

Supporting Modern User Interfaces for High Performance Computing

Jay Alameda

Extreme Science and Engineering Discovery Project

National Center for Supercomputing Applications

University of Illinois at Urbana-Champaign

Urbana, IL USA

alameda@illinois.edu

As consumer grade computing has benefited from rapid improvements in user interface design, pushing far beyond traditional windowing environments to user interfaces that use similar modalities between devices, high performance computing (HPC) has largely languished with traditional command line interfaces that are not familiar with an increasing majority of individuals. Furthermore, even though web access to projects such as XSEDE and Globus benefit from Oauth2 [Oauth2] to simplify authentication, the primary client interface for HPC (ssh) lacks any ability to take advantage of modern authentication, which leads projects like XSEDE to deploy a very clunky single sign-on hub, which requires one ssh login via the XSEDE User Portal login, and then another ssh command (gsissh) to successfully log into XSEDE integrated systems. Additionally, the single sign-on hub requires two-factor authentication, which adds an additional layer of complexity to the mere act of logging into an XSEDE integrated system. Trying to bring new users into this environment is truly daunting, and often, off-putting, as it is bringing together many things which are unfamiliar, to simply be able to log into a system.

There have been numerous attempts to improve the situation with user interfaces, eg, through science gateways [Gateways] (which are domain-specific web interfaces, primarily to enable particular science to be accomplished often with back-end high performance computers), comprehensive web interfaces for HPC such as Open OnDemand [OnDemand], and rich desktop clients such as Eclipse Parallel Tools Platform (PTP) [ParallelTools]. As a user, if you are fortunate enough to find a science gateway which supports your science, the problem of effectively using HPC resources is essentially solved. Open OnDemand provides rich functionality for HPC, with significant customization for resource managers, interactive applications such as Jupyter and RStudio, but is only supported at some XSEDE sites, using local authentication, which is not provided by default with one's XSEDE allocation. Similarly, the web interfaces JupyterHub and JupyterLab also provide remote access to HPC resources. PTP provides a rich multi-language development environment, including full support for multiple parallel paradigms including MPI, openMP, and openACC, as well as synchronized projects for effective utilization of remote HPC resources. Similar to Open OnDemand, PTP provides rich customization for resource managers, which is also the same integration mechanism for remote performance tools. GSI authentication was prototyped with PTP, but not able to be included in the release due to intellectual property concerns with the Globus libraries which provided GSI authentication.

Although the XSEDE Project supports integration of science gateways with XSEDE resources through the ECSS program, XSEDE does not support any of these interfaces (gateways, web interfaces, or rich desktop client) as a first-class user interface to XSEDE resources, even though Open OnDemand and PTP have both terminals and interfaces to simplify batch submission and monitoring. This is partially due to scope limitations on XSEDE, cultural limitations on simplifying HPC access, as well as budgetary

limitations in XSEDE to take on the project of providing a greatly enhanced user interface. As we endeavor to engage communities with more diverse needs, from more diverse backgrounds, providing a streamlined, effective, powerful user experience should become a driving factor for projects like XSEDE. For this to become reality, besides overcoming the cultural bias against providing quality alternatives to the command line, other barriers need to be overcome, including full support of OAuth2 for both the clients such as Open OnDemand and PTP *and* the integrated HPC resources, proper integration of batch systems for all fully supported platforms, and integration of a suite of applications to enable remote use. Projects like XSEDE then would need to migrate their integrated access point from the clunky single sign-on hub to embrace full support of interfaces such as Open OnDemand and PTP. This would transform the experience of a new user from struggling with unfamiliar and difficult to use tools accessed through a primitive command line interface, to successfully interact with the remote HPC resource through powerful and easy to use graphical tools, which coincidentally provide command line access, to accomplish the tasks needed for the new user to succeed with HPC.

References:

[OAuth2] OAuth 2.0, <https://oauth.net/2/>, retrieved 01 May 2019.

[Gateways] Lawrence, Katherine A., Michael Zentner, Nancy Wilkins-Diehr, Julie A. Wernert, Marlon Pierce, Suresh Marru, and Scott Michael. "Science gateways today and tomorrow: positive perspectives of nearly 5000 members of the research community," *Concurrency and Computation: Practice and Experience* 27, No. 16 (2015): 4252-4268.

[OnDemand] Hudak et al., (2018). Open OnDemand: A web-based client portal for HPC centers. *Journal of Open Source Software*, 3(25), 622, <https://doi.org/10.21105/joss.00622>

[ParallelTools] Jay Alameda, Wyatt Spear, Jeffrey L. Overbey, Kevin Huck, Gregory R. Watson, and Beth Tibbitts. 2012. The Eclipse parallel tools platform: toward an integrated development environment for XSEDE resources. In *Proceedings of the 1st Conference of the Extreme Science and Engineering Discovery Environment: Bridging from the eXtreme to the campus and beyond (XSEDE '12)*. ACM, New York, NY, USA, Article 48, 8 pages. DOI: <https://doi.org/10.1145/2335755.2335845>