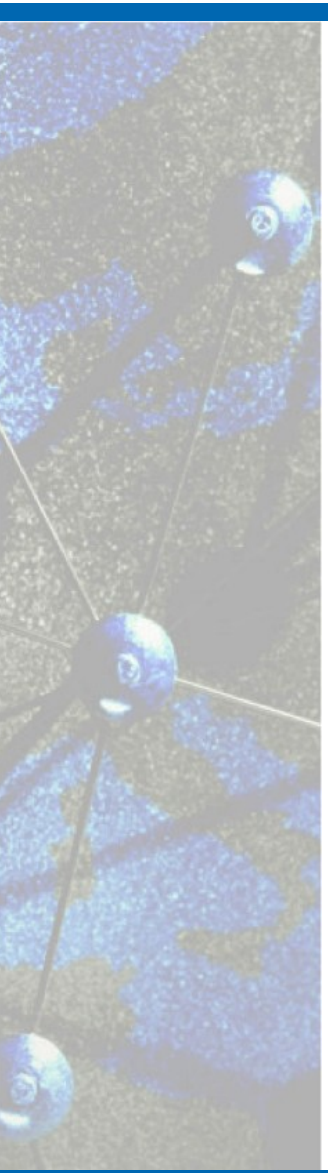

Brokering in EGI FedCloud: A Primer

Iván Díaz

CESGA

Outline

Introduction

- The EGI Fedcloud offers users access to geographically distributed cloud computing resources.
- These resources can be configured and self-managed by the user in increasingly complex ways.
- They can also be deployed in complex and flexible ways both in distribution and scaling in response to puntual or unforeseen needs.
- All this complexity demands more of the users than competing solutions, and the role of Brokering is to make their task easier

Introduction II

- The EGI Fedcloud is a IaaS cloud solution. This means the cloud lets users to create machines supported by an infrastructure, as opposed to exposing a computing platform or a service.
- Since this is a infrastructure level, users have to know how to setup an server, instead of just sending a job and waiting for the result.
- Setting up a server means selecting a OS, installing it, configuring the network correctly, installing the required applications and services, configuring them and enabling remote access for the final users.
- This means extra work, but allows users to fully customize their computing platform and software.

Overview

- In this presentation we will see the role of brokering inside the EGI Fedcloud.
- Compared to less sophisticated integration strategies.
- And also several infrastructure and brokering solutions that are supported now.

Outline

Manual Server Setup

- The simplest Integration strategy.
- The user instantiates a canned pre-installed OS image from AppDB
- Generates a set of SSH keys and uploads them to the image.
- Starts the server, logs-in and does all the setup operations needed.

Manual Server Setup II

Advantages

- Good for testing
- Also simple or one-time jobs.

Disadvantages

- Each individual server needs manual intervention.
- Non replicable or scalable.
- Time consuming and difficult to debug.
- Very vulnerable to failures.

Contextualization

- There are a lot of setup operations that must be done on each VM
- The most important of these is the credential setup.
- Also network configuration, installation, updates, patches, etc.
- This can be automated on a **contextualization** script, supported by OCCI and most VMMs.
- This is the first step in automation.

Basic OS with contextualization

- The simplest *automated* Integration strategy.
- The user instantiates a canned pre-installed OS image from AppDB
- The user associates or creates a contextualization script that
  - Generates a set of SSH keys and uploads them to the image.
  - Starts the server, logs-in and does all the setup operations needed.
- All the keys, packages and data needed by the script can be maintained on a external site.

Advantages

- The configuration step is automated and much faster.
- Changes on the configuration or supporting cloud environment are much easier.
- Perfect for applications and jobs with minimal setup and data dependences.

Disadvantages

- Some effort is needed developing or adapting the script.
- Time consuming and difficult to debug.
- If the script depends on third sites this adds an important failure point.

Fully Customized OS Image

- The user makes a disk image that is fully preconfigured.
- This image does not need to use a template or standard OS.
- Very complex configurations and static data can be added and made available instantly.

Fully Customized OS Image (II)

Advantages

- A way to ”ascend” physical machines to the cloud.
- The setup of the image takes virtually no time, as is pre-done.
- Non-standard OS and applications can be added with ease.

Disadvantages

- The upkeep and maintenance of the image can get prohibitive.
- Non updated legacy images can represent a security issue and force cloud providers to take it down.
- If the disk image is large, the transfer and instantiation times can be prohibitive, specially if frequently updated.
- Very vulnerable to misconfiguration and security problems.

Infrastructure Broker

- A broker that manages creating images and manages instantiation.
- Generally offers several interfaces, including web and graphical.
- Geared towards deploying thousands of geographically distributed images.
- Also can manage elasticity and peak processing.
- Generally uses image templates and its own implementation of contextualization.

Infrastructure Broker (II)

Advantages

- Allows very large deployments with minimal human intervention.
- Support inter-cloud deployments.
- Automatic elasticity and peak-processing.
- Guarantees better QoS for third-party final users.

Disadvantages

- Not recommended for one-shots or a low number of instances.
- Has a steeper learning curve.
- Can have vendor lock-in.

Application Broker

- The broker is abstracted as an API in the application level.
- Instances or controller processes can spawn other instances on the fly.
- Geared towards applications that can scale up or down dramatically.
- Better elasticity and peak processing managing that infrastructure brokers, since the load balancing can access local monitoring.
- Generally uses image templates and its own implementation of contextualization.

Application Broker (II)

Advantages

- Allows very large deployments with minimal human intervention.
- Support inter-cloud deployments.
- Automatic elasticity and peak-processing.
- Offers very fine cloud cost / computation efficiency trade offs, applications can go from one side of the spectrum to the other.

Disadvantages

- Not recommended for one-shots.
- Has a steeper learning curve.
- Can have vendor lock-in.

Outline

Infrastructure Brokers

- **VCycle** — A VM lifecycle management based on the vacuum model. Supports OCCI and EC2, and instantiates jobs from experiments autonomously.
- **Catania Science Gateway Framework** — A solution to create VM Images, manage auth, and deploy on different cloud providers using OCCI.
- **CompatibleOne** — All in one proprietary solution for brokering, accounting, user management, monitoring, etc..
- **SlipStream** — A Open Source 1-click solution for VM creation, deployment and contextualization, with cluster deployment support.
- **VMDirac** — An extension of the Dirac portal for cloud, it supports rOCCI, EC2 and accounting, monitoring, brokering and scheduling, with SSH contextualization.

Application Brokers

- **COMPSs** — Programming libraries and a runtime system that offers automatic parallelization and orchestration of applications and services, elasticity and auto scaling.
- **Catania Science Gateway Framework** — A solution to create VM Images, manage auth, and deploy on different cloud providers using OCCI.
- **WSPGrade** — A job and workflow submission tool for cloud. Grid applications can be ported with ease.
- **VMDirac** — An extension of the Dirac portal for cloud, it supports rOCCI, EC2 and accounting, monitoring, brokering and scheduling, with SSH contextualization.

Outline

Conclusions

- Brokering helps bridge the jump from a few virtualised servers to a true distributed computing environment.
- There are a lot of trade offs and considerations even in the simplest integration strategies.
- Users need to consider if they need brokering and EGI Fedcloud needs to make it as transparent as possible.