Scientific Discovery through Advanced Computing

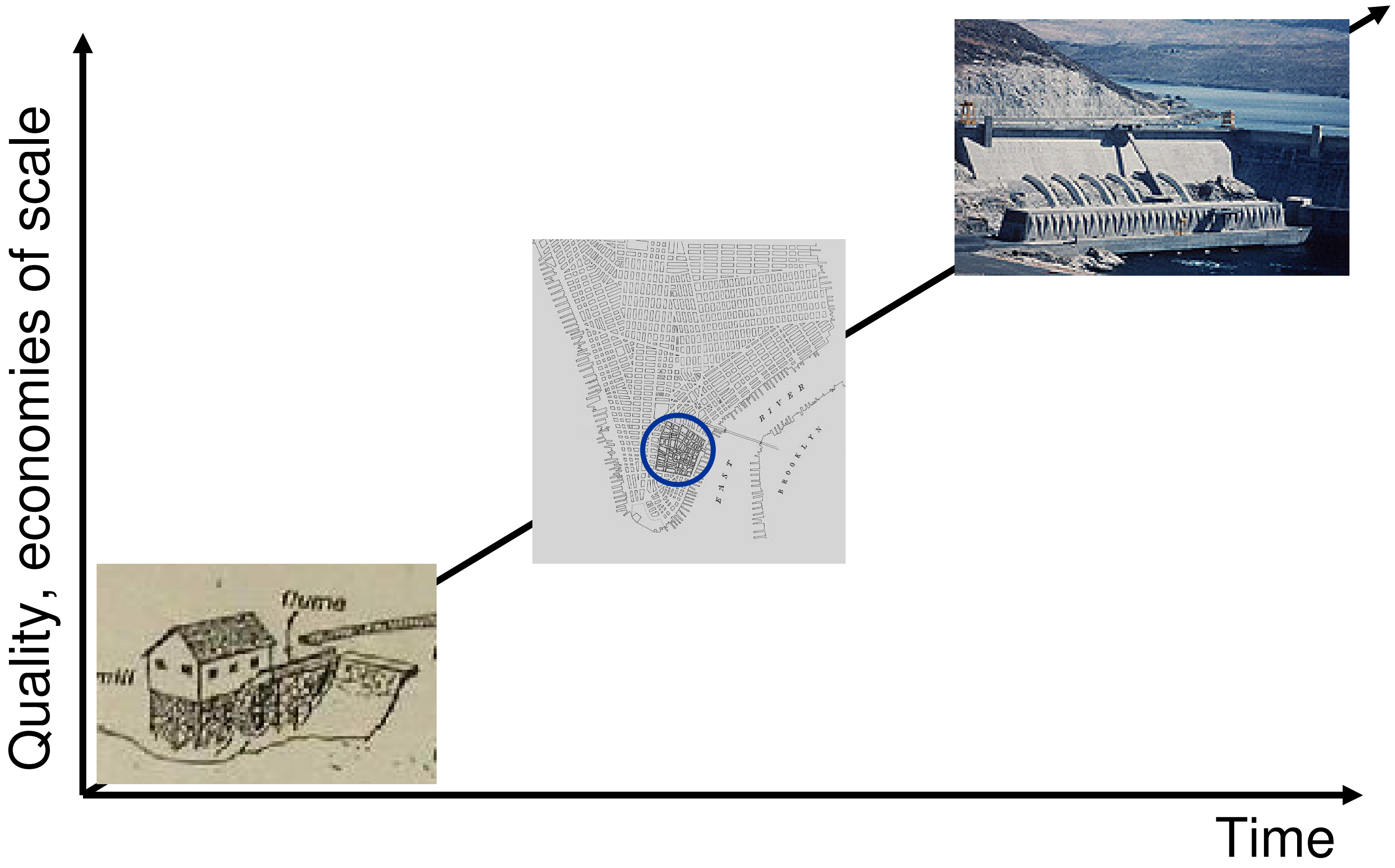
Grid

Ian Foster

Computation Institute

Argonne National Laboratory
University of Chicago

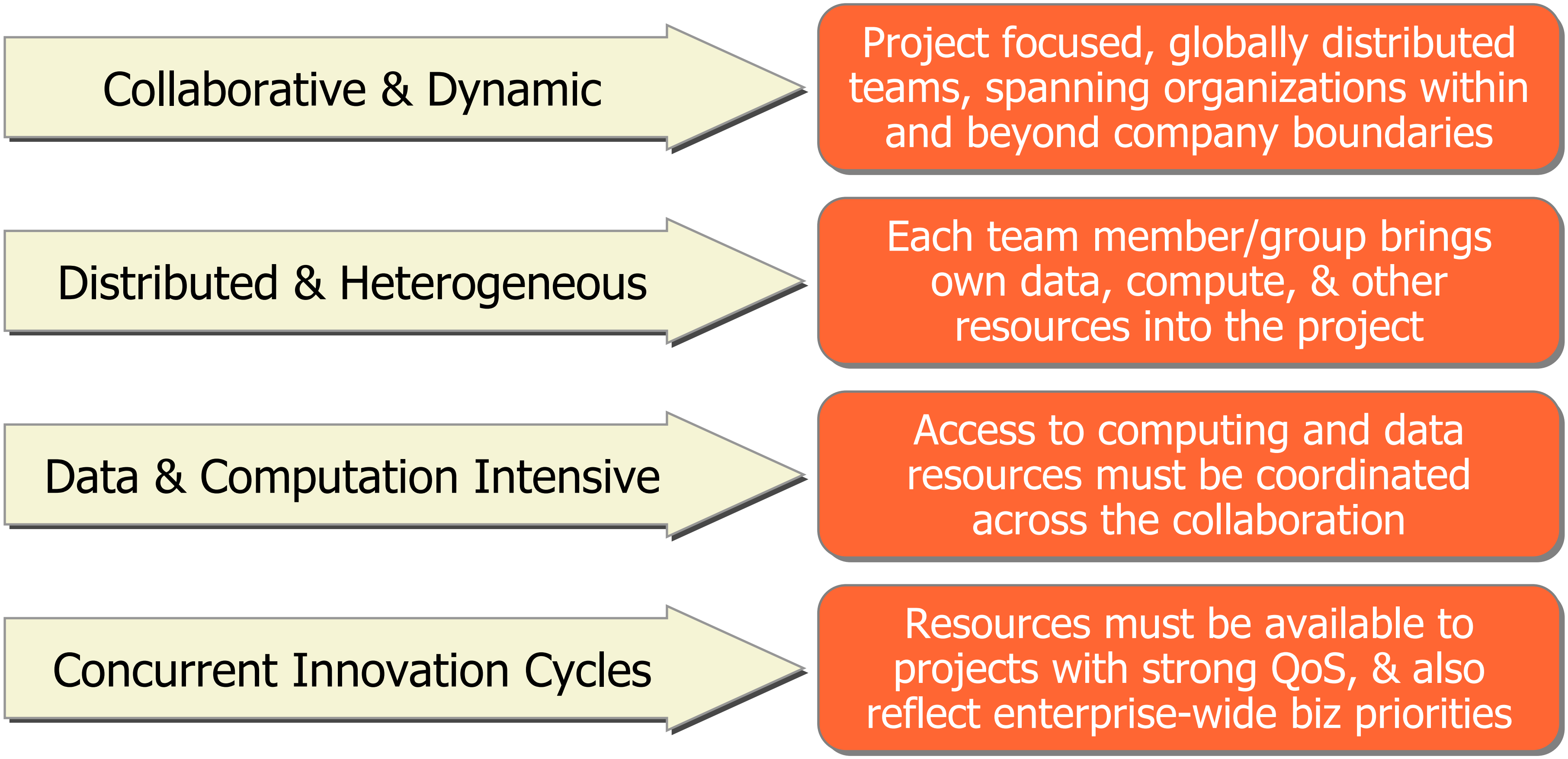
The (Power) Grid: On-Demand Access to Electricity



An Old Idea …

- “The time-sharing computer system can unite a group of investigators …. one can conceive of such a facility as an … intellectual public utility.”
  - Fernando Corbato and Robert Fano, 1966
- “We will perhaps see the spread of ‘computer utilities’, which, like present electric and telephone utilities, will service individual homes and offices across the country.”
  - Len Kleinrock, 1967

Why Grid? — The Changing Nature of Work



Collaborative & Dynamic → Project focused, globally distributed teams, spanning organizations within and beyond company boundaries

Distributed & Heterogeneous → Each team member/group brings own data, compute, & other resources into the project

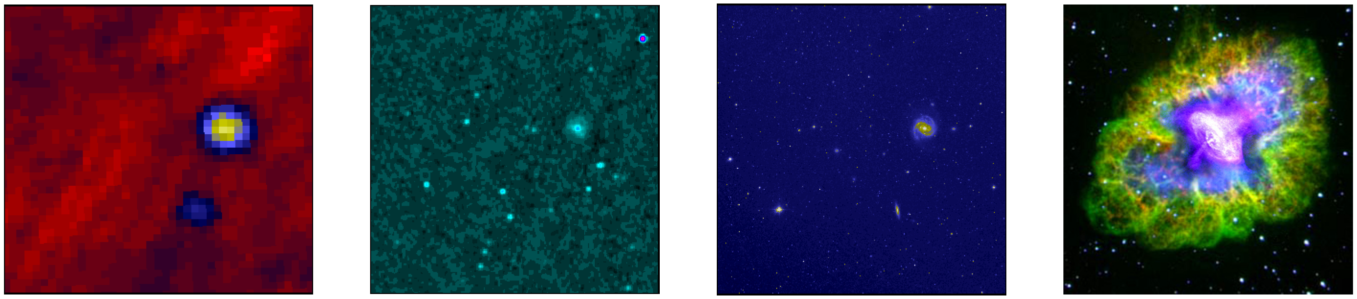
Data & Computation Intensive → Access to computing and data resources must be coordinated across the collaboration

Concurrent Innovation Cycles → Resources must be available to projects with strong QoS, & also reflect enterprise-wide biz priorities

IT must adapt to this new reality

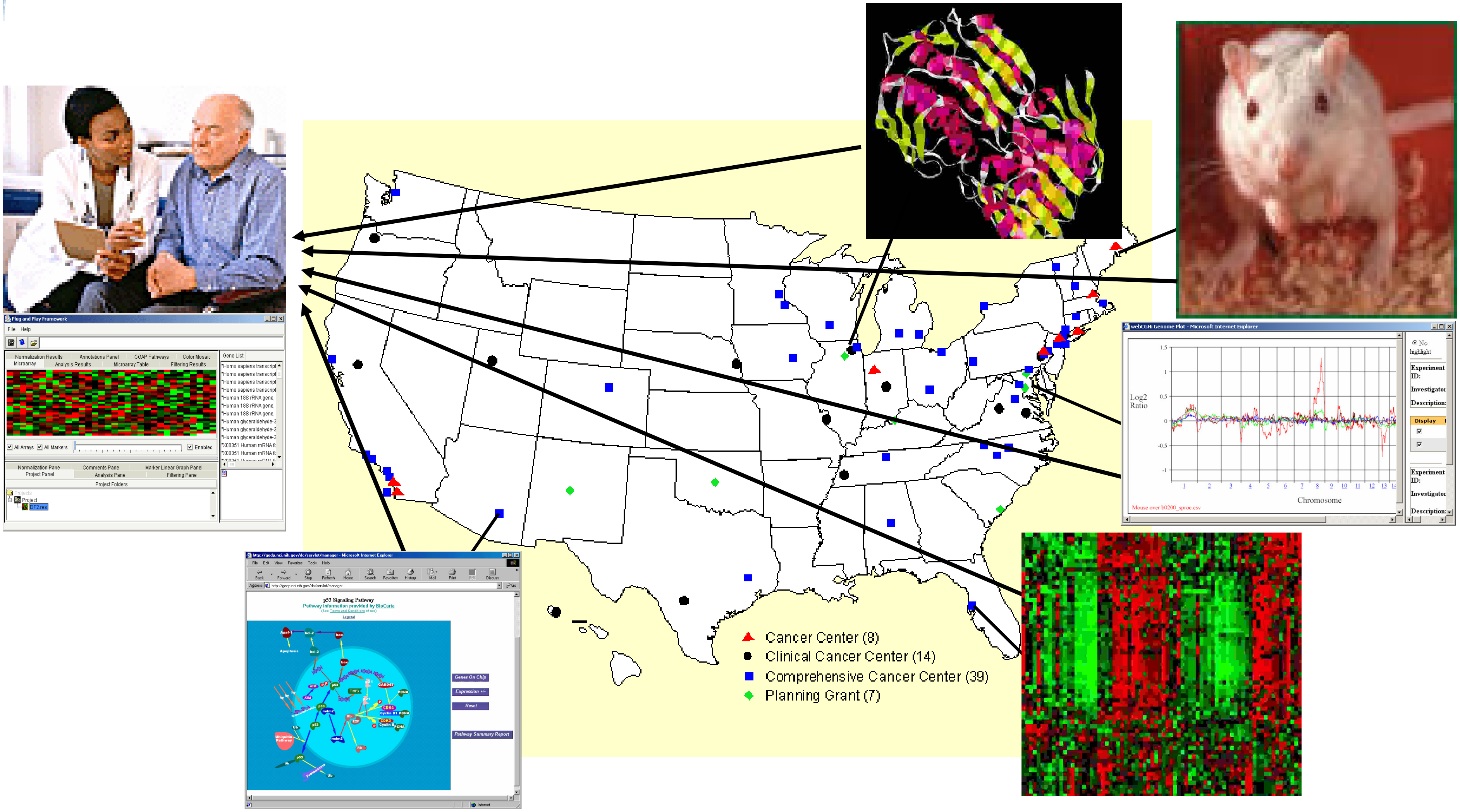
For Example: Digital Astronomy

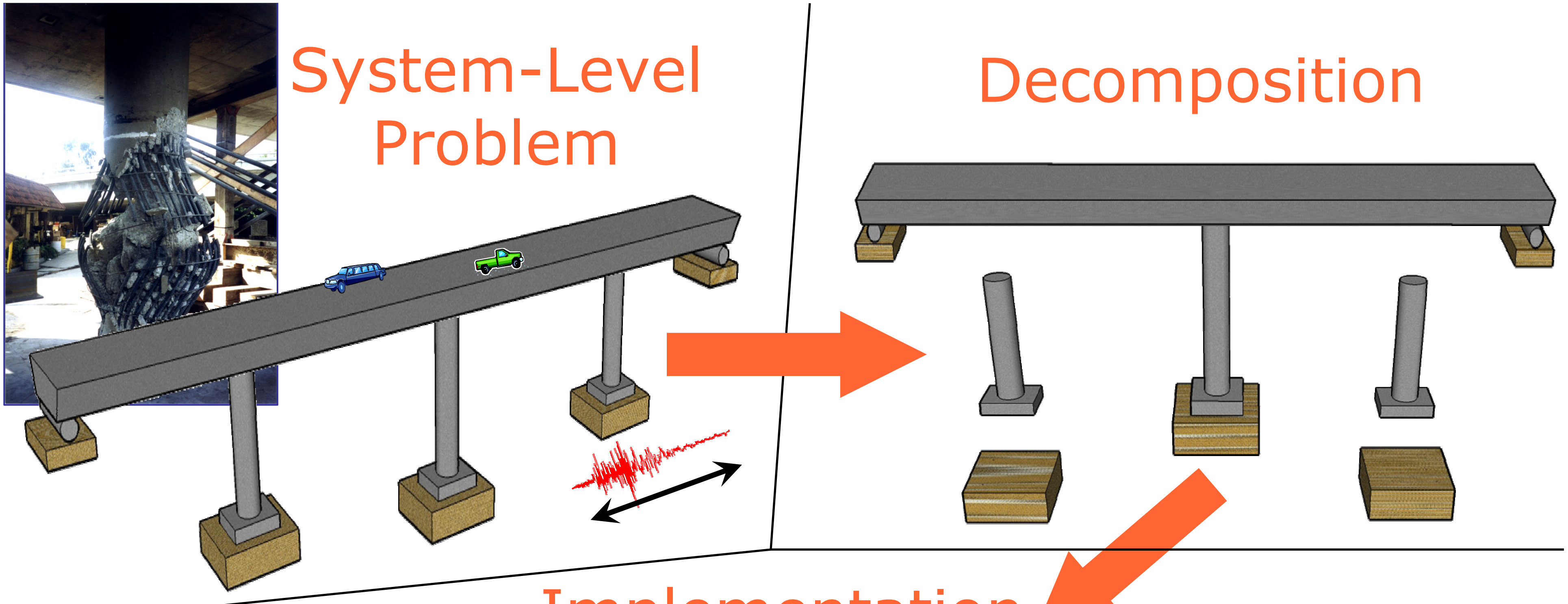
- Digital observatories provide online archives of data at different wavelengths



- Ask questions such as: what objects are visible in infrared but not visible spectrum?

For Example: Cancer Biology


Plug and Play Framework
p53 Signaling Pathway
Pathway information provided by BioCarta
webCGH: Genome Plot - Microsoft Internet Explorer
Log2 Ratio
Chromosome
Cancer Center (8)
Clinical Cancer Center (14)
Comprehensive Cancer Center (39)
Planning Grant (7)

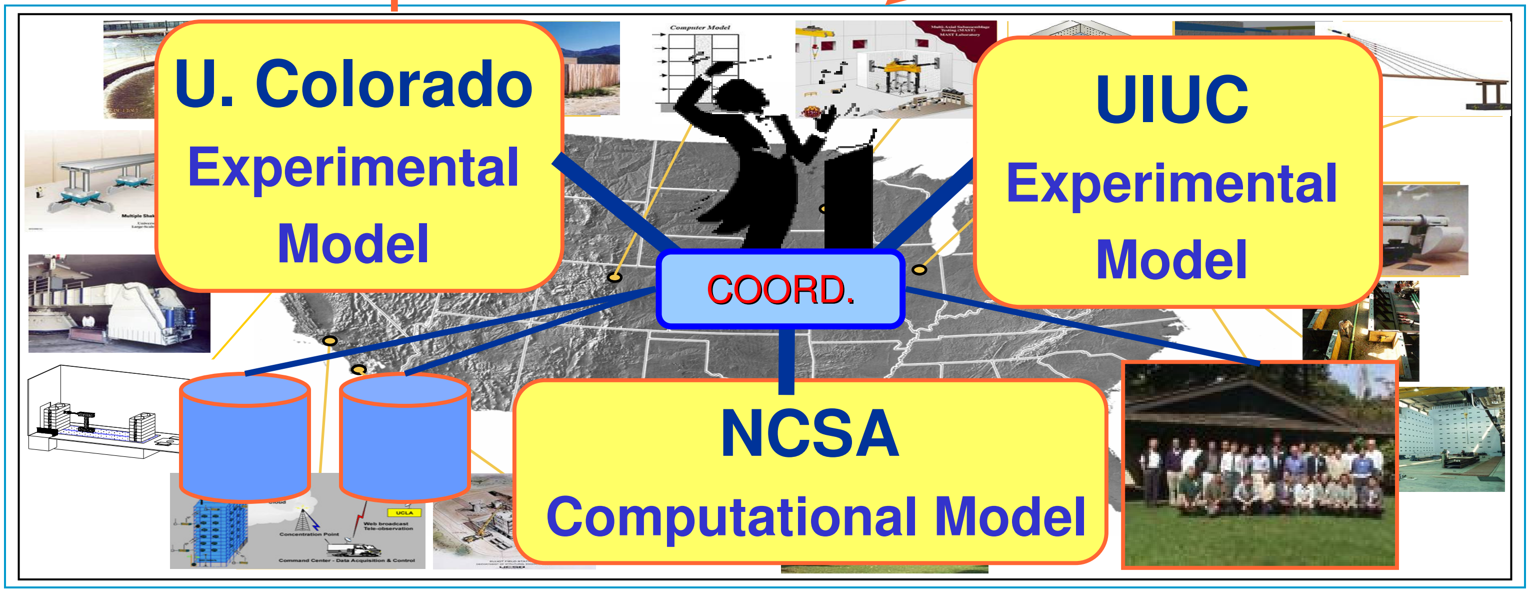


System-Level Problem

Decomposition

Implementation

Facilities
Computers
Storage
Networks
Services
Software
People

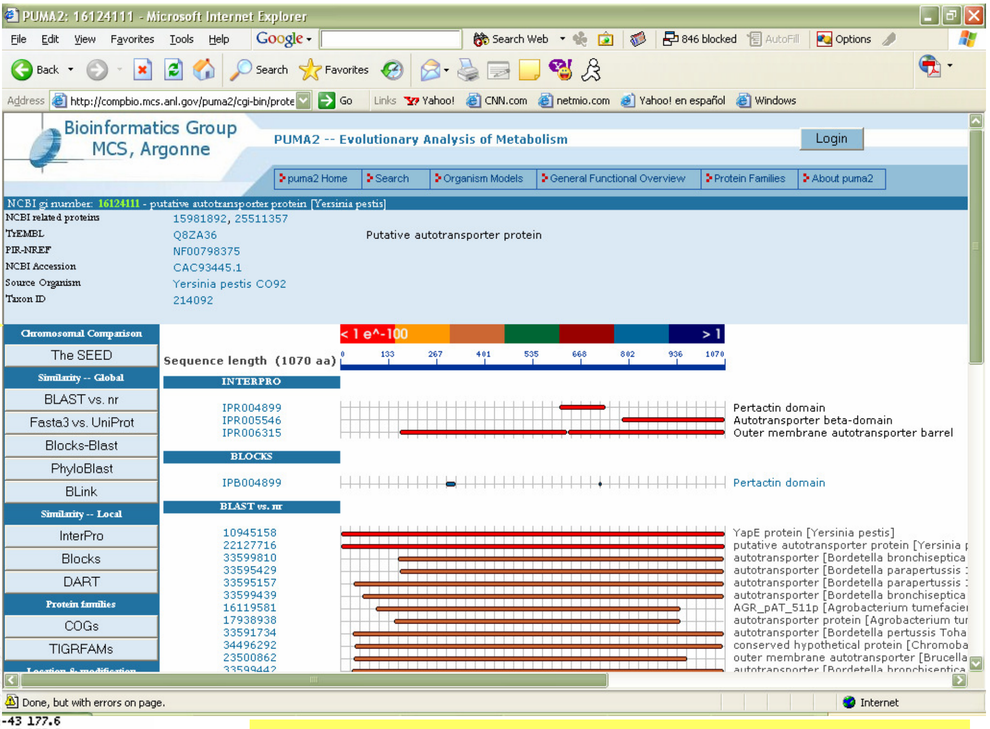


the globus alliance
www.globus.org

For Example: Bioinformatics

Public PUMA Knowledge Base

Information about proteins analyzed against ~2 million gene sequences



Bioinformatics Group
MCS, Argonne

PUMA2 -- Evolutionary Analysis of Metabolism

NCBI gi number: 16124111 - putative autotransporter protein [Yersinia pestis]

