73 (WinXP): 0.48 DOS VM (WinXP): 0.18 DOS VM (Linux): 0.18 DosBox: Wine (Linux): 0.47 DOSEMU (Linux): 0.34 GRATE (PTB): 0.41	
Costs	DosBox 0.73 (WinXP): 0.50 DOS VM (WinXP): 0.30 DOS VM (Linux): 0.30 DosBox: Wine (Linux): 0.50 DOSEMU (Linux): 0.50 GRATE (PTB): 0.50	
Object Characteristics	DosBox 0.73 (WinXP): 1.43 DOS VM (WinXP): 0.98 DOS VM (Linux): 1.05 DosBox: Wine (Linux): 1.43 DOSEMU (Linux): 0.91 GRATE (PTB): 1.08	
Context and Data Characteristics	DosBox 0.73 (WinXP): 0.28 DOS VM (WinXP): 0.28 DOS VM (Linux): 0.28 DosBox: Wine (Linux): 0.28 DOSEMU (Linux): 0.28 GRATE (PTB): 0.28	
Infrastructure	DosBox 0.73 (WinXP): 0.33 DOS VM (WinXP): 0.31 DOS VM (Linux): 0.31 DosBox: Wine (Linux): 0.36 DOSEMU (Linux): 0.20 GRATE (PTB): 0.46	
Legality	DosBox 0.73 (WinXP): 0.33 DOS VM (WinXP): 0.23 DOS VM (Linux): 0.23 DosBox: Wine (Linux): 0.33 DOSEMU (Linux): 0.33 GRATE (PTB): 0.23	
Media	DosBox 0.73 (WinXP): 1.00 DOS VM (WinXP): 0.50 DOS VM (Linux): 0.50 DosBox: Wine (Linux): 1.00 DOSEMU (Linux): 1.00 GRATE (PTB): 1.00	

Appendix D Plato preservation plan: Software art

Validate plan for Digital Preservation of Software Art

Report creation date: Apr 13, 2010 1:26:27 PM

Display Changelogs

Plan name Digital Preservation of Software Art

Current state Plan Defined

Plan description This preservation plan focuses on assessing the suitability of binary translation and virtualisation for preserving software and game art. The assessment uses material from various sources, including: scene.org, the Demoscene, runme.org, Cospiracy and private pages of artists.

Responsible planners Leo Konstantelos

Organization HATII, University of Glasgow

Identification code

Planning purpose This plan is meant to assess the suitability of different binary translation / virtualisation alternatives for preserving software and game art for various platforms.

Plan relations This is the organisation's first attempt to create a preservation plan for the software/game art collection.

Triggers

Trigger	Description
New Collection	We are evaluating the preservation potential of alternatives for a newly established collection of software and game art. The organisation has assembled the collection from material available online and from corerrespondence with artists.

[↑] Policies

No policies have been defined yet.

Document types Software and game art created for various operating systems - primarily Windows-based. Date of creation spans between 2000-2009. Each consisting of at least one executable.

Mandate For the purposes of this plan, we assume that the artworks form part of an organisation's collection of software and game art. The organisation's mission is to ensure long-term access to this collection without the necessity of maintaining obsolete systems (as in the case of

	a computer museum). Instead, the organisation wants to test how and to what degree the collected material can be rendered and/or playable on modern systems. The solution should offer capabilities for encapsulating the virtual system for easy portability in the future.
Designated community	The designated community in this instance are the users of the organisation's collection. There is no mandate to make the material available online. The users of the collection range from scholars investigating the field of software art, to artists and possibly the general public.
Applying policies	Due to IP / copyright restrictions, the organisation is planning to preserve only artworks for which it possesses the original or freeware material.
Relevant organisational procedures and workflows	Currently the organisation holds the specifications to render the artwork provided by the artist or the electronic resource from which the material was harvested. These suggest minimum requriemetns to run the artwork both in terms of software and hardware.
Contracts and agreements specifying preservation rights	Agreement with the artist to preserve the material. In cases where the material has been collected from an online resource (e.g. runme.org) the only mandate is to reference that resource in future communications regarding the material. In all other cases, artworks are freely distributed and available.
Reference to agreements of maintenance and access	
Samples description:	We have selected four examples of software art and five examples of game art. All cases can run off-line (i.e. they do not require Internet access). For software art, we selected works that range from demos (Executable linkprograms which produce, in linkreal time, engaging computer graphics and music) to interactive applications. For game art, we selected cases developed for Windows and Linux platforms that are interactive.

Collection profile

Windows plaforms.	**Collection ID:**	
	Description:	Collection of sortware and game art works downloaded from: - http://www.runme.org/ - http://www.scene.org/ - http://conspiracy.hu/ All artefacts consist of at least one file (executable). The majority of the artefacts has been created for
	Type of objects:	Executable files with some cases requiring additional data. Objects stored as compressed archives, then expanded on a hard disk drive.
	Number of objects:	
	Expected growth rate:	Moderate, provided that (a) the designated community exhibits adequate interest in the current collection and (b) further contributions are made to the current collection.

Name	Short name	Description	Original environment	Data	Object-format
[phage]	PH	Created in 1998, [phage] is a computer application which is viral– an artificial life form. [phage] filters through all available material on a specified workstation and places it in an alternate context-a visible and audible moving 3D spatialized world. By mapping a userâs unique experiences– through images, downloads, web sites visited, emails–the computer program creates spatial memory maps that not only reflect the computer and technoculture in content, but the user’s artifacts from his or her interactions. In this way, the [phage] program reflects each user as an individual. The work, in fact, becomes about the user’s experience with the particular computer. [phage] creates new living sculptures from our own data, thus mixing ideas of authorship between programmer, operating system, and users. It eradicates gender-based notions previously associated with the life creating process, and it questions the command and control paradigms which created the computer in the first place. Source and download: http://www.maryflanagan.com/phage/	Windows 98	No data	**PUID:** **Name:** **Version:** **mime-type:**

