Augmenting the User Experience in Open OnDemand

Aday Bujeda
The Agile Monkeys
Canary Islands, Spain

Sarah Duncan[†]
Harvard University
Cambridge, MA, USA

William Horka[†]
Harvard University
Cambridge, MA, USA

Emily Lawrence[†]
Harvard University
Cambridge, MA, USA

Michael Reekie[†]
Harvard University
Cambridge, MA, USA

Evan Sarmiento[†]
Harvard University
Cambridge, MA, USA

Tania Schlatter[†]
Harvard University
Cambridge, MA, USA

Leonard Wisniewski[*†]
lwisniewski@iq.harvard.edu
Harvard University
Cambridge, MA, USA

Alan Chalker[*]
alanc@osc.edu
Ohio Supercomputer Center
Columbus, OH, USA

Jeffrey Ohrstrom
Ohio Supercomputer Center
Columbus, OH, USA

ABSTRACT

The *user experience* has always been a hallmark of the Open OnDemand open-source software. User experience research has identified a number of features that would augment it further. Quick-launch buttons allow applications to start with a single click. Custom profiles offer the researcher a choice among dashboards with unique branding and sets of applications. Other single buttons enable restarting a previous job and direct submission of a support request ticket. These features have been integrated and released in Open OnDemand 3.0.

CCS CONCEPTS

• **Human-centered computing → Human computer interaction (HCI)**.

KEYWORDS

user experience (UX), dashboard, profile

ACM Reference Format:
Aday Bujeda, Sarah Duncan, William Horka, Emily Lawrence, Michael Reekie, Evan Sarmiento, Tania Schlatter, Leonard Wisniewski, Alan Chalker, and Jeffrey Ohrstrom. 2023. Augmenting the User Experience in Open OnDemand. In *Practice and Experience in Advanced Research Computing (PEARC '23), July 23–27, 2023, Portland, OR, USA.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3569951.3597546

[*]Corresponding author
[†]The Institute for Quantitative Social Science

PEARC '23, July 23–27, 2023, Portland, OR, USA

ACM ISBN 978-1-4503-9985-2/23/07…$15.00
https://doi.org/10.1145/3569951.3597546

1 INTRODUCTION

The set of research computing users is expanding to include a broader set of disciplines, each with their own unique user requirements. User experience research and design has taken on a vital role in the life cycle of the software development process. Open OnDemand is one shining example of software that has enriched the user experience for simplifying and visualizing the navigation of HPC resources, especially for those with little need or desire to deeply understand the underlying hardware and software components. Applying user experience research to research computing can help prioritize areas for software development.

In this paper, we share our experiences with applying user experience to research computing. We identify user requirements by applying user experience research and translate those requirements into new or expanded features in Open OnDemand. Furthermore, we provide descriptions of those new features in Open OnDemand 3.0. Moreover, we have developed features that can be shared with the open-source community, but also features that can offer opportunities for expansion in future development.

The rest of this paper is organized as follows. Section 2 walks through the historical evolution of user requirements for HPC and more specifically how those requirements expanded with social scientists entering the HPC arena. Section 2 also goes into the practical experiences of our team at Harvard IQSS deciding how to focus our efforts to address user requirements. Section 3 goes a little bit into our emphasis on the user experience in our development process and how we arrived at a set of features to implement in Open OnDemand. Section 4 goes into some of the details about the new features that we implemented. Section 5 wraps up with a short conclusion.

