September 19, 2017

# The Role of Standards for Risk Reduction and Inter-operation in XSEDE

Andrew Grimshaw

XSEDE

Extreme Science and Engineering
Discovery Environment

“Less is more.”
—Jerry Brown

“Perfection is achieved not when there is nothing more to add, but when there is nothing left to take away.”
—Antoine de Saint-Exupery

“Give me simple abstractions and make them work reliably”
— Kent Blackburn

# Takeaway message

- The use of standards permits XSEDE to interoperate with other infrastructures, reduces risks including vendor lock-in, and allows us to focus on higher level capabilities and less on the mundane

# What we mean by architecture

- Architecture defines the XSEDE system’s components and how they interact
  - each component is motivated by one or more requirements
  - each component is defined in terms of required capabilities: interfaces and qualities of service
- Equally important is the process by which we revise the architecture over time
  - key point: driven by new or revised requirements

# System Architecture

- What is a system architecture?
  - Set of design principles
  - A definition of the basic components
  - A definition of how the components refer to one another and interact in order to meet requirements
  - An abstraction on top of the underlying components

# Why Open Standards?

- Risk reduction
- Best-of-breed mix-and-match
- Allows innovation/competition at more interesting layers
- Facilitates interoperation with other infrastructures

# XSEDE Services Layer
# Simple services combined in many ways

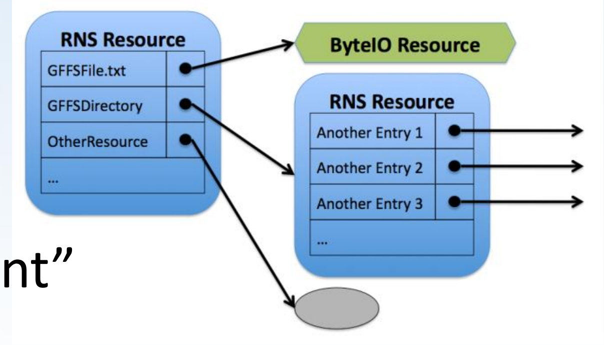
# Examples – not an complete list

- **Resource Namespace Service 1.1**
- **OGSA Basic Execution Service**
- OGSA WSRF BP – metadata and notification
- OGSA-ByteIO
- GridFTP
- WS Trust Secure Token Services
- WSI BSP for transport of credentials
- …

# OGSA Basic Execution Services

- Basic idea
  - Virtualizes a compute resource: host, queue on a supercomputer, a dynamic set of virtual machine images, a set of other BESs, …
  - Publishes characteristics of the compute resource: OS, CPU count, memory, etc.
  - accept job description in JSDL and create an activity
- Many interoperable implementations
  - GridSam, Genesis II, UNICORE 6, Globus did one

# Resource Namespace Service



- Used to create a global shared namespace
  - Analogous to Unix directories, entries "point" to grid resources
  - Most resources can be named using an intuitive path
    - /home/grimshaw/myfile or /bin/bio/blastp
  - Directories organized resources based on human relevant semantics
  - Namespace can be mapped into local file system, and grid resources accessed as if local

# Most everything can have an RNS path

- Files
- Directories
- Web pages
- Execution services – queues and BESs
- Jobs
- Applications
- Archives
- Instruments
- Relational Databases

# Why Open Standards?

- Risk reduction
- Best-of-breed mix-and-match
- Allows innovation/competition at more interesting layers
- Facilitates interoperation with other infrastructures

Our reach will forever
exceed our grasp, but,
in stretching our horizon,
we forever improve our world.

XSEDE

Extreme Science and Engineering
Discovery Environment