Name	Short name	Description	Original environment	Data	Object-format
DataDADA	DD	DataDada, version <aldkjallkdjsfa>, is an application that will turn the stored data on your hard drive into a movie complete with sound, image, and subtitles. Essentially, it reads all the data on the disk (or, optionally, only specific directories), and writes the data to your computer's sound card and video display. Additionally, it will display the name of the file being read as a human-understandable subtitle. It uses portaudio for cross-platform audio and the Simple DirectMedia Layer for cross-platform video. The work was created by August Black in 2003. It is licensed under the GNU GPL v2. Source and download: http://aug.ment.org/datadada/	Windows 2000 / Linux	No data	**PUID: Name: Version: mime-type:**
Prophecy - Project Nemesis	PR	This is an award winning 64k intro (demo) created for the Assembly 2004 demo party in Finland (for more information, see: http://en.wikipedia.org/wiki/Assembly %28demo party%29). The demo is available from the project's website in both executable and video format. Furthermore, video footage is available online in YouTube. Here, we are using the original .exe file. Conspiracy´s prophecy manages to fit a whole believable world and narrative inside sixty-four kilobytes. The technical excellence displayed here enables pushing even more content, with great if somewhat inconsistent looking scenes, evoking a good sensation of scale. That same technical excellence which now give them more room, more freedom for actual storytelling. Source: http://awards.scene.org/archive.php?cat=9&year=2004 Download: http://conspiracy.hu/release/64k/prophecy/	Windows 98 Minimum requirements: MMX compatible processor 64MB RAM Multitexturing videocard (= TNT2 / Ati Rage) Reccommended: processor (2 GHz+, AMD preferred) 512 MB memory GF4Ti/R9600-class videocard (nVidia preferred) 128 MB video memory DirectSound compatible sound card	No data	**PUID: Name: Version: mime-type:**
a.Shooter - Sonic Invaders	AS	Released in 2004, a.Shooter is the first prototype release of a.Game production: a solely acoustical ego-shooter game. a.Game is a production and research group examining acoustical mechanism of orientation and interaction by means of computer games. It produces audio games that focus and experiment with sound-driven interaction and acoustical navigation in virtual sonic environments without the support of visual information. The player's task is to shoot up sonic invaders conquering a virtual room as defined by panorama, pitch and volume. Source: http://www.agame.org/en/aShooter/ Download: http://www.agame.org/download /aSetup.exe	Windows XP	No data	**PUID: Name: Version: mime-type:**
CAERO	CR	Caero was created by the demoscene group Plant & Electromotive Force (EMF). The demo won the first place prize in the 1995 Demo Party, an annual demoscene event held from 1991 to 2002 in Denmark. It was one of the first events of its kind and set the trend for many other demoscene parties in Europe. The scenes in the demo are complex (more than 7000 polygons in the butterfly scene for example) and some of them have transparent objects and motion blur with colors. Caero was one of the first demos to use such complexity and bestoke libraries for the DOS platform. This work became popular beyond the demoscene community because it features musci composed by Brothomstates - a well-known Finish musician of IDM (Intelligent Dance Music). The version used in the experiments was acquired by the scene.org repository. Download: http://www.scene.org/file.php?id=93811	DOS (IBM PC) Minimum System Requirements: PC 486/33; DOS 5.0 or Higher; 4MB RAM; VGA Graphics Card; 4MB Hard Disk Space; SoundBlaster or SoundBlaster Pro and 100% Compatibles.	No data	**PUID: Name: Version: mime-type:**

Node	Weight	Total weight	Scale and Description
Software Art Preservation Requirements	1	1	*(An objective tree for preserving video games developed for the DOS platform. The tree is based on previous work by Mark Guttenbrunner; see: http://www.ifs.tuwien.ac.at/~becker/pubs/guttenbrunner ipres08.pdf)*
Process Characteristics	0.1	0.1	
Usability	0.6	0.06	
Installation Requirements	0.2	0.01	Ordinal With Installation, Without Installation or Solution inapplicable

Artefact Playable (if interactive)	0.8	0.05	Ordinal Yes, No, Not Interactive or Solution inapplicable
Configurability	0.4	0.04	
Artefact Configuration	0.2	0.01	Ordinal Manual, Pre-configured, Not needed or Solution inapplicable
Global System Preferences	0.5	0.02	Ordinal Manual, Pre-configured or Not applicable (single)
Ease of Configuration	0.3	0.01	Ordinal Easy (less than 15 mins), Moderate (16-60 mins) or Difficult (more than 60 mins) (single)
Costs	0.1	0.1	
Cost for acquiring preservation software	0.5	0.05	Ordinal None (freeware), Minimal or High (single)
Number of artefacts supported	0.5	0.05	Ordinal Less than 10, 10-100, 101-500, 501-2000 or more than 2000 (single)
Object Characteristics	0.3	0.3	
Artefact Renders	0.3	0.09	Ordinal Yes, No or Solution Inapplicable
Speed	0.1	0.03	Ordinal As in original, Faster than original, Slower than original, Not applicable or Solution inapplicable
Interactivity	0.2	0.06	
Input	1	0.06	
Controls with Standard PC input devices	1	0.06	
Controls resemble originals	0.15	0.01	Ordinal Yes, No, Not applicable or Solution inapplicable
Response delay	0.15	0.01	Ordinal Considerable delay, Short delay, No delay, Not applicable or Solution inapplicable
Feel of original artistic experience	0.7	0.04	Ordinal Not acceptable, Playable, Perfectly recreated, Not applicable or Solution inapplicable
Audio	0.2	0.06	
Music	1	0.06	
Quality	0.6	0.04	Ordinal No music, Unacceptable, Acceptable, Near Perfect, As in original, Not applicable or Solution inappl
Synchronous to video	0.4	0.02	Ordinal Not applicable, Severe errors, Small errors, No errors or Solution inapplicable
Graphics	0.2	0.06	
Overall Image quality	0.5	0.03	Ordinal Nothing displayed, Severe errors that affect experience, Errors but not affecting experience, Near
Colours	0.5	0.03	Ordinal Not applicable, Severe errors affecting experience, Errors but not affecting experience, Near perfe
Context and Data Characteristics	0.1	0.1	

Metadata	1	0.1	
Documentation	1	0.1	
Instuctions available	0.3	0.03	Ordinal Yes, No or Solution inapplicable
Original media available	0.4	0.04	Ordinal Yes, No or Solution inapplicable
System requirements specification	0.3	0.03	Ordinal Yes, No or Solution inapplicable
Infrastructure	0.1	0.1	
Scalability	0.4	0.04	
Modular design	0.4	0.02	Ordinal Code not available, Code not modular, Code modular (one system support) or Code modular (mu
Portability	0.6	0.02	Ordinal Not applicable, Platform dependent or Platform independent (single)
Stability	0.4	0.04	
Development status	0.2	0.01	Ordinal Commercial (open source), Commercial (non-open source), Free (open source) or Free (non-oper
Current support	0.25	0.01	Boolean Yes or No (single)
In development	0.35	0.01	Boolean Yes or No (single)
Active community	0.2	0.01	Boolean Yes or No (single)
Host-Guest OS communication	0.2	0.02	Ordinal Via Network, Native, Via virtual media or None (single)
Legality	0.1	0.1	
Alterations to original permitted	0.3	0.03	Ordinal Yes, No or Solution inapplicable
Copies of original permitted	0.4	0.04	Ordinal Yes, No or Solution inapplicable
Operating System IP status	0.3	0.03	Ordinal Freeware or Protected (single)
Media	0.2	0.2	
Keyboard/mouse support	0.3	0.06	Boolean Yes or No (single)
CD-ROM support	0.1	0.02	Ordinal Manual, Pre-configured or None (single)
Sound support	0.3	0.06	Ordinal Manual, Pre-configured or None (single)
OS pre-installed	0.3	0.06	Boolean Yes or No (single)

Importance factors
comments:
Description:

Attached files: **File name**

Evaluation comments:

Name	Description	Reason for considering	Config settings	Necessary resources
DosBox 0.73 (WinXP)	DOSBox is a binary translator working as a command-line program, configured either by a set of command-line arguments or by editing a plain text configuration file. For ease of use, several graphical front-ends have been developed by the user community. DOSBox is a full CPU emulator, capable of running DOS programs that require the CPU to be in either real mode or protected mode. Other similar programs, such as dosemu or VDMs for Windows and OS/2, provide compatibility layers and rely on virtualization capabilities of the 386 family processors. Since DOSBox can emulate its CPU by interpretation, it is independent of its host CPU. However, on systems which provide the i386 instruction set, the option to use dynamic instruction translation is available in DOSBox. Though this setting is less accurate and reliable, it is faster than interpretive CPU emulation.	It is capable of emulating many types of graphics and sound hardware. Graphics emulation includes text mode, Hercules, CGA (including composite and 160x100x16 tweaked modes), Tandy, EGA, VGA (including Mode X and other tweaks), VESA, and full S3 Trio 64 emulation.[3] Sound hardware that can be emulated includes the PC speaker, AdLib, Gravis Ultrasound, Tandy, Creative Music System/GameBlaster, Sound Blaster 1.x/2.0 /Pro/16, MPU-401, and Disney Sound Source. (MT-32/CM-32L emulation is included in unofficial builds, but not in the official source code repository due to need for copyrighted ROM images.) A component that differentiates DOSBox from other emulators is its ability to simulate peer-to-peer or Internet/Intranet networking. This includes modem simulation over TCP/IP, allowing for DOS modem games to be played over modern LANs or the Internet, and IPX network tunneling, which allows for old IPX DOS multiplayer games to be played as UDP/IP over modern LANs or the Internet. Win32 and Linux specific builds support direct serial port access. DOSBox contains its own internal DOS-like shell, rather than being a fully virtual PC emulator like Bochs. This means that it can be used without owning a license to any real DOS operating system. Most commands that are typically used in installer batch files are supported, but many of the more advanced commands of later MS-DOS versions (e.g. post-Windows 98 DOS shells) are not. In addition to its internal shell, it also supports running image files of games and software originally intended to start without any operating system. DOSBox is capable of timing-compatible implementation of the serial ports, and can run older hardware and software dependent on such; however, some USB devices that are supported by the host OS can act as a replacement for older serial port devices when using the emulator. DosBox can also be used to run many non-game DOS programs, including Windows 3.1. However, the project has a policy of not adding features that are of no use for DOS games. Also, there is no support for the emulation of post-80486 CPU features although some games that require a Pentium or higher run smoothly. Some unofficial	Default configuration, plus mounting a virtual C:\ drive	DosBox 0.73 Installation files Windows 9x / XP / Vista 5MB free space

Name	Description	Reason for considering	Config settings	Necessary resources
		CVS versions contain experimental patches that add support for these elements.		
DOS VM (WinXP)	A virtual machine built with the Sun VirtualBox. The VM features a complete installation of MS-DOS 6.2, inclusing drivers for a (virtual) CD-ROM, virtual floppy and drive and SoundBlaster sound card. The VM further includes support for mouse and international keyboard arrangements.	Virtual machines offer a distinct advantage in terms of preserving digital material - particularly in the context of cases where the material is dynamic. VirtualBox is a well established platform for creating virtual machines. The software is free and the resulting VMs can be loaded in other alternatives, such as the proprietary VMWare Workstation.	Processor: 486DX RAM: 32MB Video: VGA card (generic) Sound: Soundblaster 16 Compatible Disk Drives: 1xFloppy, 1xCD-ROM, 1xHDD 60MB Ports: 1xLPT, 2xCOM	Sun VirtualBox(or prorpietary alternative e.g. VMWare Workstation), Drivers for CD-ROM (generic), Drivers for SB-16 Mouse driver (generic)
Windows 98 VM (WinXP)	A virtual machine built with the Sun VirtualBox. The VM features a complete installation of MS Windows 98, inclusing drivers for a (virtual) CD-ROM, virtual floppy and drive and on-board sound card.	Virtual machines offer a distinct advantage in terms of preserving digital material - particularly in the context of cases where the material is dynamic. VirtualBox is a well established platform for creating virtual machines. The software is free and the resulting VMs can be loaded in other alternatives, such as the proprietary VMWare Workstation.	Processor: Pentium or equivalent RAM: 512MB Video: SVGA card (generic) Sound: on-board sound card Compatible Disk Drives: 1xFloppy, 1xCD-ROM, 1xHDD 60MB Ports: 1xLPT, 2xCOM	Sun VirtualBox(or prorpietary alternative e.g. VMWare Workstation), Drivers for CD-ROM (generic), Drivers on-board sound card
Wine for Linux	Wine is a free software application that aims to allow Unix-like computer operating systems to execute programs written for Microsoft Windows. Wine also provides a software library known as Winelib against which developers can compile Windows applications to help port them to Unix-like systems. Wine implements the Windows API entirely in user-space, rather than as a kernel module. Services normally provided by the kernel in Windows are	Test the suitability of Wine as a platform for running Windows software on a Linux system.	Wine running on Ubuntu 9.10 Karmic Koala	An installation of a Linux distribution comnpatible with Wine

Name	Description	Reason for considering	Config settings	Necessary resources
	provided by a daemon known as wineserver. Wineserver implements basic Windows functionality, as well as providing extra functions such as integration with the X Window System and translation of signals into native Windows exceptions. Although Wine implements some aspects of the Windows kernel, it is not possible to use native Windows drivers with it, due to Wine's underlying architecture. This prevents certain applications from working, such as some copy-protected titles. As of 2009, Wine runs some software packages with good stability and many others with minor issues. The developers of the Direct3D portions of Wine have continued to implement new features such as pixel shaders to increase game support. Wine can also use native DLLs directly, thus increasing functionality, but then a license for Windows is needed unless the DLLs were distributed with the application itself.			
GRATE (PTB)	GRATE (Global Remote Access to Emulation-Services) is a demontration service developed within the Planets project, which aims to show the accessibility of 'aged' digital objects through emulated systems using the Internet (Webbrowser). GRATE is developed as a component system to allow easy user access to emulation services. GRATE is coded in Java / PHP / Perl / JavaScript, GRATE is based on the TightVNC technology and calls on the	GRATE is an environment that aggregates many virtualisation / emulation alternatives. As part of the Testbed, it offers greater flexibility in accessing emulation services and uploading files remotely, rather than having to install the VM software locally. In this sense, it is a more time- and cost-effective solution for testing, executing and viewing obsolete software in its native environment.	Platform independent system loaded remotely.	Standard Webbrowser with enabled Java Virtual Machine (JSE >= 1.4.2) JavaScript enabled.

Name	Description	Reason for considering	Config settings	Necessary resources
	DROID / PRONOM services provided by the The National Archives of the United Kingdom. It provides a number of virtual machines (DOS/Windows 3.11, Windows 98), emulators and binary translators (eg. DosBox, QEMU, Hatari, Dioscuri). GRATE has been integrated into the Planets Interoperability framework for viewing objects within an emulator (Create View feature).			

Decision: GO

Reason: All the above alternatives need to be evaluated as part of the experiment for generating comparative results between approaches.

