Exploring Curation-ready Software: Use Cases

Curation Ready Software Working Group progress report, April 14, 2017

Fernando Rios, Bridget Almas, Nicole Contaxis, Paula Jabloner, Heidi Kelly

DOI 10.17605/OSF.IO/8RZ9E

The Curation Ready Software working group endeavors to develop use-case driven guidelines for improving the quality of preserved software given available resources ("curation-readiness"), including expertise, technical infrastructure, and time. Two perspectives will be explored. The first is the archival/museum perspective which is concerned with providing access to culturally important software and deals mainly with (often commercial) software that has already been released. In this perspective, software is largely addressed as a cultural heritage object or artifact, valued because of its historical, cultural, or artistic importance. The second perspective deals with software which is developed or intended for use in academic and research settings in which curation activities can take place at earlier stages in the software lifecycle.

Although we have yet to add a use case that addresses preserving software-based art, we have identified several use cases of varying levels of complexity (see Table 1 and the accompanying descriptions below). These use cases cover a wide range of software, including software at different stages of development and funding, as well as long dormant code bases or obsolete executables being addressed in archival environments.

Use cases

The [A1, A2, B1, B2, E2] use cases in Table 1 are concerned with software created as part of research while [C1, D1, E1] are more closely associated with the notion of software as an artifact. The first two cases [A1, A2] capture the research transparency, reproducibility, and reuse scenarios of software preservation. The goals are to enable other researchers to examine and understand the preserved software and enable its re-execution and reuse. The only difference between [A1] and [A2] is that in [A2], there is still opportunity to carry out curation activities within the software's planning and development stages. This is important since these stages of the software lifecycle are where curation activities have the potential to have the most impact in terms of enabling others to understand, reproduce and reuse the code as well as enabling more effective preservation over longer timeframes. This working group's exploration into what curation activities might be carried out at different stages of the software lifecycle is currently ongoing.

In the next two cases [B1, B2], the goal is the curation of software that is part of a network of distributed services and linked data to support an online digital publication of a scholarly work product. Many digital publications take advantage of distributed services and data to present enhanced and contextualized views of research. The software supporting these publications often have many complex dependencies, on both software and data, which must be fulfilled in order to recreate and sustain complete representations of the original publication. The choices for curation and preservation of this software are not straightforward and the options available will be restricted or expanded depending upon how early in the development lifecycle curation concerns are taken into account. Important factors include the extent to which software development best practices are adhered to throughout development; whether curation and sustainability concerns are considered when introducing software, service and data dependencies; and whether the software driving the digital publication is itself essential for reproducibility of the research and thus of primary interest for curation and preservation, or if it is secondary to the work product itself. Examining the importance of these factors in relation to the other use cases presented is currently ongoing.

The [C1] cases are based on external and internal research inquiries to access software artifacts preserved in a historical museum environment. As part of the larger museum and archival community actively involved in preserving our digital past, we are documenting these current use cases as tests to build our institutional software access policies. Therefore, the use cases dig deep into providing as much access as our resources will allow. As users and software dependencies are myriad, do we provide a working system (allowing running of the application in its native environment), emulation on a virtual machine, or simply a disk image? If we have a running version of historic software, how can we be certain that it renders and operates appropriately? Unexpected issues include: copy protected 1980s commercial software, and not having or being unable to obtain common early operating systems.

The [D1] case illustrates how a software producer may, after the fact, attempt to preserve its intellectual property both as a part of its institutional archives and as a cultural heritage object. In this use case, a federal institution that has created software since the 1960's sought to add software to its Institutional Archives. The [D1] case is an attempt to rectify when software wasn't considered worthy of attention, documentation, and inclusion in record retention policies. A lack of thorough documentation - in part due to the age of the software - required significant backtracking, including archival research and developer interviews. Access to the preserved executable was facilitated through a pre-existing digital library website, where it was made available for download with instructions about using a MS-DOS emulator. While the pilot project has been completed, significant work in creating a manageable workflow for the backlog is necessary.

The [E1, E2] cases relate to the varied roles that a large library plays in archiving software related to university faculty's research, both as end products of that research as well as primary sources fueling new research. The [E1] case is the most mature instance at the university so far: a legacy PC game preservation and emulation project, which aims to provide researchers and students with access to video games donated by a faculty member in order to support new research into gaming, computing history, media studies, and related areas. The [E2] case relates to some of the previously stated cases ([A1, A2, B1, B2]), where the need to preserve software as part of faculty research has been identified. Much of this work is still in early phases, in large part because of the diffuse nature of related responsibilities.

Ongoing work

For each use case, we are in the process of defining curation-readiness and determining the commonalities and differences between the cases. Working group outputs of these stages will include:

- Definitions of curation-readiness from the perspective of each use case
- How curation-readiness may be improved for those cases
- Steps a curator or archivist can take to increase curation-readiness (e.g,. Actions taken, questions asked, suggestions made, criteria met, etc.)

