

## Evaluating CIPRES Science Gateway Integration Options with Apache Airavata

Saminda Wijeratne, kswijera@iu.edu, IU; Mark A. Miller, mmiller@sdsc.edu, SDSC; Terri Schwartz, terri@sdsc.edu, SDSC; Suresh Marru, smarru@iu.edu, IU; Marlon Pierce, marpierc@iu.edu, IU

Science Gateways democratize access to use of cyberinfrastructure, enabling users around the world to incorporate sophisticated analysis, computational modeling and simulation techniques into their research. Apache Airavata is a service-based software framework designed to support a wide range of such science gateway systems, including desktop clients and Web browser-based gateways. The Science Gateway Platform as a service (SciGaP) project is furthering the development of Apache Airavata to provide a hosted gateway platform as a service that is designed to provide the generic middleware functionalities required by all gateways. This will enable wider adoption by lowering the overhead for gateway creation.

As our first step, we have chosen to build production quality middleware services that meet the needs of existing heavily used gateways: the CIPRES Science Gateway (CSG), the UltraScan Science Gateway (UltraScan) and the Neuroscience Gateway (NSG). In this poster we evaluate strategies for integrating the Workbench Framework, the software package used to create CIPRES, NSG and other gateways, with the Apache Airavata software framework. A key SciGaP milestone is for current and future gateways using the Workbench Framework to be able to adopt hosted SciGaP services easily. We examine two architectural choices for achieving this goal.

**Integration with the Tool Registry:** One approach is to integrate Airavata with the Workbench Framework's tool registry. The tool registry defines service provider interfaces for job management and file management functionalities. Integration through tool registry carries benefits such as seamless integration without any changes to existing Workbench code, support of application abstractions, transparent deployments and workflows (lifting gateway responsibility to manage them) and advance status monitoring. Drawbacks include exposure of features limited to what tool registry offers (such as workflow management and advanced scheduling capabilities) and static code generation at compile time for tools added/removed/modified.

**Direct Integration of Airavata with the Workbench:** A second integration approach is to re-architect the Workbench Framework to bypass the tool registry abstraction. This approach facilitates a complete delegation of job management to Airavata, allowing the gateway developers to focus only upon the user-oriented capabilities (increasing the science done through the gateways). Benefits include one less layer of abstraction (Tool Registry), support for workflow management and advance scheduling capabilities, direct interface support with the gateway database and user management functionality. While significantly more effort is needed, this approach is more forward-looking and will enable Workbench-based gateways to collaborate with fellow gateways in adding new capabilities.

**Evaluation and Conclusion:** The initial goal of the integration is to provide proof-of-concept for the premise that Apache Airavata can provide a set of middleware services that in total constitutes a general purpose gateway framework. As an immediate outcome, we are testing Airavata with heavy loads of CIPRES gateway usage. With the ability to quickly spawn new, fully capable gateways based on the Workbench Framework, we are also evaluating Airavata's use as a multi-tenanted service.

# Poster Mockup

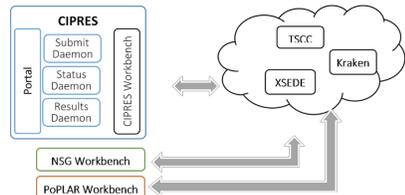
Note: The following mockup does not illustrate the format of the final poster. The completed poster will have more descriptive diagrams along side helper text. I'm providing a few larger resolution images for reference purposes.

## Evaluating CIPRES Science Gateway Integration Options with Apache Airavata

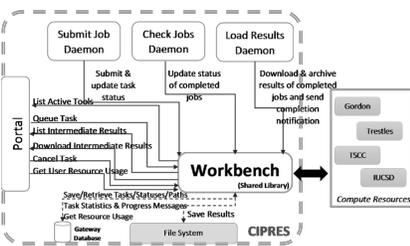
xxx xxxx    xxx xxxx    xxx xxxx    xxx xxxx    xxx xxxx  
 xx@xxx    xx@xxx    xx@xxx    xx@xxx    xx@xxx

### Introduction

quick brown fox jumped over the fence quick brown fox jumped over the fence quick brown fox jumped over the fence

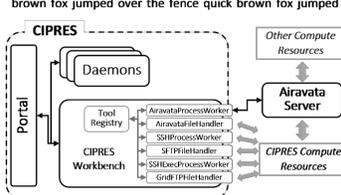


quick brown fox jumped over the fence quick brown fox jumped over the fence



### Method 1

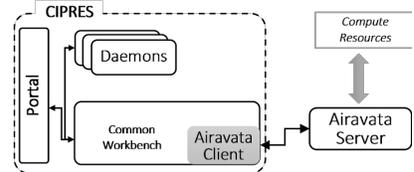
quick brown fox jumped  
 • over the fence quick brown  
 • fox jumped over the fence quick brown fox jumped  
 • over the fence quick brown fox  
 • jumped over the fence quick  
 brown fox jumped over the fence quick brown fox jumped



quick brown fox jumped  
 • over the fence quick brown  
 • fox jumped over the fence quick brown fox jumped  
 brown fox jumped over the fence quick brown fox jumped

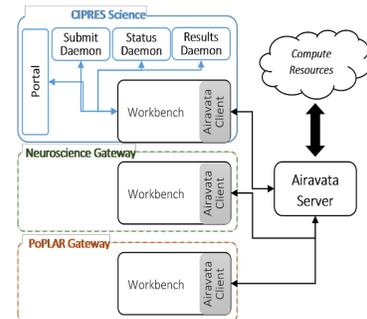
### Method 2

quick brown fox jumped  
 • over the fence quick brown  
 • fox jumped over the fence quick brown  
 fox jumped brown fox jumped over the fence quick brown fox jumped over the fence quick brown fox jumped over the fence quick brown fox jumped over  
 • over the fence quick brown  
 • the fence quick brown fox jumped over the  
 • fence quick brown fox jumped over the fence quick



• over the fence quick brown  
 • fox jumped over the fence quick brown  
 fox jumped brown fox jumped over the fence quick brown

### Conclusion



quick brown fox jumped over the fence quick brown fox jumped over the fence quick brown fox jumped over the fence