2 PREVIOUS WORK

High-performance computing has a tradition of aspiring to highest-performance computing. The complex and scalable nature of the computation and data required the user to think about how to map

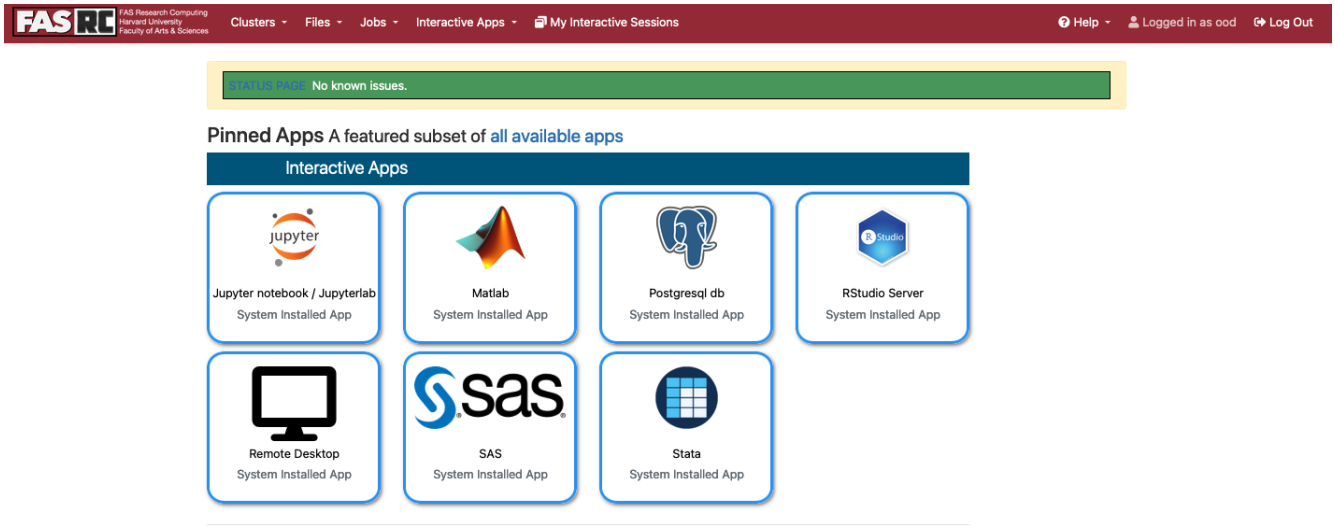

Figure 1: Example Open OnDemand dashboard with quick-launch application buttons.

their problem onto parallel compute and data resources. To achieve this, developers of scientific applications attempted to extract the innate performance of the latest hardware and software innovations by directly programming to exploit those features. This implied that scientists in one domain needed to translate their science directly to the resources by acquiring detailed knowledge about those resources.

Tools were developed to ease the burden of knowledge acquisition by the scientist. Parallel compilers allowed automatic mappings of well-known optimizations to be applied across parallel resources with little or no intervention by the scientist. Communication software helped the scientist distribute and share data among parallel processes working together in a coordinated fashion to solve a single higher-level problem. Parallel debuggers and performance analyzers helped to optimize the underlying activity among the resources.

2.1 Sid: Research computing for the social sciences

At the Institute of Quantitative Social Science (IQSS) at Harvard University, we aim to "solve society's greatest challenges through bold and collaborative social science." To do this, in addition to programs and services, we accommodate access to computational and data resources by offering infrastructure in conjunction with FAS Research Computing at Harvard and developing software to ease the hurdles to using those resources in a manner that accelerates the research process.

High-performance computing for social scientists often means trying to solve a problem with resources that exceed what is available on their basic laptop or desktop computer. Whereas a physicist or chemist might see hardware and software optimizations as another form of a physics or chemistry problem, respectively, a social scientist generally would rather just try to use the same methods for solving a problem on their laptop or desktop, but at a much larger scale. Of course, this does not mean that no social scientist digs into the complexity of the hardware and software and that no physical scientist is ever frustrated by needing to do the same. The long-time great challenge of high-performance computing has been to seamlessly bridge the gap between the paradigms needed to solve problems on complex hardware and software and the paradigms that are needed to solve problems in other domains.

To this end, our team at IQSS provided the Research Computing Environment (RCE) for many years as a cluster resource for social scientists. Through integration of a full stack of open-source and commercial products, we always remained focused on ease of use in the choices. For the next generation of the RCE, we started to develop a full-stack hybrid-cloud solution [1] that we eventually evolved to a fully cloud-native solution called Sid, named after beloved Harvard Government professor, Sid Verba, who had a knack for unifying people across campus.

