Active Curation of Artifacts and Experiments is Changing the Way Digital Libraries will Operate

Bruce R. Childers*, Jack W. Davidson†, Wayne Graves‡, Bernard Rous‡ and David Wilkinson*

*Department of Computer Science
University of Pittsburgh, Pittsburgh, Pennsylvania 15260 USA
Email: {childers@cs.pitt.edu, dwilk@cs.pitt.edu}

†Department of Computer Science
University of Virginia, Charlottesville, Virginia 22904 USA
Email: jwd@virginia.edu

‡Association for Computing Machinery
New York, New York 10121 USA
Email: {graves@acm.org, rous@acm.org}

***Abstract*—A new type of "active curation system" is becoming prevalent in computer science that provides executable access to artifacts and experiments behind published results and enables their reuse. These systems allow changing and repeating experiments to understand how an innovation behaves in conditions beyond the ones described in a paper. Active curation systems also enable accountability and accelerate research progress by giving access to complete experimental details. As these systems take hold, it is important to understand their capabilities and how digital libraries (DLs) should be integrated with them. In this position paper, we describe a study underway to explore how best to do this integration. The study uses case studies to understand how the systems and one particular DL, the Association for Computing Machinery's Digital Library, should work together to package, deliver and interact with artifacts and experiments as digital objects that can be executed. This study is a step toward developing the approaches required for production integration of active curation systems and digital libraries.**

I. Introduction

Innovation in computer science (CS) typically relies heavily on experimentation with software tools, data sets, parameters, and a myriad of other information. These *artifacts* are used to implement prototypes, empirically evaluate new ideas and analyze implications. The artifacts are used in *experiments* to produce *results*. An experiment incorporates all data, workflows, setups and tools for an empirical study. A result is the outcome of the study. Results are typically analyzed and summarized for dissemination of a new idea to the broader community through peer-reviewed articles.

Within this context of experimentally-driven CS, a far-reaching trend has started to emerge: there is recognition of the importance and benefit of gaining access to the artifacts, experiments and results underlying published articles for research accountability and credibility (important aspects of the scientific method). In the United States, federal mandates are driving the need for this access, such as data management plans spearheaded by the National Science Foundation (NSF) and now adopted by the Department of Energy (DOE). Likewise, the National Institutes of Health (NIH) has put in place data access mandates. Similar mandates and efforts have appeared in the European Union, such as the Horizon 2020 Open Research Data Pilot, and elsewhere. Simultaneously, a grass-roots conversation is advocating and incentivizing improved experimental quality to increase research productivity through access to full information and tools behind publications. For example, Artifact Evaluation (AE) rewards production of high-quality artifacts and experiments [1], [2]. In AE, authors provide the materials used to generate experimental results for peer review separate from paper review. Several efforts are underway for curation of digital artifacts and experiments to similar ends. This trend will gain even more momentum as CS enters the age of open access and accountability.

Specifically, a new type of system, which we call a *active curation system*, is emerging to curate digital objects that can be executed (i.e., run on computational resources) to facilitate all steps of research, including formation of new ideas, implementation, experimentation, dissemination, and archiving. These systems provide direct, executable access to experiments, which may be shared, used in AE, archived and/or accessed from a published article. They go beyond code, result and/or experiment, repositories, which store and share fixed/static information (like `github` or `Bitbucket`), to allow modifying and running experiments.

Active curation systems (e.g., DataMill [3], OCCAM [4], and EUDAT [5]) offer a "try before you buy" approach to online sharing of interactive and modifiable experiments as digital objects; these systems allow reviewers and readers to try different parameters, alternative data sets, etc., to create and run new experiments. The systems provide the computational capabilities to support executing the original experiments from a paper, changing the experiments, and running entirely new ones. The system may be hosted in the cloud and structured as centralized, distributed or federated structures that encompass underlying repositories (ideally built with existing efforts for sharing research data, like MyExperiment [6], Research Object [7], Figshare [8], Zenodo [9], DataVerse [10], and Open Science Framework [11]) and computational resources. The computational resources may be marshalled on-demand from cloud providers (e.g., Microsoft Azure, Amazon AWS, and Google Cloud Platform), community systems (e.g., XSEDE [12], CloudLab [13], Chameleon [14], and PSC Bridges [15]), or institutional (private) resources. Controls may be provided to restrict what can be done with an experiment, such as limiting an experiment's scale to be runnable in a short time interval with minimal resources for a "quick look", rather than full experimental runs. Active curation systems are a compelling step in the evolution of research repositories and community computational systems to tie both together and then link them to the scholarly record.

