Scaling eScience Impact

Ian Foster

Computation Institute
Argonne National Lab & University of Chicago
The Big Questions

Nature of the universe

Consciousness

Life & death

Future of the planet

Nature of the universe

Consciousness
How Do We Answer Them?

Empirical

Theory

Simulation

Data

<0 1700 1950 1990
The Same is True of Smaller Questions

● Designing new chemical catalysts
● Selling advertising
● Creating entertainment
● Finding parking
eScience: Science in an Exponential World

- “Large-scale science carried out through distributed collaborations—often leveraging access to large-scale data & computing” [John Taylor, UK EPSRC]

- “When brute force doesn’t work anymore” [Alex Szalay]

- Science accelerated, decentralized, integrated, & democratized via the Internet
Grid: An Enabler of eScience

The dubious electrical power grid analogy

Must we buy (or travel to) a power source?

Or can we ship power to where we want to work?

Enable on-demand access to, and integration of, diverse resources & services, regardless of location
1st Generation Grids

Focus on aggregation of many resources for massively (data-)parallel applications

EGEE

Open Science Grid
Second-Generation Grids

- Empower many more users by enabling on-demand access to **services**
- Science gateways (TeraGrid)
- Service oriented science
- Or, “Science 2.0”

“Service-Oriented Science”, *Science*, 2005
“Web 2.0”

- Software as services
  - Data- & computation-rich network services
- Services as platforms
  - Easy composition of services to create new capabilities ("mashups")—that themselves may be made accessible as new services
- Enabled by massive infrastructure buildout
  - Google projected to spend $1.5B on computers, networks, and real estate in 2006
  - Many others are spending substantially
- Paid for by advertising
Science 2.0: E.g., Virtual Observatories

User → Discovery tools → Gateway → Analysis tools → Data Archives

Figure: S. G. Djorgovski
People **create** services (data or functions) ... which I **discover** (& decide whether to use) ... & **compose** to create a new function ... & then **publish** as a new service.

→ I find “someone else” to **host** services, so I don’t have to become an expert in operating services & computers!

→ I hope that this “someone else” can **manage** security, reliability, scalability, ...
The Importance of “Hosting” and “Management”

Tell me about this star

Tell me about these 20K stars

Support 1000s of users

E.g., Sloan Digital Sky Survey, ~10 TB; others much bigger
Load Comes from All Over ...

A few hours of Globus.org access, early one morning ...
Skyserver Sessions
(Thanks to Alex Szalay)
The Two Dimensions of Service Oriented Science

- **Decompose** across network
  - Clients **integrate** dynamically
    - Select & compose services
    - Select “best of breed” providers
    - Publish result as new services

- **Decouple** resource & service providers

Fig: S. G. Djorgovski
Hosting & Management: Application Hosting Services

Application providers

AppIn Code

Application deployment

Users

Application client

Admins

AHS management

Policy management

Authorization

Persistence

PDP

Hosting Service

AppIn Code

Resource Provider

Provisioning
Web Services Resource Framework in a Nutshell

- **Service**
- **State representation**
  - Resource
  - Resource Property
- **State identification**
  - Endpoint Reference
- **State Interfaces**
  - GetRP, QueryRPs, GetMultipleRPs, SetRP
- **Lifetime Interfaces**
  - SetTerminationTime
  - ImmediateDestruction
- **Notification Interfaces**
  - Subscribe
  - Notify
- **ServiceGroups**
Apache Tomcat
Globus Toolkit
Web Services Container

- Service Container
- Globus Toolkit Web Services Container
- RPs Resource Service
- GetRP
- GetMultRPs
- SetRP
- QueryRPs
- Subscribe
- SetTermTime
- Destroy
- EPR
- ResourceHome
- RPs
- WorkManager
- DB Conn Pool
- JNDI Directory
- PIP
- PDP
- Security
- Authorization
- State
- Management
- Persistence