The goal is to articulate a set of characteristics of curation-ready software, as well as activities and responsibilities of various stakeholders in addressing those characteristics, across a variety of different scenarios. Achieving this goal will allow people involved with software preservation to better allocate responsibilities between developers, archivists, and other support staff and prepare the way for mutually agreeable and beneficial workflows for software preservation. All of the working group's output is located at https://osf.io/crfyv/.

If you or your institution are addressing software preservation and would like to participate in this working group, please contact any of the working group members. In particular, we are seeking out use cases that deal with commercial software being preserved at its originating company or software-based-art.

Table 1: Listing of software preservation use cases

Use case	Stakeholders	Goals	Responsibilities	Example(s)
[A1] Software developed via funded research intended for use by other researchers in a similar field of study. Development of the software is complete and the source code is available.	● Funders ● Researchers ● Institutions ● Publishers ● Archives / repositories ● curator/data management specialist	● Funders, publishers, institutions: Enable research transparency, reproducibility, reuse, track research impact. ● Archives/repositories: Keep preservation copies of software. Potentially enable re-execution. ● Researchers: Have others use/extend the software, receive credit, and possibly have others reproduce and verify the work. ● Curator: The focus is on enhancing the suitability of the software to address the preservation objectives. In addition, help ensure that the goals of all parties are met by providing support and guidance to address their needs.	● Funders, institutions, publishers: provide clear software sharing and preservation requirements (including archival location). ● Archives/repositories: meet the preservation needs of funders, institutions, publishers, and researchers (e.g., availability, citability, runnability etc). ● Researchers: provide the code and be willing to work with the curator to make it suitable for the goals. ● Curator: Be knowledgeable in key aspects of the needs of each stakeholder. Be prepared to work with minimal stakeholder interaction.	● Purpose-built code which is used to support claims associated with a publication. Preserving the code is meant to address reproducibility of the results. ● Research where the software itself is the primary research output. Preservation of the code is essentially preservation of the scholarship.
[A2] Same as [A1] except development has **NOT** started or is in the early stages, and the source code is available.	Same as [A1]	Same as [A1]	Same as [A1]. In addition: ● Researcher and Curator: balance software development best practices, preservation, and sharing needs, with available resources. ● Curator: develop a planning mechanism which will address preservation and sharing needs	Same as [A1]
[B1] Software developed via funded research that supports other research outputs (e.g., online digital publications). The software is part a larger ecosystem and makes use of web based service APIs and linked data to provide components of its	Same as [A1]	● Funders, publishers, institutions: same as [A1]. Additionally, ensure that the online publication can be sustained by the institution beyond the period of funding for development. ● Archives/repositories: same as [A1]	● Funders, institutions, publishers: same as [A1]. ● Archives/repositories: same as [A1] ● Researchers/Developers: same as [A1]. Additionally, ensure that the dependencies between the research product and the	A platform that presents an online digital publication of scholarly work product (such as a digital edition).

functionality. It also uses 3rd party open source libraries. The software itself is not the primary research output but the research product is dependent upon the software for its presentation. The source code is available in GitHub. Building the software requires use of a package manager/build tool to retrieve 3rd party dependencies from published open source repositories, as well as ongoing accessibility of those libraries. Development is either not started, is at an early stage, or has at least some continued funding to support preservation activities.		● Researchers/developers: Same as [A1]. Additionally, ensure that the online publication can be rebuilt, deployed and hosted. ● Curator: ensure that the goals of the other stakeholders are met by enabling others to find, use, and cite the software over time.	software are explicitly accounted for. ● Curator: enhance the suitability of the software for addressing the goals by working with the researcher and implementing a suitable digital repository.	
[B2] Same as [B1] except development is finished and no further funding is available.	Same as [A1]	Same as [B1]	Same as [B1]. In addition: ● Curator: interview developers / researchers to identify priorities and mitigation options for software preservation.	Same as [B1]
[C1] Historic software test use cases as the Museum determines the appropriate level of internal and external researcher access. The Museum holds a large collection of software that is currently being preserved and interpreted as part of the newly created Center for Software History. How can this be made scalable over a large collection? It has been pointed out that repositories	● Museum curators, archivists and educators. ● Outside researchers (academic, hobbyist, legal, media, educators) ● Artifact donors ● Funders ● Other libraries, archives & museums	● Museum use: for exhibits, blogs, published papers, educational programs ● Researchers: Make our extensive collection available for viewing/ use by the general public ● Make donors feel comfortable that their digital artifacts will be preserved and made available. ● Funders: Provide good stewardship to Museum funders to guarantee steady stream of funding over time ● Participate in cultural community	● Museum: Provide preservation, description and access to digital materials ● Researchers: Provide enough reference assistance to help researchers understand the 'black box' that historic software might be over time. ● Donors: Provide digital preservation services and access ● Funders: same as above ● Share our methods & findings	● Contacted about access to historic software where the requester needed to see the functionality of the software as it was originally intended. A disk image was not enough. Repurposed our reproduction agreement to use for this purpose. Created disk image. Built research laptop that prevented copying. Installed VirtualBox on laptop & installed Windows '98 to recreate environment. The software was originally written for Windows '95 but could not find a complete version of '95 to install. Were able to boot Real Networks Developer but wouldn't run..Our copy was a developer disk it