Regardless of the specific capabilities and implementation choices, active curation systems help users understand how innovations behave under conditions beyond the ones described in the original papers. By giving executable access to complete experimental details of an innovation and allowing interactive modification, active curation systems also enable accountability and accelerate research. These systems allow other researchers to extend the original experiments and to use the original artifacts for comparison. Furthermore, they enable a new style of scientific dissemination, where the *product*, i.e., the artifacts and experiments themselves, take center stage, in addition to, or possibly even replacing, the peer-reviewed article.

Given the diversity of CS research, it is unlikely that any one active curation system can serve all needs; every community has different requirements, ranging from relatively small algorithmic descriptions (say in R) for data analytics, to complex, multi-stage workflows in computer systems research, to instrumented systems in cyber-physical and embedded computing, to large-scale parallel execution in high-performance computing (HPC), and everything and anything in between! DLs must be prepared for this change to handle diverse digital entities and the ways in which they are identified, described, reviewed, delivered and manipulated. The researchers using the platforms and developing the active curation systems must also be informed about the requirements, standards and needs of the DLs. Interfaces and metadata will be needed for interoperability between the DL and the curation systems.

II. Integrating Active Curation Systems

To help publishers, such as the Association for Computing Machinery, the Institute of Electrical and Electronics Engineers (IEEE), and the Society for Industrial and Applied Mathematics (SIAM), prepare for active curation systems and the objects they will curate, we are carrying out a study to understand the landscape of community and commercial efforts. The study is designed to be objective, inclusive and systematic to consider a range of systems relevant to different communities. Rooted in specific use cases, the study will assist publishers in developing the approaches and procedures needed for integration. The case studies will trial specific active curation systems that are integrated with the Association for Computing Machinery's (ACM) digital library, a preeminent discovery service and archive of CS publications. The studies may also be used as part of AE at selected conferences to learn about new review approaches for evaluating software artifacts and experimental outcomes. The information gathered will influence artifact review, the DL and the curation platforms themselves to better support community evaluation, dissemination of results and interoperability of digital libraries with external active curation systems. The trials will help test and refine approaches to determine what does and does not work, and demonstrate active curation for feedback from the CS community.

The study, which is already underway and has made initial progress (see Section III), has several steps:

1) Identify relevant active curation systems to catalog capabilities and targeted audiences. The identification of these systems considers AE and archiving experiments for access through the DL.
2) Develop "use cases" to test/evaluate the active curation systems for a spectrum of research communities, ranging from simple use cases to more complex ones (e.g., workflows involving multiple tools and data sets).
3) Evaluate exemplar active curation systems in the context of the use cases to learn requirements and capabilities, with an eye toward the impact on both AE and the DL.
4) From the evaluation, identify the primary capabilities and issues that the DL, developers and authors need to consider to incorporate and use active curation systems with digital libraries.
5) Develop guidelines for technical best practices for active curation systems, preparing software artifacts and experiments for curation as digital objects that can be executed, integrating the DL with the systems, and achieving service guarantees/standards.
6) Determine prototype interfaces to "plug-in" specific active curation systems into the DL, e.g., readers can access papers in the DL and click on experiments in papers and/or associated with DL metadata to access original experiments, which can then be modified and re-evaluated.
7) Using the prototype interfaces, integrate the exemplar systems with the DL to refine and improve interfaces, learn what does and does not work, and demonstrate the benefits to the broader community.
8) Seek feedback from the community to refine initial approaches and interfaces.

Although the study's aim is technically oriented towards integrating DLs and active curation systems, policy and procedural issues are inherently being considered to a limited extent. Likewise, we recognize the importance of facilitating AE by further integration between *submission platforms*, DLs and active curation systems. Although these issues are outside this study's scope, they are important to consider in a future study.

