Helix Nebula
Proof of Concepts results

A view from the Supplier side

Frascati, January 2013

Your business technologists. **Powering progress**

Differing technologies and interfaces deployed

	Cloud stack	Availability	Hypervisor	API	Networking
Atos	StratusLab	Custom	KVM	OpenNebula	Specific
CloudSigma	Proprietary	General	KVM	Proprietary	Open
Interoute	Abiquo	General	KVM/ESX	n/a	Open
T-Systems	Zimory/ vCloud	Custom	VMware	Zimory	Specific

- Cloud is a rapidly-merging phenomenon, with few real standards as yet
  - so it is not surprising that the implementations differ
  - but organisations such as the ODCA and others are working on that
- So far, all customer interfaces were different

Helix Nebula Proofs of Concept
Supplier perspective

- The technology works: required workloads from participating scientific organisations can be successfully deployed in cloud environments
- The Demand sides each had slightly different requirements, and Supply-side clouds all differ significantly in implementation, even those sharing common hypervisors
  - so each user-supplier case was effectively a bespoke deployment
- Software requirements ("legacy") from the demand side can be complex and are critical to understanding POC requirements and ensuring a successful outcome
- A common API is desired and would significantly streamline driving deployments across multiple clouds
- The ability to move VM images between clouds and potentially convert VM formats is needed for better cross cloud deployment
- Differences between private and public deployment models on the supply side are important and influence various aspects of PoC's directly
  - i.e. WAN, open Internet or VPN access
- Networking between demand-side institutions and supply-side providers, and between multiple supply-side providers, is an important success criteria
- Areas such as service structure/architecture require further exploration

Proof of Concepts were successful

- All PoC environments completed successfully but at least one supplier
- Clouds can deliver the facilities required to fulfil the scientific requirements
  - starting at the IaaS level
- Inter-Supplier communication (open and often) was crucial to success
- We now/then need(ed) to move on to improving the overall experience
  - making the multi-supplier environment more coherent
  - performance optimisation and price/performance not yet considered

- Points of learning for further addressing
  - common interfaces: e.g. unified API
  - server image management and conversion, especially to cope with bespoke/ legacy software usage
  - inter-cloud workload assignment and scaling
  - networking: bandwidth and interfaces, charging algorithms
  - breaking the demand-supply scaling “chicken and egg” syndrome
  - common charging and billing mechanisms not yet in sight