Action needed: None

Alternative	Experiment description	Run description	Experiment data uploaded
DosBox 0.73 (WinXP)			No
DOS VM (WinXP)			No
Windows 98 VM (WinXP)			No
Wine for Linux			No
GRATE (PTB)			No

Hide Evaluation & Transformation

Usability > Installation Requirements

Results

Alternatives	1	2	3	4	5
DosBox 0.73 (WinXP)	Solution inapplicable	Solution inapplicable	Solution inapplicable	Solution inapplicable	Without Installation
DOS VM (WinXP)	Solution inapplicable	Solution inapplicable	Solution inapplicable	Solution inapplicable	With Installation
Windows 98 VM (WinXP)	Without Installation	Without Installation	Without Installation	Solution inapplicable	Without Installation
Wine for Linux	Without Installation	Without Installation	Without Installation	With Installation	Without Installation
GRATE (PTB)	Without Installation	Without Installation	Without Installation	Solution inapplicable	Without Installation

Transformer

Ordinal Value	Target Value
With Installation	-> 2.0
Without Installation	-> 5.0
Solution inapplicable	-> 1.0

Transformed Results

Alternatives	1	2	3	4	5	Aggregated	Comments
DosBox 0.73 (WinXP)	1	1	1	1	5	1.8	
DOS VM (WinXP)	1	1	1	1	2	1.2	
Windows 98 VM (WinXP)	5	5	5	1	5	4.2	
Wine for Linux	5	5	5	2	5	4.4	
GRATE (PTB)	5	5	5	1	5	4.2	

Aggregation mode: Arithmetic mean

Usability > Artefact Playable (if interactive)

Results

Transformer

Transformed Results

Alternatives	1	2	3	4	5
DosBox 0.73 (WinXP)	Solution inapplicable	Solution inapplicable	Solution inapplicable	Solution inapplicable	Not Interactive
DOS VM (WinXP)	Solution inapplicable	Solution inapplicable	Solution inapplicable	Solution inapplicable	Not Interactive
Windows 98 VM (WinXP)	Not Interactive	Not Interactive	Not Interactive	Solution inapplicable	Not Interactive
Wine for Linux	Not Interactive	Not Interactive	Not Interactive	Yes	Not Interactive
GRATE (PTB)	Not Interactive	Not Interactive	Not Interactive	Solution inapplicable	Not Interactive

Ordinal Value	Target Value
Yes	-> 5.0
No	-> 0.0
Not Interactive	-> 0.0
Solution inapplicable	-> 1.0

Alternatives	1	2	3	4	5	Aggregated	Comments
DosBox 0.73 (WinXP)	1	1	1	1	0	0.8	
DOS VM (WinXP)	1	1	1	1	0	0.8	
Windows 98 VM (WinXP)	0	0	0	1	0	0.2	
Wine for Linux	0	0	0	5	0	1	
GRATE (PTB)	0	0	0	1	0	0.2	

Aggregation mode: Arithmetic mean

Configurability > Artefact Configuration

Results

Alternatives	1	2	3	4	5
DosBox 0.73 (WinXP)	Solution inapplicable	Solution inapplicable	Solution inapplicable	Solution inapplicable	Pre-configured
DOS VM (WinXP)	Solution inapplicable	Solution inapplicable	Solution inapplicable	Solution inapplicable	Pre-configured
Windows 98 VM (WinXP)	Not needed	Not needed	Not needed	Solution inapplicable	Pre-configured
Wine for Linux	Not needed	Not needed	Not needed	Manual	Pre-configured
GRATE (PTB)	Not needed	Not needed	Not needed	Solution inapplicable	Pre-configured

Transformer

Ordinal Value	Target Value
Manual	-> 1.0
Pre-configured	-> 3.0
Not needed	-> 5.0
Solution inapplicable	-> 1.0

Transformed Results

Alternatives	1	2	3	4	5	Aggregated	Comments
DosBox 0.73 (WinXP)	1	1	1	1	3	1.4	Caero has pre-configured settings for detcting sound card
DOS VM (WinXP)	1	1	1	1	3	1.4	Caero has pre-configured settings for detcting sound card
Windows 98 VM (WinXP)	5	5	5	1	3	3.8	Caero has pre-configured settings for detcting sound card
Wine for Linux	5	5	5	1	3	3.8	Caero has pre-configured settings for detcting sound card
GRATE (PTB)	5	5	5	1	3	3.8	

Aggregation mode: Arithmetic mean

Configurability > Global System Preferences

Results

Alternatives	Single
DosBox 0.73 (WinXP)	Pre-configured
DOS VM (WinXP)	Manual
Windows 98 VM (WinXP)	Manual

Transformer

Ordinal Value	Target Value
Manual	-> 1.0
Pre-configured	-> 5.0
Not applicable	-> 5.0

Transformed Results

Alternatives	Single (=Aggregated)	Comments
DosBox 0.73 (WinXP)	5	
DOS VM (WinXP)	1	
Windows 98 VM (WinXP)	1	

Alternatives	Single
Wine for Linux	Pre-configured
GRATE (PTB)	Pre-configured

Alternatives	Single (=Aggregated)	Comments
Wine for Linux	5	
GRATE (PTB)	5	

Configurability > Ease of Configuration

Results

Alternatives	Single
DosBox 0.73 (WinXP)	Easy (less than 15 mins)
DOS VM (WinXP)	Difficult (more than 60 mins)
Windows 98 VM (WinXP)	Difficult (more than 60 mins)
Wine for Linux	Moderate (16-60 mins)
GRATE (PTB)	Easy (less than 15 mins)

Transformer

Ordinal Value	Target Value
Easy (less than 15 mins)	-> 5.0
Moderate (16-60 mins)	-> 3.0
Difficult (more than 60 mins)	-> 1.0

Transformed Results

Alternatives	Single (=Aggregated)	Comments
DosBox 0.73 (WinXP)	5	
DOS VM (WinXP)	1	
Windows 98 VM (WinXP)	1	
Wine for Linux	3	
GRATE (PTB)	5	

Costs > Cost for acquiring preservation software

Results

Alternatives	Single
DosBox 0.73 (WinXP)	None (freeware)
DOS VM (WinXP)	None (freeware)
Windows 98 VM (WinXP)	None (freeware)
Wine for Linux	None (freeware)
GRATE (PTB)	None (freeware)

Transformer

Ordinal Value	Target Value
None (freeware)	-> 5.0
Minimal	-> 3.0
High	-> 1.0

Transformed Results

Alternatives	Single (=Aggregated)	Comments
DosBox 0.73 (WinXP)	5	
DOS VM (WinXP)	5	
Windows 98 VM (WinXP)	5	
Wine for Linux	5	
GRATE (PTB)	5	

Costs > Number of artefacts supported

Results

Alternatives	Single
DosBox 0.73 (WinXP)	more than 2000
DOS VM (WinXP)	501-2000
Windows 98 VM (WinXP)	501-2000
Wine for Linux	101-500
GRATE (PTB)	501-2000

Transformer

Ordinal Value	Target Value
Less than 10	-> 1.0
10-100	-> 2.0
101-500	-> 3.0
501-2000	-> 4.0
more than 2000	-> 5.0

Transformed Results

Alternatives	Single (=Aggregated)	Comments
DosBox 0.73 (WinXP)	5	
DOS VM (WinXP)	4	
Windows 98 VM (WinXP)	4	
Wine for Linux	3	
GRATE (PTB)	4	