A. Study Outcomes

Several case studies will be the primary outcome from the study. The case studies will integrate selected exemplar active curation systems with the DL in trial runs to access specific digital objects that can be executed on specific computational resources. The case studies will show how the exemplar active curation systems can (1) package and deliver artifacts and experiments as digital objects that can be modified and executed in an interactive fashion, and (2) provide access and interact with artifacts and experiments associated with papers. The case studies will also give insight into how the end-to-end process should be structured and what technical capabilities are required.

While these outcomes are useful to the ACM DL, more broadly, upon completion of the study, the outcomes will provide insight to software developers building active curation systems and other publishers and providers of digital library services (e.g., IEEE, SIAM, etc.) that wish to also integrate the systems. The completed study will also assist AE evaluators and authors in deciding how to package their artifacts/experiments and what active curation systems are appropriate to their particular situation. The study will give understanding to the developers and maintainers of the ACM DL and the curation systems about the technical capabilities required to bring together AE, curating digital objects that can be executed, and archiving that involves interoperability of several systems. We note that many scientific communities are facing the same challenges and opportunities as computer science with software products and experimental results. The outcomes from this study will be both informed by work in these other communities, as well as providing insight to those communities from our experience in experimental computer science. Relevant outcomes will be made publicly available for open use/modification by the broader community, including the specific integrations developed for each use case.

III. Initial Work on the Study

In this section, we give an overview of our initial work in performing the study and the preliminary outcomes from that work. We have examined the landscape of available active curation systems to characterize their capabilities. With the assistance of an Advisory Group appointed by ACM, we determined common use cases where these systems might be used by authors and readers to interact, modify or build upon published results from a scholarly article in computer science.

Below, we describe our use cases, why they were selected and how we plan to implement them. The situations covered by the use cases are not exhaustive; there are many other ways in which authors and readers might want to deploy and interact with software artifacts and experiments beyond these particular case studies. They are, however, designed to cover common ways in which readers of proceedings papers from ACM Special Interest Group (SIG) conferences might access and use artifacts and experimental results. This position paper is intended to solicit feedback to improve and refine the use cases and their implementation.

Case Study 1: Repeating Comparisons with Varying Datasets

In experimental CS, a typical evaluation compares different algorithms (in the broadest sense, i.e., an algorithm might be a "system", a "software implementation", etc.). The comparison may consider different variants of an algorithm, or be a comparison of a new technique versus a previous one. Many metrics might be considered, such as performance, energy, reliability, memory size, storage capacity, and topic-specific ones. The purpose of the evaluation is to understand the behavior, benefits and pitfalls of the different choices. The evaluation may use several situations, such as different fault injection scenarios for analysis of resilience. Authors usually consider situations that achieve broad coverage and address specific advantages/disadvantages of the algorithms.

Given the possibly *huge* space of comparison, it is natural that authors cannot cover every possibility, nor anticipate every situation, including ones that are not yet known. After all, research comparisons are usually about pointing to promising directions (or avoiding false paths!), rather than specifics or absolutes. Ideally, the algorithm implementation, with all experimental situations, should be made available for other researchers to study more carefully and build upon in new ways. In this case study, we consider this scenario: a reader of a published article wants to repeat comparisons under the original published situations, as well as new ones for the reader's own work. We imagine that a reader comes to the ACM DL, accesses a paper, and then decides to repeat comparisons in the paper. Further, the reader wants to do the comparisons with new data sets.

There are many ways in which this capability might be achieved. For this first case study, we consider what may be the simplest approach: the reader downloads packaged artifacts from the DL and re-runs comparisons on his/her own computer. We make a twist to this simple situation to further imagine that the reader works in high-performance computing and has access to an HPC cluster to run a parallel algorithm. This case study then covers 1) replicating algorithm comparisons; 2) packaging, accessing and distributing artifacts to local computational resources; 3) using unique hardware resources (a cluster) to run packaged artifacts; and 4) repeating experiments with different data sets.

