Extended Abstract: The DesignSafe-CI Architecture

Stephen Mock,* Matthew Hanlon, Dan Stanzione; Texas Advanced Computing Center;
and Ellen Rathje, The University of Texas at Austin
* Texas Advanced Computing Center, The University of Texas at Austin, Austin, Texas, 78758-4497, USA; mock@tacc.utexas.edu

Abstract: *Natural hazards threaten life and property across the United States and the world. The DesignSafe cyberinfrastructure (CI) supports multi-hazard engineering by addressing the big challenges associated with multi-hazard research. The DesignSafe CI was recently deployed and is under active development. This paper outlines the existing and planned architecture of the DesignSafe CI.*

1. Introduction

DesignSafe-ci.org [1] is the cyberinfrastructure (CI) component of the NSF-supported Natural Hazards Engineering Research Infrastructure (NHERI). Beyond the CI, NHERI includes shared-use experimental facilities, a computational modeling and simulation center (SimCenter), a post-disaster rapid response research facility (RAPID), and a network coordinating office (NCO). DesignSafe will support integration across the NHERI components and enable research discoveries of the broader natural hazards community.

The DesignSafe CI enables web-based access to data management, simulation, and analysis tools, and will be adding data curation and publication services in the near future. The DesignSafe CI is available to all stakeholders with an interest in natural hazards engineering and has been architected to provide a powerful yet approachable interface to high performance computing and data resources. Furthermore, experience and lessons learned from CyVerse [2] (previously iPlant), Araport [3], NEES [4], and many other CI and gateway projects informed building the architecture for a successful CI. This abstract presents the high level view of the architecture of the DesignSafe CI.

Presented at Gateways 2016, San Diego Supercomputer Center, La Jolla, CA, November 2-3, 2016.
https://gateways2016.figshare.com/

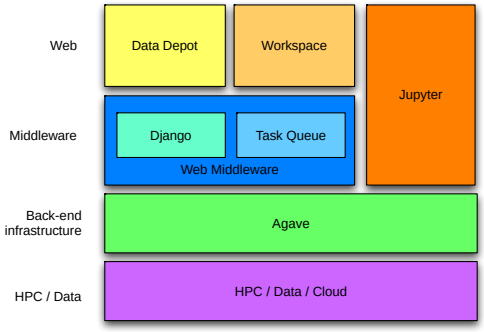

Figure 1: The DesignSafe CI architecture is divided into 4 layers; the web layer, the middleware layer, the back-end services infrastructure layer, and the HPC and Data layer.

2. Data Depot and Discovery Workspace

The Data Depot and Discovery applications enable access to data management, simulation, and analysis tools for interacting with data *in situ* within the DesignSafe CI. Both are AngularJS [5] applications, designed to provide a rich and powerful interface within a web browser.

The Data Depot is where users go to move data into and out of the CI, manage, organize, and share data, and annotate data with metadata. There are currently four sections to the Data Depot: My data, Shared data, Public data, and third party data (at this time Box.com). Users can upload data via drag and drop from their computer or transfer data from Box.com. Users can also download data to their computer or transfer data from DesignSafe to Box.com.

The Discovery Workspace provides access to simulation and analysis tools such as OpenSees [6], OpenFOAM [7], ADCIRC [8], and MATLAB [9]. Users can run any of these applications from within their web browser using data from the Data Depot. Generated outputs are archived back to the Data Depot. Some applications, like MATLAB, have an interactive GUI interface. To enable access

to this interface DesignSafe utilizes Web-VNC solution based on the TACC Visualization Portal [9]. Most applications can be run either as batch jobs on a HPC resource or utilize an on-demand cloud execution model. New tools can be added both by administrators, and soon by the users to ensure a relevant and up-to-date set of tools an algorithms are available to the community.

3. Middleware

Backing the Data Depot and Discovery Workspace is a web middleware consisting of a Django [11] webserver, a websockets server, and a task queue.

The Django webserver provides a content management system (CMS) and server code to orchestrate requests between the Data Depot and Workspace user applications and the backend infrastructure. In addition to typical HTTP requests, DesignSafe uses websockets to enable real-time, bi-directional communication between the server and the browser client applications. This allows the server to "push" information to the client when events occur, such as a change in job status in the Discovery Workspace, or when a large file transfer completes in the Data Depot.