don’t provide a translator for foreign language manuscripts. Do we need to ‘translate’ the disk image?				was set to expire in 3 months. We needed to set the calendar on Windows ’95 back to spoof the disk into running. Are uncertain if the look and feel is the same as originally used back in the day. ● Historic software used for exhibitions both online and on-site. Blog posts already published with historic source code available. ● Implement and maintain Museum’s Digital Repository
[D1] Preserving software at institutions with a history of developing software in-house: A federal institution wanted to preserve and make accessible the software they developed from 1964 forward as a part of their institutional archives. Significant work was required to locate existent copies and compile contemporaneous documentation because software and software documentation was not included in record retention policies. Documentation was supplemented with oral histories with developers.	● Institution ● Academic researchers, including humanities researchers ● Younger researchers, including school groups ● Curator/data management specialist	● Institutions: Maintain a historical record of software development at the institution and the use of software in medical practice, research, and education more generally ● All researchers: Understand how to access historic software and contextualize it within the history of technological development, media and cultural histories, and institutional histories ● Curator: Enable others to find, use, and cite the software over time; Provide adequate contextualizing information including documentation of the software (i.e. user manuals) and documentation of the development process (i.e. meeting minutes, etc).	● Funders and institutions: provide clear software sharing and preservation requirements and priorities; provide record retention policies that include software and related documentation ● Developers: provide code and documentation if available and be willing to work with the curator to make it suitable for the goals, including providing oral histories as necessary ● Curator: enhance the suitability of the software for addressing the goals by working with the researcher and implementing a suitable digital repository.	
[E1] Legacy Video Games Preservation: provide access and preservation of legacy video games as collected by researchers and faculty affiliated with the university in order to provide primary sources for researchers in gaming and computer history	● Current faculty and researchers ● Current students ● Libraries faculty and staff ● Outside researchers visiting campus with appropriate technical credentials	● Institution: provide preservation and access of unique research collection ● Institution: adhere to established copyright laws related to collection while providing as much access as possible ● Faculty and researchers: undertake unique research	● Institution: provide ongoing funding, resources, and technical infrastructure to support preservation and access ● Library Technologies Department: provide ongoing migrations of emulated environments as needed ● Digital Preservation unit: create	**Legacy PC Research Collection:** Media Services department (which includes the liaison librarian to the Media School) approached Digital Preservation about a collection of legacy PC video games donated to the Libraries by a new faculty member. Working with the Copyright Librarian to ensure legal adherence, we are working to emulate the collection on a

		using legacy PC games as primary sources ● Curator: facilitate ongoing access in a way that provides an experience as near to the original as possible given the new computing environment	archival information packages for disk images of games and establish necessary technical and preservation metadata requirements ● Media Services: provide contextual information and descriptive metadata; provide information about user community	set of four computers in the Media Services area as well as creating preservation AIPs and depositing them into our long-term storage. We've been running into DRM issues with this content and can't preserve the source code. We're using different operating systems for the emulation - DOSBox for DOS games, as well as various versions of Windows installed within virtual machines to run the other games.
[E2] Software preservation to provide ongoing access to data: provide preservation of software associated with unique datasets and 3D objects so that future researchers can fully access and utilize them	● Future faculty and researchers ● Libraries faculty and staff ● Future students ● Outside researchers	● Institution: provide ongoing access to the unique data being created locally ● Institution: adhere to established copyright laws ● Libraries faculty and staff: provide ongoing curation for software and maintain relationships to corresponding datasets	● Institution: provide ongoing funding, resources, and technical infrastructure to support preservation and access ● Digital Preservation unit: establish workflows and necessary technical and preservation metadata requirements ● Metadata unit: establish metadata frameworks to support curation ● Digital Collections Services: provide repository environment for software and establish mechanisms for migration ● Research Data Management/SchoIComm: work with faculty and researchers to obtain software and contextual information	● **Uffizi**: The Uffizi collection in Italy is currently partnering to preserve 3D scans of sculptures and place AIPs into the Digital Preservation Network (DPN). This work will hopefully eventually integrate software preservation in order to ensure long-term access to 3D objects. ● **Imago**: The Imago repository project (developed in Hydra) is being built in collaboration with the Indiana Center for Biological Research Collections (CBRC) and will eventually include 3D scans of research objects. This project will also hopefully include a software preservation component. ● **Research Data Management**: This use case is the least clear at the moment, but we expect that researcher-created software will be deposited along with datasets in future. This will hopefully present the easiest way to access and preserve source code, but might also be the messiest.