To implement this case study, we selected a candidate application and active curation system. The application is a parallel DNA assembly algorithm that processes very large graph structures, which is similar to many big data applications [16]. It is an interesting application because the associated article describes a classic A vs. B algorithmic comparison. It also operates on big data sets that exceed one terabyte, introducing a challenge in setting up and repeating an experiment. The software implementation of the algorithm is available and uses the widely-adopted MPI parallel programming environment. For the active curation system, we selected Collective Knowledge (CK) [17], [18]. CK can "wrap" an experiment into a packaged archive that, when extracted, automatically re-runs experiments, pulling down data sets as required from independent data repositories. The data sets required for the DNA algorithm are contained in an open-access repository, where CK can grab them. CK also easily facilitates using data sets other than the ones used in the original evaluation. It is open source software with strong support from its development team. Lastly, because the artifact requires particular hardware resources (a cluster with sufficient memory and core count) and software requirements (MPI library), the ACM DL is being enhanced to display the detailed re-run requirements.

Case Study 2: Sharing and Modifying Experiments

This second case study examines another common scenario: modifying an experimental configuration to understand how an innovation behaves in a situation different from what was evaluated in the original paper. Access will be given to the original software used for evaluation, along with all experiment metadata (e.g., configurations, data sets, benchmarks, etc.). A reader should then be able to modify and run the experiments to test new conditions.

This case study looks at the other end of the spectrum of case study 1. It considers an active curation system that is seamlessly and tightly integrated into the ACM DL. The DL will extract and serve digital entities that can be manipulated from a paper's display web page. The extracted digital entities might highlight critical results from the paper, such as a summary graph of performance for some new algorithm. A reader could then interact with that graph, changing initial conditions and running new experiments, which would then become part of the archive associated with that paper. The provenance of experiments would be extracted from the active curation system and kept with new experimental runs through the DL. While the DL would provide this "overview access", to dig deeper, the reader would then access the artifacts and experiments directly through the active curation system. In essence, this case study considers an interactive paper displayed and manipulated via the DL. Detailed access would be supported by a hand-off to the active curation system itself. Because the active curation system and the DL will dynamically interact, the curation system needs to be online (in the cloud), rather than locally hosted as in case study 1.

We identified a candidate paper [19] from ACM SIGSAC (security and access control) for this case study. The paper describes probabilistic actor modeling to analyze the expressiveness and overhead costs associated with various access controls when used in specific applications. A simulator is used that implements the probabilistic model. Because the modeling has many design choices and the simulator generates a huge number of results, it is a good candidate to illustrate the benefits of interactive access. It is also a good candidate due to the size of the design space; the paper cannot consider nor anticipate every parameter choice that a reader may wish to explore (which varies with the application of the access control). For this case study, we need a active curation system that supports interactivity, workflows, provenance, cloud hosting, and results visualization. Our candidate is OCCAM [4], which is a prototype system to support the full life-cycle of research from initial innovation, to paper reviewing, to artifact evaluation, to preservation and subsequent access and modification of experiments. The ACM DL will be enhanced to interact with OCCAM to extract highlight results and allow changes to parameters, which are passed back to OCCAM for new runs of the simulator. Results would then be delivered from OCCAM to the DL as newly added digital entities (e.g., graphs that can be manipulated) accessible from the paper's display page but clearly distinguished from the author's own work via their own provenance display.

Case Study 3: Deriving a New Artifact from an Existing One

In this third case study, we consider the situation where an existing software artifact is modified to derive a new software artifact. When access is given to an innovation, the software implementation can be changed to build on it with new capabilities. Alternatively, a new idea might be compared against a previous one, tackling the same issue. The original implementation can be changed with the new idea, allowing for fair comparison in the same software framework. The new implementation may then be deployed along with the original one through an active curation system. The integration of case study 3 will be similar to case study 2, except support will now be required to allow modifying the software artifact itself, tracking and displaying provenance of source changes, and adding the new artifact to the digital library.

This case study will use a prototype commercial system, which can host a software artifact and associated experiments from a paper; facilitate and track modifications to the source code and experiments; and store/retrieve the new changes. The system also permits comparing results before and after the source code change, delivering digital entities to the DL for interactive display with the paper.

Our candidate software artifact for this case study comes from the computer architecture community (SIGMICRO and SIGARCH). The artifact is a simulator of a computer memory sub-system. The simulator will be modified to support a technique that permutes memory addresses to reduce contention in the memory for improved performance. The technique is based on a classic XOR-address remapping method [20], which requires only a small number of source code changes to implement. The modification will be made to the simulator and new experiments will be run, both of which will be recorded and made available by the active curation system. For the simulator, we plan to use DRAMsim2 [21], which can faithfully model a range of memory architectures, such as DDR4 and DDR3 memory. Using the simulator, a new memory sub-system will be created that has XOR-address mapping. The simulator will be run with traces from several benchmark programs to generate results that compare memory performance without and with XOR remapping. Using the commercial system, graphs will be generated and extracted that can be incorporated into the ACM DL for interactivity and further source and experiment modification and experimental runs.