NCBI related proteins	15981892, 25511357	
TrEMBL	Q8ZA36	Putative autotransporter protein
PIR-NREF	NF00798375	
NCBI Accession	CAC93445.1	
Source Organism	Yersinia pestis CO92	
Taxon ID	214092	

Sequence length (1070 aa)

INTERPRO

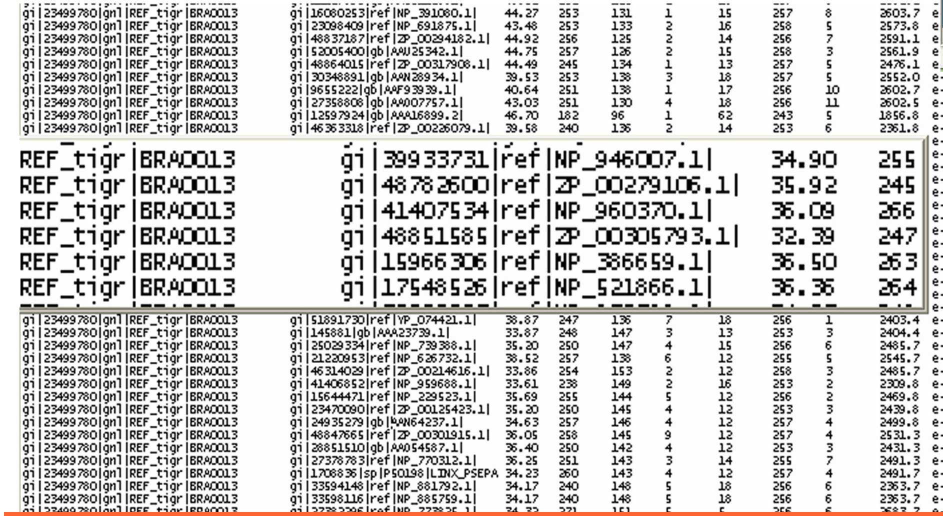
IPR004899	Pertactin domain
IPR005546	Autotransporter beta-domain
IPR006315	Outer membrane autotransporter barrel

BLOCKS

IPB004899	Pertactin domain

BLAST vs. nr

10945158	YapE protein [Yersinia pestis]
22127716	putative autotransporter protein [Yersinia p
33599810	autotransporter [Bordetella bronchiseptica
33595429	autotransporter [Bordetella parapertussis 1
33595157	autotransporter [Bordetella parapertussis 1
33599439	autotransporter [Bordetella bronchiseptica
16119581	AGR_pAT_511p [Agrobacterium tumefacien
17938938	autotransporter protein [Agrobacterium tur
33591734	autotransporter [Bordetella pertussis Toha
34496292	conserved hypothetical protein [Chromoba
23500862	outer membrane autotransporter [Brucella
33599442	autotransporter [Bordetella bronchiseptica



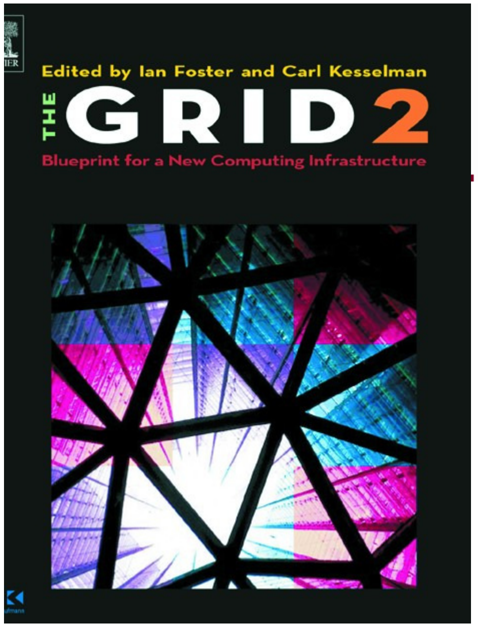
```
gi|23499780|gnl|REF_tigr|BRA0013     gi|16080253|ref|NP_391080.1|    44.27  253  131  1  15  257  8   2603.7  e
gi|23499780|gnl|REF_tigr|BRA0013     gi|23098409|ref|NP_691875.1|    43.48  253  133  2  16  258  5   2573.8  e
gi|23499780|gnl|REF_tigr|BRA0013     gi|48837187|ref|ZP_00294182.1|  44.92  256  125  2  14  256  7   2591.1  e
gi|23499780|gnl|REF_tigr|BRA0013     gi|52005400|gb|AAU25342.1|      44.75  257  126  2  15  258  3   2561.9  e
gi|23499780|gnl|REF_tigr|BRA0013     gi|48864015|ref|ZP_00317908.1|  44.49  245  134  1  13  257  5   2476.1  e
gi|23499780|gnl|REF_tigr|BRA0013     gi|30348891|gb|AAN28934.1|      39.53  253  138  3  18  257  5   2552.0  e-43 177.6
gi|23499780|gnl|REF_tigr|BRA0013     gi|9655222|gb|AAF93939.1|       40.64  251  138  1  17  256  10  2602.7  e-43 177.2
gi|23499780|gnl|REF_tigr|BRA0013     gi|27358808|gb|AAO07757.1|      43.03  251  130  4  18  256  11  2602.5  e-41 170.6
gi|23499780|gnl|REF_tigr|BRA0013     gi|12597924|gb|AAA16899.2|      46.70  182  96   1  62  243  5   1856.8  e-39 162.5
gi|23499780|gnl|REF_tigr|BRA0013     gi|46363318|ref|ZP_00226079.1|  39.58  240  136  2  14  253  6   2361.8  e-36 154.5

REF_tigr|BRA0013      gi|39933731|ref|NP_946007.1|     34.90   255
REF_tigr|BRA0013      gi|48782600|ref|ZP_00279106.1|   35.92   245
REF_tigr|BRA0013      gi|41407534|ref|NP_960370.1|     36.09   266
REF_tigr|BRA0013      gi|48851585|ref|ZP_00305793.1|   32.39   247
REF_tigr|BRA0013      gi|15966306|ref|NP_386659.1|     36.50   263
REF_tigr|BRA0013      gi|17548526|ref|NP_521866.1|     36.36   264

e-33 142.9
e-33 142.5
e-32 141.4
e-32 140.2
e-32 139.4
e-32 139.0
e-31 137.9
e-31 137.1
e-30 135.2
e-30 134.8
e-30 134.8
e-30 134.4
e-30 134.
e-30 134.

gi|23499780|gnl|REF_tigr|BRA0013     gi|51891730|ref|YP_074421.1|    38.87  247  136  7  18  256  1   2403.4  e-30 133.7
gi|23499780|gnl|REF_tigr|BRA0013     gi|145881|gb|AAA23739.1|        33.87  248  147  3  13  253  3   2404.4  e-30 133.3
gi|23499780|gnl|REF_tigr|BRA0013     gi|25029334|ref|NP_739388.1|    35.20  250  147  4  15  256  6   2485.7  e-30 132.9
gi|23499780|gnl|REF_tigr|BRA0013     gi|21220853|ref|NP_626732.1|    38.52  257  138  6  12  255  5   2545.7  e-30 132.9
gi|23499780|gnl|REF_tigr|BRA0013     gi|46314029|ref|ZP_00214616.1|  33.86  254  153  2  12  258  3   2485.7  e-30 132.9
gi|23499780|gnl|REF_tigr|BRA0013     gi|41406852|ref|NP_959688.1|    33.61  238  149  2  16  253  2   2308.8  e-30 132.1
gi|23499780|gnl|REF_tigr|BRA0013     gi|15644471|ref|NP_229523.1|    35.69  255  144  5  12  256  2   2469.8  e-30 132.1
gi|23499780|gnl|REF_tigr|BRA0013     gi|23470090|ref|ZP_00125423.1|  35.20  250  145  4  12  253  3   2439.8  e-30 132.1
gi|23499780|gnl|REF_tigr|BRA0013     gi|24935279|gb|AAN64237.1|      34.63  257  146  4  12  257  4   2499.8  e-30 132.1
gi|23499780|gnl|REF_tigr|BRA0013     gi|48847665|ref|ZP_00301915.1|  36.05  258  145  9  12  257  4   2531.3  e-29 131.7
gi|23499780|gnl|REF_tigr|BRA0013     gi|28851510|gb|AAO54587.1|      36.40  250  142  4  12  253  3   2431.3  e-29 131.7
gi|23499780|gnl|REF_tigr|BRA0013     gi|27378783|ref|NP_770312.1|    36.25  251  143  3  14  255  7   2491.3  e-29 131.7
gi|23499780|gnl|REF_tigr|BRA0013     gi|1708836|sp|P50198|LINX_PSEPA 34.23  260  143  4  12  257  4   2491.7  e-29 131.3
gi|23499780|gnl|REF_tigr|BRA0013     gi|33594148|ref|NP_881792.1|    34.17  240  148  5  18  256  6   2363.7  e-29 130.2
gi|23499780|gnl|REF_tigr|BRA0013     gi|33598116|ref|NP_885759.1|    34.17  240  148  5  18  256  6   2363.7  e-29 130.2
```

Back Office Analysis

Millions of BLAST, BLOCKS, etc., on OSG and TeraGrid

Natalia Maltsev et al., http://compbio.mcs.anl.gov/puma2

The Grid


Edited by Ian Foster and Carl Kesselman
THE GRID 2
Blueprint for a New Computing Infrastructure

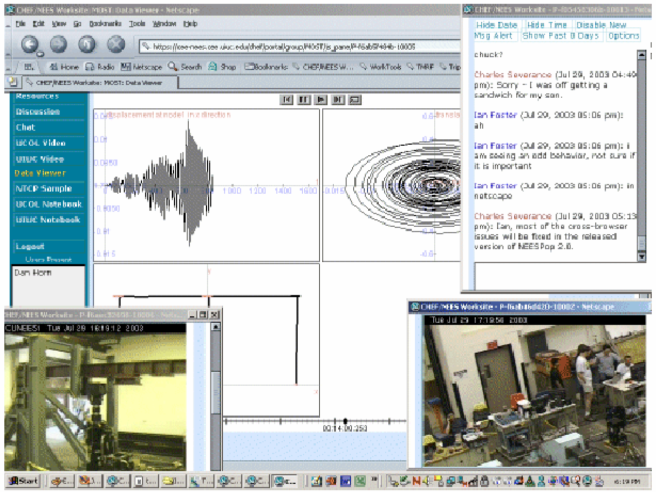
Enable "*coordinated resource sharing & problem solving in dynamic, multi-institutional virtual organizations.*"
(Source: **"The Anatomy of the Grid"**)

- Access to shared resources
  → Virtualization, allocation, management
- With predictable behaviors
  → Provisioning, quality of service
- In dynamic, heterogeneous environments
  → Standards-based interfaces and protocols

More Specifically, I May Want To …

- Create a service for use by my colleagues
- Manage who is allowed to access my service (or my experimental data or …)
- Ensure reliable & secure distribution of data from my lab to my partners
- Run 10,000 jobs on whatever computers I can get hold of
- Monitor the status of the different resources to which I have access

Underlying Problem: The Application-Infrastructure Gap



chuck?

Charles Severance (Jul 29, 2003 04:49 pm): Sorry - I was off getting a sandwich for my son.

Ian Foster (Jul 29, 2003 05:06 pm): ah

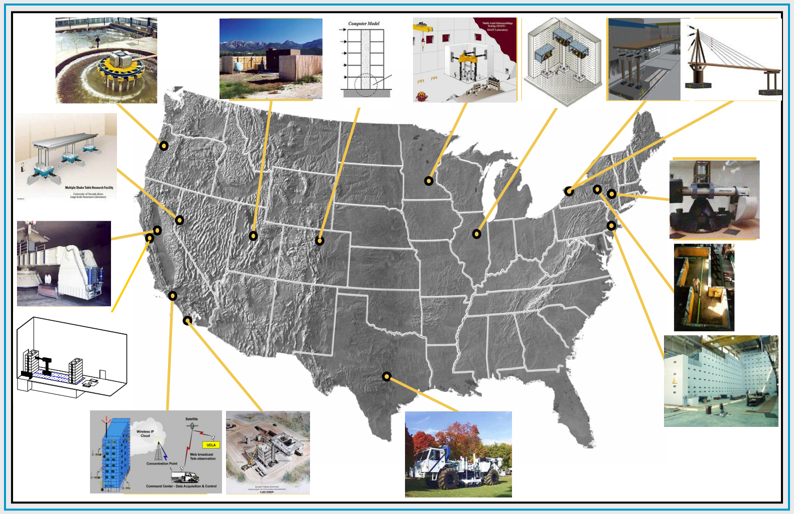
Ian Foster (Jul 29, 2003 05:06 pm): I am seeing an odd behavior, not sure if it is important

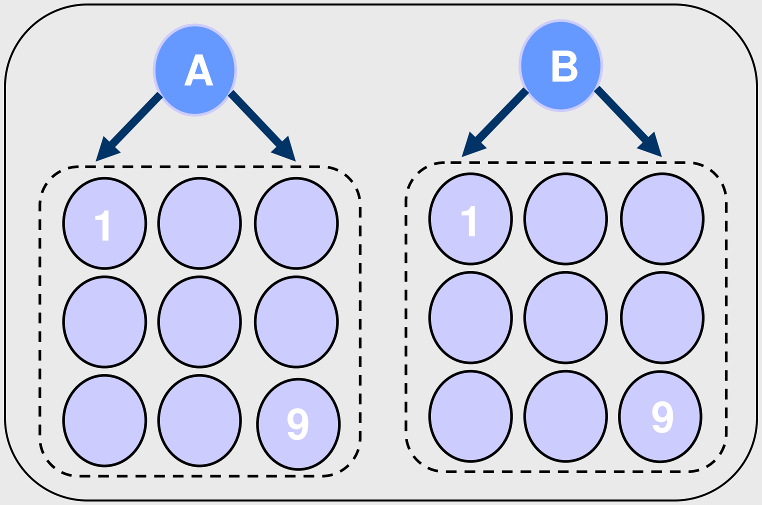
Ian Foster (Jul 29, 2003 05:06 pm): in netscape

Charles Severance (Jul 29, 2003 05:13 pm): Ian, most of the cross-browser issues will be fixed in the released version of NEESPop 2.0.

Dynamic and/or Distributed Applications

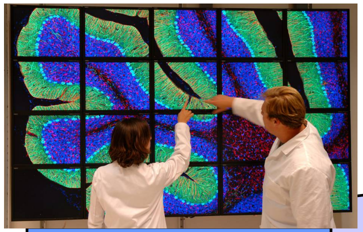
Shared Distributed Infrastructure

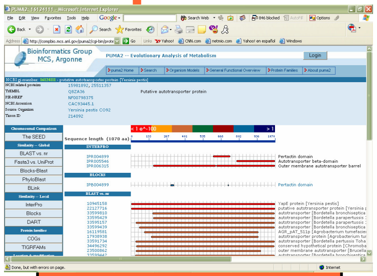




the globus alliance
www.globus.org

Bridging the Application-Resource Gap





PUMA2: 16124111 - Microsoft Internet Explorer

http://compbio.mcs.anl.gov/puma2/cgi-bin/prote

Bioinformatics Group
MCS, Argonne

PUMA2 -- Evolutionary Analysis of Metabolism

NCBI gi number: 16124111 - putative autotransporter protein [Yersinia pestis]

NCBI related proteins	15981892, 25511357	
TrEMBL	Q8ZA36	Putative autotransporter protein
PIR-NREF	NF00798375	
NCBI Accession	CAC93445.1	
Source Organism	Yersinia pestis CO92	
Taxon ID	214092	

INTERPRO

IPR004899	Pertactin domain
IPR005546	Autotransporter beta-domain
IPR006315	Outer membrane autotransporter barrel

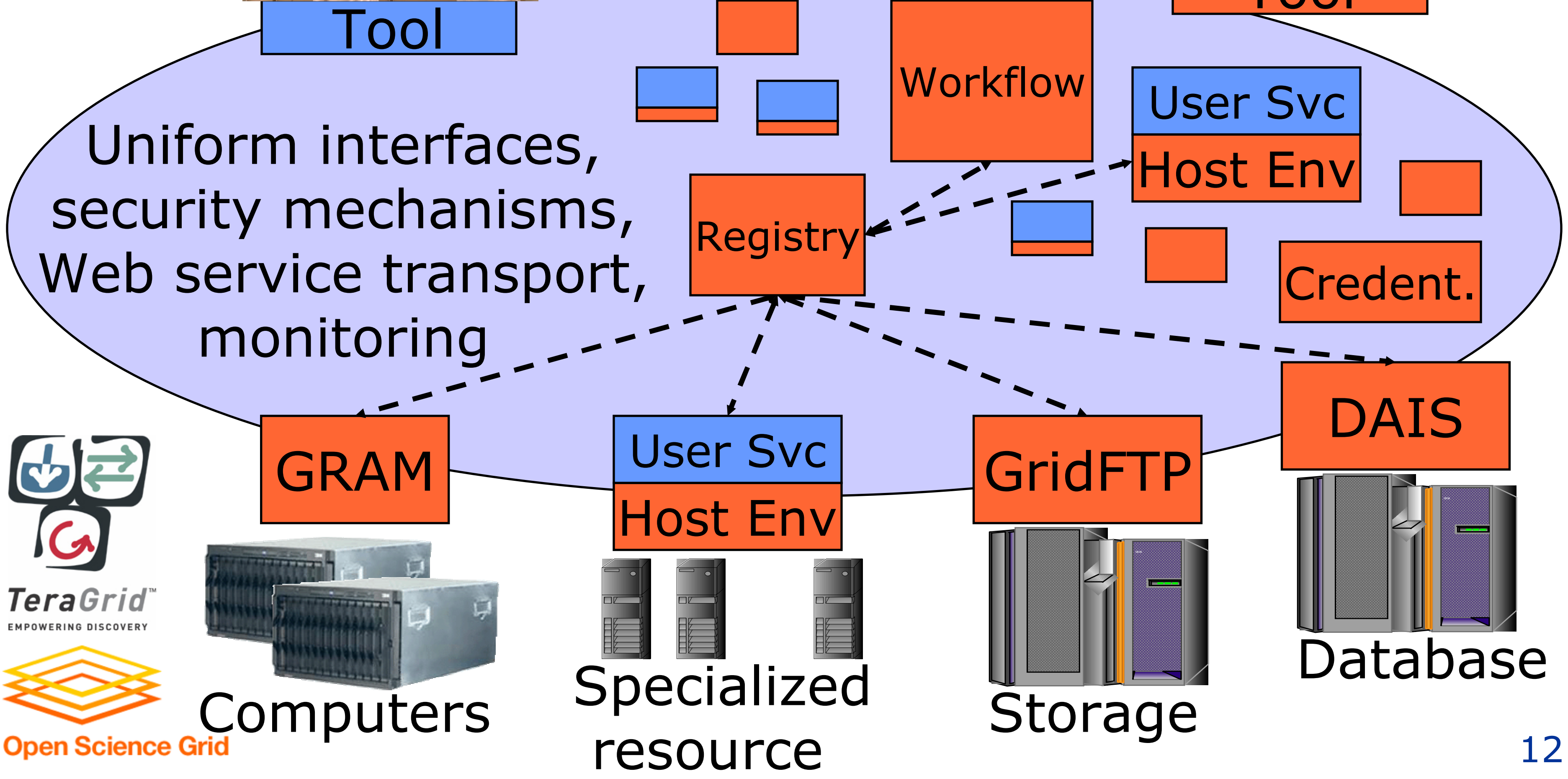
BLOCKS

IPB004899	Pertactin domain

BLAST vs. nr

10945158	YapE protein [Yersinia pestis]
22127716	putative autotransporter protein [Yersinia
33599810	autotransporter [Bordetella bronchiseptica
33595429	autotransporter [Bordetella parapertussis
33595157	autotransporter [Bordetella parapertussis
33599439	autotransporter [Bordetella bronchiseptica
16119581	AGR_pAT_511p [Agrobacterium tumefacien
17938938	autotransporter protein [Agrobacterium tur
33591734	autotransporter [Bordetella pertussis Toha
34496292	conserved hypothetical protein [Chromoba
23500862	outer membrane autotransporter [Brucella

Done, but with errors on page.



