Retrocomputing as Preservation

Yuri Takhteyev
University of Toronto
140 St. George Street
Toronto, ON M5S 3G6
yuri.takhteyev@utoronto.ca

Quinn DuPont
University of Toronto
140 St. George Street
Toronto, ON M5S 3G6
quinn.dupont@utoronto.ca

ABSTRACT

This project explores the world of retrocomputing, a constellation of largely—though not exclusively—non-professional practices involving old computing technology. Retrocomputing includes many activities that can be seen as constituting "preservation," and in particular digital preservation. At the same time, however, it is often transformative, producing assemblages that "remix" fragments from the past with newer elements or joining historic components that were never previously combined. While such "remix" may seem to undermine preservation, it allows for fragments of computing history to be reintegrated into a living, ongoing practice, contributing to preservation in a broader sense. The seemingly unorganized nature of retrocomputing assemblages also provides space for alternative "situated knowledges" and histories of computing, which can be quite sophisticated. Recognizing such alternative epistemologies in turn paves the way for alternative approaches to preservation. The institutional digital preservation community may have a lot to gain from paying closer attention to retrocomputing. This gain, however, should not just involve looking for ways to make use of the knowledge and labor of retrocomputing enthusiasts. Rather, it is important to recognize the value of their projects on their own terms and ask in what ways institutional efforts can support such projects.

Keywords

retrocomputing, software preservation, remix

In late March of 2012 Jordan Mechner received a shipment from his father, a box full of old floppies. Among them was a 3.5 inch disk labelled "Prince of Persia Source Code (Apple)." Mechner's announcement of this find on his blog the next day took the world of nerds by storm. *Prince of Persia*, a game that Mechner developed in the late 1980s, revolutionized the world of computer games through its surprisingly realistic representation of human movement. After being ported to DOS and Apple's Mac OS in the early 1990s the game sold 2 million copies.

Mechner's original 1989 version, however, was written for Apple II, a platform already somewhat outdated at the time, and featured much more modest graphics and sound than the later DOS and Mac versions. This early version is still remembered—and played—by the aficionados, being easily available on the Internet in the form of disk image files derived from a "crack" of the game produced around 1990, credited to "The Crasher" and associates, and bearing a dedication to "Nebraska Cement Factory."

The easiest way to run such images is to load them on one of the many Apple II emulators available online. For the more dedicated fans, however, there is the option of using original hardware. For some, this original hardware, is of course, Apple II. For others, original hardware can mean *other* 1980s computers, including some that could not run the game at the time. For example, in 2011 a programmer known as "mrsid" successfully completed the project of porting the Apple II version of *Prince of Persia* to Commodore 64, a task that took him two and a half years. Projects such as mrsid's would be much easier if the source code of the game were available. Yet, the code had long been presumed lost. Mechner's discovery of the floppy thus generated much excitement.

The find, however, also presented a challenge. "I will now begin working with a digital-archeology-minded friend to attempt to figure out how to transfer 3.5″ Apple ProDOS disks onto a MacBook Air and into some kind of 21st-century-readable format," Mechner wrote on his blog. Mechner's call for assistance brought two men to his door a few weeks later. One was Jason Scott, best known as the maintainer of textfiles.com, a website originally dedicated to preserving thousands of ASCII files shared on bulletin-board systems (BBS) in the 1980s and early 1990s, but then expanded to collect shareware CD images, audio files, and other digital artifacts from the era. The other man was Tony Diaz, a collector of Apple II hardware and the maintainer of the website apple2.org, dedicated to images of Apple II. Each man came with somewhat different tools. Scott brought DiscFerret, a small open-hardware device designed to read raw pattern of magnetism from a floppy, leaving the analysis and digitization of the pattern to a software tool, thus allowing flexible support for a wide range of approaches for storing data, as well as an ability to circumvent many antique copy-protection schemes. Diaz arrived with a van full of Apple II hardware—original, though rigged with substantial hardware and software modifications, including support for Ethernet, not available on the original Apple II.

With their help, Mechner's original files were transferred to his MacBook Air, in a live-tweeted session tagged "#popsource" that attracted so much attention that Mechner's website collapsed from the traffic. The source code was then quickly made available on GitHub, a site widely used for sharing open source code. Within hours, GitHub user "st3fan" made a modification commenting out the copy-protection code. This move was purely symbolic, since the posted code was incomplete at the time and could not actually be compiled and run. A few days later, however, a programmer working on an Apple II emulator credited the posted source code as a source of information that helped improve the emulator.

The story presented above provides a glimpse into the world of retrocomputing, a set of diverse practices involving contemporary engagement with old computer systems, which we have explored through a year long study combining online observation of retrocomputing projects and *in situ* interaction with the participants. Such practices are primarily private and non-professional, though this is not always the case—there is also a substantial economy providing products and services. And to the extent that retrocomputing participants are "hobbyists," in the

iPRESS2012, Oct 1-5, 2011, Toronto, ON, Canada.

sense of not paid for their work, they are hardly unskilled amateurs. Rather, their practice often demonstrates deep sophistication. In other words, many of them are "hobbyists" only in the same sense as many of the contributors to open source software, which today underlies much of the world's computing infrastructure. Retrocomputing often involves old games, yet many participants also engage with non-gaming technology.

