**VM Management – raw notes**

chair: Alexander Papaspyrou
Questions:

* are there standards we should be using?
* are there best practice we should adopt?
* what software is available at the market?
* what is the maturity of the software?
* what is the availability of the software?
* what are the priorities?
* what are the gaps, issues, and concerns?
* what work is needed to remove these?

Discussion items:

* specify bandwidth for a vm - for data-intensive applications
* expire and revocation of images - probably out of scope
* trust/endorsement/approval policy and model (certainly out of scope)
* authentication mechanisms (preferably X, 509v3 or VOMS proxy)
* bulk operations

MR: grid appliances and other appliances (community) - differences, relation to one another

* differences
	+ grid appliances is the traditional middleware stack packages as VM
	+ community provided VMs
* differences from VM management?
	+ in general no; maybe on regularity of updates

back to what is VM management?

* at what level?
	+ someone needs to address policy issues, (e.g. you can run a vm for a specific time) - not here, but a recommendation.

Requirements:

* HIGH: parameters to instantiate a single VM
	+ bandwidth for a VM (for data-intensive application), network type memory, running time, disk space, IP address, # cores, CPU, firewall rules, running time
* LOW: SLA: exclusive usage of physical machine, requested uptime
* MEDIUM: Instantiate *n* VMs as cluster
	+ shared disk space, connectivity
	+ dependencies between VMs
		- e.g. deploy one VM implies deploying other VMs
		- e.g. identified by a tag
		- provisiong VMs should be focused on the infrastructure level, anything further down is the problem for the communities
* Leads to two levels of deployment scenarios

LOW: Do we need advanced management  (advanced reservation?) HIGH: in a testing environment

* maybe nothing now, or more for the future
* VO and site level needs to be able to schedule
	+ load balancing and fair share
* does the interface need scheduling functionalities?
	+ need it internally, but via the interface?
* NB: if i give you a VM, we must be able to ask, how long will you have it?
	+ From request to instantiation - for single and bulk submission
* i need 500, can i even get 500, tell me, cause if not, i will go to another site where i can - a capability we dont have at the moment - good time to introduce this.
* in a federated scenario, makes sense

what kind of VMs are we providing

* those in batch system mode
* or on-demand
* Jeff: they need to be there, cause that is what the new users will have had on their computers before.

what is the difference between starting a process and starting a VM

* should be faster using virtualisation
* redhat provides features to tune process resource consumption
* processes can be checkpointed, migrated, …

List general requirements now, then prioritize later.

MED: expiry and revocation of images

* revocation can be done now by policy decision at the site - covered by EGI scenario 6

trusting images, is similar to trusting physical machine

* restart

policies will be different throughout the sites

* 1.) can only start golden images, then limit the run time, restart from only the original trusted image
* 2.) sites are more trusted to let user restart vms from saved snapshots of images

need to have a notification of bad images
balance risk and usability - we have mechanisms in place

* so we need formal information to differentiate between the two.

HIGH: state view from the VM Mgt interface of what is running on the system

* A user wants to know which VMs are running on a certain systems
* A sys admin wants to know who is running what

MED: a user can request a snapshot of its running VM the API if enabled by the site

HIGH: authentication mechanisms

* perferably x.509v3 or VOMS proxy)
* evaluate SAML/Shibboleth, EDUGain
* 2 level of authenication (out of scope of this meeting)
	+ super-users starting and stopping VMs at a site
	+ users of services running in a VM
* the decision of which hypervisors should be supported by sites is a policy decision
* VM mgt is agnostic to image format (supported hypervisors), but expose which hypervisors are supported.

Standards for VM Mgt

* std for vm mgt
	+ occi for vm mgt (create/start/stop)
	+ ovf for vm format
		- occi and ovf need integration
	+ moving vm between sites
		- cdmi can be a possible solution
	+ connecting OCCI/OVF/CDMI
* ec2? maybe higher level
	+ at the moment it is the API on the resource provider side
	+ messaging (out of scope)
	+ cloud from dmtf, similar goal of occi
* maturity?
	+ tcloud siliar to occi
		- stratuslab has one implementation
	+ CDMI: reference implmentation + SARA
	+ Move rest til tomorrow for other implementations
* standards of network

state of VM Mgt Implementations

* the requirements should be on the interfaces; given this
* what vm management solutinos can we consider?
	+ opennebula, openstack, nimbus, eucalyptus, batch schedulers (in the area of user jobs as VM), Platform ISF, VMware

\*\*Redhat has just announced its cloud architecture
differentiate between batch/HTP VMs and on-demand VMs