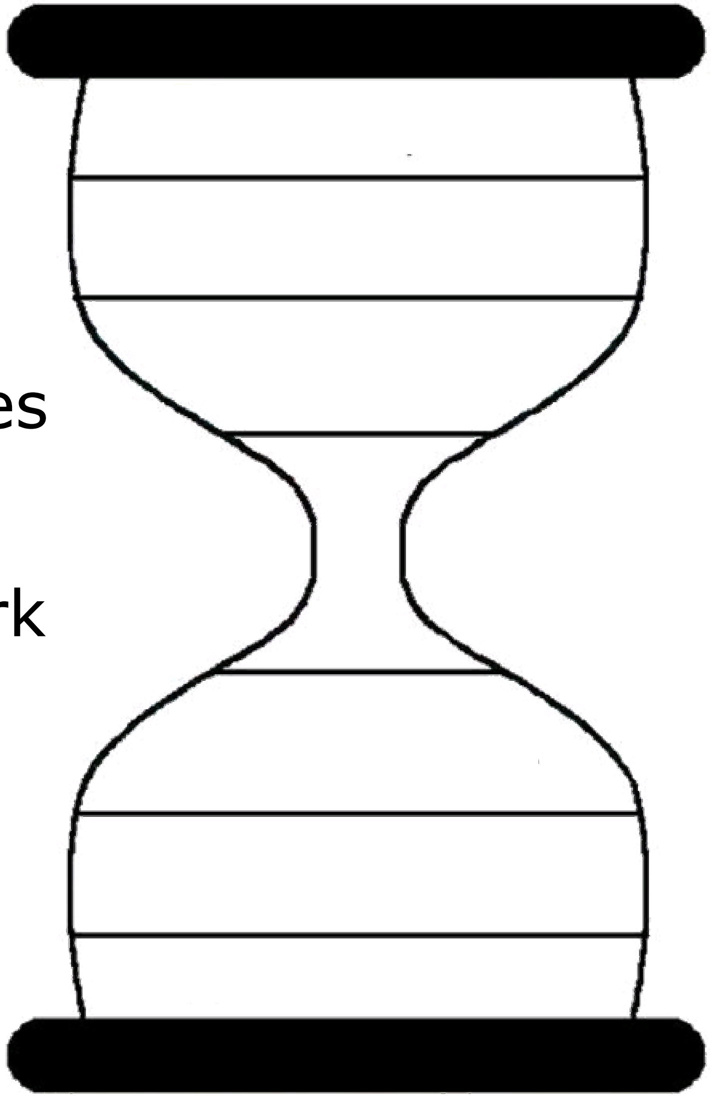
Uniform interfaces, security mechanisms, Web service transport, monitoring

TeraGrid™
EMPOWERING DISCOVERY

Open Science Grid

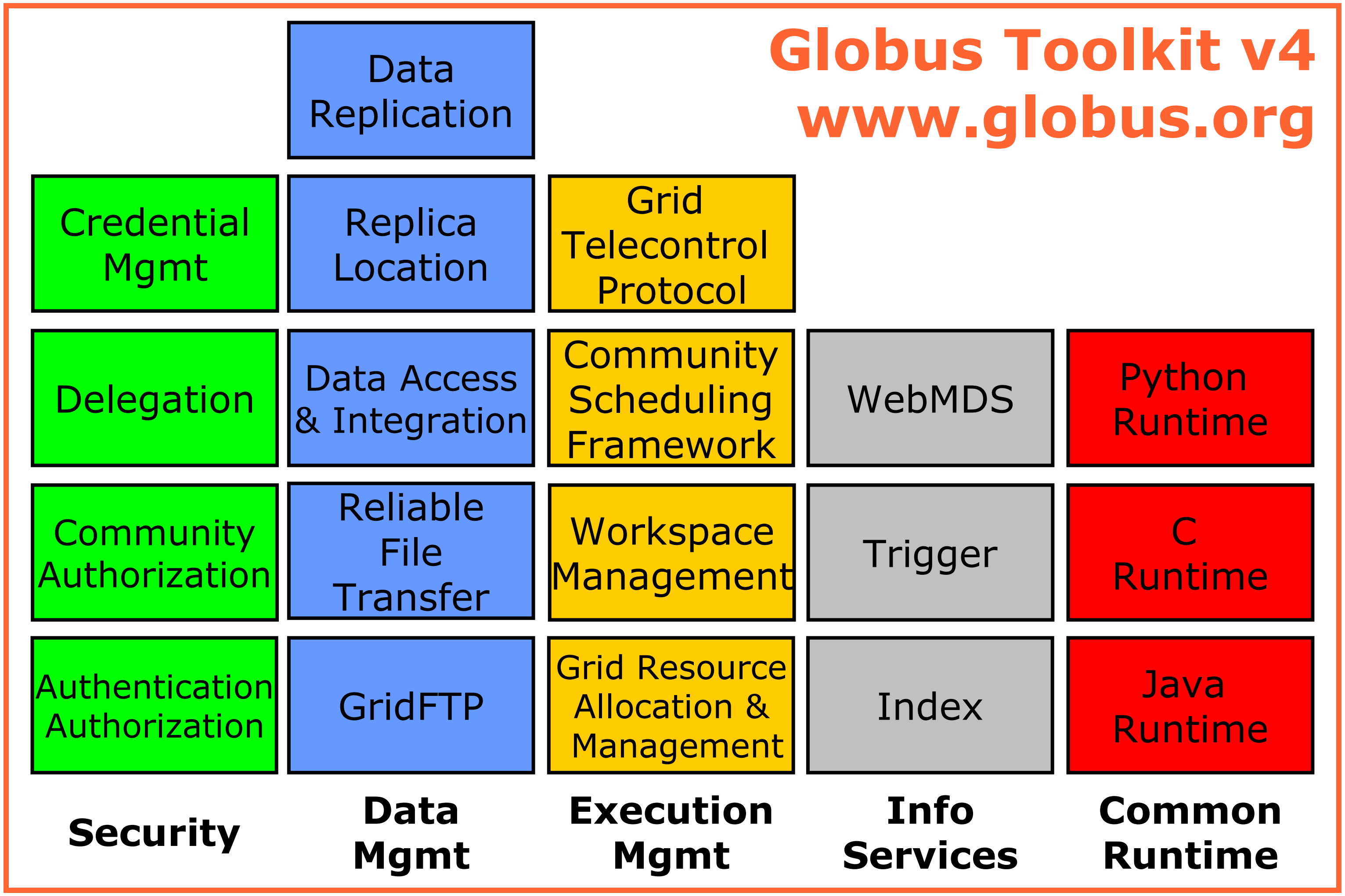
Grid Infrastructure

- Distributed management
  - Of physical resources
  - Of software services
  - Of communities and their policies
- Unified treatment
  - Build on Web services framework
  - Use WS-RF, WS-Notification (or WS-Transfer/Man) to represent/access state
  - Common management abstractions & interfaces



the globus alliance
www.globus.org

Globus Toolkit: Open Source Grid Infrastructure



Globus Toolkit v4
www.globus.org

	Data Replication			
Credential Mgmt	Replica Location	Grid Telecontrol Protocol		
Delegation	Data Access & Integration	Community Scheduling Framework	WebMDS	Python Runtime
Community Authorization	Reliable File Transfer	Workspace Management	Trigger	C Runtime
Authentication Authorization	GridFTP	Grid Resource Allocation & Management	Index	Java Runtime
Security	**Data Mgmt**	**Execution Mgmt**	**Info Services**	**Common Runtime**

I. Foster, Globus Toolkit Version 4: Software for Service-Oriented Systems, LNCS 3779, 2-13, 2005

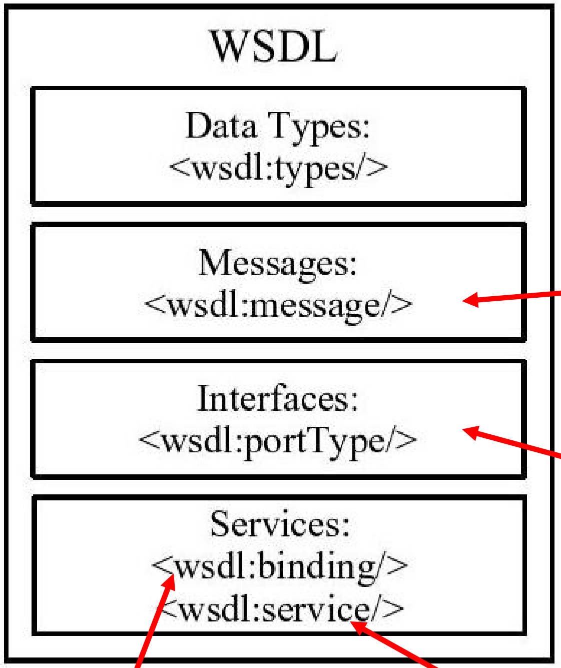
More Specifically, I May Want To …

- Create a service for use by my colleagues
- Manage who is allowed to access my service (or my experimental data or …)
- Ensure reliable & secure distribution of data from my lab to my partners
- Run 10,000 jobs on whatever computers I can get hold of
- Monitor the status of the different resources to which I have access

Web Services

- Standards for defining & accessing services
  - WSDL: Web Services Description Language
  - SOAP: Simple Object Access Protocol
  - Also other standards for security, state access, etc., etc.
- Technology for hosting services, e.g.:
  - Apache Axis (Java)
  - Microsoft (C#)
  - Others in other languages (C, Python, etc.)

WSDL: Web Services Description Language



WSDL

Data Types:
<wsdl:types/>

Messages:
<wsdl:message/>

Interfaces:
<wsdl:portType/>

Services:
<wsdl:binding/>
<wsdl:service/>

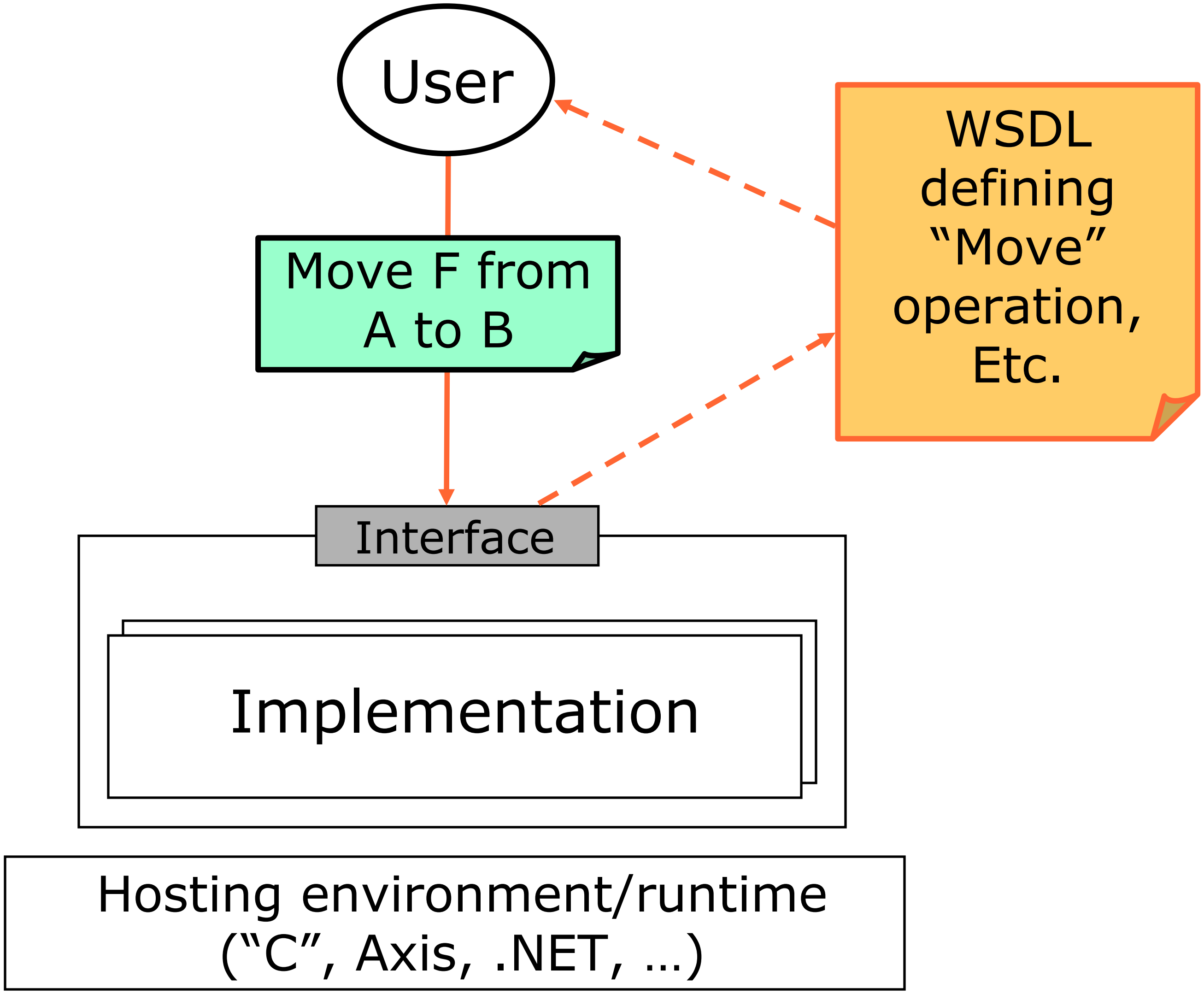
Define expected messages for a service, and their (input or output parameters)

An interface groups together a number of messages (operations)

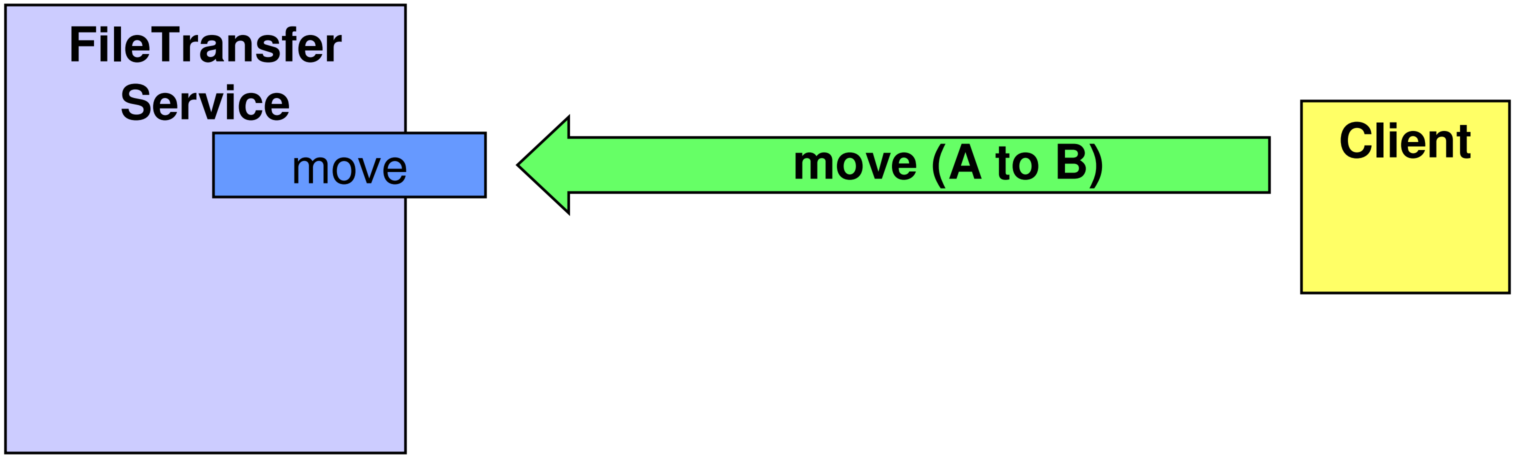
Bind an Interface via a definition to a specific transport (e.g. HTTP) and messaging (e.g. SOAP) protocol

The network location where the service is implemented , e.g. http://localhost:8080

Web Services:
E.g., File Transfer Service


User
Move F from A to B
WSDL defining "Move" operation, Etc.
Interface
Implementation
Hosting environment/runtime ("C", Axis, .NET, …)

"Stateless" vs. "Stateful" Services



FileTransfer Service

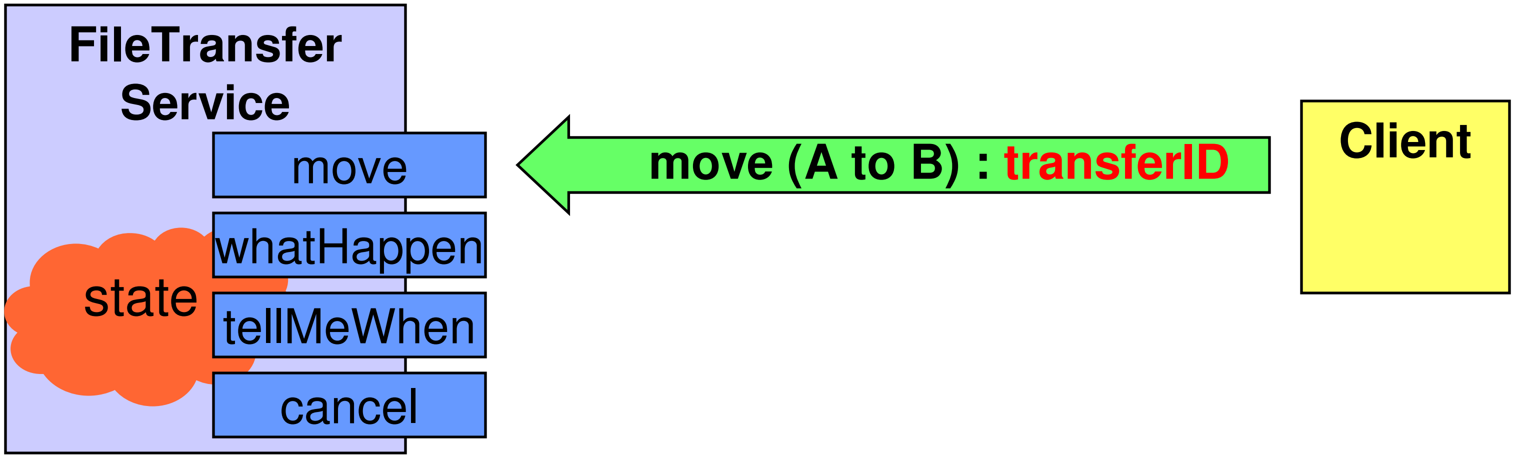
move

move (A to B)

Client

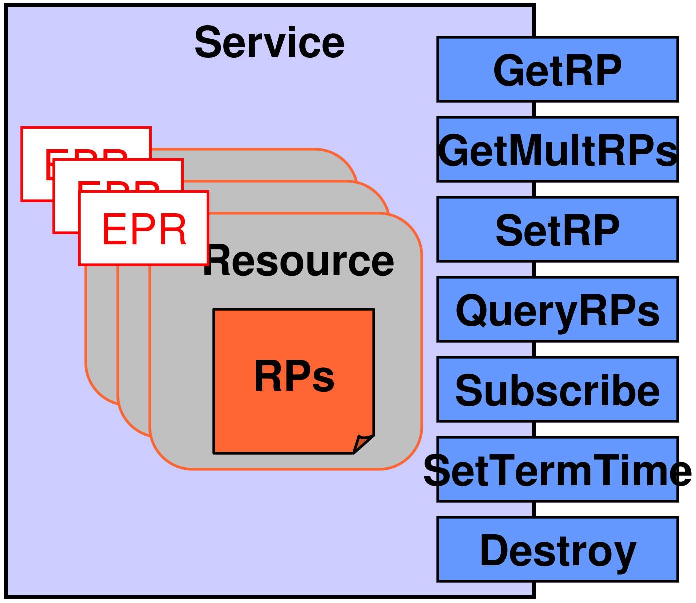
- Without state, how does client:
  - Determine what happened (success/failure)?
  - Find out how many files completed?
  - Receive updates when interesting events arise?
  - Terminate a request?
- Few useful services are truly "stateless", but WS interfaces alone do not provide built-in support for state

FileTransferService (without WSRF)



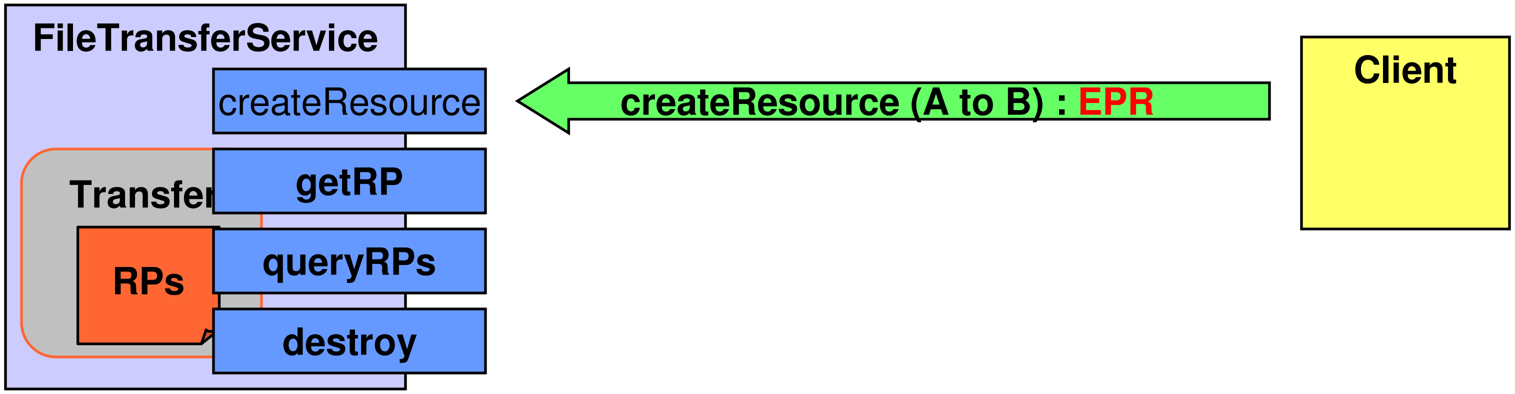
- Developer reinvents wheel for each new service
  - Custom management and identification of state: **transferID**
  - Custom operations to inspect state synchronously (**whatHappen**) and asynchronously (**tellMeWhen**)
  - Custom lifetime operation (**cancel**)

WSRF in a Nutshell



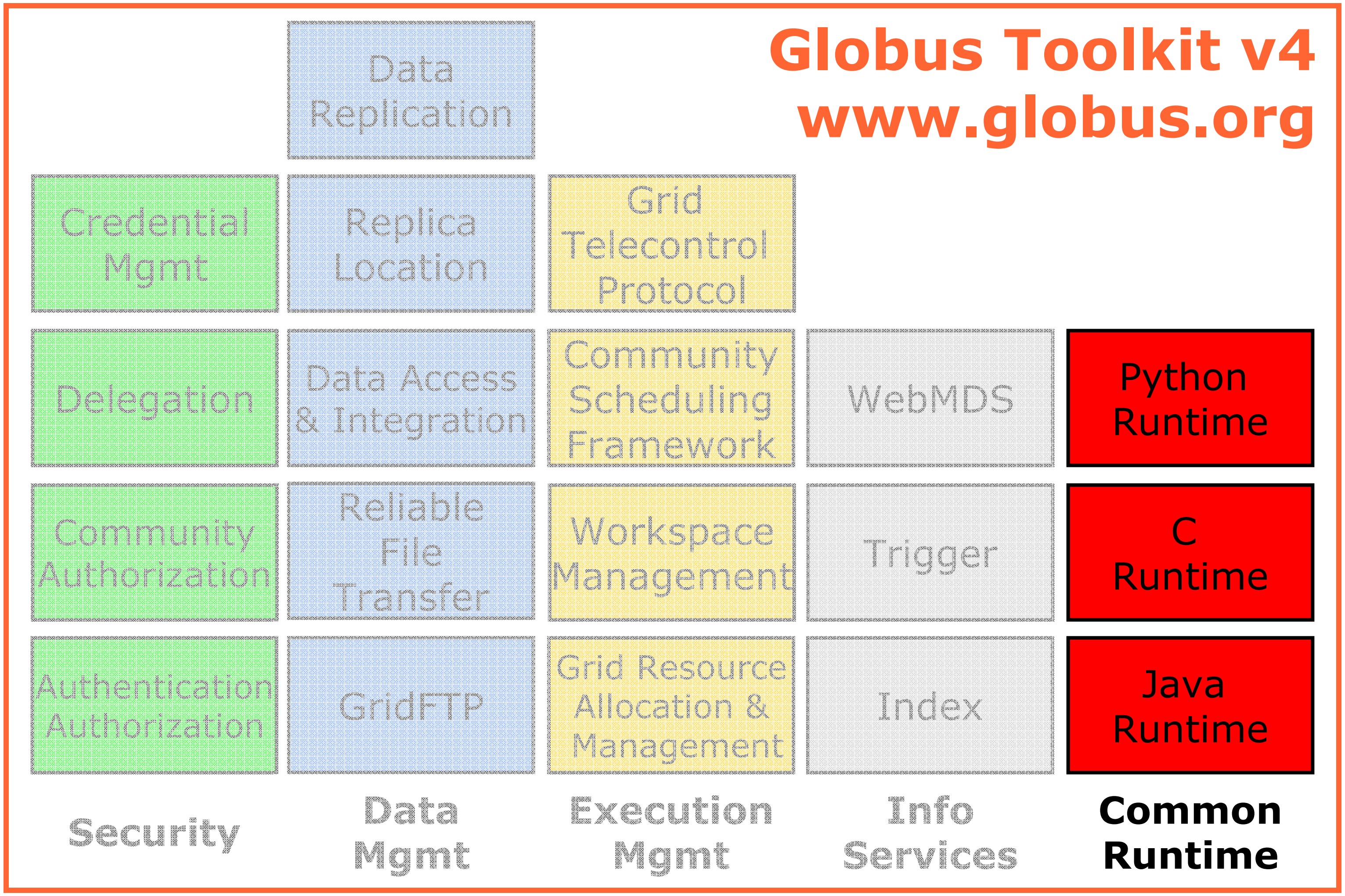
- Service
- State representation
  - Resource
  - Resource Property
- State identification
  - Endpoint Reference
- State Interfaces
  - GetRP, QueryRPs, GetMultipleRPs, SetRP
- Lifetime Interfaces
  - SetTerminationTime
  - ImmediateDestruction
- Notification Interfaces
  - Subscribe
  - Notify
- ServiceGroups