Developing a full-stack solution was going to require more resources than we could apply for implementation and maintenance. In line with other products at IQSS such as Dataverse [6] and OpenScholar, we aspired to share our full-stack solution as open-source and form a community around it. In the meantime, Open OnDemand [2][3][4][5][9][10][11] was already gaining traction in the HPC space as an open-source technology that bridged the gap between complex resources and scientists running their applications. We decided to pivot from developing a full-stack solution on our own to collaborating with the expertise of the Open OnDemand team for the front-end software and the FAS Research Computing (FASRC) at Harvard for their expertise on implementing and managing scalable integrated back-end solutions. We would migrate our social scientists from RCE and Sid to the FASRC systems and focus our software development efforts on Open OnDemand.

A team at Texas A&M had a similar set of requirements from their researchers and built extensions to Open OnDemand to address some of these.[8] We aimed to do similar modifications by contributing new features to the Open OnDemand community so they are available to a wide audience with similar requirements.

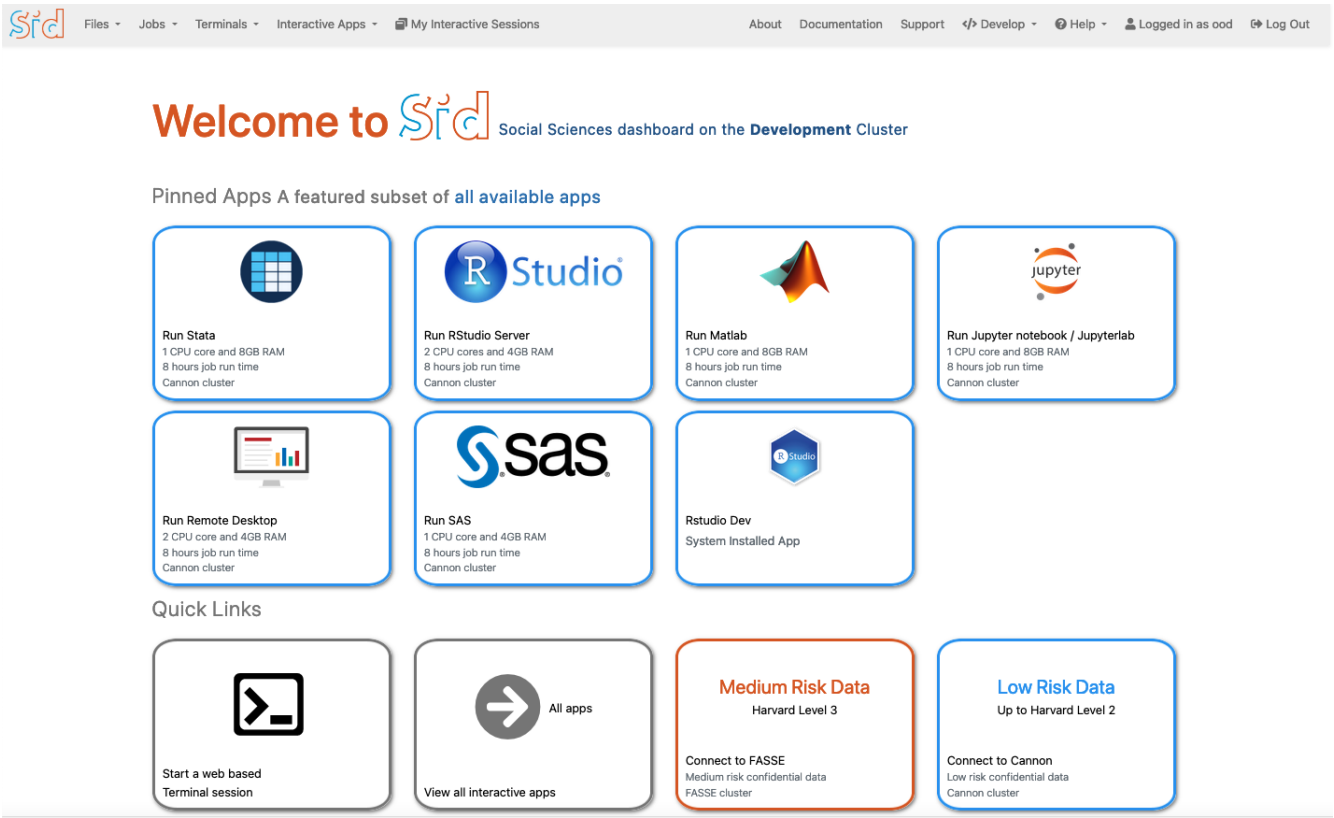

Figure 2: Example custom branding enabling a uniquely-branded dashboard for a school, division or department.

3 USER EXPERIENCE (UX)

Identifying features and subsequently designing and implementing for a simplified and optimized user experience has emerged as an important field of study in and of itself over the past several decades. Being able to do impactful things with as little as the single push of a button is the ultimate optimization. Apps on devices such as smartphones have made us all aware of how easy it can be to do things that were complex in the past. The key element for identifying features was to get feedback directly from the users, to find out what they are thinking while trying to accomplish a task, and to translate and convert their aspirations to simple features that accomplish those goals.