Object Characteristics > Artefact Renders

Results

Alternatives	1	2	3	4	5
DosBox 0.73 (WinXP)	Solution Inapplicable	Solution Inapplicable	Solution Inapplicable	Solution Inapplicable	Yes
DOS VM (WinXP)	Solution Inapplicable	Solution Inapplicable	Solution Inapplicable	Solution Inapplicable	Yes

Transformer

Ordinal Value	Target Value
Yes	-> 5.0
No	-> 0.0

Transformed Results

Alternatives	1	2	3	4	5	Aggregated	Comments
DosBox 0.73 (WinXP)	1	1	1	1	5	1.8	
DOS VM (WinXP)	1	1	1	1	5	1.8	

Alternatives	1	2	3	4	5
Windows 98 VM (WinXP)	Yes	Yes	Yes	Solution Inapplicable	Yes
Wine for Linux	Yes	Yes	Yes	Yes	Yes
GRATE (PTB)	Yes	Yes	Yes	Solution Inapplicable	Yes

Ordinal Value	Target Value
Solution Inapplicable	-> 1.0

Alternatives	1	2	3	4	5	Aggregated	Comments
Windows 98 VM (WinXP)	5	5	5	1	5	4.2	
Wine for Linux	5	5	5	5	5	5	
GRATE (PTB)	5	5	5	1	5	4.2	

Aggregation mode: Arithmetic mean

Object Characteristics > Speed

Results

Alternatives	1	2	3	4	5
DosBox 0.73 (WinXP)	Solution inapplicable	Solution inapplicable	Solution inapplicable	Solution inapplicable	As in original
DOS VM (WinXP)	Solution inapplicable	Solution inapplicable	Solution inapplicable	Solution inapplicable	Slower than original
Windows 98 VM (WinXP)	Slower than original	As in original	Slower than original	Solution inapplicable	Slower than original
Wine for Linux	Slower than original	As in original	Slower than original	As in original	Slower than original
GRATE (PTB)	Slower than original	Slower than original	Slower than original	Slower than original	Slower than original

Transformer

Ordinal Value	Target Value
As in original	-> 5.0
Faster than original	-> 2.0
Slower than original	-> 2.0
Not applicable	-> 0.0
Solution inapplicable	-> 1.0

Transformed Results

Alternatives	1	2	3	4	5	Aggregated	Comments
DosBox 0.73 (WinXP)	1	1	1	1	5	1.8	
DOS VM (WinXP)	1	1	1	1	2	1.2	
Windows 98 VM (WinXP)	2	5	2	1	2	2.4	
Wine for Linux	2	5	2	5	2	3.2	
GRATE (PTB)	2	2	2	2	2	2	

Aggregation mode: Arithmetic mean

Controls with Standard PC input devices > Controls resemble originals

Results

Alternatives	1	2	3	4	5
DosBox 0.73 (WinXP)	Solution inapplicable	Solution inapplicable	Solution inapplicable	Solution inapplicable	Not applicable
DOS VM (WinXP)	Solution inapplicable	Solution inapplicable	Solution inapplicable	Solution inapplicable	Not applicable
Windows 98 VM (WinXP)	Not applicable	Not applicable	Not applicable	Solution inapplicable	Not applicable
Wine for Linux	Not applicable	Not applicable	Not applicable	Yes	Not applicable
GRATE (PTB)	Not applicable	Not applicable	Not applicable	Solution inapplicable	Not applicable

Transformer

Ordinal Value	Target Value
Yes	-> 5.0
No	-> 1.0
Not applicable	-> 0.0
Solution inapplicable	-> 1.0

Transformed Results

Alternatives	1	2	3	4	5	Aggregated	Comments
DosBox 0.73 (WinXP)	1	1	1	1	0	0.8	
DOS VM (WinXP)	1	1	1	1	0	0.8	
Windows 98 VM (WinXP)	0	0	0	1	0	0.2	
Wine for Linux	0	0	0	5	0	1	
GRATE (PTB)	0	0	0	1	0	0.2	

Aggregation mode: Arithmetic mean

Controls with Standard PC input devices > Response delay

Results

Alternatives	1	2	3	4	5

Transformer

Ordinal	Target

Transformed Results

Alternatives	1	2	3	4	5	Aggregated	Comments

Alternatives	1	2	3	4	5
DosBox 0.73 (WinXP)	Solution inapplicable	Solution inapplicable	Solution inapplicable	Solution inapplicable	Not applicable
DOS VM (WinXP)	Solution inapplicable	Solution inapplicable	Solution inapplicable	Solution inapplicable	Not applicable
Windows 98 VM (WinXP)	Not applicable	Not applicable	Not applicable	Solution inapplicable	Not applicable
Wine for Linux	Not applicable	Not applicable	Not applicable	No delay	Not applicable
GRATE (PTB)	Not applicable	Not applicable	Not applicable	Solution inapplicable	Not applicable

Value	Value
Considerable delay	-> 1.0
Short delay	-> 3.0
No delay	-> 5.0
Not applicable	-> 0.0
Solution inapplicable	-> 5.0

Alternatives	1	2	3	4	5	Aggregated	Comments
DosBox 0.73 (WinXP)	5	5	5	5	0	4	
DOS VM (WinXP)	5	5	5	5	0	4	
Windows 98 VM (WinXP)	0	0	0	5	0	1	
Wine for Linux	0	0	0	5	0	1	
GRATE (PTB)	0	0	0	5	0	1	

Aggregation mode: Arithmetic mean

Controls with Standard PC input devices > Feel of original artistic experience

Results

Alternatives	1	2	3	4	5
DosBox 0.73 (WinXP)	Solution inapplicable	Solution inapplicable	Solution inapplicable	Solution inapplicable	Perfectly recreated
DOS VM (WinXP)	Solution inapplicable	Solution inapplicable	Solution inapplicable	Solution inapplicable	Playable
Windows 98 VM (WinXP)	Playable	Perfectly recreated	Playable	Solution inapplicable	Playable
Wine for Linux	Perfectly recreated	Perfectly recreated	Playable	Perfectly recreated	Playable
GRATE (PTB)	Playable	Playable	Playable	Solution inapplicable	Playable

Transformer

Ordinal Value	Target Value
Not acceptable	-> 1.0
Playable	-> 3.0
Perfectly recreated	-> 5.0
Not applicable	-> 0.0
Solution inapplicable	-> 1.0

Transformed Results

Alternatives	1	2	3	4	5	Aggregated	Comments
DosBox 0.73 (WinXP)	1	1	1	1	5	1.8	
DOS VM (WinXP)	1	1	1	1	3	1.4	
Windows 98 VM (WinXP)	3	5	3	1	3	3	
Wine for Linux	5	5	3	5	3	4.2	
GRATE (PTB)	3	3	3	1	3	2.6	

Aggregation mode: Arithmetic mean

Music > Quality

Results

Alternatives	1	2	3	4	5
DosBox 0.73 (WinXP)	Solution inapplicable	Solution inapplicable	Solution inapplicable	Solution inapplicable	As in original
DOS VM (WinXP)	Solution inapplicable	Solution inapplicable	Solution inapplicable	Solution inapplicable	Unacceptable
Windows 98 VM (WinXP)	No music	Unacceptable	Unacceptable	Solution inapplicable	Unacceptable
Wine for Linux	No music	As in original	Acceptable	As in original	Acceptable
GRATE (PTB)	Unacceptable	Unacceptable	Unacceptable	Unacceptable	Unacceptable