GT4 Web Services Container

Globus Toolkit Version 4: Software for Service-Oriented Systems, LNCS 3779, 2-13, 2005
**RAVE**

- **Remote Application Virtualization Environment**
- Builds on Introduce
  - Define service
  - Create skeleton
  - Discover types
  - Add operations
  - Configure security
- Wrap arbitrary executables

Ravi Madduri et al., Argonne/U.Chicago & Ohio State University
Defining Community: Membership and Laws

- Identify VO participants and roles
  - For people and services
- Specify and control actions of members
  - Empower members → delegation
  - Enforce restrictions → federate policy

Access granted by community to user
Policy of site to community
Effective Access
Site admission-control policies
(1) Use-conditions are Imposed by Independent Stakeholders

Stakeholders provide and maintain and use-conditions

Memo
- exclude “bad” countries
- Memo
- include all LBNL staff and guests
- Memo
- must have X-ray safety training
- Memo
- must have approved protocol
- Memo
- must be group member

(2) Users have Attributes that Match the Use-conditions

Attribute certifiers that are trusted by the stakeholders

(3) Access is Granted after Verifying that User Attributes Match the Required Use-Conditions

Access control

1 Societal Access Control Model

SAML
Security Services for VO Policy

- **Attribute Authority (ATA)**
  - Issue signed attribute assertions (incl. identity, delegation & mapping)

- **Authorization Authority (AZA)**
  - Decisions based on assertions & policy

- **Use with message- or transport-level security**
Globus Authorization Framework

- VOMS
- Shibboleth
- LDAP
- PDP
- Attributes
- Authorization Decision
- GT4 Client
- PIP
- GT4 Server
- PDP
- PERMIS
Service-Oriented Science & Cancer Biology

caBIG: sharing of infrastructure, applications, and data.
Cancer Bioinformatics Grid

Grid Data Service

caArray

Grid Portal

Microarray

Analytical Service

Tool 1

Tool 2

NCICB

Grid-Enabled Client

Research Center

Gene Database

Protein Database

Tool 3

Tool 4

Grid Services Infrastructure (Metadata, Registry, Query, Invocation, Security, etc.)

Image

Research Center

Tool 2

Tool 3

All Globus-based, by the way ...
Composing Services: E.g., BPEL Workflow System

See also Kepler & Taverna

Provisioning: Astro Portal Stacking Service

- **Purpose**
  - On-demand “stacks” of random locations within ~10TB dataset

- **Challenge**
  - Rapid access to 10-10K “random” files
  - Time-varying load

- **Solution**
  - Dynamic acquisition of compute, storage

Joint work with Ioan Raicu & Alex Szalay
Dynamic Provisioning: Swift Architecture

Specification
- Abstract computation
- SwiftScript Compiler
- Virtual Data Catalog
- SwiftScript

Scheduling
- Execution Engine (Karajan w/ Swift Runtime)
  - Swift runtime callouts
  - Status reporting

Execution
- Virtual Node(s)
- Provenance data
  - launcher
  - App F1
  - file1
  - Provenance data
  - launcher
  - App F2
  - file2
  - Provenance data
  - launcher
  - App F1
  - file3

Provisioning
- Falkon Resource Provisioner
- Amazon EC2

Yong Zhao, Mihael Hatigan, Ioan Raicu, Mike Wilde, Ben Clifford
Synthetic Benchmark

- 18 Stages
- 1000 tasks
- 17,820 CPU seconds
- 1,260 total time on 32 machines

Ioan Raicu & Yong Zhao, U.Chicago
Release after 15 Seconds Idle

- Allocated
- Registered
- Active

# of Workers

Time (sec)

0 580.386 1156.853 1735.62
Release after 180 Seconds Idle
Montage

Yong Zhao and Ioan Raicu, U.Chicago
BI server applications started and decommissioned by a Grid-enabled dispatcher
Computation as a First-Class Entity