FileTransferService (w/ WSRF)



- Developer specifies custom method to createResource and leaves the rest to WSRF standards:
  - State exposed as Resource + Resource Properties and identified by Endpoint Reference (EPR)
  - State inspected by standard interfaces (GetRP, QueryRPs)
  - Lifetime management by standard interfaces (Destroy)

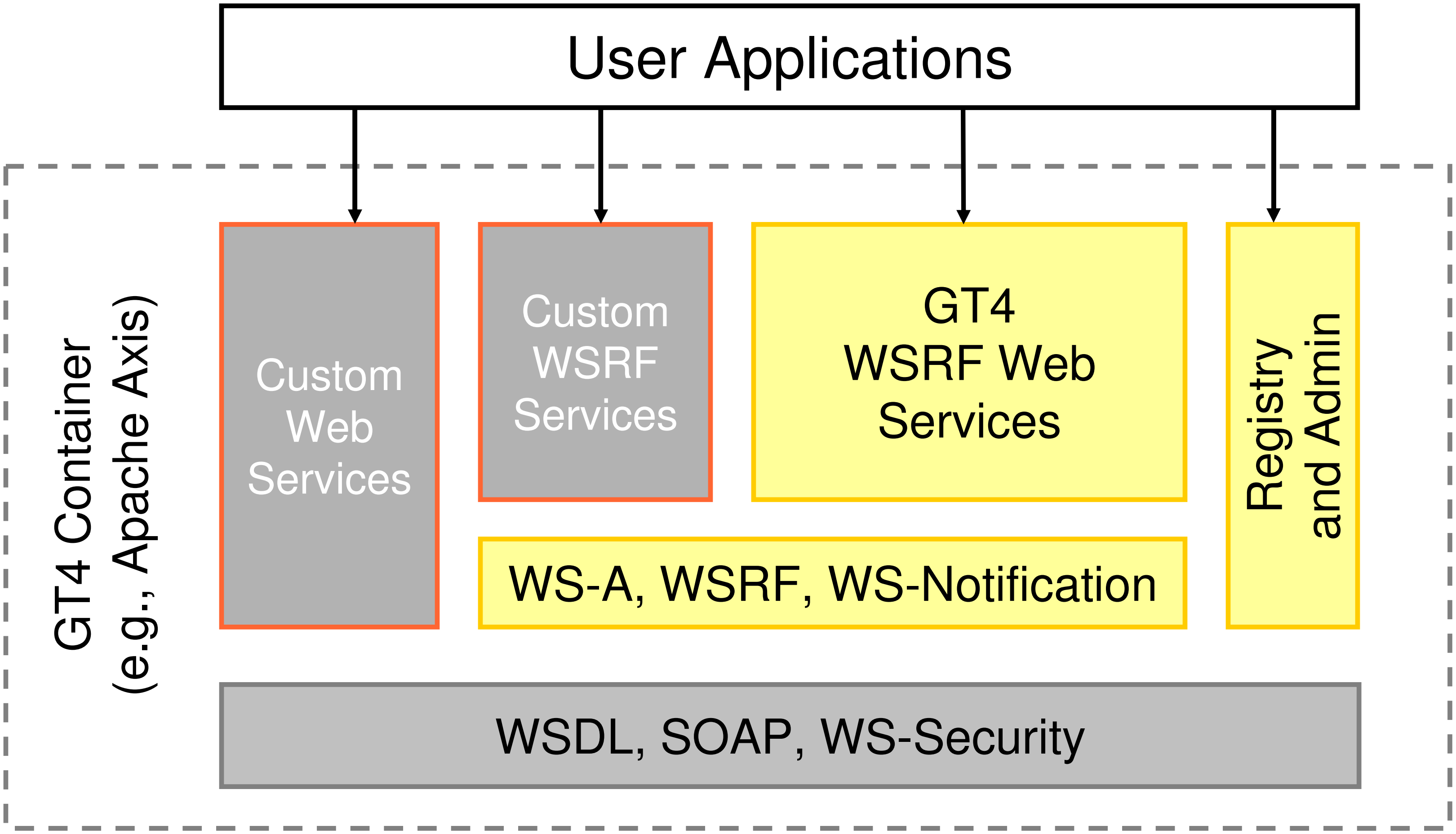
Globus Toolkit: Open Source Grid Infrastructure



Globus Toolkit v4
www.globus.org

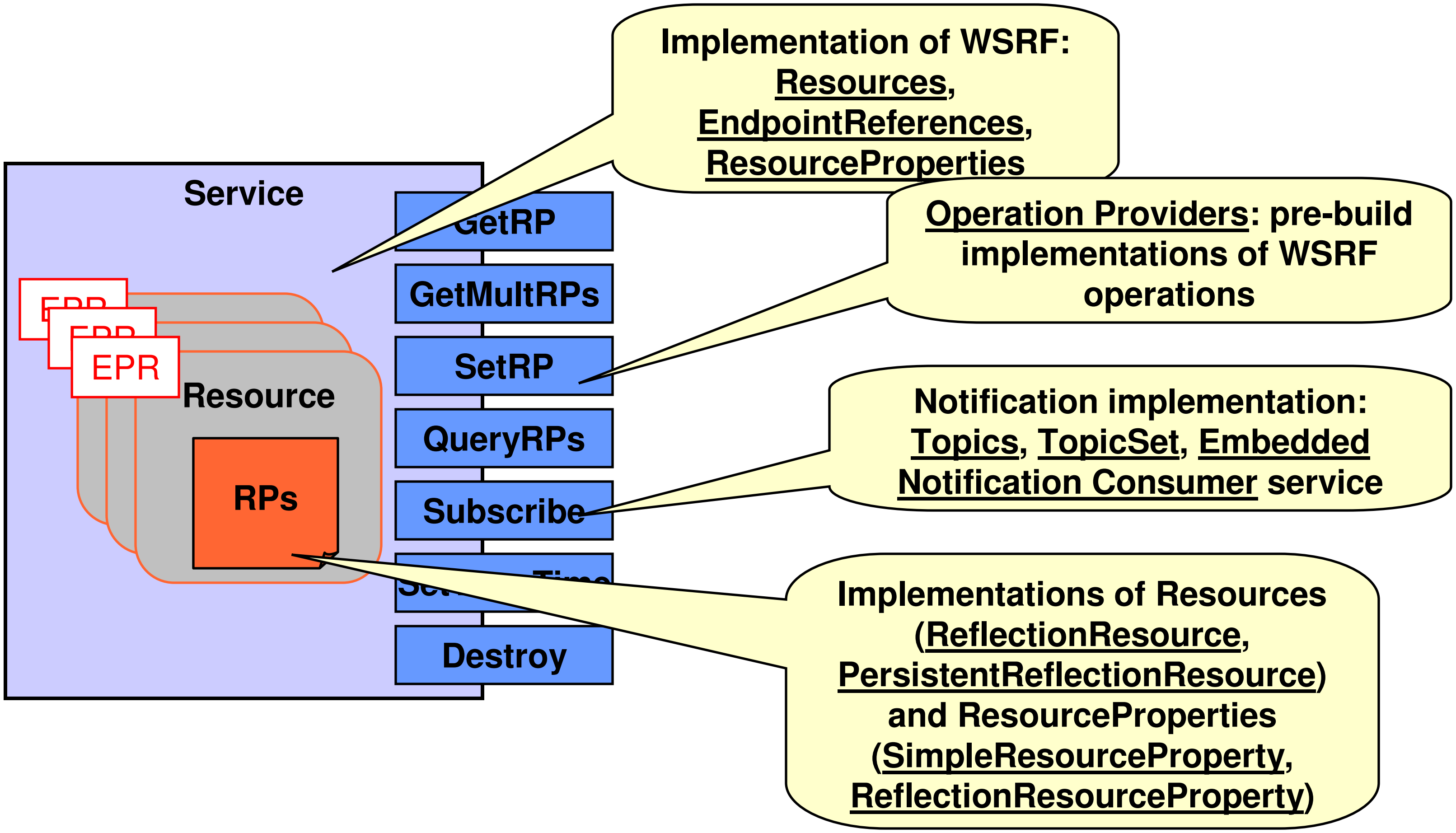
Security	Data Mgmt	Execution Mgmt	Info Services	Common Runtime
	Data Replication			
Credential Mgmt	Replica Location	Grid Telecontrol Protocol		
Delegation	Data Access & Integration	Community Scheduling Framework	WebMDS	Python Runtime
Community Authorization	Reliable File Transfer	Workspace Management	Trigger	C Runtime
Authentication Authorization	GridFTP	Grid Resource Allocation & Management	Index	Java Runtime

GT4 and Web Services



Java (standard Apache Axis), C (fast, small footprint), Python

GT4 WS Core in a Nutshell



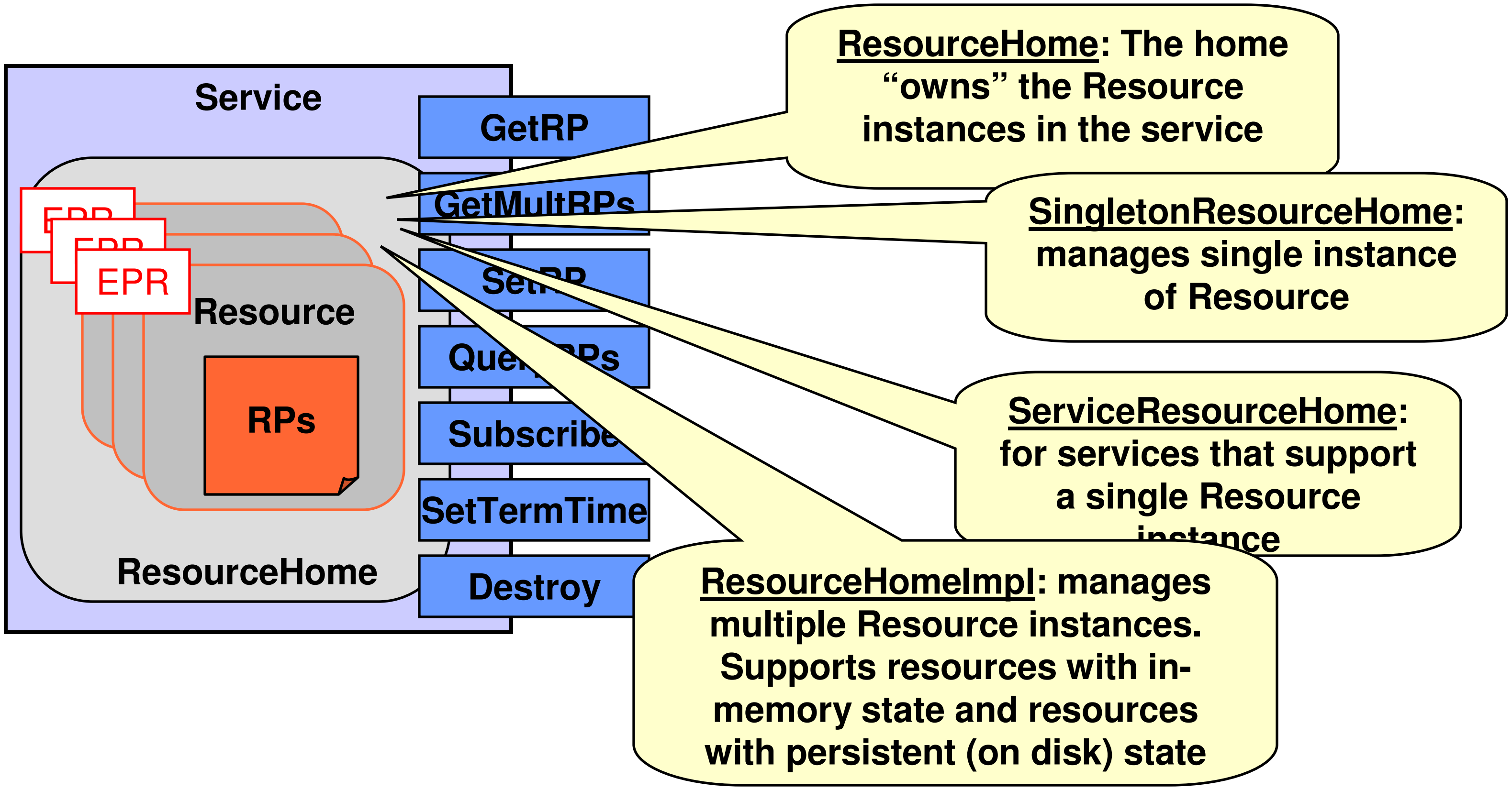
Implementation of WSRF: Resources, EndpointReferences, ResourceProperties

Operation Providers: pre-build implementations of WSRF operations

Notification implementation: Topics, TopicSet, Embedded Notification Consumer service

Implementations of Resources (ReflectionResource, PersistentReflectionResource) and ResourceProperties (SimpleResourceProperty, ReflectionResourceProperty)

GT4 WS Core in a Nutshell



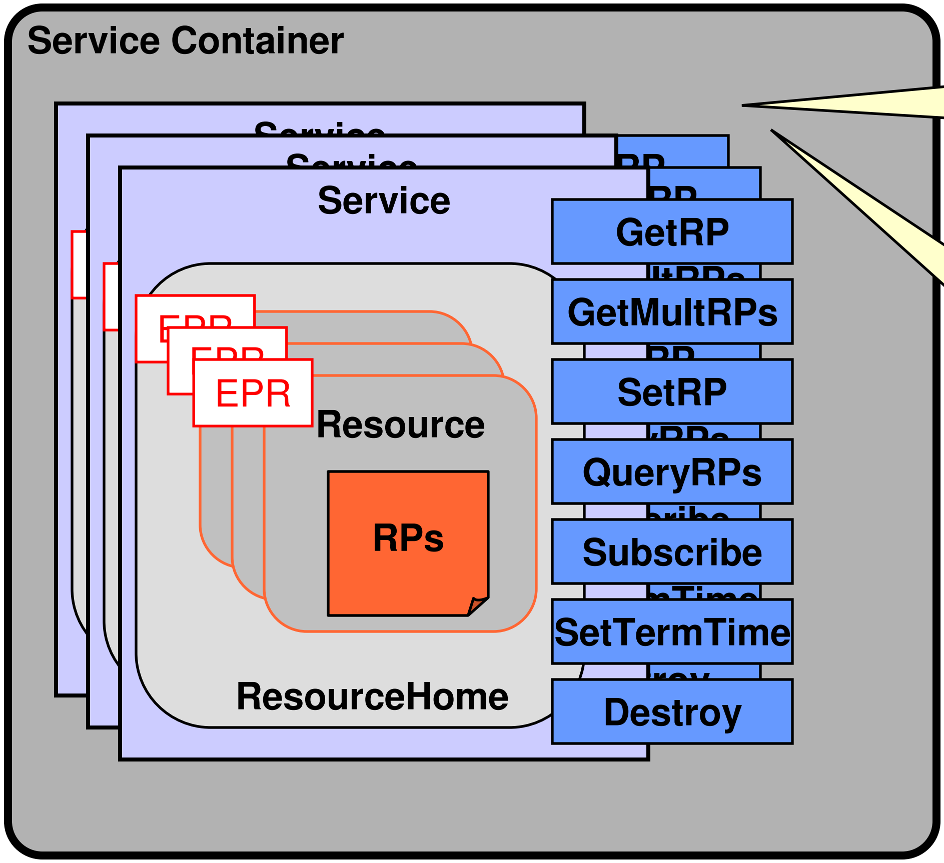
ResourceHome: The home “owns” the Resource instances in the service

SingletonResourceHome: manages single instance of Resource

ServiceResourceHome: for services that support a single Resource instance

ResourceHomeImpl: manages multiple Resource instances. Supports resources with in-memory state and resources with persistent (on disk) state

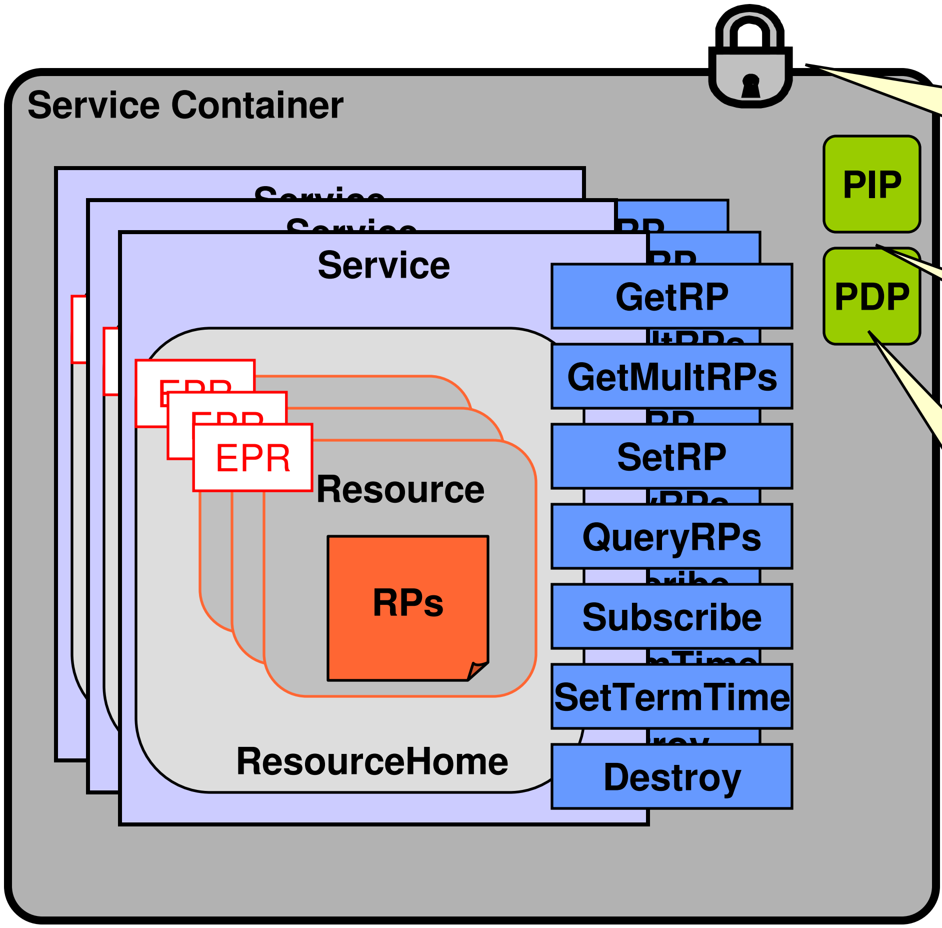
GT4 WS Core in a Nutshell



<u>Service Container</u>: host multiple services in container; one JVM process

…more details: based on AXIS service container, processes SOAP messages, ResourceContext extension.