Transformer

Ordinal Value	Target Value
No music	-> 5.0
Unacceptable	-> 1.0
Acceptable	-> 3.0
Near Perfect	-> 4.0
As in original	-> 5.0
Not applicable	-> 0.0
Solution inapplicable	-> 1.0

Transformed Results

Alternatives	1	2	3	4	5	Aggregated	Comments
DosBox 0.73 (WinXP)	1	1	1	1	5	1.8	
DOS VM (WinXP)	1	1	1	1	1	1	
Windows 98 VM (WinXP)	5	1	1	1	1	1.8	Only mono sound for Caero
Wine for Linux	5	5	3	5	3	4.2	
GRATE (PTB)	1	1	1	1	1	1	No sound produced through GRATE

Aggregation mode: Arithmetic mean

Music > Synchronous to video

Results

Alternatives	1	2	3	4	5
DosBox 0.73 (WinXP)	Solution inapplicable	Solution inapplicable	Solution inapplicable	Solution inapplicable	No errors
DOS VM (WinXP)	Solution inapplicable	Solution inapplicable	Solution inapplicable	Solution inapplicable	Severe errors
Windows 98 VM (WinXP)	Not applicable	No errors	Small errors	Solution inapplicable	Severe errors
Wine for Linux	Not applicable	No errors	No errors	No errors	Severe errors
GRATE (PTB)	Severe errors	Severe errors	Severe errors	Severe errors	Severe errors

Transformer

Ordinal Value	Target Value
Not applicable	-> 0.0
Severe errors	-> 1.0
Small errors	-> 3.0
No errors	-> 5.0
Solution inapplicable	-> 1.0

Transformed Results

Alternatives	1	2	3	4	5	Aggregated	Comments
DosBox 0.73 (WinXP)	1	1	1	1	5	1.8	
DOS VM (WinXP)	1	1	1	1	1	1	
Windows 98 VM (WinXP)	0	5	3	1	1	2	
Wine for Linux	0	5	5	5	1	3.2	
GRATE (PTB)	1	1	1	1	1	1	No sound produced through GRATE

Aggregation mode: Arithmetic mean

Graphics > Overall Image quality

Results

Alternatives	1	2	3	4	5
DosBox 0.73 (WinXP)	Solution inapplicable	Solution inapplicable	Solution inapplicable	Solution inapplicable	As in original
DOS VM (WinXP)	Solution inapplicable	Solution inapplicable	Solution inapplicable	Solution inapplicable	Errors but not affecting experience
Windows 98 VM (WinXP)	As in original	As in original	Errors but not affecting experience	Solution inapplicable	Severe errors that affect experience
Wine for Linux	As in original	As in original	Errors but not affecting experience	As in original	Errors but not affecting experience
GRATE (PTB)	As in original	As in original	As in original	Solution inapplicable	As in original

Transformer

Ordinal Value	Target Value
Nothing displayed	-> 0.0
Severe errors that affect experience	-> 1.0
Errors but not affecting experience	-> 3.0
Near perfect	-> 4.0
As in original	-> 5.0
Solution inapplicable	-> 1.0

Transformed Results

Alternatives	1	2	3	4	5	Aggregated	Comments
DosBox 0.73 (WinXP)	1	1	1	1	5	1.8	
DOS VM (WinXP)	1	1	1	1	3	1.4	
Windows 98 VM (WinXP)	5	5	3	1	1	3	
Wine for Linux	5	5	3	5	3	4.2	
GRATE (PTB)	5	5	5	1	5	4.2	

Aggregation mode: Arithmetic mean

Graphics > Colours

Results

Alternatives	1	2	3	4	5
DosBox 0.73 (WinXP)	Solution inapplicable	Solution inapplicable	Solution inapplicable	Solution inapplicable	As in original
DOS VM (WinXP)	Solution inapplicable	Solution inapplicable	Solution inapplicable	Solution inapplicable	Severe errors

Transformer

Ordinal Value	Target Value
Not applicable	-> 0.0

Transformed Results

Alternatives	1	2	3	4	5	Aggregated	Comments
DosBox 0.73 (WinXP)	1	1	1	1	5	1.8	
DOS VM (WinXP)	1	1	1	1	1	1	At certain points, the

Alternatives	1	2	3	4	5
					affecting experience
Windows 98 VM (WinXP)	As in original	As in original	Near perfect	Solution inapplicable	Severe errors affecting experience
Wine for Linux	As in original	As in original	As in original	As in original	As in original
GRATE (PTB)	As in original	As in original	As in original	As in original	As in original

Ordinal Value	Target Value
Severe errors affecting experience	-> 1.0
Errors but not affecting experience	-> 3.0
Near perfect	-> 4.0
As in original	-> 5.0
Solution inapplicable	-> 1.0

Alternatives	1	2	3	4	5	Aggregated	Comments
							colour palette is completely wrong
Windows 98 VM (WinXP)	5	5	4	1	1	3.2	At certain points, the colour palette is completely wrong for Caero
Wine for Linux	5	5	5	5	5	5	
GRATE (PTB)	5	5	5	5	5	5	

Aggregation mode: Arithmetic mean

Documentation > Instuctions available

Results

Alternatives	1	2	3	4	5
DosBox 0.73 (WinXP)	Solution inapplicable	Solution inapplicable	Solution inapplicable	Solution inapplicable	Yes
DOS VM (WinXP)	Solution inapplicable	Solution inapplicable	Solution inapplicable	Solution inapplicable	Yes
Windows 98 VM (WinXP)	Yes	Yes	Yes	Solution inapplicable	Yes
Wine for Linux	Yes	Yes	Yes	Yes	Yes
GRATE (PTB)	Yes	Yes	Yes	Solution inapplicable	Yes

Transformer

Ordinal Value	Target Value
Yes	-> 5.0
No	-> 1.0
Solution inapplicable	-> 1.0

Transformed Results

Alternatives	1	2	3	4	5	Aggregated	Comments
DosBox 0.73 (WinXP)	1	1	1	1	5	1.8	
DOS VM (WinXP)	1	1	1	1	5	1.8	
Windows 98 VM (WinXP)	5	5	5	1	5	4.2	
Wine for Linux	5	5	5	5	5	5	
GRATE (PTB)	5	5	5	1	5	4.2	

Aggregation mode: Arithmetic mean

Documentation > Original media available

Results

Alternatives	1	2	3	4	5
DosBox 0.73 (WinXP)	Solution inapplicable	Solution inapplicable	Solution inapplicable	Solution inapplicable	Yes
DOS VM (WinXP)	Solution inapplicable	Solution inapplicable	Solution inapplicable	Solution inapplicable	Yes
Windows 98 VM (WinXP)	Yes	Yes	Yes	Solution inapplicable	Yes
Wine for Linux	Yes	Yes	Yes	Yes	Yes
GRATE (PTB)	Yes	Yes	Yes	Solution inapplicable	Yes

