WEB SYSTEM FOR CREATING AND MANAGING VIRTUAL HIGH PERFORMANCE COMPUTING ENVIRONMENTS

Pedro Adriano Pessoa Teixeira

Project/Dissertation developed under supervising of Prof. Jorge Manuel Gomes Barbosa and Prof. Tito Carlos Soares Vieira

in Faculty of Engineering of Porto University

1. Motivation

Some computing infra-structures, namely Grids and Clusters, can be rather inflexible when compared to Clouds, as the latter are supposed to allow the user to take advantage of a myriad of services, and not just computing power.

Current Grid infrastructures are generally not very flexible when it comes to the users' needs and the user must adapt its code to the infrastructure. *Cloud Computing* on the other hand, is associated with an extreme flexibility allowing the infrastructure to adapt itself to the users' requirements.

Having this in mind, FEUP has started developing a private cloud project at its Informatics centre (CICA - Centro de Informática Prof. Correia de Araújo). This project aims at increasing the usability of the current computing system that exists at FEUP and with this, increase its usage and stop the lack of growth. In order to achieve this goal, it was proposed that a web portal would be developed which would simplify the access to the system.

2. Main Goals

This project has the following objectives:

1. Create a web interface for FEUP's private cloud project;

2. Allow for the *on-the-fly* creation of Virtual Machine (VM) images;

3. This creation must be dynamic, i.e. the images must be created according to user specification;

4. The users must be helped when choosing which VM image to use;

5. The web system must provide a way of managing the VM images.

3. Work Description

The development of the project can be divided in the following parts:

1. Developing a web system;

2. Implementing the VM image creation;

3. Implementing the VM image management;

4. Helping the users choose a suitable VM image to launch.

3.1. The web system

This project involved the implementation of a web system that helped users decide which VM they should use for their computing jobs. This web system was fully built in *Django* and *Python*, using some helpful modules.

3.2. VM image creation

The VM image creation was implemented using *Ubuntu*'s *vmbuilder*, using a *bash* script that is called in the web system. This script gathers the information that the user inputs in the web system and creates a new VM image.

3.3. VM image management

Users are able to manage their created VM images and system administrators are able to manage both users and VM images.

A specific section was created inside the web system for this purpose. System administrators are able to make their decisions based on information provided, information which is collected from the system, such as VM images date of creation and date of last usage.

3.4. Helping the users

Users are provided with information considered relevant so they can perform their choices. A set of statistics collected from both the system and the users actions is presented, such as most used VM images.

A search function was developed, so the users can look for an image that may already exist. This search function uses ``tags'' attached to the VM image objects. These ``tags'' were implemented using a *Django* module named *django-tagging*. This module attaches ``tags'' to any instance of a model defined by *Django*.

Using this ``tag'' capability, statistics such as most used and most searched ``tags'' are also presented to the user.

A ``tag'' cloud is also presented so the user can obtain a visual representation of the ``tag'' distribution in the system.

3.5. Integrating with *OpenStack*

A possible integration with the cloud *middlware OpenStack* is also documented, showing the basis of that integration and how it can be achieved.

4. Conclusions

The objectives were completed and the system was successfully developed. VM image creation and customization is implemented, showing that users can in fact take advantage of VM images created by other users, thus reducing deployment time.

Advantages over *OpenStack* are also depicted, namely the ability of searching for a VM image and actually managing VM images and users in a user-friendly way (*OpenStack's* interface is very technical, whereas the web system developed tries to be as less technical as possible and fully describing what option does what so the user has no problems using it).