Our team had done some extensive user experience (UX) research by observing users as they used RCE and also as we developed and released beta versions of Sid. In anticipation of our migration of social scientists to the FASRC systems, we also observed how Open OnDemand addressed some of their pain points and identified new areas where we might be able to help.

3.1 User requirements

One common interest among users is the desire to have one button to do common things. The user not only wants a single button, but wants that readily accessible on the main dashboard. In fact, the user seems to derive comfort from staying on the main dashboard while having a set of buttons and menu choices that clearly achieve most or all of their common activities.

Moreover, another common request from the user is to only have available buttons or menu choices that are relevant to them in their context. That is, they do not want to make choices when unnecessary or among options unknown to them. Mirroring this user sentiment is that of the administrator who does not want to explain options that particular users will not need.

Tying these common sentiments together is the observation that even with a simplified dashboard, the research computing environment may nonetheless include complex components and workflows, so anything that can accommodate that back-and-forth relationship between the user and the administrator is appreciated.

From those observations, we identified requirements for the following capabilities:

- Easily accessing resources
- Quick launching of an application
- Branding of an Open OnDemand dashboard
- Offering multiple customized dashboards
- Submitting a support ticket from the dashboard

4 NEW FEATURES

With these requirements in mind, we can now discuss the features that we developed in Open OnDemand 3.0. For more detailed information on how to configure and deploy these new features, please consult the Open OnDemand 3.0 documentation. [7]