Transformer

Ordinal Value	Target Value
Yes	-> 5.0
No	-> 1.0
Solution inapplicable	-> 1.0

Transformed Results

Alternatives	1	2	3	4	5	Aggregated	Comments
DosBox 0.73 (WinXP)	1	1	1	1	5	1.8	
DOS VM (WinXP)	1	1	1	1	5	1.8	
Windows 98 VM (WinXP)	5	5	5	1	5	4.2	
Wine for Linux	5	5	5	5	5	5	
GRATE (PTB)	5	5	5	1	5	4.2	

Aggregation mode: Arithmetic mean

Documentation > System requirements specification

Results

Alternatives	1	2	3	4	5
DosBox 0.73 (WinXP)	Solution inapplicable	Solution inapplicable	Solution inapplicable	Solution inapplicable	Yes
DOS VM (WinXP)	Solution inapplicable	Solution inapplicable	Solution inapplicable	Solution inapplicable	Yes
Windows 98 VM (WinXP)	Yes	Yes	Yes	Solution inapplicable	Yes
Wine for Linux	Yes	Yes	Yes	No	Yes
GRATE (PTB)	Yes	Yes	Yes	Solution inapplicable	Yes

Transformer

Ordinal Value	Target Value
Yes	-> 5.0
No	-> 1.0
Solution inapplicable	-> 1.0

Transformed Results

Alternatives	1	2	3	4	5	Aggregated	Comments
DosBox 0.73 (WinXP)	1	1	1	1	5	1.8	
DOS VM (WinXP)	1	1	1	1	5	1.8	
Windows 98 VM (WinXP)	5	5	5	1	5	4.2	
Wine for Linux	5	5	5	1	5	4.2	
GRATE (PTB)	5	5	5	1	5	4.2	

Aggregation mode: Arithmetic mean

Scalability > Modular design

Results

Alternatives	Single
DosBox 0.73 (WinXP)	Code modular (one system support)
DOS VM (WinXP)	Code modular (one system support)
Windows 98 VM (WinXP)	Code modular (one system support)
Wine for Linux	Code modular (one system support)
GRATE (PTB)	Code modular (multiple system support)

Transformer

Ordinal Value	Target Value
Code not available	-> 1.0
Code not modular	-> 2.0
Code modular (one system support)	-> 3.0
Code modular (multiple system support)	-> 5.0

Transformed Results

Alternatives	Single (=Aggregated)	Comments
DosBox 0.73 (WinXP)	3	
DOS VM (WinXP)	3	
Windows 98 VM (WinXP)	3	
Wine for Linux	3	
GRATE (PTB)	5	

Scalability > Portability

Results

Alternatives	Single
DosBox 0.73 (WinXP)	Platform dependent
DOS VM (WinXP)	Platform dependent
Windows 98 VM (WinXP)	Platform dependent
Wine for Linux	Platform dependent
GRATE (PTB)	Platform independent

Transformer

Ordinal Value	Target Value
Not applicable	-> 0.0
Platform dependent	-> 1.0
Platform independent	-> 5.0

Transformed Results

Alternatives	Single (=Aggregated)	Comments
DosBox 0.73 (WinXP)	1	
DOS VM (WinXP)	1	
Windows 98 VM (WinXP)	1	
Wine for Linux	1	
GRATE (PTB)	5	

Stability > Development status

Results

Alternatives	Single
DosBox 0.73 (WinXP)	Free (open source)
DOS VM (WinXP)	Free (open source)

Transformer

Ordinal Value	Target Value
Commercial (open source)	-> 3.0
Commercial (non-open source)	-> 2.0

Transformed Results

Alternatives	Single (=Aggregated)	Comments
DosBox 0.73 (WinXP)	5	
DOS VM (WinXP)	5	

Alternatives	Single
Windows 98 VM (WinXP)	Free (open source)
Wine for Linux	Free (open source)
GRATE (PTB)	Free (open source)

Ordinal Value	Target Value
Free (open source)	-> 5.0
Free (non-open source)	-> 4.0

Alternatives	Single (=Aggregated)	Comments
Windows 98 VM (WinXP)	5	
Wine for Linux	5	
GRATE (PTB)	5	

Stability > Current support

Results

Alternatives	Single
DosBox 0.73 (WinXP)	Yes
DOS VM (WinXP)	Yes
Windows 98 VM (WinXP)	Yes
Wine for Linux	Yes
GRATE (PTB)	Yes

Transformer

Ordinal Value	Target Value
Yes	-> 5.0
No	-> 1.0

Transformed Results

Alternatives	Single (=Aggregated)	Comments
DosBox 0.73 (WinXP)	5	
DOS VM (WinXP)	5	
Windows 98 VM (WinXP)	5	
Wine for Linux	5	
GRATE (PTB)	5	

Stability > In development

Results

Alternatives	Single
DosBox 0.73 (WinXP)	Yes
DOS VM (WinXP)	Yes
Windows 98 VM (WinXP)	Yes
Wine for Linux	Yes
GRATE (PTB)	Yes

Transformer

Ordinal Value	Target Value
Yes	-> 5.0
No	-> 1.0

Transformed Results

Alternatives	Single (=Aggregated)	Comments
DosBox 0.73 (WinXP)	5	
DOS VM (WinXP)	5	
Windows 98 VM (WinXP)	5	
Wine for Linux	5	
GRATE (PTB)	5	

Stability > Active community

Results

Alternatives	Single
DosBox 0.73 (WinXP)	Yes
DOS VM (WinXP)	Yes
Windows 98 VM (WinXP)	Yes
Wine for Linux	Yes
GRATE (PTB)	Yes

Transformer

Ordinal Value	Target Value
Yes	-> 5.0
No	-> 1.0

Transformed Results

Alternatives	Single (=Aggregated)	Comments
DosBox 0.73 (WinXP)	5	
DOS VM (WinXP)	5	
Windows 98 VM (WinXP)	5	
Wine for Linux	5	
GRATE (PTB)	5	

Infrastructure > Host-Guest OS communication

Results

Alternatives	Single
DosBox 0.73 (WinXP)	Via virtual media
DOS VM (WinXP)	Via virtual media
Windows 98 VM (WinXP)	Via virtual media

Transformer

Ordinal Value	Target Value
Via Network	-> 2.0
Native	-> 5.0
Via virtual media	-> 3.0

Transformed Results

Alternatives	Single (=Aggregated)	Comments
DosBox 0.73 (WinXP)	3	
DOS VM (WinXP)	3	
Windows 98 VM (WinXP)	3	

Alternatives	Single
Wine for Linux	Native
GRATE (PTB)	Via virtual media

Ordinal Value	Target Value
None	-> 1.0

Alternatives	Single (=Aggregated)	Comments
Wine for Linux	5	
GRATE (PTB)	3	

Legality > Alterations to original permitted

Results

Alternatives	1	2	3	4	5
DosBox 0.73 (WinXP)	No	No	No	No	No
DOS VM (WinXP)	No	No	No	No	No
Windows 98 VM (WinXP)	No	No	No	No	No
Wine for Linux	No	No	No	No	No
GRATE (PTB)	No	No	No	No	No

Transformer

Ordinal Value	Target Value
Yes	-> 5.0
No	-> 1.0
Solution inapplicable	-> 1.0

Transformed Results

Alternatives	1	2	3	4	5	Aggregated	Comments
DosBox 0.73 (WinXP)	1	1	1	1	1	1	
DOS VM (WinXP)	1	1	1	1	1	1	
Windows 98 VM (WinXP)	1	1	1	1	1	1	
Wine for Linux	1	1	1	1	1	1	
GRATE (PTB)	1	1	1	1	1	1	