- Capture information about relationships among
  - Data (varying locations and representations)
  - Programs (& inputs, outputs, constraints)
  - Computations (& execution environments)

- Apply this information to:
  - Discovery of data and programs
  - Computation management
  - Provenance
  - Planning, scheduling, performance optimization

A Virtual Data System for Representing, Querying & Automating Data Derivation [SSDBM02]
Example: fMRI Analysis

First Provenance Challenge, http://twiki.ipaw.info/ [CCPE06]
Query Examples

- **Query by procedure signature**
  - Show procedures that have inputs of type `subjectImage` and output types of `warp`

- **Query by actual arguments**
  - Show `align_warp` calls (including all arguments), with argument `model=rigid`

- **Query by annotation**
  - List anonymized subject images for young subjects:
    - Find datasets of type `subjectImage`, annotated with `privacy=anonymized` and `subjectType=young`

- **Basic lineage graph queries**
  - Find all datasets derived from dataset ‘5a’

- **Graph pattern matching**
  - Show me all output datasets of `softmean` calls that were aligned with `model=affine`
Welcome

This is the new home Globus software development; it is still under construction. The current status of our efforts to build this environment can be found on this page. Comments regarding this site can be sent to info@globus.org. Thank you for your interest in Globus development!

Globus was first established as an open source software project in 1996. Since that time, the Globus development team has expanded from a few individuals to a distributed, international community. In response to this growth, the Globus community (the “Globus Alliance”) established in October 2005 a new source code development infrastructure and meritocratic governance model, which together make the process by which a developer joins the Globus community both easier and more transparent.

The Globus governance model and infrastructure are based on those of Apache Jakarta. In brief, the governance model places control over each individual software component (project) in the hands of its most active and respected contributors (committer) with a Globus Management Committee (GMC) providing overall guidance and conflict resolution. The infrastructure comprises repositories, email lists, Wikis, and bug trackers configured to support per-project community access and management.

For more information, see:

- The Globus Alliance Guidelines, which address various aspects of the Globus governance model and the Globus community.
- A description of the Globus Alliance Infrastructure.
- A list of current Globus projects.
Summary: Service-Oriented Science

People **create** services (data or functions) ... which I **discover** (& decide whether to use) ... & **compose** to create a new function ... & then **publish** as a new service.

→ I find “someone else” to **host** services, so I don’t have to become an expert in operating services & computers!

→ I hope that this “someone else” can **manage** security, reliability, scalability, ...

“Service-Oriented Science”, *Science*, 2005
People **create** services (data or functions) ... which I **discover** (& decide whether to use) ... & **compose** to create a new function ... & then **publish** as a new service.

**Profoundly revolutionary:**

- Accelerates the pace of enquiry
- Introduces a new notion of “result”
- Requires new reward structures, training, infrastructure

“Service-Oriented Science”, *Science*, 2005
Science 1.0 → Science 2.0

Megabytes & gigabytes → Terabytes & petabytes
Tarballs → Services
Journals → Wikis
Individuals → Communities
Community codes → Science gateways
Supercomputer centers → Campus & national grids...
Makefile → Workflow
Computational science → Science as computation
Mostly physical sciences → All sciences (& humanities)
1000s of computationalists → Millions of scientists
Government funded → Government funded
Thanks!

- DOE Office of Science
- NSF Office of Cyberinfrastructure
- Colleagues at Argonne, U.Chicago, USC/ISI, and elsewhere
- Many members of the German DGrid community
Service-Oriented Science Challenges

- A need for new **technologies, skills, & roles**
  - Creating, publishing, hosting, discovering, composing, archiving, explaining ... services

- A need for substantial **software development**
  - “30-80% of modern astronomy projects is software”—S. G. Djorgovski, Caltech

- A need for more & different **infrastructure**
  - Computers & networks to host services

- And certainly profound **research challenges**
  - In every part of the service & science lifecycle

For more information: [http://ianfoster.typepad.com](http://ianfoster.typepad.com)