

Plugin architecture for The Mass Open Cloud GUI - Project Proposal

Team Members:

Dana Aljawder
Lama AlSuwayan
Everett Carson
Igibek Koishybayev
Hung Vong

Project Mentor:

Jon Bell

**EC 500 – Cloud Computing
Spring 2015**

1. Vision and Goals Of The Project:

The plugin for MOC-UI will function as a web application that has an interface to control physical machines (HaaS), and will interact with HaaS APIs to allocate resources for the use cases. Our goals include:

- enabling an interface to request for resources with minimal wait time
- creating a handler for large amounts of requests
- simplifying the process so that end-users can be either non-experts or experts

Users/Personas of the Project:

We expect the users of the plugin to be users of HaaS. This means we expect our end users to be system administrators, research students from the different universities (BU, MIT, HU, NU, UMass) and groups that want to run their experiments on the physical nodes.

2. Scope and Features Of The Project:

The end-result of this project aspires to create a plugin that can both interface with the existing MOC-UI and utilize HaaS APIs to control resource allocation. The interface will be a very simple one. The hope is that users can use the interface to request for services, or create their own based on the architecture provided.

3. Solution Concept

This section provides a high-level outline of the solution.

Global Architectural Structure of the Project:

The following diagram provides an overview of the interaction between the web app we are going to develop and the existing Haas Server.

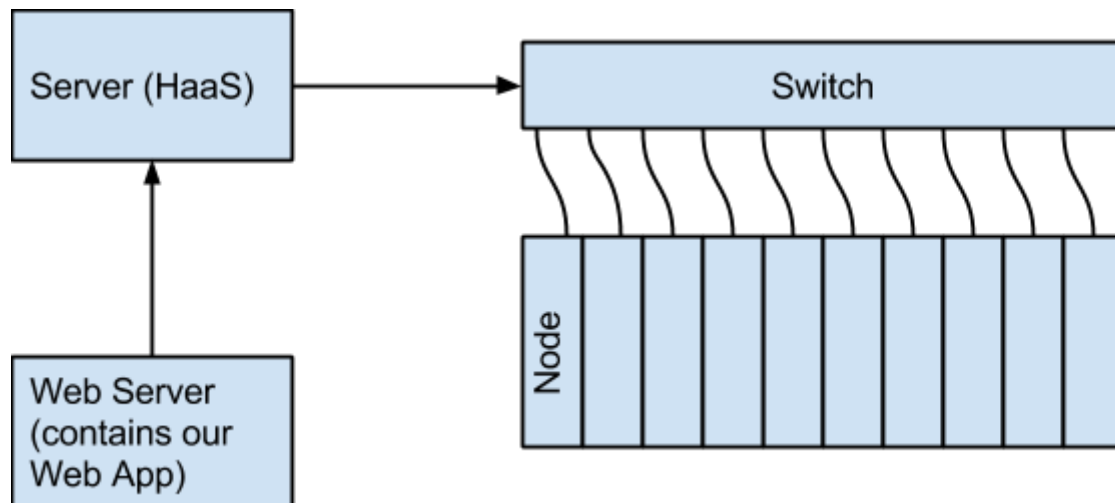


Figure 1 - Project Overview

To accomplish our goal, the following needs to be achieved:

- Setup a web server (including installation of software)
- Develop web pages that connect to the HaaS API
- Add an iframe to the MOC-UI that displays our web application

Design Implications:

The web application should allow the user to perform the following functions:

- Get authenticated using existing HaaS user database
- View projects
- View action history
- Create Project
- Create Headnode
- Create/Manage Networks
- View available nodes
- Manage nodes (add/remove them from projects)

4. Acceptance criteria

Minimum acceptance criteria:

- Plugin must be accessed using an iframe in the MOC UI.
- Basic UI should allow the end-user to utilize basic HaaS API services to manage their projects (create project, create networks, associate nodes).

Stretch:

- UI should be able to authenticate users.

5. Release Planning:

Release #1(Week 6):

Create a template web page that can traverse pages by pressing dummy buttons.

Release #2(Week 8):

Integrate the create/delete project HaaS functionality to the web page.

Release #4(Week 10):

Integrate adding nodes functionality to the web page.

Release #4(Week 12):

Integrating the create/manage networks functionality to finalize the API.

Release #5(Week 14):

Integrating the plugin/web application into the MOC-UI using an iframe.