

Virtualization for Preservation of Executable Art

Kam Woods

Indiana University
Department of Computer Science
kamwoods@cs.indiana.edu

Abstract. In this work I present performance data and failure analysis related to the use of emulated environments to view “demoscene” audio-visual presentations originally intended for execution on a variety of computing platforms. Open-source tools and operating systems are used exclusively in order to evaluate the efficacy of this preservation technique.

1 Introduction

The use of machine emulation to enable access to executable audio-visual content in legacy environments has been notably [4], [5] - if somewhat sparsely - documented. In this project I demonstrate an emulation/virtual-ization environment constructed using open-source tools and intended to provide one-click access to original “demoscene” executables. These presentations represent a multi-decade span of artistic endeavor which have over time gained full recognition for their impact on new media art [3]. The project analyzes the execution of a representative selection of these programs on a Linux host system using emulation tools including Wine [7], UAE [6], and Hatari [1] with respect to audio, video, and timing performance. As demoscene software frequently exploits low-level programming routines such as custom interrupt processing to maximize performance, failure modes are recorded and addressed.

These executable objects typically encapsulate various issues encountered in handling complex media within a single file, along with implicit expectations about the composition of their execution environment and its limitations. They are therefore uniquely suited to an emulation strategy, as the art *is* the real-time output of the program [2]. The use of open-source software for the emulation environment, along with scripted installation and execution, mitigates a number of issues associated with the future maintenance and operation of both object and toolset. These include providing a unified front end for otherwise disparate emulation modes, leveraging the considerable work that has been invested in construction of emulators for these environments, and providing a natural packaging of the executable along with extended metadata covering usage criteria into a well-defined preservation object.

References

1. About Hatari. <http://hatari.sourceforge.net/>. Accessed May, 2008.
2. J. Hunter and S. Choudhury. Implementing Preservation Strategies for Complex Multimedia Objects. *ECDL 2003: The Seventh European Conference on Research and Advanced Technology for Digital Libraries*, 2003.
3. Collected BibTeX Entries of Demoscene Publications. <http://www.kameli.net/demoresearch/>.
4. Pil Mellor. CaMiLEON: emulation and BBC doomsday. *RLG DigiNews*, 7(2), 2003.
5. Seeing double emulation theory and practice. <http://www.variablemedia.net/e/seeingdouble/home.html>. Accessed November, 2006.
6. About UAE. <http://uae.coresystems.de/>. Accessed May, 2008.
7. About Wine. <http://www.foolabs.com/xpdf/>. Accessed September, 2006.