GT4 WS Core in a Nutshell

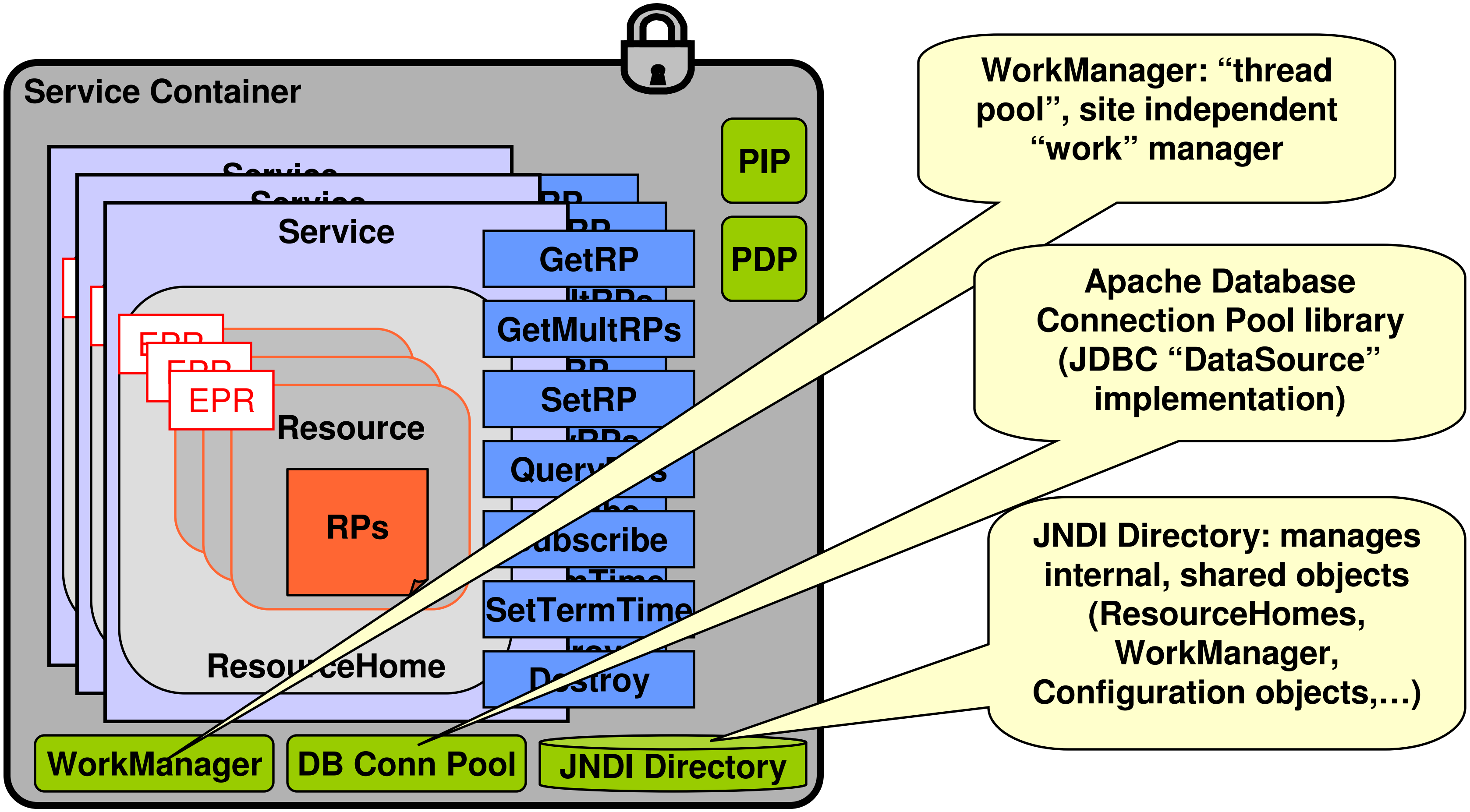


<u>Secure Communication</u>: Transport, Message, Conversation (Transport demonstrates best performance)

Configurable Security Policies: <u>Policy Information Points</u> (PIPs), <u>Policy Decision Points</u> (PDP) -- chained

Example authorization PDPs: GridMap, SAML implementations, XACML policies

GT4 WS Core in a Nutshell

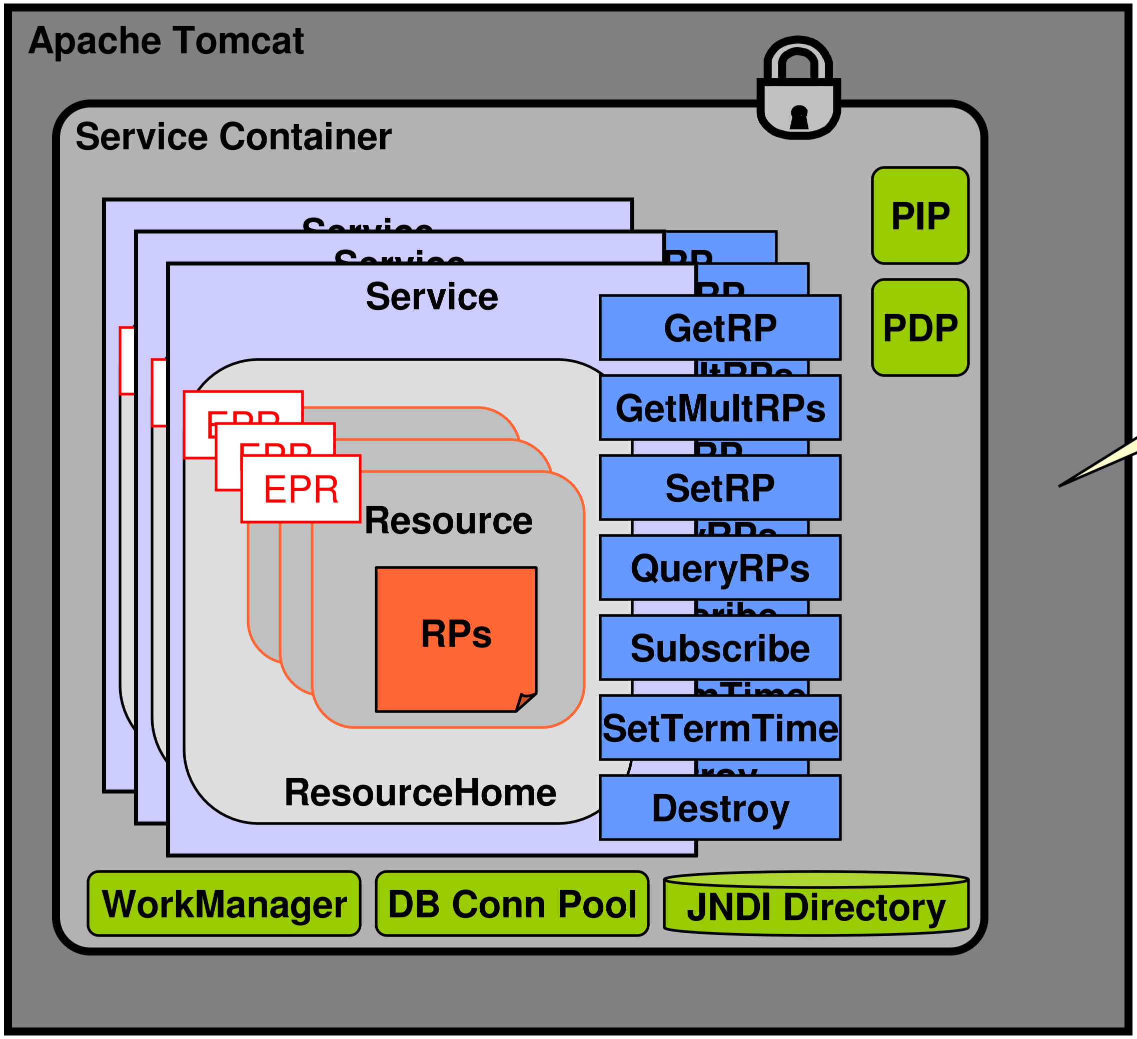


WorkManager: “thread pool”, site independent “work” manager

Apache Database Connection Pool library (JDBC “DataSource” implementation)

JNDI Directory: manages internal, shared objects (ResourceHomes, WorkManager, Configuration objects,…)

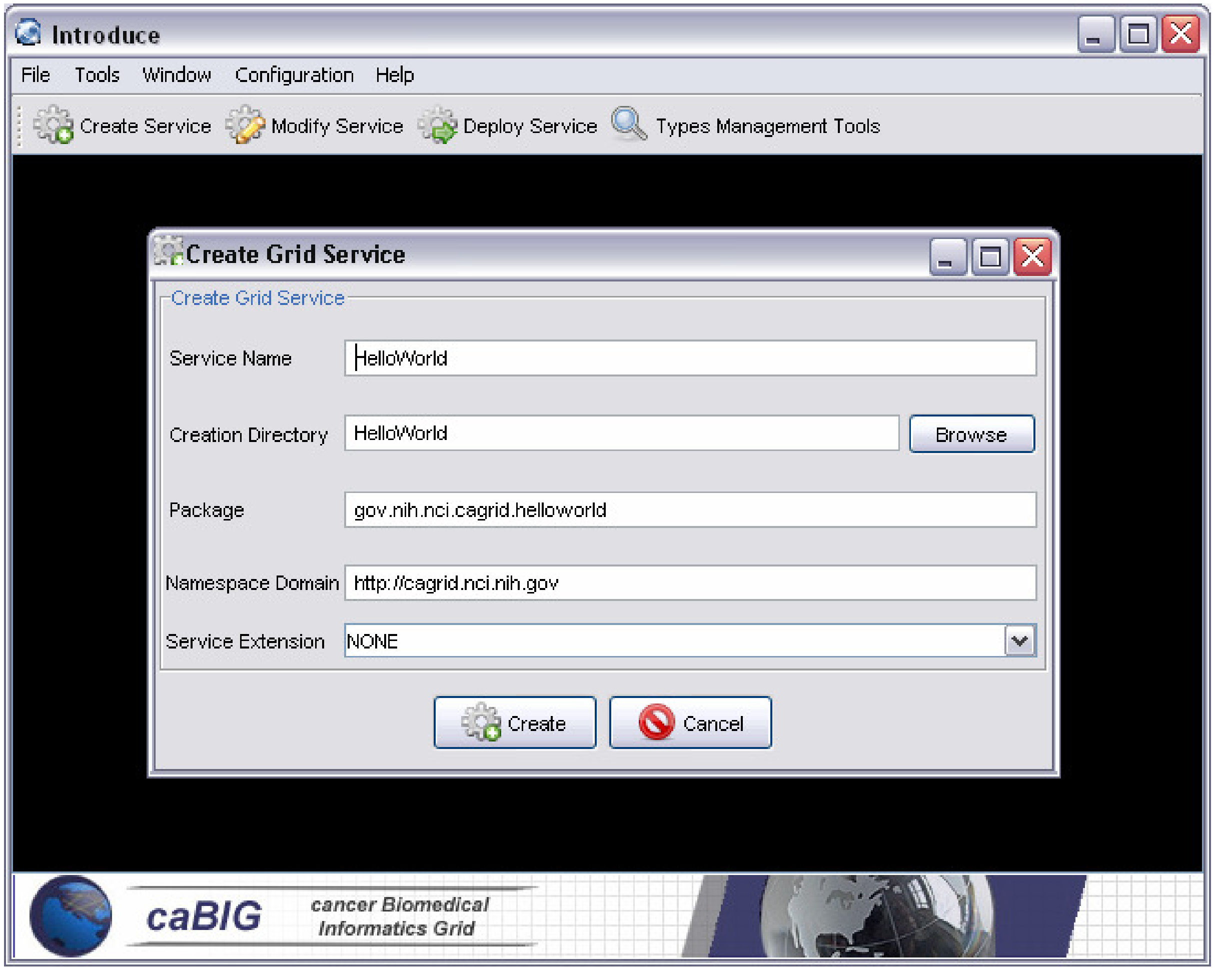
GT4 WS Core in a Nutshell



Deploy Service Container "standalone" or within Apache Tomcat

The Introduce Authoring Tool

- Define service
- Create skeleton
- Discover types
- Add operations
- Configure security
- Modify service



Create Grid Service

Service Name: HelloWorld

Creation Directory: HelloWorld

Package: gov.nih.nci.cagrid.helloworld

Namespace Domain: http://cagrid.nci.nih.gov

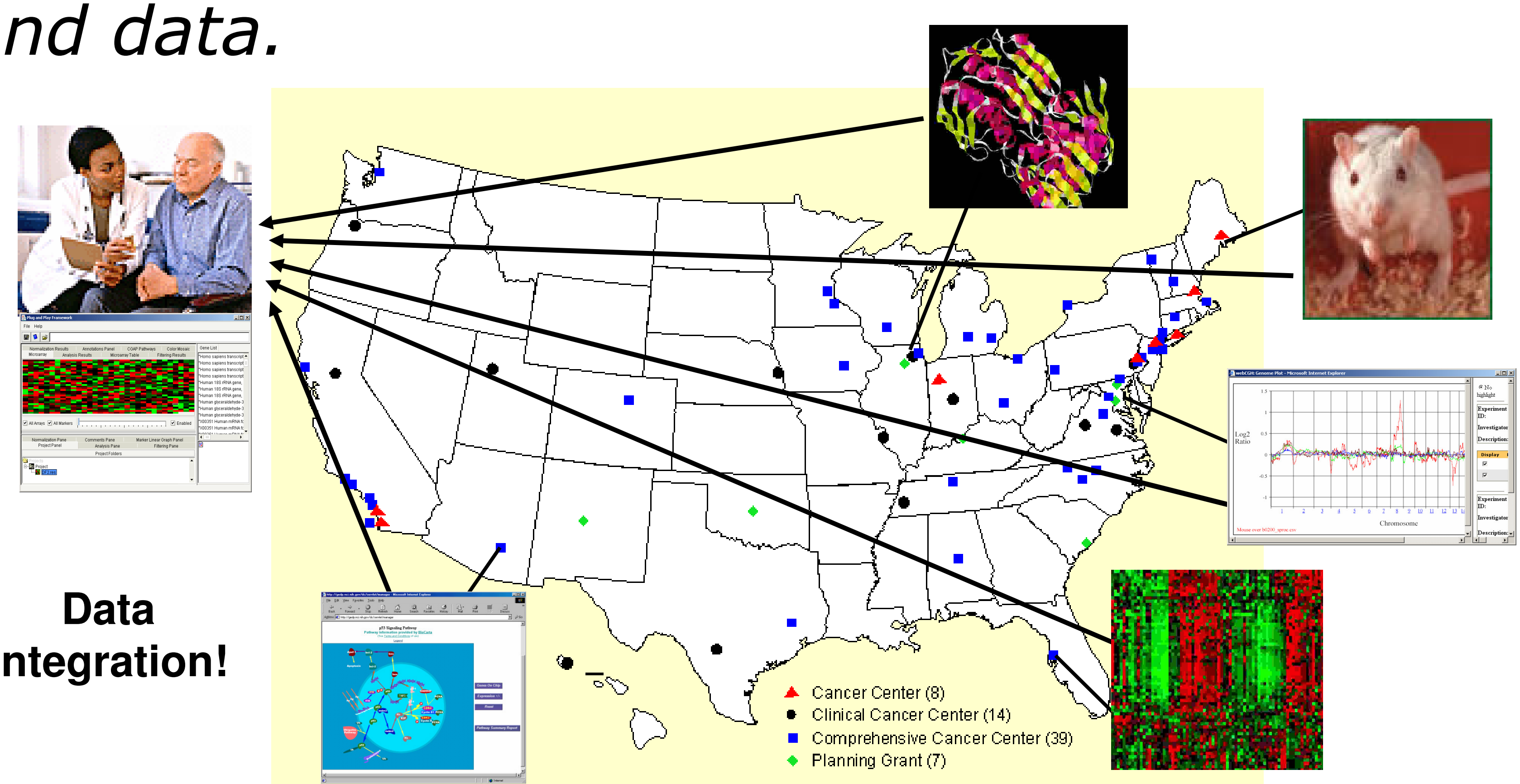
Service Extension: NONE

See also: SOAPLab, OPAL, pyGlobus, Gannon, etc.

Introduce: Hastings, Saltz, et al., Ohio State University

For Example: Cancer Biology

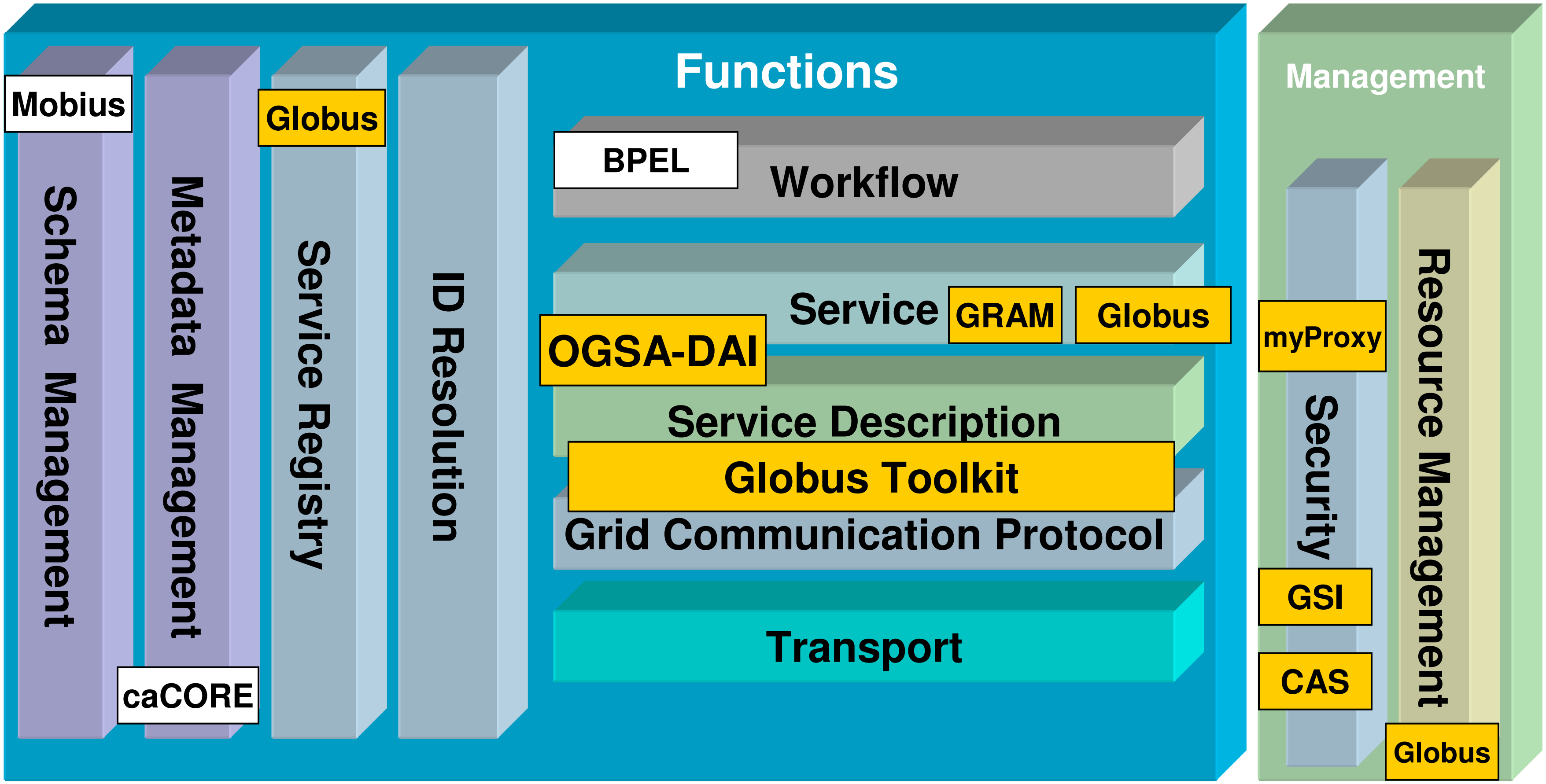
caBIG: sharing of infrastructure, applications, and data.



Data Integration!

caBIG cancer Biomedical Informatics Grid

Cancer Biomedical Informatics Grid



Spans 60 NIH cancer centers across the U.S.

caBIG™ cancer Biomedical Informatics Grid™ an initiative of the National Cancer Institute

Slide credit: Peter Covitz, National Institutes of Health

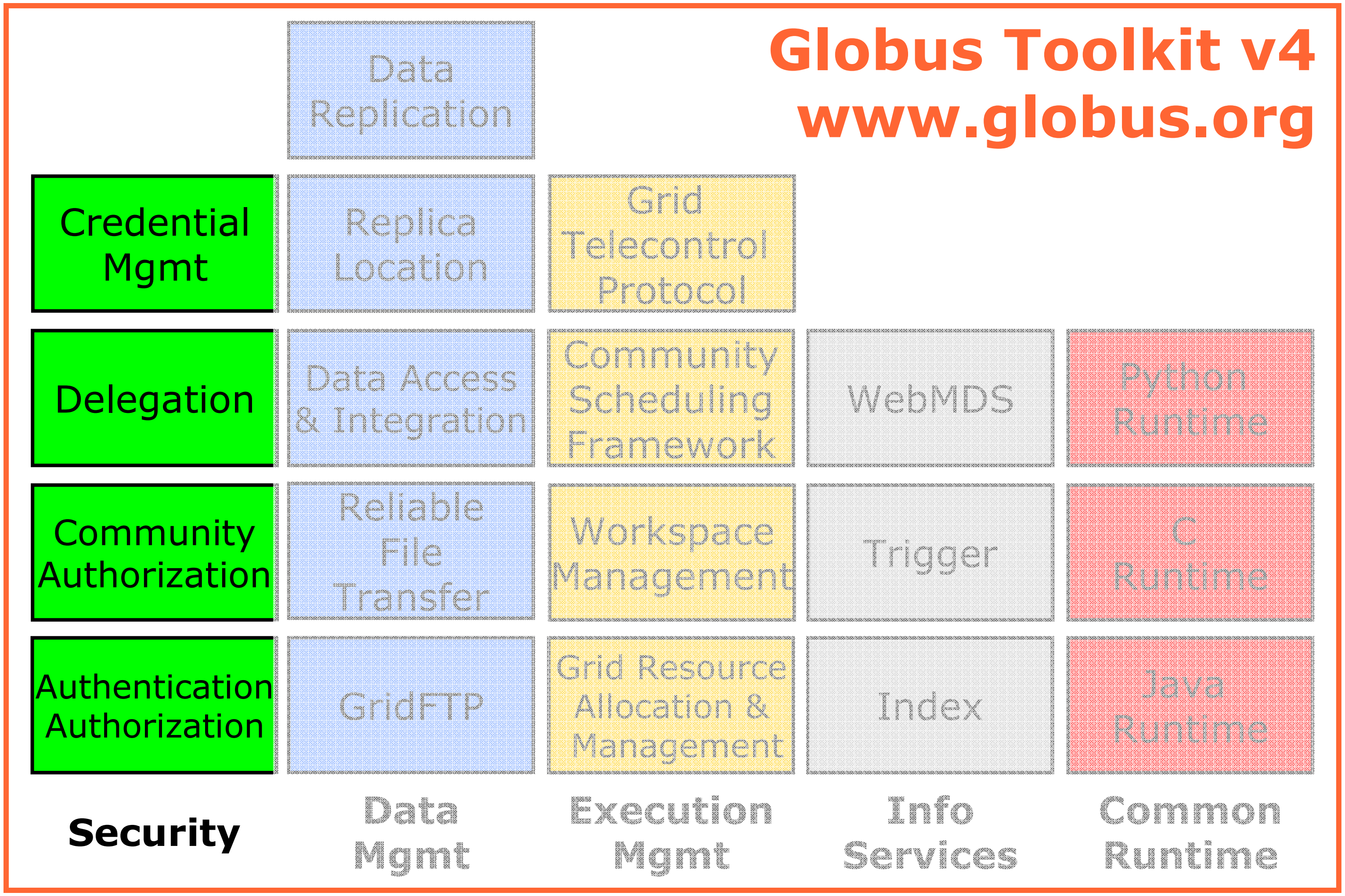
More Specifically, I May Want To …

- Create a service for use by my colleagues
- Manage who is allowed to access my service (or my experimental data or …)
- Ensure reliable & secure distribution of data from my lab to eight partners
- Run 10,000 jobs on whatever computers I can get hold of
- Monitor the status of the different resources to which I have access

Grid Security Concerns

- Control access to shared services
  - Address autonomous management, e.g., different policy in different work groups
- Support multi-user collaborations
  - Federate through mutually trusted services
  - Local policy authorities rule
- Allow users and application communities to set up dynamic trust domains
  - Personal/VO collection of resources working together based on trust of user/VO

