ARTCE: the Archival Repository of Tools for Computing Education

Jeremiah Blanchard and John R. Hott

1 Overview

In Computing Education, researchers and educators often build a tool to support either their educational or research goals; however, archival-quality links are not generally provided in publications leading to missing and abandoned software [1]. Prior work has also shown that not only are these tools difficult to find, but many links in computer science and computing education literature are "dead" or redirect to generic landing pages due to deletion or page movement [1]. In contrast, the longevity of archival objects and facilitating their discovery has been the topic of work and study by librarians and archivists for hundreds of years [4, 3]. They have created multiple systems to address these challenges for various digital (and physical) artifacts. The Digital Object Identifier (DOI) was developed to provide archival-quality links to digital literary objects, such as papers, and is typically applied to modern peer-reviewed journals and other scientific literature [5, 2]. Similarly, the Archival Resource Key was created by archivists to provide a persistent identifier for objects [3] and applied broadly to the descriptions of persons, corporate bodies, and families [4]. While there has been some work in archiving software such as the Internet Archive's repository[1], and extensive use of software repositories such as GitHub and SourceForge[2], no such system has been developed specifically for the archival and discovery of software for Computing Education [1].

2 Functionality and Structure

To address the lack of a clear, archival-driven system for software projects presented and utilized in Computing Education research, we propose ARTCE, the Archival Repository of Tools for Computing Education. ARTCE is a static archival identifier system for software projects, which seeks to provide structure and means to archive and discover digital software relating to education and the research of education. We envision ARTCE providing the following features:

- Static identifier for software projects[3]
- Database of project snapshot, description, license, and deployment information
- Relevant branch, fork, and/or other variant data for project versions
- Tagging system for keyword assignment to facilitate discovery based on researcher and educator needs
- Optional link(s) to repositories and other relevant information
- Browseable source code (when applicable)

ARTCE's tagging system will include identification of programming languages, course topic and level targets, and other relevant instructional information. It will also allow inclusion of tags existing in the ACM's Computing Classification System (CCS)[4].

Initially, we envision ARTCE providing a mechanism for including archival-quality links into publications in Computing Education venues. However, we hope to expand the system with editorial and review support to archive peer-reviewed and high-quality tools used throughout the discipline, similar to approach taken by ACM EngageCSEdu[5] and Nifty Assignments[6] for instructional materials.

ARTCE's goal is to ensure long-term viability, via snapshot and metadata archival, of software references in computing literature and other media in a discoverable way. Just as the DOI system ensures a static, long-term link for literary publications, ARTCE will provide long-term static references for software. By providing a thorough set of tagging options, ARTCE will support educators, researchers, and developers in filtering and searching through software projects based on specific instructional needs.

[1] `https://archive.org`
[2] `https://www.github.com`, `https://sourceforge.net/`
[3] Eventual compatibility with DOI is planned.
[4] `https://dl.acm.org/ccs`
[5] `https://engage-csedu.org`
[6] `http://nifty.stanford.edu/`

3 Alignment with SPLICE Goals

ARTCE will help to avoid "re-inventing the wheel" of software for Computer Science courses; a searchable repository improves access to code, not just immediately, but in the long term via static archival links. By ensuring instructors, researchers, and developers can access software projects in the future, ARTCE will increase reuse of those projects rather than creation of redundant new systems, thereby reducing unnecessary duplication of effort in the community.

4 Value Proposition

By providing computing educators and researchers with a straightforward, easy-to-access collection of prior software and and documentation regarding its use and deployment, ARTCE will free practitioner time and effort for more innovative and constructive efforts in computing education. ARTCE will also provide the community archival-quality links to include in publications and other material going forward.

5 Project Team

- Dr. Jeremiah Blanchard is Associate Instructional Professor in the Department of Engineering Education at the University of Florida, where he also serves as the Director of the Computer Engineering program. His research focuses on how students learning programming languages.
- John R. Hott is Assistant Professor of Computer Science at the University of Virginia on the Academic General Faculty Teaching Track. His research includes student use of collaboration policies, engagement on course forums, academic integrity, and tools to support Computing Education. Before joining the University of Virginia faculty, Dr. Hott served as Technical Lead on the open source Social Networks and Archival Context project.

6 Budget

Funding received will be used primarily to fund for-hire development work by undergraduate and graduate students, plus minimum PI effort, fringe benefits, and fees as described below.

$\$5,000$	:	Student Development Hours at University of Florida
$\$5,000$	:	Minimum PI effort, plus associated fringe benefits; Undergraduate Student hours; Departmental Fees; F&A at University of Virginia

References

[1] Jeremiah Blanchard, John R. Hott, Vincent Berry, Rebecca Carroll, Bob Edmison, Richard Glassey, Oscar Karnalim, Brian Plancher, and Seán Russell. Stop reinventing the wheel! promoting community software in computing education. In *Proceedings of the 2022 Working Group Reports on Innovation and Technology in Computer Science Education*, ITiCSE-WGR '22, page 261–292, New York, NY, USA, 2022. Association for Computing Machinery.

[2] Rajesh Chandrakar. Digital object identifier system: an overview. *The Electronic Library*, 24(4):445–452, 2006.

[3] John Kunze and Richard Rodgers. The ark identifier scheme. 2008.

[4] Ray R Larson and Krishna Janakiraman. Connecting archival collections: the social networks and archival context project. In *International Conference on Theory and Practice of Digital Libraries*, pages 3–14. Springer, 2011.

[5] Norman Paskin. Digital object identifier (doi®) system. *Encyclopedia of library and information sciences*, 3:1586–1592, 2010.