IV. Community Engagement and Assessment

This effort is engaging the computer science community through the ACM's Publications Board and Special Interest Group Governing Board. The effort is being conducted in concert with an ACM Task Force on Data, Software and Reproducibility in Publication, convened by the Publications Board. This task force aims to investigate and understand the needs for new services and capabilities in the ACM DL to support the changing landscape of review and scholarly dissemination practices and methods. Through this task force, community feedback and guidance is being incorporated in the study. The ACM SIG Governing Board also established a second task force on issues about reproducibility in published results, which has also influenced the study. Feedback from the broader community will be also sought through the SIG Governing Board and the chairs of SIGs. We expect the study's outcomes will be reviewed by both the Publications Board and the SIG Governing Board.

Importantly, we will make the pilot integrations for the case studies available to members of several SIGs for community feedback in "trial runs", where SIG members are invited to try out the case studies. Through these trials, we will solicit feedback with a questionnaire. The questionnaire will be designed to gather information useful for assessing how much effort authors expect and are willing to expend to encapsulate artifacts and experiments as digital objects for active curation, the degree to which the authors view having executable, interactive access will accelerate research agendas and improve accountability, and the benefits that authors and readers expect from the access. We will also solicit input of survey participants on their views about what kinds of capabilities are needed from new services in the DL that incorporate digital objects that can be executed.

Finally, we will make videos of demonstrations of the case studies. The videos will show the perspective of both authors and readers of papers carrying out actions in the case studies (e.g., packaging materials into digital objects that can be executed and interacting with the packaged digital objects). We will use the videos to demonstrate the case studies for further feedback from the ACM community.

V. Summary

Traditional digital libraries and their publications will need to change to accommodate new methods of identifying, describing and disseminating research outcomes, including the software, data sets, benchmarks, configurations and other information that are used in experimental evaluation, right along with user-generated content. These emerging methods represent a type of "active curation system" that allows interactively modifying, deriving and running new experiments beyond the ones described by a paper. In this paper, we present a study (currently underway) with the ACM Digital Library to understand how to integrate active curation systems and to demonstrate their benefits to the experimental computer science research community. The study's outcomes will provide valuable insight to authors, developers of active curation systems, and publishers for preparing articles of the future.

VI. Acknowledgments

The ACM Advisory Group overseeing this effort includes Jack Davidson, University of Virginia (ex officio); Juliana Freire, New York University; Sheila Morrissey, Ithaka S+R; Bernard Rous, ACM; Michela Taufer, University of Delaware; and Alex Wade, Microsoft.

The ACM Task Force on Data, Software and Reproducibility in Publication is led by Michael Lesk of Rutgers and Alex Wade of Microsoft Research. The ACM SIG Governing Board Replication Taskforce is led by Simon Harper.

The application for case study 1 was suggested by Michela Taufer. Stephen Harrell, Paul Peltz and John Johnson also gave helpful insight into this application for the case study. Grigori Fursin provided software improvements to Collective Knowledge that are useful for case study 1. Debashis Gangley, Adam Lee, Daniel Mosse', and Bill Garrison contributed to the integration of the access control simulator with OCCAM, which will be used in case study 2.

This project is funded by the Alfred P. Sloan Foundation and conducted in conjunction with the Association for Computing Machinery. Some material in this document is based in part upon work supported by the National Science Foundation under grant numbers ACI-1535232 and CNS-1305220. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the NSF.

VII. Disclosure

Bruce Childers and David Wilkinson are the principal investigator and developer, respectively, of OCCAM. Because OCCAM is part of the study, an Advisory Group appointed by ACM is overseeing the effort to manage potential conflict of interest.

References

[1] "Artifact Evaluation," http://www.artifact-eval.org.

[2] S. Krishnamurthi and J. Vitek, "The real software crisis: Repeatability as a core value," *Commun. ACM*, vol. 58, no. 3, pp. 34–36, Feb. 2015.

