Analysing SAGA as the Access Layer for OCCI backed Cloud Infrastructures

Shantenu Jha, Andre Merzky, Thijs Metsch
Outline

- Motivation
- Use Cases
- Standards based approach to UC scenarios
Motivation

• lessons from the Grid experience SHOULD be transferred as infrastructure evolve...
• … and extend to the Cloud paradigm.
• Complex Systems fail for complex reasons
• Complex Systems fail for simple reasons
• Standards simplify systems
• Standards support interop
Why SAGA + OCCI?

- This is an OGF workshop! ;-)  
- We are biased  
- Note: this is one possible approach to a standards stack for applications to use IaaS
Outline

• Motivation: interoperation

• Use Case

• Standards based approach to support Use Case

• Discuss!
Motivation: Interop

- System Level Interop:
  - Services communicate, exchange, share
  - Example: http, ftp, dns, posix
  - 'Federation'
  - mostly standards based (de-facto / de-jure)

- Application Level Interop:
  - application can use different systems concurrently, w/o system level support
  - example: MPI, blast
  - often ad-hoc, portability required
Motivation: Interop

• SAGA
  • standards based Application Level Interop
  • stable syntax and semantics on client side
  • portability of programming abstractions

• OCCI
  • standards based System Level Interop
  • stable syntax and semantics on servicer side
  • portability of programming abstractions

• SAGA and OCCI support different domains!
Use Case

• Ensemble based MD Simulations
Use Case Challenges

- O(10,000) simulations per ensemble
  - 1 .. 1024 cores per simulation
  - running min-hrs-days
  - loosely coupled
  - dump O(1) MB/min

- Challenges:
  - job startup and management
  - data tracking and management
  - scale
  - interop
Approach to support UC
Summary

• Interoperable Standards Stack is *easily* possible

• We will be implementing that stack

• Other stacks are very viable
Summary