Many of the activities that make up retrocomputing can be seen as constituting collection and preservation, and many retrocomputing enthusiasts in fact recognize preservation of computer history as one of their key goals. Such activities involve efforts to collect and restore old hardware, develop emulators, and build substantial collections of old software. For example, it does not take long to find on the Internet disk images for *Prince of Persia* for Apple II, as well as a *variety* of emulators that can run them. Some of the emulators are produced by open source projects in the full sense of the term, many are "not quite open source," for example distributed under licenses that prohibits commercial use. The differences highlight the distinct historic origins of retrocomputing and free software and the need to recognize retrocomputing itself as a historically situated practice.

Emulation requires images of original software. This means images of the original system software, as well as the application software that is to be emulated. Images collected and circulated by the hobbyists come from sundry sources. Some are actually quite old: the most commonly available image for the Apple II *Prince of Persia* appears to have originated around 1990. Some are more recent, but still produced by running image ripping software on old machines—or at least *older* machines. For example, one can read Apple II disks using early PC hardware with special software. This method can be quite challenging due to copy protection, as well as the gradual disappearance of older hardware. Perhaps the most sophisticated solution for this problem is exemplified by DiscFerret and KryoFlux—both hardware solutions that sit between a floppy disk drive and a contemporary computer, allowing the latter to scan the raw pattern of magnetization from a disk's surface, defeating many of the copy-protection methods employed in the 1980s. Both projects are run by private groups, in case of DiskFerret—as an open source and "open hardware" project.

At the same time, closer attention to those retrocomputing projects reveals that they cannot be easily understood as just a matter of preservation in the narrow sense of keeping objects from the past fixed in their "original" form. Instead, retrocomputing is often transformative and involves construction of assemblages that "remix" fragments of old systems with newer elements, such as old software running on freshly debugged emulators or original hardware enhanced with contemporary networking. It can also involve a mixture of historic components that were never combined in the past, as in the case of mrsid's porting of *Prince of Persia* to Commodore 64.

We conceptualize these transformative aspects of retrocomputing as a form of "remix"—a term popularized by Lessig (2008). Like the closely related concept of "collage," the term "remix" refers to a creative and often playful reassembly of fragments of earlier works into something new. While the reasons for chimeric assemblages described above are sometimes pragmatic, at other times they are simply playful, carried out for fun. At a gathering of Commodore enthusiasts attended by one of the authors, a participant demonstrated an old Commodore 64C system that he had skillfully painted bright blue. He explained that the computer was originally made of white plastic, but had turned an "ugly" yellow over time. Repainted blue, it looked "awesome." Quite often, though, the pursuit of fun and beauty cannot be easily separated from the "pragmatic" motivation for remixing fragments of old computing. Much like Linus Torvalds describing his development of Linux as "just for fun" (Torvalds 2001), this notion of fun usually means getting satisfaction in finding solutions to technical problems, thus fusing "pragmatic" and "playful" motivations.

Playful remix inherent in much of retrocomputing may at first blush seem to be in contradiction to efforts preserving the history of computing. This contradiction, however, dissipates with further analysis. Even in the seemingly extreme case of re-painting an old machine to a new color—a step that cannot be undone—the work is preservative in that it restores the "awesomeness" that the artifact once possessed. A machine that was once a source of joy becomes capable of bringing joy once again. More generally, the remix inherent in retrocomputing allows continuous reintegration of elements of past computing systems to ongoing, living practice. We understand practice as the system of activities comprised of people, ideas, and material objects, tied by shared meanings and joint projects (see also Takhteyev 2012). Computing artifacts are born into such systems and have power and meaning because of their linkages to other elements. Over time, however, some elements of these systems disintegrate or enter into new relationships and abandon the old ones. With the dissolution of relationships comes the fading of tacit knowledge that once made its use possible. Such processes of social decomposition may often be much more damaging to old computing systems than the physical breakdown of the hardware or storage media, and it cannot be stopped by isolating the fragments and shielding them from sunlight or improper temperature and humidity.

The decay of the socio-technical practice in which computing is embedded is partly stopped or even undone in retrocomputing, as ancient fragments are reintegrated into ongoing activities, becoming part of a *contemporary* living practice. Such integration allows for maintenance and even recovery of tacit knowledge (see also Galloway 2011), as well as continuous circulation of tools and resources. Retrocomputing is undergirded by a complex ecology of commercial, hobby, and grey market products and services, and it is this complex, interrelated ecosystem that allowed Mechner's call to be answered so quickly, and his files to be transferred with such seeming ease from 1980s floppies to GitHub, where they will be both preserved and further remixed.

However, appreciating retrocomputing just for the resources it can provide may miss its deeper value. The practice of retrocomputing also provides space for ongoing circulation of meaning and divergent "situated knowledges" (Haraway 1988) and for alternative histories of computing. Consequently, it may not only provide us with new insights into the *how* of digital preservation, but also into the *what* and *why*. We may therefore need to recognize the value of retrocomputing projects on their own terms and look for ways to provide them with support.

ACKNOWLEDGMENTS

This paper is based on work supported by the Social Science and Humanities Research Council of Canada.

REFERENCES

Galloway, P. (2011). Retrocomputing, archival research, and digital heritage preservation: a computer museum and iSchool collaboration. *Library Trends*, *59*(4), 623–636. doi:10.1353/lib.2011.0014

Haraway, D. (1988). Situated knowledges: the science question in feminism and the privilege of partial perspective. *Feminist Studies*, *14*(3), 575–599. doi:10.2307/3178066

Lessig, L. (2008). *Remix*. London: Bloomsbury Publishing PLC. Retrieved from http://www.bloomsburyacademic.com/view/Remix_9781849662505/book-ba-9781849662505.xml

Takhteyev, Y. (2012) *Coding Places: Software Practice in a South American City*. MIT Press.