Enabling User Driven Web Applications on Remote Computing Resource

Weijia Xu Ruizhu Huang Yige Wang
Texas Advanced Computing Center
The University of Texas at Austin
Austin, Texas, USA
xwj@tacc.utexas.edu rhuang@tacc.utexas.edu yige.wang@utexas.edu

***Abstract*—While CI providers have continued success with the infrastructure-as-a-service model (IaaS), there are increasing demands to offer more user driven service models from domain scientists. We propose a user driven web application that empowers users to run their interactive analytic tasks using CI resources dynamically. Ad-hoc analysis routines can be described with multiple pre-defined task modules in a configuration file that can be shared and re-used. A user can run the web application on CI resource without alleviated privilege or additional service deployment by administrators. The functions and user interface of the web application are automatically initialized based on the configuration file. Therefore, the framework offers a new way for a user to access and utilize remote resources. This new model can effectively reduce the access barrier to remote computing resource offered by CI. In this paper, we describe the proposed architecture of this framework and give a use case example.**

Keywords-component: high performance computing; cyberinfrastructure; reconfigurable web service; service computing

I. INTRODUCTION

Across many domain science fields in academia, there is an increasing trend to move computational expensive analytics to remote advanced computing resources [1] [2] [3] [4]. However, the complexity of infrastructure and existing service models presents challenges to users.

To improve the CI resource accessibility for wide variety use cases and domain fields, we propose a user driven web application that can dynamically support a various analysis pipeline by enabling users to set up their own interactive web application. This application effectively bridges existing resource and services with specific needs from users. Common operations are abstracted as *task*s. Multiple tasks can be composed to an *application* through a simple configuration file in JSON format. The framework can then generate a web user interface to allow user run the analysis interactively on the remote resources.

In this paper, we describe the design and implementation of proposed framework and discuss how our proposed approach offers complementary services to existing approaches.

II. RELATED WORK

There are three most common CI utilization models (job submission with secure shell, remote software session, web portal and gateway) for open science. The proposed application framework offers features complementary to the existing service models. With IaaS model, users can start a customized web application with improved usability and interactivity. CI providers have the flexibility to deploy the proposed framework with specific workflow defined as a software service for a targeted user community. The framework can also be deployed and made available as part of platform to support dynamically created workflows.

The framework is also related to widely available existing workflow management system from open source community and industry [5] [6] [7] [8]. However, a key difference is that the proposed framework is designed to run as a standalone web application. It doesn't rely on server tools and software installation at remote resources. Hence the proposed framework can be easily run on different remote resources, even in a heterogonous resources environment. This component enables not only sharable *workflows* but also multi-user analytic instances.

III. ARCHITECHURE DESIGN AND DATA MODEL

A. Architecture Overview

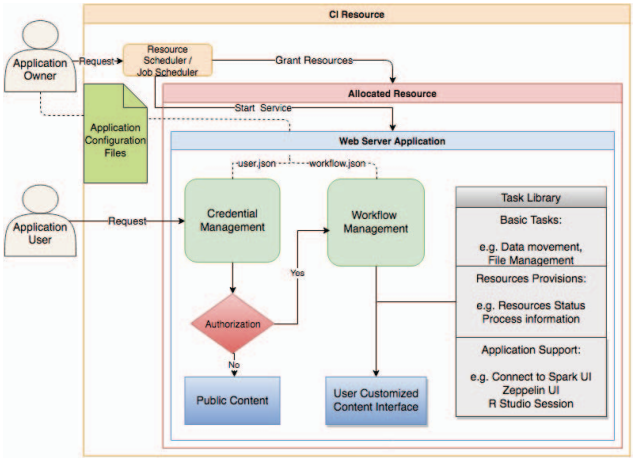

Figure 1. Overview of framework architecture and access workflow

An overview of the system architecture of proposed framework and startup workflow is shown in Figure 1. To start the web application, application owner will first request resources from remote cyberinfrastructure. Once the application and its configuration files are accessible from allocated resource, a web application server can be started by on the allocated resources for application users to interact through web interface.

978-1-5386-7374-4/18/$31.00 ©2018 IEEE
DOI 10.1109/SERVICES.2018.00038

There are four key components of the proposed web application framework: credential management, application management, pre-defined task libraries and application configuration files. The application configuration files are central in the proposed web framework. Many of the features and content of the web application can be dynamically and customized through configuration files.

B. Preliminary Implementation

We have implemented the proposed framework using Play Web Framework (version 2.6). The authentication and authorization services are implemented using Silhouette authentication library (version 5.0). The application has been tested with computing resources hosted at Texas Advanced Computing Center.

C. A Simple Example

An exemplar application configuration file consisting of three tasks is shown in Figure-2 (top left). The result of this configuration file is a web interface with three steps each of which corresponds to one task (Figure-2 right). This exemplar application is designed to enable users to run processing tasks in parallel. The first task creates a widget for users to interactively upload script files to be executed to the remote resource. The second task is to enable user running distributed copies of script using MPI [9]. The third step will help users to inspect the combined results.