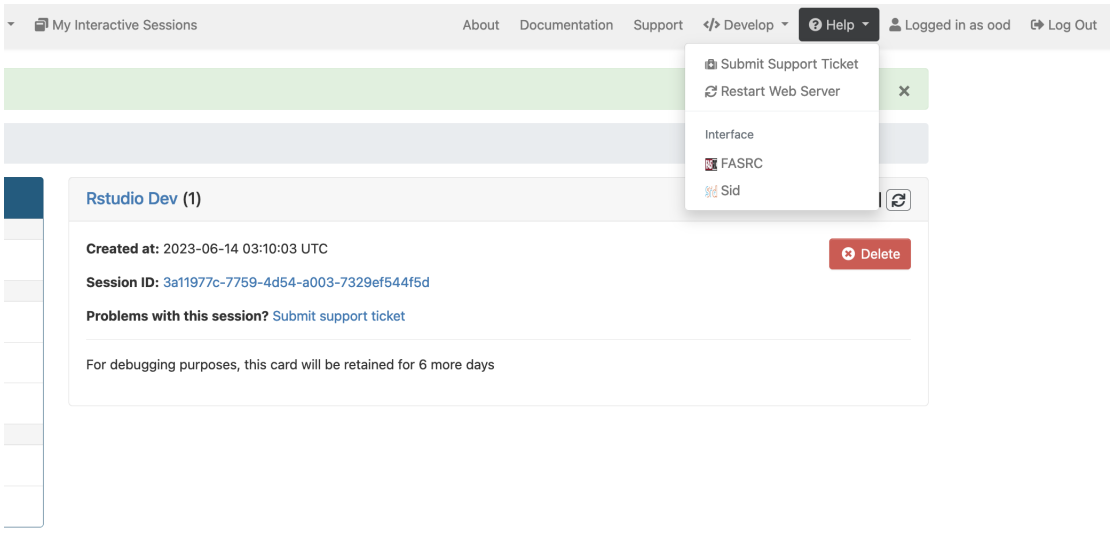

Figure 3: Example configuration profiles enabling selection among different unique dashboards. In this example, the FASRC profile provides the user the dashboard shown in Figure 1 and the Sid profile switches to the dashboard shown in Figure 2.

4.1 Quick-launch buttons

Quick-launch buttons allow the user to launch a particular application with a single push of a button. Based on user feedback and various prototype methods, these quick-launch buttons were implemented by the OSC team using pinned apps. This is a static operation in that a specific number of cores and specific amount of memory is always allocated to the process launched by the button. Future versions could certainly be expanded to allow a more dynamic configuration of the buttons by the user. Workload analysis could be done for individuals or certain sets of users to identify what might be the most popular statically-sized buttons that should be offered by default. Figure 1 shows some examples of quick-launch apps.

4.2 Custom profiles

Quick-launch buttons can be created for any application. In an environment where there are numerous applications available, this can provide some difficult choices if every user must select from all the applications. Some users may only use one or two applications and therefore would not want to see any others. For groups of users like schools or departments, there may be subsets of applications only relevant in those domains.

The custom profiles feature allows definition of named profiles, each providing a unique dashboard with only a certain set of applications. This allows schools, departments, or even research groups to request creation of a profile of dashboard applications that suits their needs. When configuring multiple profiles, the user can dynamically switch among profiles without restarting or changing the system configuration. Figure 3 shows an example of custom profiles.

Furthermore, perhaps in the past, administrators offered this service but in a more brute-force manner by deploying separate Open OnDemand servers for each school or department. If there are many schools or departments, this results in potentially a significant number of systems to manage. The custom profiles feature allows these multiple unique dashboards to be managed as a set of expanded configuration under a single server.

4.3 Custom branding

Now that there is the possibility of multiple dashboards running off a single Open OnDemand server, there may be a desire to brand the different dashboards based on their requestor, whether it be a school, department, research group, or even user. This brings assurance to the researcher that they are working on a dashboard uniquely tailored to their needs. Figure 2 shows an example of the Sid branding.

4.4 Creating support request tickets

Since our users wanted a quick way to submit a support request ticket, we looked at various options to put this option onto the dashboard. This can be done either as a menu option like in the Help menu in Figure 3 or as a separate button on the navigation menu or as a link on the job status panel. If requesting a support ticket via the job status panel, the support ticket can then include metadata from the job that caused the behavior inspiring the support ticket request.

Initially, we also were looking to support connection with the Request Tracker (RT) support ticket system. After discussing with the OSC team, we determined that the default method of e-mail forms would be best. However, this feature was implemented extensibly such that integration with the API of your favorite support ticketing system can be achieved.

4.5 Restarting jobs

In Open OnDemand 3.0, recently_used_apps is a new widget that, like the quick launchers, enables users to efficiently access their

commonly used applications. This widget will show the quick-launch buttons of the four most recently run applications. Alternatively, if you halt an application, you can go to the job tile and click a button there to restart it.

5 CONCLUSIONS

User experience research has proved to be invaluable for prioritizing the features to develop for Open OnDemand. Collaborating with researchers, developers on the OSC Open OnDemand team, the broader set of researchers across our own University, and the Open OnDemand community enables identification of features that can be broadly used and implemented for future extension. We plan to continue active engagement and collaboration in the Open OnDemand community as we continue to develop simple ways to help bridge the ability to use complex resources to solve complex research problems.