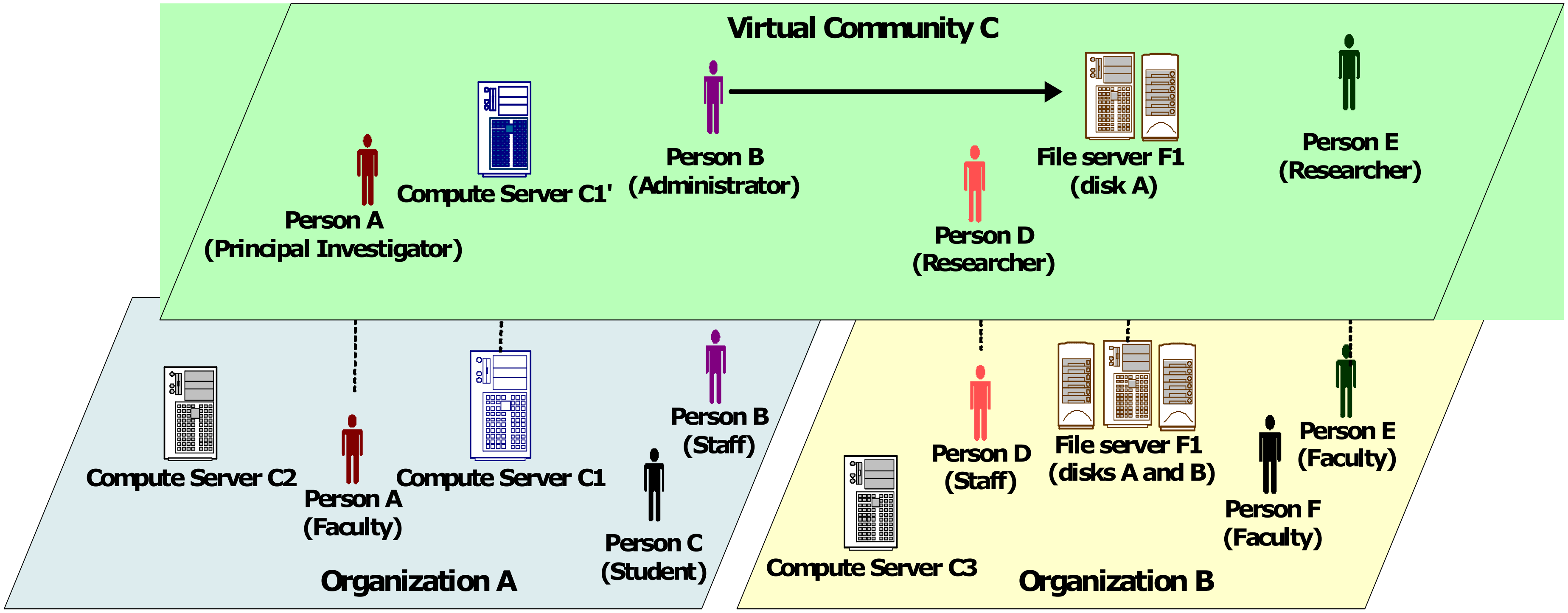
Globus Toolkit: Open Source Grid Infrastructure



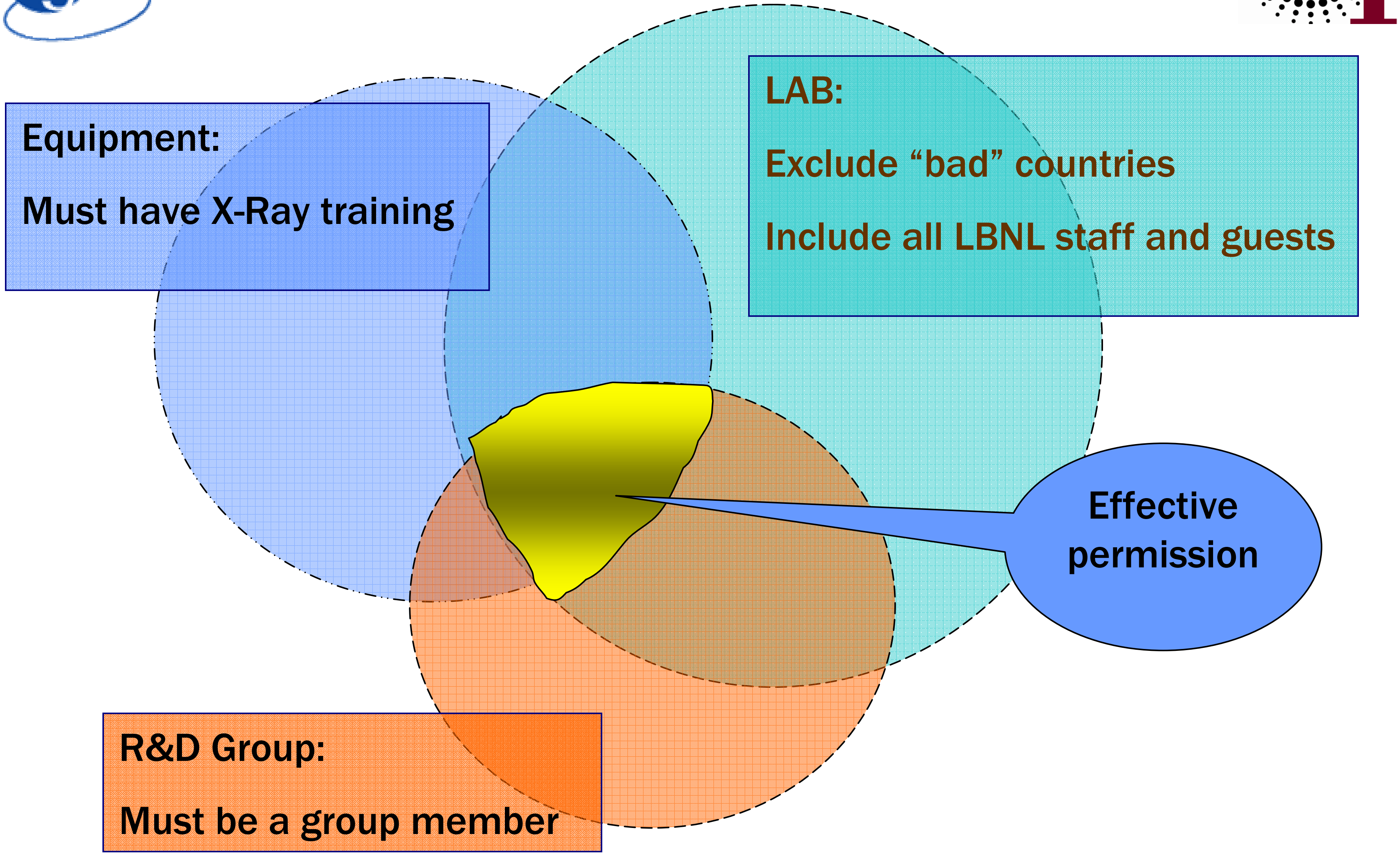
Globus Toolkit v4
www.globus.org

	Data Replication			
Credential Mgmt	Replica Location	Grid Telecontrol Protocol		
Delegation	Data Access & Integration	Community Scheduling Framework	WebMDS	Python Runtime
Community Authorization	Reliable File Transfer	Workspace Management	Trigger	C Runtime
Authentication Authorization	GridFTP	Grid Resource Allocation & Management	Index	Java Runtime
Security	**Data Mgmt**	**Execution Mgmt**	**Info Services**	**Common Runtime**

Virtual Organization (VO) Concept



- VO for each application or workload
- Carve out and configure resources for a particular use and set of users



Equipment:

Must have X-Ray training

LAB:

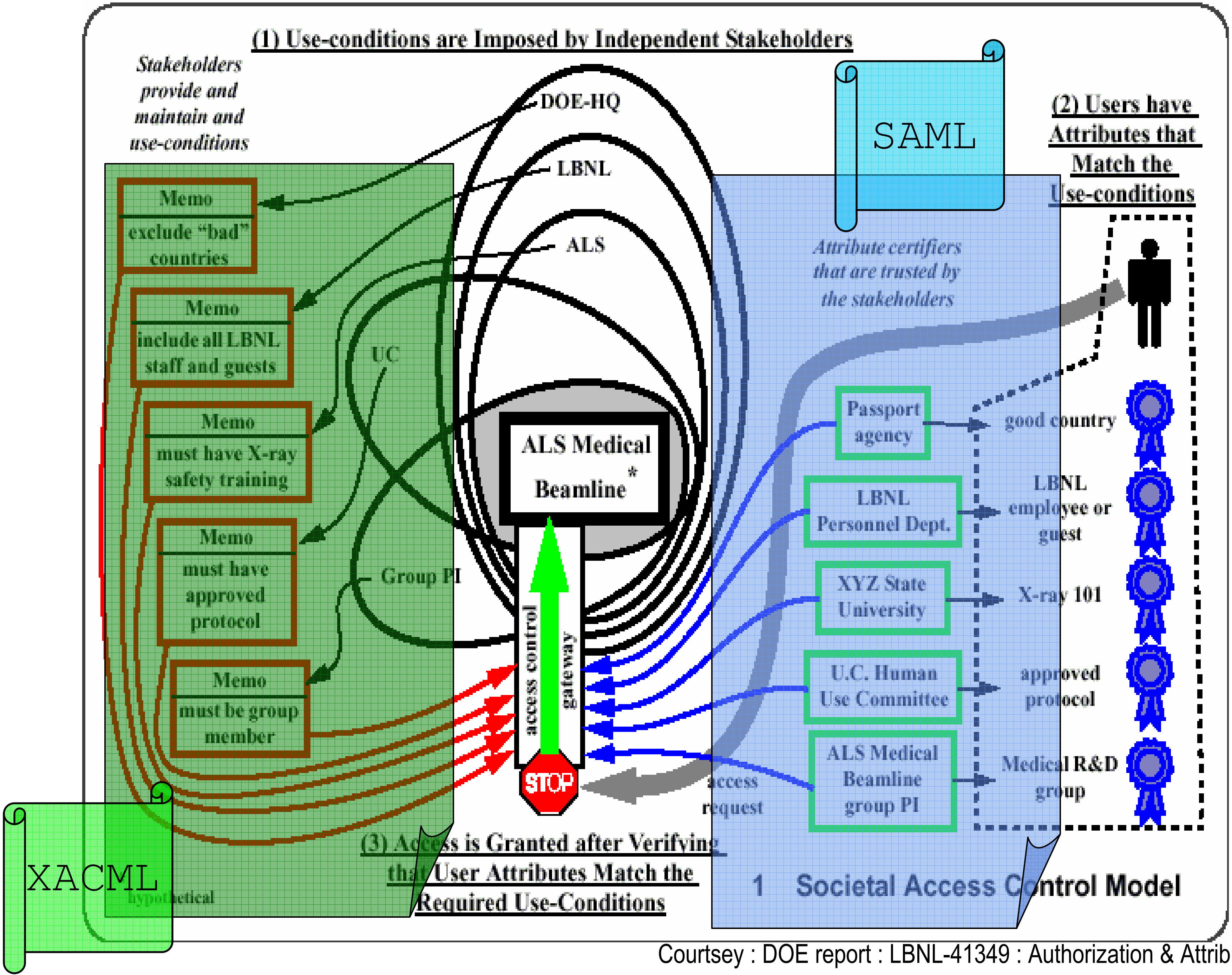
Exclude “bad” countries

Include all LBNL staff and guests

Effective permission

R&D Group:

Must be a group member



(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

DOE-HQ

LBNL

ALS

UC

Group PI

ALS Medical Beamline *

access control gateway

STOP

SAML

Attribute certifiers that are trusted by the stakeholders

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

Passport agency

LBNL Personnel Dept.

XYZ State University

U.C. Human Use Committee

ALS Medical Beamline group PI

good country

LBNL employee or guest

X-ray 101

approved protocol

Medical R&D group

access request

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

XACML

hypothetical

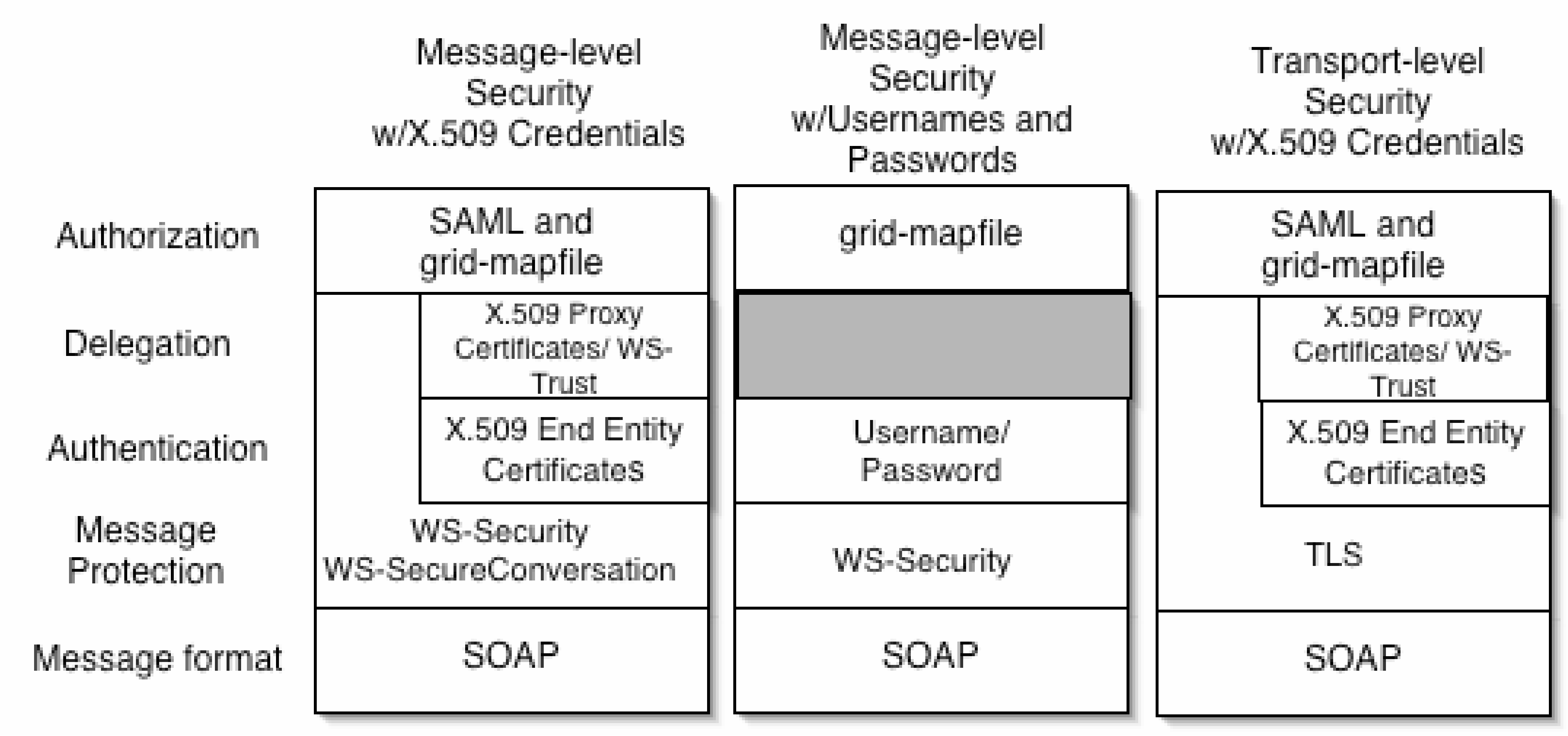
1 Societal Access Control Model

Courtsey : DOE report : LBNL-41349 : Authorization & Attribute Certificates for Widely Distributed Access Control

GT4 Security

- Public-key-based authentication
- Extensible authorization framework based on Web services standards
  - SAML-based authorization callout
    - As specified in GGF OGSA-Authz WG
  - Integrated policy decision engine
    - XACML (eXtensible Access Control Markup Language) policy language, per-operation policies, pluggable
- Credential management service
  - MyProxy (One time password support)
- Community Authorization Service
- Standalone delegation service

GT4’s Use of Security Standards



	Message-level Security w/X.509 Credentials	Message-level Security w/Usernames and Passwords	Transport-level Security w/X.509 Credentials
Authorization	SAML and grid-mapfile	grid-mapfile	SAML and grid-mapfile
Delegation	X.509 Proxy Certificates/ WS-Trust		X.509 Proxy Certificates/ WS-Trust
Authentication	X.509 End Entity Certificates	Username/ Password	X.509 End Entity Certificates
Message Protection	WS-Security WS-SecureConversation	WS-Security	TLS
Message format	SOAP	SOAP	SOAP

Supported,
but slow

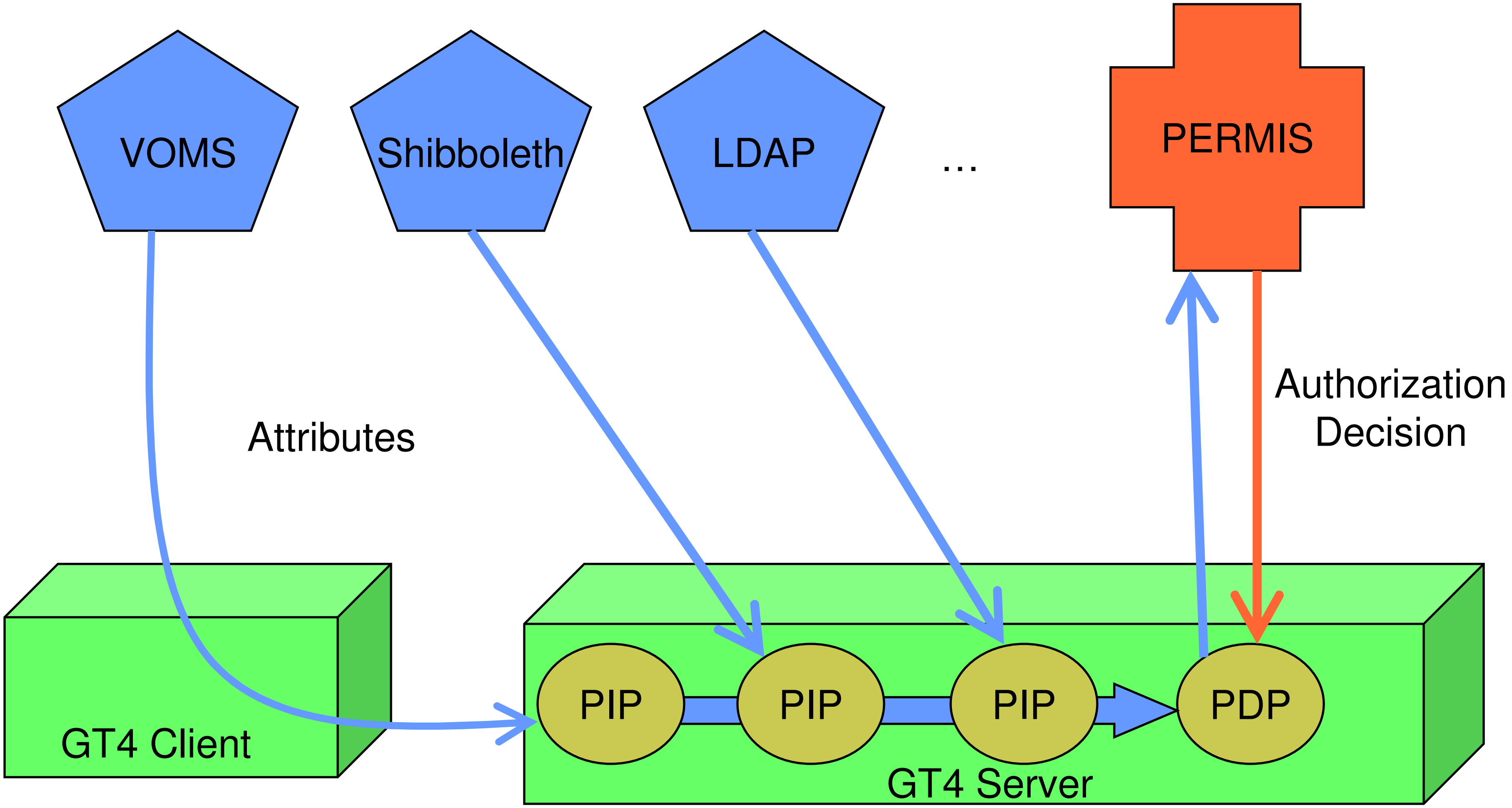
Supported,
but insecure

**Fastest,
so default**

GT-XACML Integration

- eXtensible Access Control Markup Language
  - OASIS standard, open source implementations
- XACML: sophisticated policy language
- Globus Toolkit ships with XACML runtime
  - Included in every client and server built on GT
  - Turned-on through configuration
- … that can be called transparently from runtime and/or explicitly from application …
- … and we use the XACML-”model” for our Authz Processing Framework

GT Authorization Framework



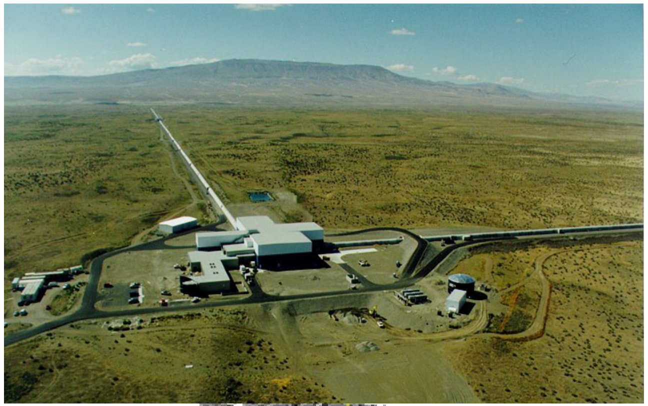
More Specifically, I May Want To …

- Create a service for use by my colleagues
- Manage who is allowed to access my service (or my experimental data or …)
- Ensure reliable & secure distribution of data from my lab to my partners
- Run 10,000 jobs on whatever computers I can get hold of
- Monitor the status of the different resources to which I have access