Alongside the Django application, DesignSafe operates a task queue. The task queue is utilized for managing the execution of long-running, asynchronous tasks. Examples of such tasks are managing the execution of a job on an HPC resource, or copying a large amount of data between the CI and a third party cloud storage service. Rather than executing tasks directly, tasks are executed through the queue. Tasks can then be monitored separately for status changes and retried in the case of a failure. The user is not blocked from continuing to use the application and notifications are sent when a tasks finish.

4. JupyterHub

In addition to the Data Depot and Workspace, DesignSafe has also deployed a JupyterHub instance [12]. JupyterHub is a multi-user server for running Jupyter notebooks, which are in turn web applications for editing and executing documents with accompanying code, computations, text, and rich media. DesignSafe's JupyterHub currently supports notebooks containing code in Python (versions 2 and 3) and R. The DesignSafe JupyterHub utilizes the same authentication layer as the main DesignSafe Portal and provides direct access to the user's own data as well as public data from the data depot. In addition to the notebook, users can access a virtual command-line terminal. Along with basic Linux command-line tools, DesignSafe has further customized this environment with software tools common to the community, such as OpenSees.

5. Agave Science-as-a-Service Platform

The DesignSafe CI leverages the Agave Science-as-a-Service platform [13] for managing user authentication, job execution, and data and metadata management. The Agave platform enables DesignSafe to schedule job execution and manage data across multiple HPC and cloud resources.

6. HPC and Data

Simulation jobs are run on the HPC and cloud systems at TACC. Where appropriate the jobs on the VMs have been packaged to run in Docker [14] containers, enabling them to be easily run on different clouds with little change. Jobs are configured, launched, monitored, and managed using the Agave Apps and Jobs APIs.

The Data Depot files sit at rest on TACC's high performance storage and data management system, Corral. This filesystem on Corral is automatically replicated to an offsite copy at UT Arlington. Corral exports NFS to targeted systems for direct access, and Agave's Files API provides additional access via a RESTful web service interface.

7. Conclusion

The DesignSafe CI expands upon lessons learned from other large scale CI projects, and lays the foundation for a capable platform for natural hazard engineering research. The platform is both extensible and generic enough that it need not be limited to natural hazards engineering but could be applied to other fields of study.

8. Acknowledgments

DesignSafe-CI is generously supported by a grant from the National Science Foundation

(#1520817).

9. References

[1] https://www.designsafe-ci.org/

[2] Goff, Stephen A., et al., "The iPlant Collaborative: Cyberinfrastructure for Plant BIology," *Frontiers in Plant Science 2* (2011), doi: 10.3389/fpls.2011.00034.

[3] Hanlon, M. R., Vaughn, M., Mock, S., Dooley, R., Moreira, W., Stubbs, J., Town, C., Miller, J., Krishnakumar, V., Ferlanti, E., and Pence, E. (2015). Araport: an application platform for data discovery. *Concurrency Computat.: Pract. Exper.*, doi: 10.1002/cpe.3542.

[4] https://nees.org/

[5] https://angularjs.org/

[6] McKenna, F. (2011). "OpenSees: A Framework for Earthquake Engineering Simulation,." Computing in Science and Engineering, 13(58).

[7] H. G. Weller, G. Tabor, H. Jasak, C. Fureby, *"A tensorial approach to computational continuum mechanics using object-oriented techniques,"* COMPUTERS IN PHYSICS, VOL. 12, NO. 6, NOV/DEC 1998.

[8] Luettich, R., and Westerink, J. (2004). "Formulation and Numerical Implementation of the 2D/3D ADCIRC Finite Element Model Version 44.XX." ADCIRC.

[9] http://www.mathworks.com/products/matlab/

[10] https://vis.tacc.utexas.edu/

[11] https://www.djangoproject.com/

[12] http://jupyter.org/

[13] Dooley, R., M. Vaughn, D. Stanzione, S. Terry, and E. Skidmore (2012). "Software-as-a-service: The iPlant Foundation API." 5th IEEE Workshop on many-task computing on grids and supercomputers.

[14] Dirk Merkel. 2014. Docker: lightweight Linux containers for consistent development and deployment. *Linux J.* 2014, 239, pages.