

EGI Training Session – Federated Cloud

Gergely Sipos² Boris Parák¹ Enol Fernández² Diego Scardaci²

¹CESNET

²EGI.eu

Nov 18, 2015



Introduction

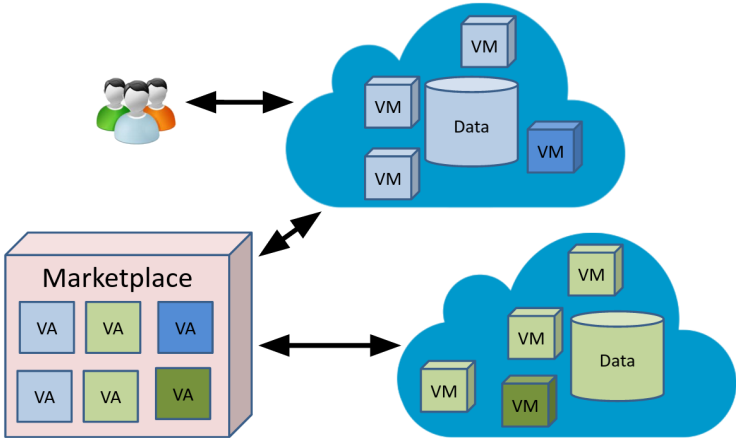
Cloud Use Case(s)

- highly-specific software environments
- need for auto-scaling or cloud-bursting features
- occasional resource usage (low long-term utilization)
- experimental or development environments
- service hosting (application servers, databases, ...)

Software as a Service (SaaS)

Platform as a Service (PaaS)

Infrastructure as a Service (IaaS)



VM → Compute Instance

VA → Virtual Appliance

- Computing Services → Virtual Machines
- Block Storage Services → Virtual Machine Disks
- Object Storage Services → FedCloud or **EUDAT**
- VA Management Services → **The EGI AppDB**

Catania Science Gateway	<ul style="list-style-type: none">• SaaS• Identity Federation	VMDIRAC	<ul style="list-style-type: none">• Abstraction on top of various HPC/HTC/cloud
Slipstream	<ul style="list-style-type: none">• PaaS for automating deployments• Helix Nebula	WS-PGRADE	<ul style="list-style-type: none">• Workflow development and enactment
COMPSs	<ul style="list-style-type: none">• Framework for auto-parallelisation	Vcycle	<ul style="list-style-type: none">• VM lifecycle manager

Today focusing on *IaaS*!

Compute Management

- provisioning a compute instance from a VA
- retrieving details of a compute instance
- triggering actions on a compute instance
 - *restart*
 - *suspend* ↔ *start*
 - *stop* ↔ *start*
- deprovisioning a compute instance

Additional Capabilities

- boot-time compute instance contextualization
 - *user-data* (**cloud-init**)
 - *public key*
- run-time (un)linking of *storage* instances
- run-time (un)linking of *network* instances
- ... other features demanded by users coming soon!
 - run-time compute instance resizing
 - saving running compute instances

- a way to “change” a compute instance on boot
- useful for adjusting configuration, providing instance-specific metadata or small files
- using a *de facto* standard → **cloud-init**
- YAML configuration file, extensive documentation
<https://goo.gl/sP5kr9>

```
#cloud-config  
  
phone_home:  
  url: http://my.example.com/$INSTANCE_ID/  
  post: [ pub_key_dsa, pub_key_rsa, pub_key_ecdsa, instance_id ]
```

- built on OGF's **O**pen **C**loud **C**omputing **I**nterface
- abstracts platform-specific *CMF* concepts
- authN/authZ with *VOMS*, accessible through
 - raw protocol → *OCCL*
 - libraries → *ruby, java, python*
 - clients → *shell/cmdline*
 - brokers → *CSG, SlipStream, COMPSs, ...*

(Block) Storage Management

- block storage represents additional disks for compute instances
- tightly coupled with *Compute Management*, additional resource for the same use cases

Ephemeral Storage:

- “scratch” space, automatically provisioned
- tied to the life-cycle of a compute instance
→ **non-persistent**

Persistent Storage:

- storage space for long-lived data, input data or results
- life-cycle separate from the life-cycle of a compute instance
→ **persistent**

Basic Capabilities

- provisioning a persistent volume
- attaching a volume to a compute instance
- detaching a volume from a compute instance
- deprovisioning a persistent volume

Technical Details the same as for Compute Management.

Network Management

- rudimentary network management capabilities
- similar to *floating* public IP addresses
- advanced features in development right now

Basic Capabilities

- attaching a pre-existing public network to a compute instance
- detaching a pre-existing public network from a compute instance

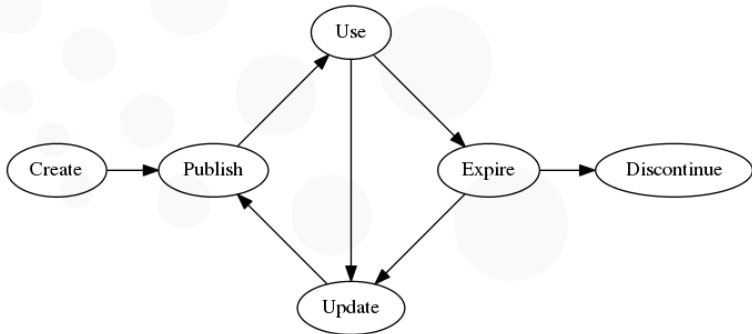
Technical Details the same as for Compute Management.

Virtual Appliance Management

Virtual Appliance

- a set of one or more virtual resource descriptors
- in the cloud context, virtual resource \rightarrow VA
- metadata & binary data (disks) of included appliances
- pre-installed and *partially* configured software
- simplifies and speeds up resource deployment for users

VA Life-cycle



The EGI AppDB

- store metadata and manage your VA's life-cycle
- does not store VA files, always referenced by location
- tracks availability of VAs on sites
- provides details for *Compute Management* tools

Workflow Example

Simplified Example

1. create a *VA* and register it in *AppDB*
2. ask a *VO Manager* to endorse it for your *VO*
3. use *AppDB* to check availability on sites
4. use *Compute Management* to provision compute instances
 - include configuration with contextualization
 - attach public IP addresses with *Network Management*
 - attach storage for data with *Storage Management*
5. perform computation, transfer results
6. deprovision allocated resources (*compute* and *storage*)

Live Demo – The AppDB

Quick AppDB Summary

Example VA: <https://goo.gl/brvCXS>

Register VA: <https://goo.gl/2cCCjl>

Populate VA: <https://goo.gl/YikgZN>

Notify VO(s): <https://goo.gl/OxUTz7>

– That's All Folks! –

...

Do you have any questions?

- ask **NOW!**
- ask us directly at parak@cesnet.cz or gergely.sipos@egi.eu
- send your questions to ucst@egi.eu