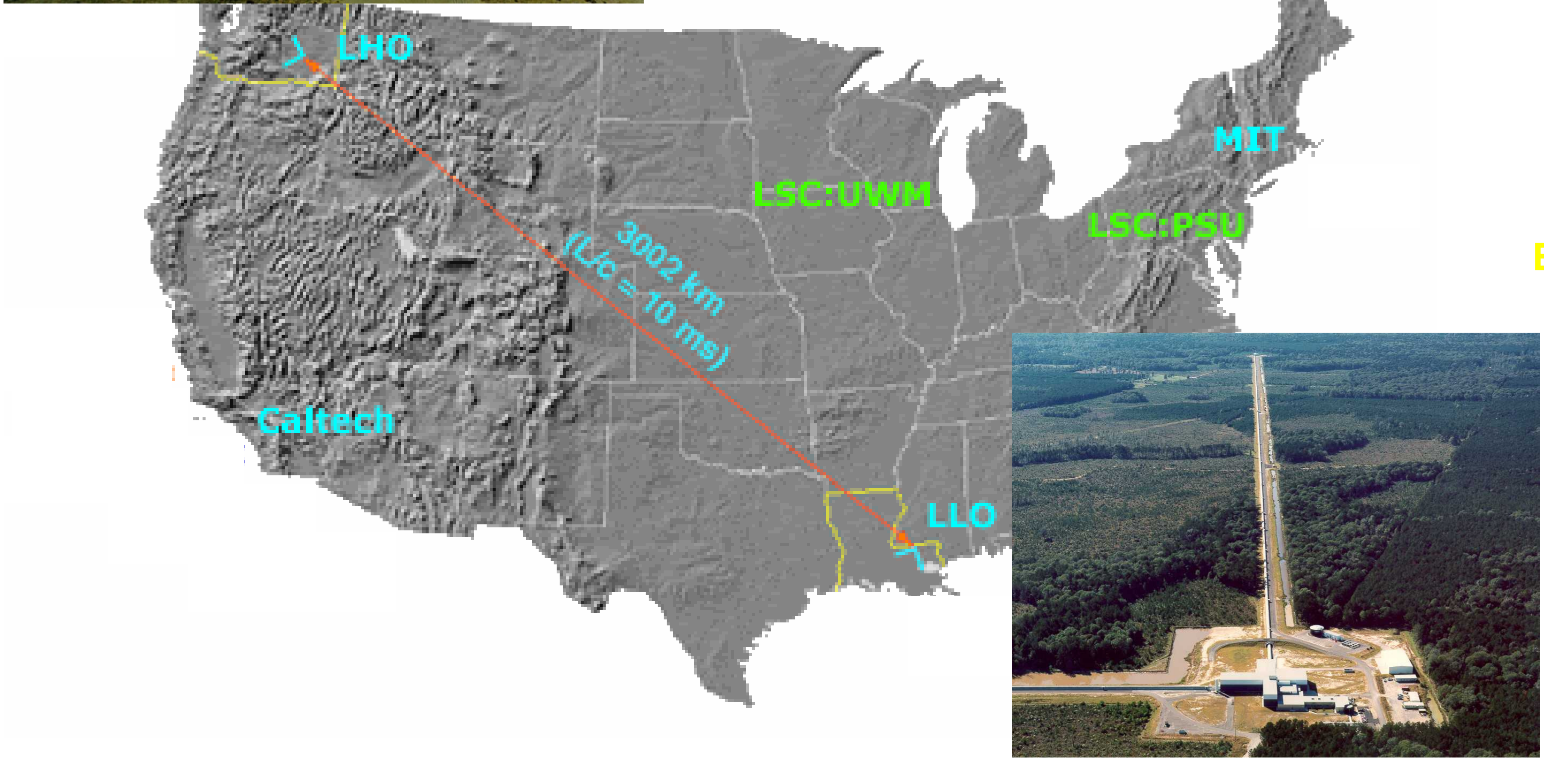
Use Case: Distribution of a New Data Set

- Typical requirement for a distributed team performing iterative design tasks
- Data set is logically defined
  - Domain-specific name: ‘System Design V1.32’
  - Map to physical files: directories, catalog, etc.
- Users, applications, workflows request latest data
  - Scripted or through a web service interface.
  - Requests are recorded and failures are retried.
- Global policies are applied
  - Bandwidth usage is managed
  - Access policies are enforced
- Replicas at each site are tracked
  - Redundant transfers are avoided
  - Files are copied from cheapest up-to-date source

Reliable Wide Area Data Replication



LIGO Gravitational Wave Observatory



SCOTLAND
ENGLAND
Birmingham•
▪Cardiff

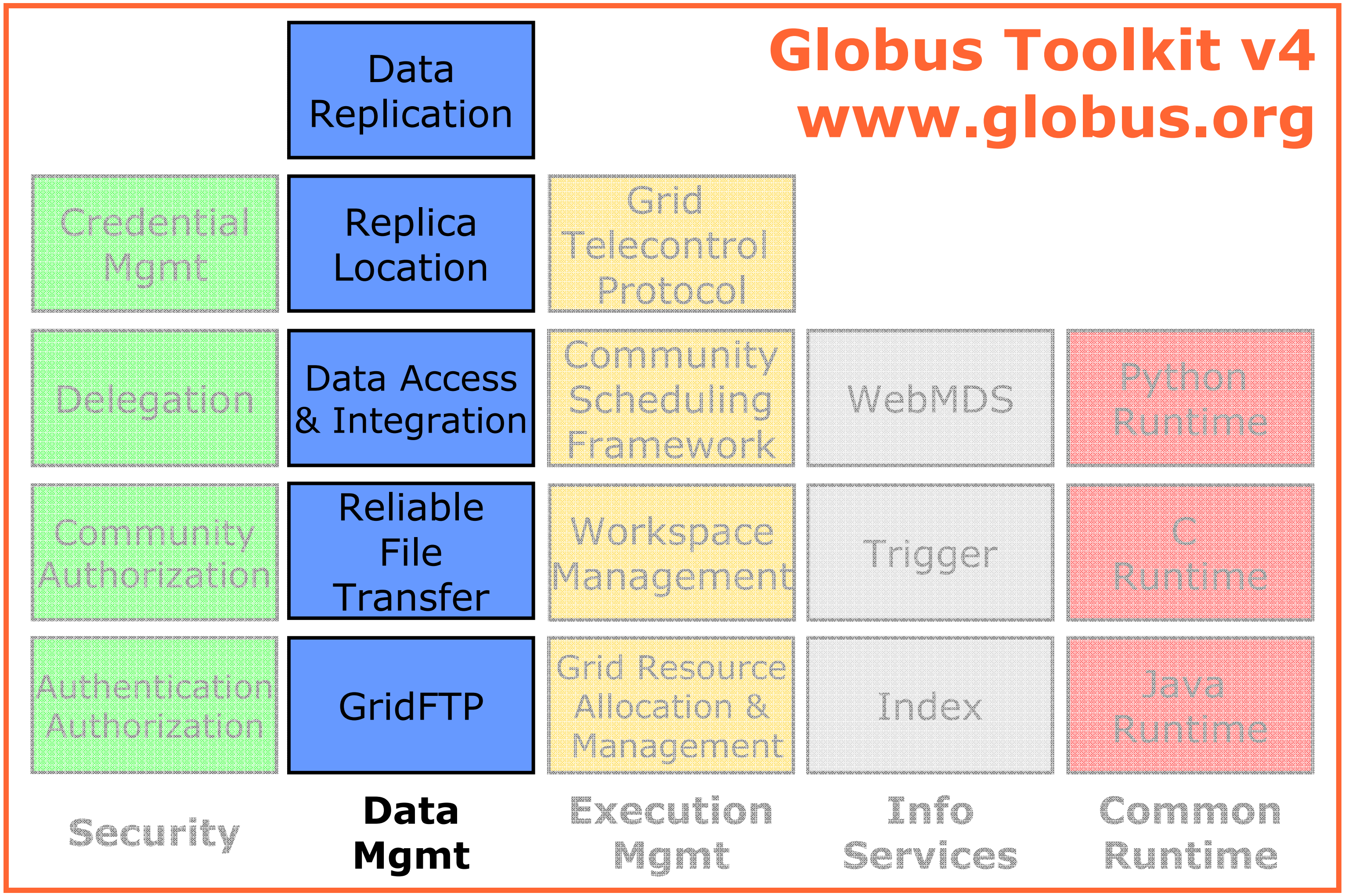
Replicating >1 Terabyte/day to 8 sites

>30 million replicas so far

MTBF = 1 month

www.globus.org/solutions

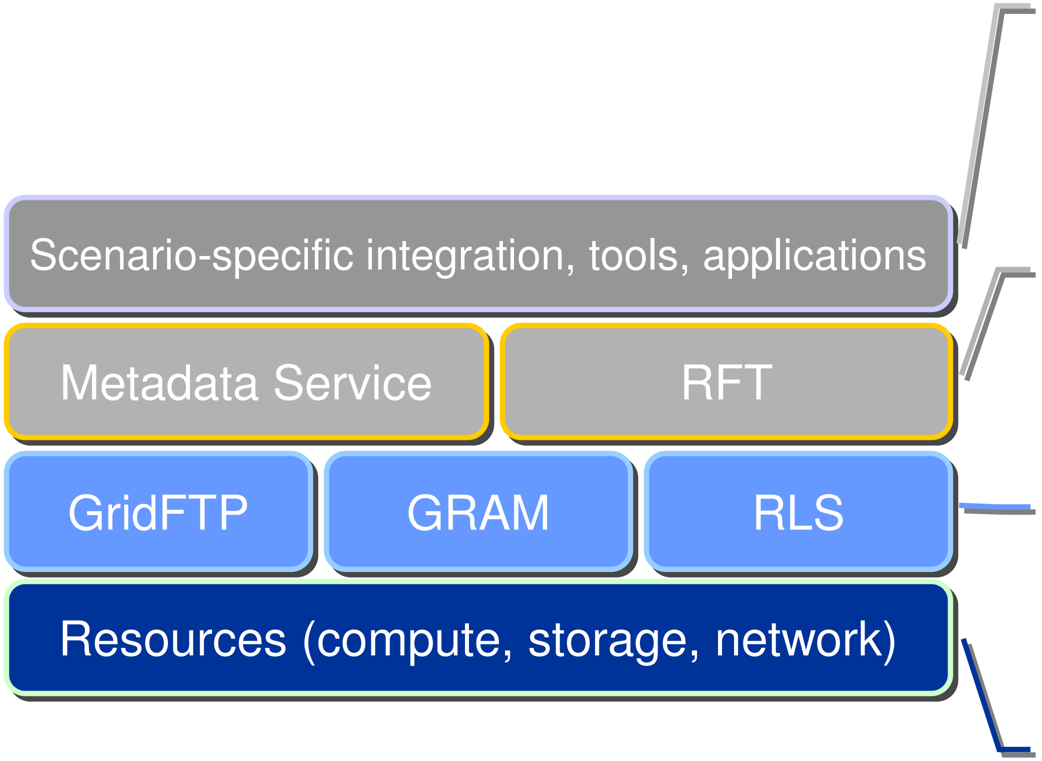
Globus Toolkit: Open Source Grid Infrastructure



Globus Toolkit v4
www.globus.org

Security	Data Mgmt	Execution Mgmt	Info Services	Common Runtime
	Data Replication			
Credential Mgmt	Replica Location	Grid Telecontrol Protocol		
Delegation	Data Access & Integration	Community Scheduling Framework	WebMDS	Python Runtime
Community Authorization	Reliable File Transfer	Workspace Management	Trigger	C Runtime
Authentication Authorization	GridFTP	Grid Resource Allocation & Management	Index	Java Runtime

Data Services Foundation



Scenario-specific integration, tools, applications

Metadata Service

RFT

GridFTP

GRAM

RLS

Resources (compute, storage, network)

- Applications leverage well designed APIs and interfaces
- Integration and tools built using language of choice

- Metadata catalog enables data discovery using domain metadata
- Reliable File Transfer ensures robust replication

- GridFTP interface to storage
- GRAM interface to computation
- RLS for file replica cataloging

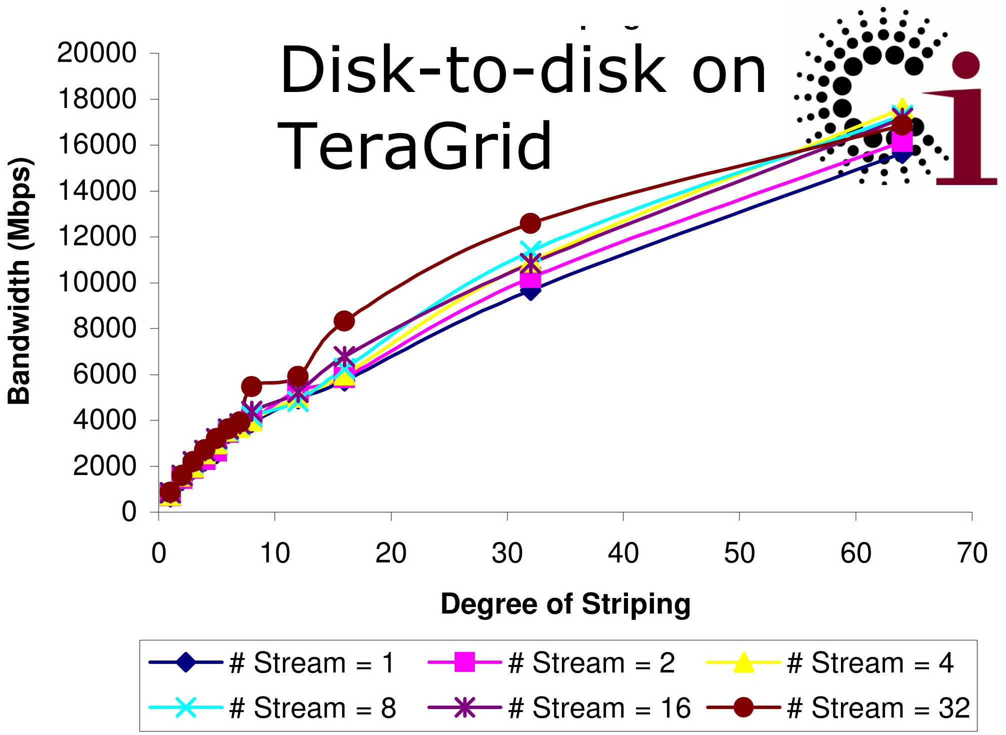
- Multiple Linux clusters
- Disk and tape storage
- Wide spectrum of network bandwidth

GT4 Data Management

- **Stage/move** large data to/from nodes
  - GridFTP, Reliable File Transfer (RFT)
  - Alone, and integrated with GRAM
- **Locate** data of interest
  - Replica Location Service (RLS)
- **Replicate** data for performance/reliability
  - Distributed Replication Service (DRS)
- Provide **access** to diverse data sources
  - File systems, parallel file systems, hierarchical storage: GridFTP
  - Databases: OGSA DAI

GridFTP in GT4

- 100% Globus code
  - ◆ No licensing issues
  - ◆ Stable, extensible
- IPv6 Support
- XIO for different transports
- Striping → multi-Gb/sec wide area transport
  - ◆ 27 Gbit/s on 30 Gbit/s link
- Pluggable
  - ◆ Front-end: e.g., future WS control channel
  - ◆ Back-end: e.g., HPSS, cluster file systems
  - ◆ Transfer: e.g., UDP, NetBLT transport



Disk-to-disk on TeraGrid

GridFTP: Secure, High Performance Data Transport

- Integrated instrumentation: Developers can use client API and plug-in mechanism to leverage different instrumentation
  - Performance markers
  - Restart markers
  - Throughput performance
  - Netlogger style performance
- Logging/audit trail: Extensive logging in the server
  - Multiple log levels: ERROR, WARN, INFO, DUMP, ALL
  - Log to stdio, syslog, file, …
  - Log all connections/transfers to single file or unique files
  - Netlogger style logging
  - Control permissions on log files

GridFTP: Secure, High Performance Data Transport

- Parallel data streams
  - Multiple TCP streams between sender and receiver
  - Sender pushes multiple blocks in parallel streams
  - Blocks reassembled at receiving side and put into correct order
  - Protection against dropped packets for each stream
- TCP buffer size control
  - Tune buffers to latency of network
  - Regular FTP optimized for low latency networks, not tunable
- Dramatic improvements for high latency WAN transfers
  - 90% of network utilization possible
  - 27 GB/s achieved with commodity hardware

GridFTP: Secure, High Performance Data Transport

- Server-side computation
  - Extended retrieve (ERET), Extended store (ESTO)
  - Simple pre-processing (partial get, sub-sampling)
  - Can greatly reduce network load
  - Client must also support ESTO/ERET functionality
- Striped server configurations
  - Multiple server back ends act as single server
  - Underlying parallel file system accessible to all nodes
  - High performance requires capable parallel file system
  - Each node must read/write its blocks of file
  - Allows multiple levels of parallelism (CPU, bus, NIC, disk, etc.)
  - Client sees a single logical server

GridFTP: Secure, High Performance Data Transport

- Data Storage Interface (DSI)
  - Interfaces to various storage types
  - Implement simple functions such as send, receive, mkdir,…
  - DSI modules available for HPSS and SRB
- Globus FTP client library (API):
  - Integration of data transport capabilities directly into applications
  - Plug-in architecture for installing fault recovery and performance tuning algorithms
  - Asynchronous programming model

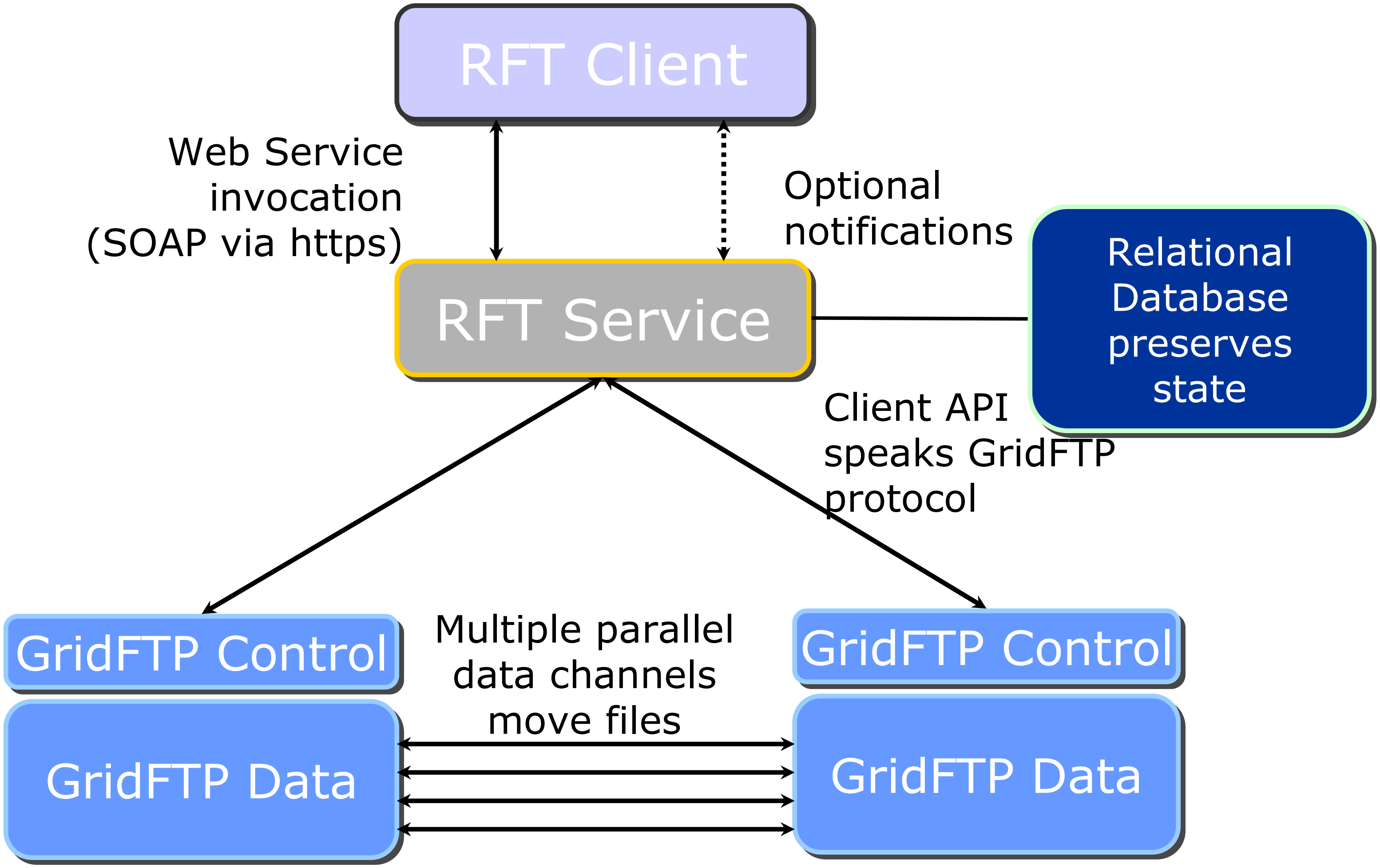
GridFTP: Client API

- Simple client flow comprises:
  1. Setup transfer details including number of parallel data channels, TCP buffer size, local buffer number and size
  2. Open connection to server URL and provide a “completion callback” function to be called when transfer complete
  3. Setup local buffers to hold read/write
  4. Register “data callback” function to be called for filling/flushing buffers
  5. Set “not done flag”
  6. Loop/wait until “completion callback” clears not done flag
- Work is done inside the “data callback” function
  - Local buffer filled with data (receiver) & ready to be flushed
  - Receive the offset into the file and any error code
  - fseek() to the correct place and fwrite() to file
  - Register another empty buffer/callback combination

GridFTP: Tool Mechanics

- Server mechanics
  - `globus-gridftp-server`
  - Usually runs as root
  - Usually run as a daemon; connections fork new process and setuid
  - Can run inetd/xinetd if so desired
  - Port 2811 is standard but is configurable
  - Logging and security highly configurable
- Client mechanics
  - `globus-url-copy`
  - Options for parallel channels, TCP buffer size, data buffer size, debugging, recursive directory transfers, etc.

Reliable File Transfer (RFT)


RFT Client
Web Service invocation (SOAP via https)
Optional notifications
RFT Service
Relational Database preserves state
Client API speaks GridFTP protocol
GridFTP Control
GridFTP Data
Multiple parallel data channels move files
GridFTP Control
GridFTP Data

Has transferred >900,000 files.