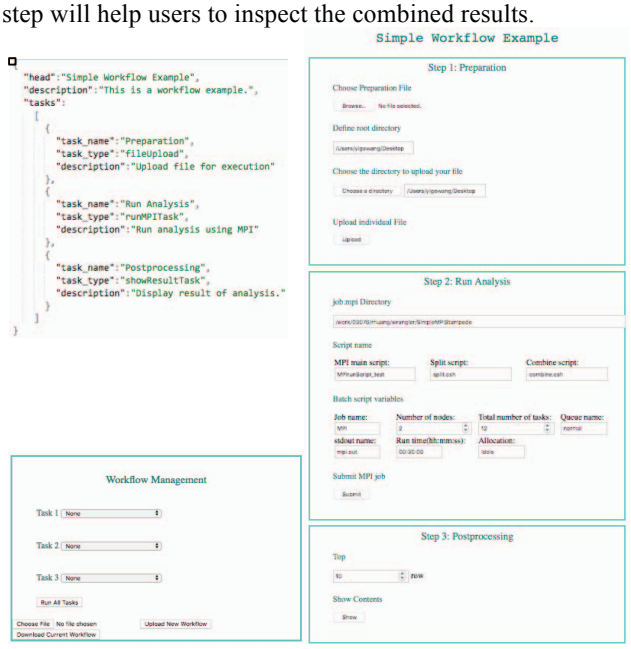

Figure 2. Example of workflow configuration and result web interface

In addition to three processing steps specified by configuration file, the workflow management component also generates control elements for users to interactive with the workflow itself (Figure-2 bottom left). There are three workflow interactions currently implemented: 1) download the current workflow as a JSON file; 2) upload a new workflow configuration file to change the workflow dynamically; 3) specify dependency among tasks.

IV. Summary and Ongoing Work

In this paper, we have presented a web application framework proposal to enhance the accessibility of large cyberinfrastructure to users from diverse domain fields. The framework includes a set of pre-built task modules to help bridging users with remote hardware and software resources. By specifying JSON formatted configuration files, users can transform an ad-hoc analytic workflow into a dynamically composed multi-user interactive web application. The generated web application is self-contained with minimum system dependency on remote system. The composed workflow can also be exported, preserved and shared by other users. While it is still an ongoing development, we have successfully used with several ongoing projects. Only a simplest example is included here due to space constraint.

Our ongoing work includes extending the usability and extensibility of the framework by implementing additional re-usable task modules and supports of more remote resources and analysis tools. The framework can be used as a way to set up cloud based virtual laboratory for training and education on advanced computing technology.

Acknowledgment

This work has been support through funding from National Science Foundation (Award# 1726816). Software testing have been supported with Wrangler, a NSF funded cyberinfrastructure resource (Award # 1341711).

References

[1] Merchant, Nirav, et al., "The iPlant Collaborative: Cyberinfrastructure for Enabling Data to Discovery for the Life Sciences," *PLOS Biology*, 2016.

[2] Stephen A. Goff et al., "The iPlant collaborative: cyberinfrastructure for plant biology," *Frontiers in plant science*, no. 2, p. 34, 2011.

[3] E.M. Rathje et al., "DesignSafe: new cyberinfrastructure for natural hazards engineering.," *Natural Hazards Review*, vol. 18, no. 3, 2017.

[4] Avita Katal and Mohammad Wazid and R. H. Goudar, "Big data: issues, challenges, tools and good practices," in *In Contemporary Computing (IC3), 2013 Sixth International Conference on.*, 2013, pp. 404-409.

[5] Diimitrios Georgakopoulos and Mark Hornick and Amit Sheth, ""An overview of workflow management: From process modeling to workflow automation infrastructure," *Distributed and parallel Databases*, vol. 3, no. 2, pp. 119-153, 1995.

[6] Suresh Marru et al., "Apache airavata: a framework for distributed applications and computational workflows," in *In Proceedings of the 2011 ACM workshop on Gateway computing environments*, 2011, pp. 21-28.

[7] Ilkay Altintas, Chad Berkley, Efrat Jaeger, Matthew Jones, and Bertram Ludascher and Steve Mock, "Kepler: an extensible system for design and execution of scientific workflows," in *16th Scientific and Statistical Database Management 2004* , 2004, pp. 423-424.

[8] Malcolm Atkinson, Sandra Gesing, Johan Montagnat, and Ian Taylor, "Scientific workflows: Past, present and future," *Future Generation Computer Systems*, no. 75, pp. 216-227, 2017.

[9] William Gropp, Ewing Lusk, Nathan Doss, and Anthony Skjellum, "A high-performance, portable implementation of the MPI message passing interface standard," Parallel computing, vol. 22, no. 6, pp. 789-828, 199