Aggregation mode: Arithmetic mean

Legality > Copies of original permitted

Results

Alternatives	1	2	3	4	5
DosBox 0.73 (WinXP)	Yes	Yes	Yes	Yes	Yes
DOS VM (WinXP)	Yes	Yes	Yes	Yes	Yes
Windows 98 VM (WinXP)	Yes	Yes	Yes	Yes	Yes
Wine for Linux	Yes	Yes	Yes	Yes	Yes
GRATE (PTB)	Yes	Yes	Yes	Yes	Yes

Transformer

Ordinal Value	Target Value
Yes	-> 5.0
No	-> 1.0
Solution inapplicable	-> 1.0

Transformed Results

Alternatives	1	2	3	4	5	Aggregated	Comments
DosBox 0.73 (WinXP)	5	5	5	5	5	5	
DOS VM (WinXP)	5	5	5	5	5	5	
Windows 98 VM (WinXP)	5	5	5	5	5	5	
Wine for Linux	5	5	5	5	5	5	
GRATE (PTB)	5	5	5	5	5	5	

Aggregation mode: Arithmetic mean

Legality > Operating System IP status

Results

Alternatives	Single
DosBox 0.73 (WinXP)	Freeware
DOS VM (WinXP)	Protected
Windows 98 VM (WinXP)	Protected
Wine for Linux	Freeware
GRATE (PTB)	Freeware

Transformer

Ordinal Value	Target Value
Freeware	-> 5.0
Protected	-> 1.0

Transformed Results

Alternatives	Single (=Aggregated)	Comments
DosBox 0.73 (WinXP)	5	
DOS VM (WinXP)	1	
Windows 98 VM (WinXP)	1	
Wine for Linux	5	
GRATE (PTB)	5	

Media > Keyboard/mouse support

Results

Alternatives	Single
DosBox 0.73 (WinXP)	Yes
DOS VM (WinXP)	Yes

Transformer

Ordinal Value	Target Value
Yes	-> 5.0
No	-> 1.0

Transformed Results

Alternatives	Single (=Aggregated)	Comments
DosBox 0.73 (WinXP)	5	
DOS VM (WinXP)	5	

Alternatives	Single
Windows 98 VM (WinXP)	Yes
Wine for Linux	Yes
GRATE (PTB)	Yes

Alternatives	Single (=Aggregated)	Comments
Windows 98 VM (WinXP)	5	
Wine for Linux	5	
GRATE (PTB)	5	

Media > CD-ROM support

Results

Alternatives	Single
DosBox 0.73 (WinXP)	Pre-configured
DOS VM (WinXP)	Manual
Windows 98 VM (WinXP)	Pre-configured
Wine for Linux	Pre-configured
GRATE (PTB)	None

Transformer

Ordinal Value	Target Value
Manual	-> 1.0
Pre-configured	-> 5.0
None	-> 1.0

Transformed Results

Alternatives	Single (=Aggregated)	Comments
DosBox 0.73 (WinXP)	5	
DOS VM (WinXP)	1	
Windows 98 VM (WinXP)	5	
Wine for Linux	5	
GRATE (PTB)	1	

Media > Sound support

Results

Alternatives	Single
DosBox 0.73 (WinXP)	Pre-configured
DOS VM (WinXP)	Manual
Windows 98 VM (WinXP)	Manual
Wine for Linux	Pre-configured
GRATE (PTB)	None

Transformer

Ordinal Value	Target Value
Manual	-> 1.0
Pre-configured	-> 5.0
None	-> 1.0

Transformed Results

Alternatives	Single (=Aggregated)	Comments
DosBox 0.73 (WinXP)	5	
DOS VM (WinXP)	1	
Windows 98 VM (WinXP)	1	
Wine for Linux	5	
GRATE (PTB)	1	

Media > OS pre-installed

Results

Alternatives	Single
DosBox 0.73 (WinXP)	Yes
DOS VM (WinXP)	No
Windows 98 VM (WinXP)	No
Wine for Linux	Yes
GRATE (PTB)	Yes

Transformer

Ordinal Value	Target Value
Yes	-> 5.0
No	-> 1.0

Transformed Results

Alternatives	Single (=Aggregated)	Comments
DosBox 0.73 (WinXP)	5	
DOS VM (WinXP)	1	
Windows 98 VM (WinXP)	1	
Wine for Linux	5	
GRATE (PTB)	5	

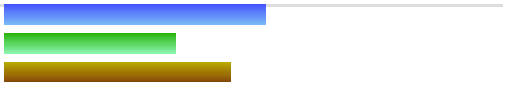
Evaluation comments:

Transformation comments:

How do the aggregation mechanisms work?

Result-Tree with all Alternatives, Aggregation method: Weighted multiplication

Node	Results
Software Art Preservation Requirements	DosBox 0.73 (WinXP): 2.68 DOS VM (WinXP): 1.75 Windows 98 VM (WinXP): 2.30



	Wine for Linux:	3.76
	GRATE (PTB):	2.77
Process Characteristics	DosBox 0.73 (WinXP):	1.05
	DOS VM (WinXP):	0.99
	Windows 98 VM (WinXP):	0.95
	Wine for Linux:	1.08
	GRATE (PTB):	1.00
Costs	DosBox 0.73 (WinXP):	1.17
	DOS VM (WinXP):	1.16
	Windows 98 VM (WinXP):	1.16
	Wine for Linux:	1.14
	GRATE (PTB):	1.16
Object Characteristics	DosBox 0.73 (WinXP):	1.19
	DOS VM (WinXP):	1.10
	Windows 98 VM (WinXP):	1.34
	Wine for Linux:	1.51
	GRATE (PTB):	1.31
Context and Data Characteristics	DosBox 0.73 (WinXP):	1.06
	DOS VM (WinXP):	1.06
	Windows 98 VM (WinXP):	1.15
	Wine for Linux:	1.17
	GRATE (PTB):	1.15
Infrastructure	DosBox 0.73 (WinXP):	1.11
	DOS VM (WinXP):	1.11
	Windows 98 VM (WinXP):	1.11
	Wine for Linux:	1.12
	GRATE (PTB):	1.16
Legality	DosBox 0.73 (WinXP):	1.12
	DOS VM (WinXP):	1.07
	Windows 98 VM (WinXP):	1.07
	Wine for Linux:	1.12
	GRATE (PTB):	1.12
Media	DosBox 0.73 (WinXP):	1.38
	DOS VM (WinXP):	1.10
	Windows 98 VM (WinXP):	1.14
	Wine for Linux:	1.38
	GRATE (PTB):	1.21

Result-Tree with all Alternatives, Aggregation method: Weighted sum.
This tree contains only strategies that do not have knock-out evaluation criteria; see above

Node	**Results**	
Software Art Preservation Requirements	DosBox 0.73 (WinXP):	3.17
	DOS VM (WinXP):	2.19
	Windows 98 VM (WinXP):	2.99
	Wine for Linux:	4.14
	GRATE (PTB):	3.53