[3] A. Oliveira, J.-C. Petkovich, T. Reidemeister, and S. Fischmeister, "Datamill: Rigorous performance evaluation made easy," in *Proc. of the 4th ACM/SPEC International Conference on Performance Engineering (ICPE)*, Prague, Czech Republic, April 2013, pp. 137–149.

[4] "Open Curation for Computer Architecture Modeling," http://occam.cs.pitt.edu.

[5] "EUDAT: the collaborative Pan-European infrastructure providing research data services, training and consultancy for Researchers, Research Communities and Research Infrastructures and Data Centers," https://www.eudat.eu/.

[6] D. D. Roure, C. Goble, and R. Stevens, "The design and realisation of the virtual research environment for social sharing of workflows," *Future Generation Computer Systems*, vol. 25, no. 5, pp. 561 – 567, 2009. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167739X08000939

[7] K. Belhajjame, J. Zhao, D. Garijo, M. Gamble, K. Hettne, R. Palma, E. Mina, O. Corcho, J. M. Gmez-Prez, S. Bechhofer, G. Klyne, and C. Goble, "Using a suite of ontologies for preserving workflow-centric research objects," *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 32, pp. 16 – 42, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1570826815000049

[8] "figshare - Credit for all your research," https://figshare.com.

[9] "Zenodo. Research. Shared." http://zenodo.org.

[10] M. Crosas, "The dataverse network: An open-source application for sharing, discovering and preserving data," *D-Lib Magazine*, vol. Volume 17, 2011. [Online]. Available: http://www.dlib.org/dlib/january11/crosas/01crosas.html

[11] "Open Science Framework – A scholarly commons to connect the entire research cycle," http://osf.org.

[12] J. Towns, T. Cockerill, M. Dahan, I. Foster, K. Gaither, A. Grimshaw, V. Hazlewood, S. Lathrop, D. Lifka, G. D. Peterson, R. Roskies, J. R. Scott, and N. Wilkens-Diehr, "Xsede: Accelerating scientific discovery," *Computing in Science and Engineering*, vol. 16, no. 5, pp. 62–74, 2014.

[13] R. Ricci, E. Eide, and The CloudLab Team, "Introducing CloudLab: Scientific infrastructure for advancing cloud architectures and applications," *USENIX ;login:*, vol. 39, no. 6, Dec. 2014. [Online]. Available: https://www.usenix.org/publications/login/dec14/ricci

[14] "Chameleon: A configurable experimental environment for large-scale cloud research," https://www.chameleoncloud.org/.

[15] N. Nystrom, "Building Bridges to the Future," *insideHPC*, June 2016. [Online]. Available: http://insidehpc.com/2016/06/building-bridges-to-the-future/

[16] P. Flick, C. Jain, T. Pan, and S. Aluru, "A parallel connectivity algorithm for De Bruijn graphs in metagenomic applications," in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, ser. SC '15. New York, NY, USA: ACM, 2015, pp. 15:1–15:11.

[17] "Collective Knowledge," https://github.com/ctuning/ck.

[18] G. Fursin, R. Miceli, A. Lokhmotov, M. Gerndt, M. Baboulin, A. D. Malon, Z. Chamski, D. Novillo, and D. D. Vento, "Collective Mind: Towards Practical and Collaborative Auto-Tuning," *Scientific Programming*, vol. 22, no. 4, pp. 309–329, 2014.

[19] W. C. Garrison, A. J. Lee, and T. L. Hinrichs, "An actor-based, application-aware access control evaluation framework," in *Proceedings of the 19th ACM Symposium on Access Control Models and Technologies*, ser. SACMAT '14. New York, NY, USA: ACM, 2014, pp. 199–210.

[20] Z. Zhang, Z. Zhu, and X. Zhang, "A permutation-based page interleaving scheme to reduce row-buffer conflicts and exploit data locality," in *Proceedings of the 33rd Annual ACM/IEEE International Symposium on Microarchitecture*, ser. MICRO 33. New York, NY, USA: ACM, 2000, pp. 32–41.

[21] P. Rosenfeld, E. Cooper-Balis, and B. Jacob, "DRAMSim2: A cycle accurate memory system simulator," *Computer Architecture Letters*, vol. 10, no. 1, pp. 16–19, 2011.