REFERENCES

[1] Steven Abramson, William Horka, and Leonard Wisniewski. 2014. A Hybrid Cloud Architecture for a Social Science Research Computing Data Center. In *ICDCSW '14: Proceedings of the 2014 IEEE 34th International Conference on Distributed Computing Systems Workshops*. IEEE Computer Society, Washington, DC, 45–50. https://doi.org/10.1109/ICDCSW.2014.32

[2] Alan Chalker, Eric Franz, Morgan Rodgers, Trey Dockendorf, Doug Johnson, Doris Sajdak, Joseph P. White, Benjamin D. Plessinger, Mohammad Zia, Steven M. Gallo, Robert E. Settlage, and David E. Hudak. 2021. Open OnDemand: State of the platform, project, and the future. *Concurrency and Computation: Practice and Experience* 33, 19 (2021), e6114. https://doi.org/10.1002/cpe.6114 arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/cpe.6114

[3] Alan Chalker, Robert Settlage, and David Hudek. 2021. Open OnDemand App Development and Integration. https://doi.org/10.5281/zenodo.5570225

[4] Dave Hudak, Doug Johnson, Alan Chalker, Jeremy Nicklas, Eric Franz, Trey Dockendorf, and Brian L. McMichael. 2018. Open OnDemand: A web-based client portal for HPC centers. *Journal of Open Source Software* 3, 25 (2018), 622. https://doi.org/10.21105/joss.00622

[5] David E. Hudak, Thomas Bitterman, Patricia Carey, Douglas Johnson, Eric Franz, Shaun Brady, and Piyush Diwan. 2013. OSC OnDemand: A Web Platform Integrating Access to HPC Systems, Web and VNC Applications. In *Proceedings of the Conference on Extreme Science and Engineering Discovery Environment: Gateway to Discovery* (San Diego, California, USA) *(XSEDE '13)*. ACM, New York, NY, USA, Article 49, 6 pages. https://doi.org/10.1145/2484762.2484780

[6] Gary King. 2007. An Introduction to the Dataverse Network as an Infrastructure for Data Sharing. *Sociological Methods and Research* 36 (2007), 173–199.

[7] Ohio Supercomputer Center 2023. *Open OnDemand Documentation*. Ohio Supercomputer Center. https://osc.github.io/ood-documentation/latest/

[8] Duy Pham, Kyle Hsu, Marinus Pennings, Tri Pham, and Phi Au. 2022. Extending Functionalities on a Web-based Portal for Research Computing. In *PEARC '22: Practice and Experience in Advanced Research Computing*. Association for Computing Machinery, New York, NY, Article 37, 4 pages. https://doi.org/10.1145/3491418.3535182

[9] Robert Settlage, Alan Chalker, Eric Franz, Doug Johnson, Steve Gallo, Edgar Moore, and David Hudak. 2019. Open OnDemand: HPC for Everyone. In *High Performance Computing*, Michèle Weiland, Guido Juckeland, Sadaf Alam, and Heike Jagode (Eds.). Springer International Publishing, Cham, 504–513.

[10] Robert Settlage, Alan Chalker, Jeff Ohrstrom, Eric Franz, Doug Johnson, and David Hudak. 2021. Open OnDemand as a Platform for Virtual Learning in Higher Education. In *Proceedings of Sixth International Congress on Information and Communication Technology: ICICT 2021, London, Volume 4 (Lecture Notes in Networks and Systems)*, X.S. Yang, S. Sherratt, N. Dey, and A. Joshi (Eds.). Springer Nature Singapore, Singapore, 323–331. https://doi.org/10.1007/978-981-16-1781-2

[11] Robert Settlage, Srijith Rajamohan, Kevin Lahmers, Alan Chalker, Eric Franz, Steve Gallo, and David Hudak. 2020. Portals for Interactive Steering of HPC Workflows. In *Tools and Techniques for High Performance Computing*, Guido Juckeland and Sunita Chandrasekaran (Eds.). Springer International Publishing, Cham, 179–189.