Globus RFT for Robust Data Management

- Supports concurrency
  - Multiple files in transit at any given time
  - Useful when transferring many small files
- Restart markers saved by service in database
  - Failed transfers restarted from where left off
- Client need not stay connected during transfers
  - Submit RFT transfer then grab laptop and go
- Clients check status in two ways
  - Subscribe to notifications from RFT service
  - Poll service to find status of transfers

Globus RFT for Robust Data Management

- Exposes WSRF compliant interface
  - Code RFT client using favorite Web services tools
- Single RFT service fronts multiple RFT resources
  - Each “user” can have separate resource
  - Each resource maintains own queue, notifications, lifetime
- Delete sets of files/directories on a GridFTP server
- Configurable exponential back off before retrying failed transfer
- Transfer all or none option
- Configurable # of concurrent transfers per container, request
- Configurable number of retries for failed transfers per request

RFT: Tool Mechanics

- RFT Service
  - Runs in Globus Java WS container/Tomcat
  - Uses JDBC capable database; PostgreSQL and MySQL most widely tested and used
- RFT clients
  - rft and rft-delete: simple clients, not intended for production use
  - Recommend application-specific Web Services clients developed against the service WSDL

Globus RLS for File Replica Management

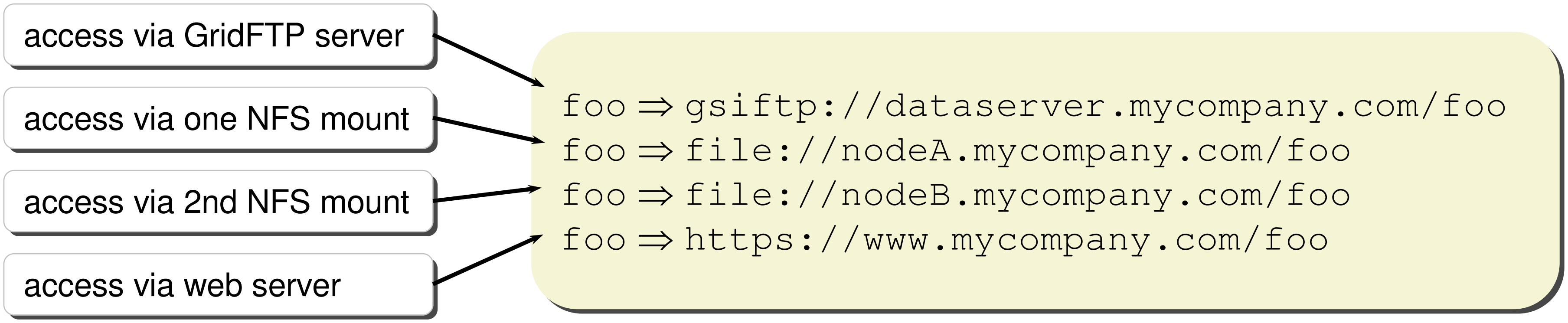
- Why replicate files?
  - Fault tolerance: avoid single points of failure
  - Reduce latency: use “nearest” copy
- Use GridFTP and RFT to move the files
  - Fast, robust transfer but no replica management
- Globus Replica Location Service (RLS)
  - Registry recording file locations
  - Enables discovery of replicas
  - Distributed catalog for scalability/fault tolerance
  - Capable of tracking tens of millions of files across distributed sites

Globus RLS for File Replica Management

- Maintains mappings between logical identifiers and target names
- Logical identifier or Logical File Name (LFN)
  - Location-independent identifier (name)
  - Example: `foo`
- Target name or Physical File Name (PFN)
  - Specific file identifier such as a URL
  - E.g.: `gsiftp://myserver.mycompany.com/foo`
- RLS maps between LFNs and PFNs
  - `foo` ⇒ `gsiftp://myserver.mycompany.com/foo`

Globus RLS for File Replica Management

- LFN to PFN mappings are often many-to-one
- Multiple PFNs may indicate different access to a file



```
foo ⇒ gsiftp://dataserver.mycompany.com/foo
foo ⇒ file://nodeA.mycompany.com/foo
foo ⇒ file://nodeB.mycompany.com/foo
foo ⇒ https://www.mycompany.com/foo
```

Globus RLS for File Replica Management

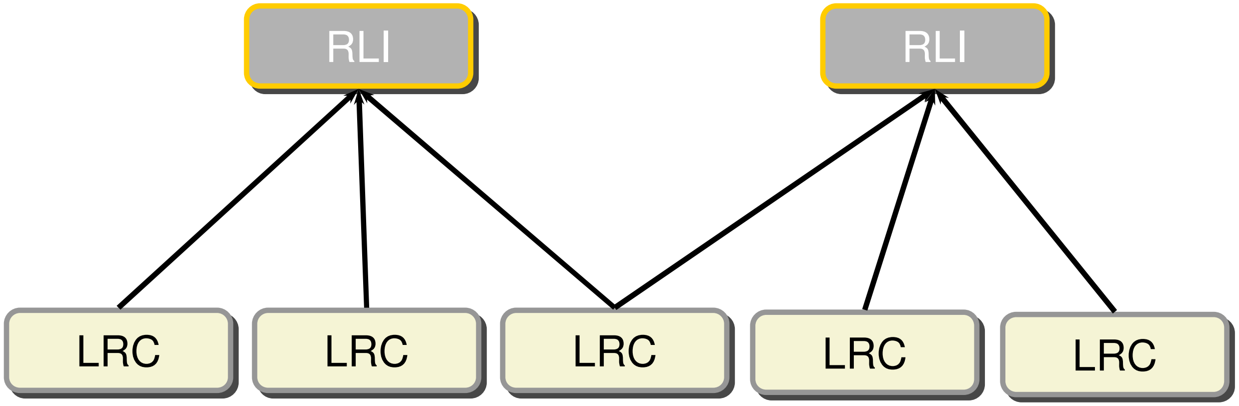
- Local replica catalog (LRC): Catalog of LFN to PFN mappings
- LRCs contain consistent information about local to target mappings

Local Replica Catalog (LRC)

fee ⇒ gsiftp://dataserver.mycompany.com/fee
fii ⇒ file://nodeA.mycompany.com/fii
foo ⇒ file://nodeB.mycompany.com/foo
fum ⇒ https://www.mycompany.com/fum

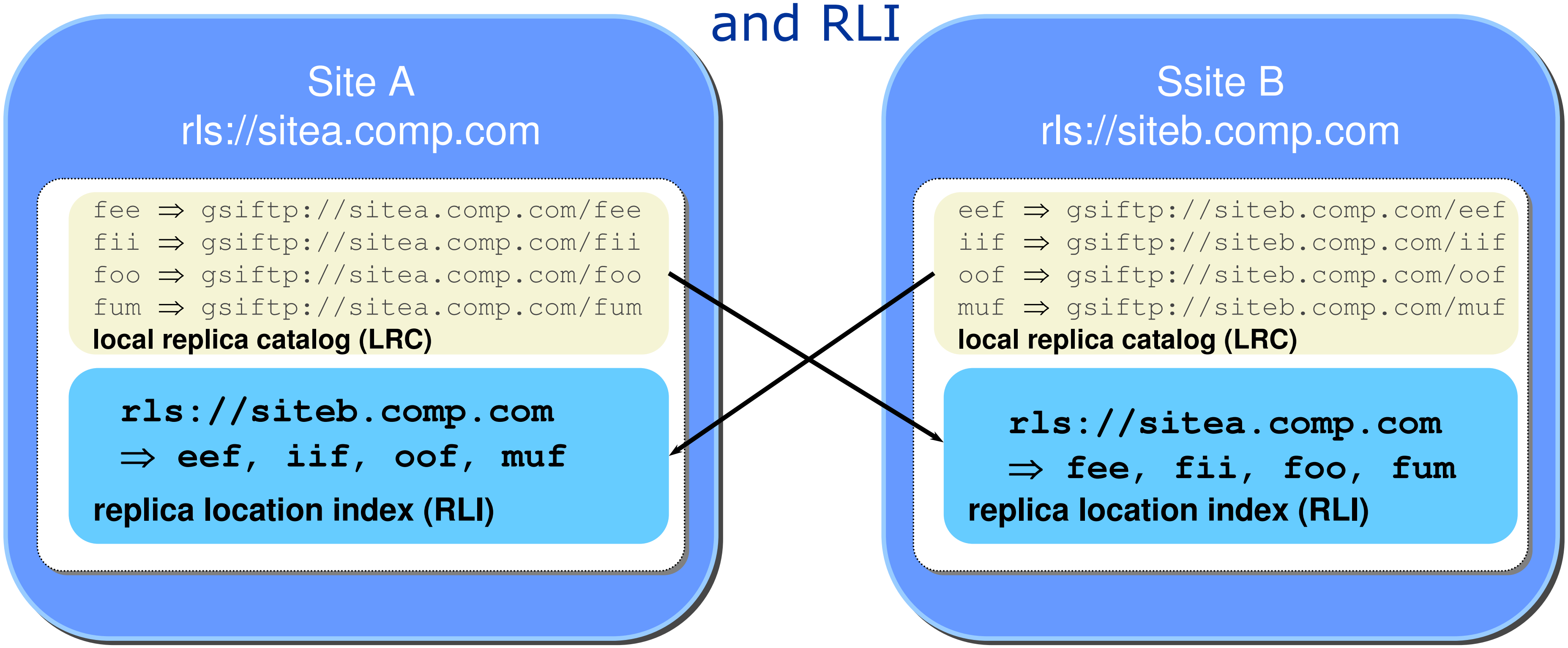
Globus RLS for File Replica Management

- Replica Location Index (RLI): Aggregate information about one or more LRCs
- Only the LFN content for LRC is aggregated
  - Each configured LRC sends list of LFNs to LRCs
  - PFNs and mappings **not** aggregated



Globus RLS for File Replica Management

Each *site* represented by a RLS server instance with both LRC and RLI



Site A
rls://sitea.comp.com

fee ⇒ gsiftp://sitea.comp.com/fee
fii ⇒ gsiftp://sitea.comp.com/fii
foo ⇒ gsiftp://sitea.comp.com/foo
fum ⇒ gsiftp://sitea.comp.com/fum
local replica catalog (LRC)

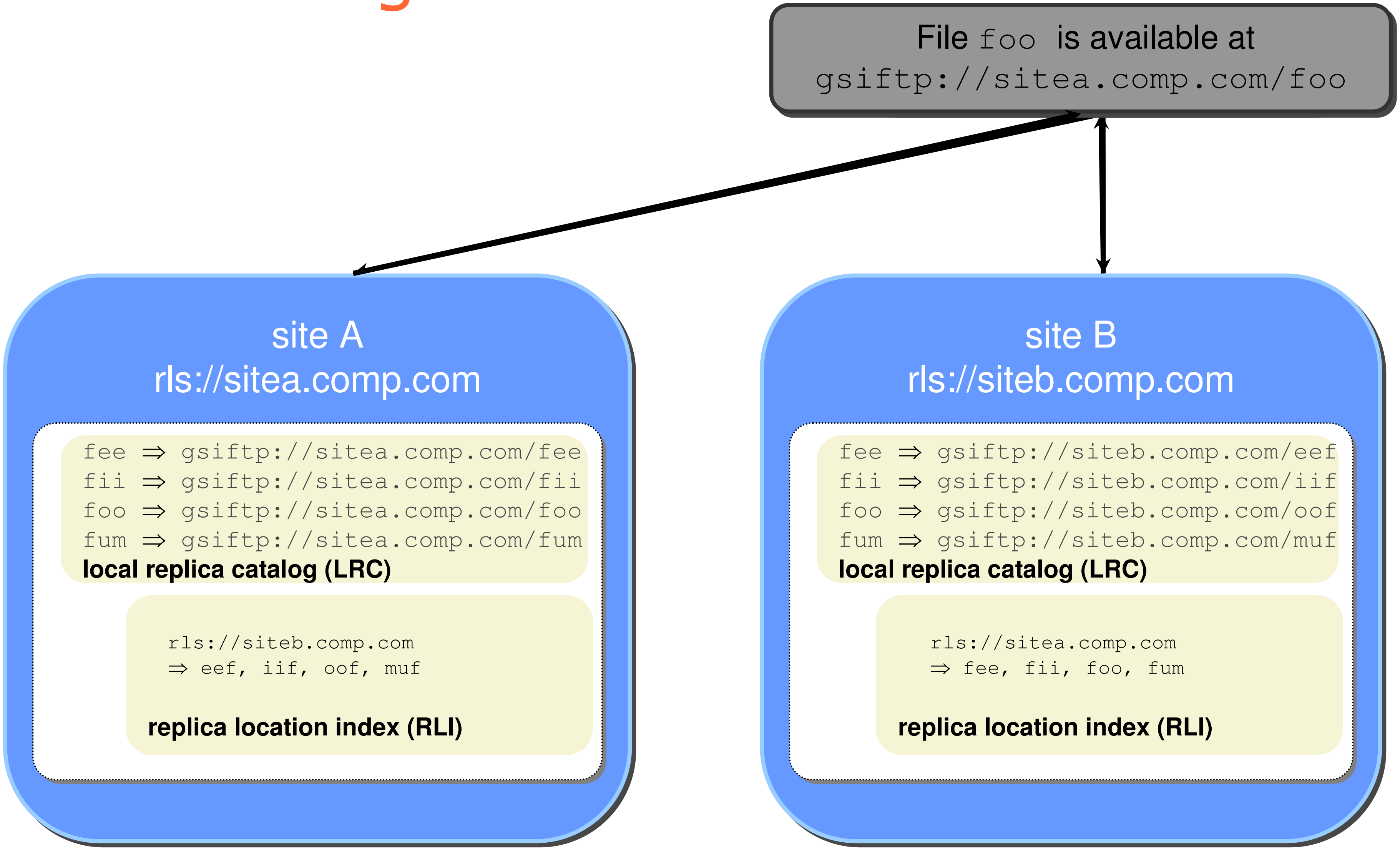
rls://siteb.comp.com
⇒ eef, iif, oof, muf
replica location index (RLI)

Ssite B
rls://siteb.comp.com

eef ⇒ gsiftp://siteb.comp.com/eef
iif ⇒ gsiftp://siteb.comp.com/iif
oof ⇒ gsiftp://siteb.comp.com/oof
muf ⇒ gsiftp://siteb.comp.com/muf
local replica catalog (LRC)

rls://sitea.comp.com
⇒ fee, fii, foo, fum
replica location index (RLI)

Finding Files Across the Grid



File `foo` is available at
`gsiftp://sitea.comp.com/foo`

site A
rls://sitea.comp.com

```
fee ⇒ gsiftp://sitea.comp.com/fee
fii ⇒ gsiftp://sitea.comp.com/fii
foo ⇒ gsiftp://sitea.comp.com/foo
fum ⇒ gsiftp://sitea.comp.com/fum
```

local replica catalog (LRC)

```
rls://siteb.comp.com
⇒ eef, iif, oof, muf
```

replica location index (RLI)

site B
rls://siteb.comp.com

```
fee ⇒ gsiftp://siteb.comp.com/eef
fii ⇒ gsiftp://siteb.comp.com/iif
foo ⇒ gsiftp://siteb.comp.com/oof
fum ⇒ gsiftp://siteb.comp.com/muf
```

local replica catalog (LRC)

```
rls://sitea.comp.com
⇒ fee, fii, foo, fum
```

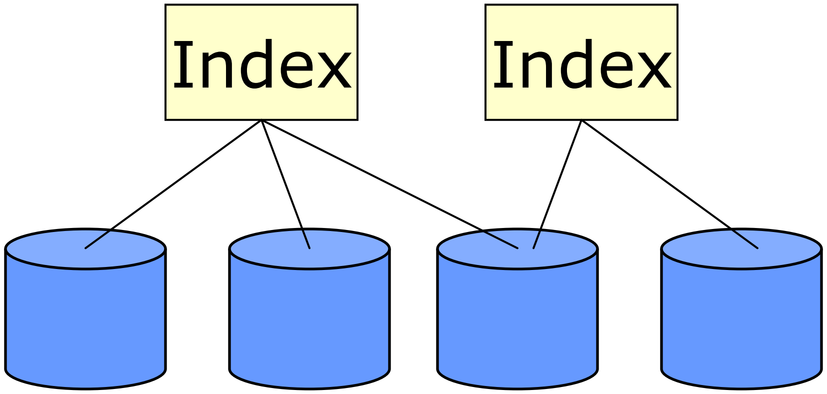
replica location index (RLI)

Globus RLS for File Replica Management

- **Soft state update** from LRCs to RLIs
  - Relaxed consistency of index
  - Tunable depending on desired load
- Two alternative update methods supported
  - Full list updates send entire list of LFNs periodically, partial updates in between
    - Complete list means always accurate
    - Large lists put drain on network, CPU, storage
  - Optional compressed bloom filter or hash
    - Compression relieves load on network, CPU, storage
    - False positives are possible (tunable rate)

Replica Location Service

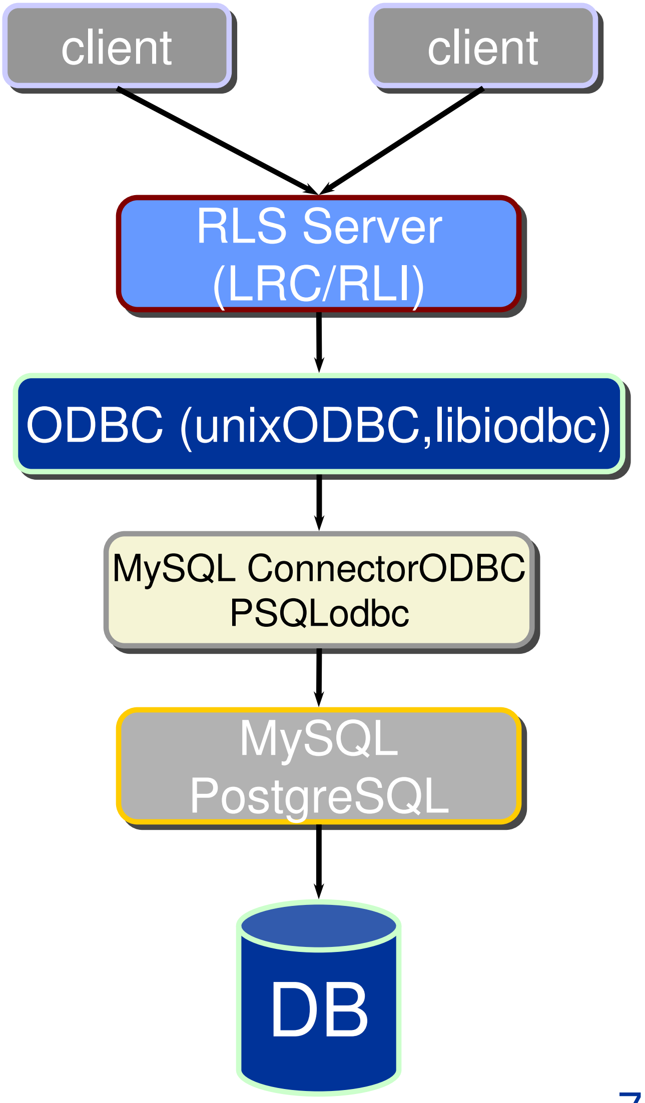
- Identify location of files via logical to physical name map
- Distributed indexing of names, fault tolerant update protocols
- GT4 version scalable & stable
- Managing ~40 million files across ~10 sites



Local DB	Update send (secs)	Bloom filter (secs)	Bloom filter (bits)
10K	<1	2	1 M
1 M	2	24	10 M
5 M	7	175	50 M

RLS Mechanics

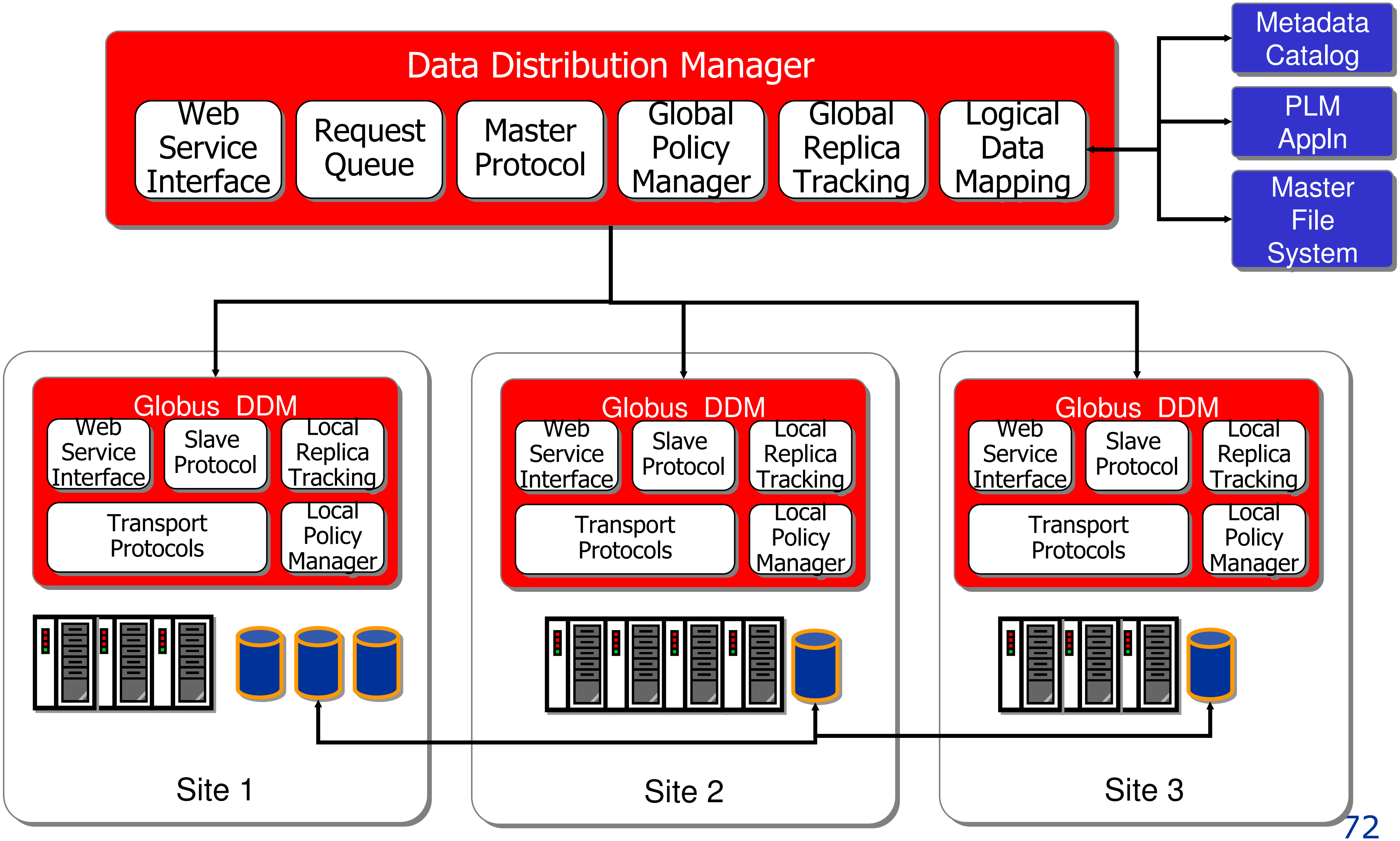
- Server runs as daemon
- Usually not run as root
- Use with any ODBC RDBMS
  - MySQL, PostgreSQL, Oracle most tested
- Multi-threaded, written in C
- GSI socket server
  - Single interface for both LRC and RLI
  - Differentiated by API calls
- ACL for types of access (admin, update, query, write, all)



RLS Mechanics

- Command line tools
  - ◆ `globus-rls-admin`: administration and on the fly configuration changes
  - ◆ `globus-rls-cli`: simple command line client for interacting with both LRC and RLI part of server
- Client APIs
  - ◆ C and Java APIs available
  - ◆ Functions to publish mappings, query, wildcard queries, administration tasks
  - ◆ “Bulk” versions of functions for publishing and queries on many objects

Data Management Architecture



LIGO Data Grid: Before & After

Before:

- Data replication via “FedEx” Grid
- Ad-hoc site-by-site idioms for finding data in storage
- Ad-hoc error prone mapping from metadata to file names
- Workflow limited to a single compute resource site

After:

- 24 x 7 x 365 continuous fault tolerant data streaming
- Single client tool for scientists and applications to find data
- Scientists concentrate on metadata and forget file names
- Multi-site planning of workflows across LIGO Data Grid

LIGO scientists searching for signals from neutron stars and black holes run **more jobs** across **more resources** and access **more data** using the LIGO Data Grid built on Globus**.**

Papers are published faster due to Globus and the LIGO Data Grid.

More Specifically, I May Want To …

- Create a service for use by my colleagues
- Manage who is allowed to access my service (or my experimental data or …)
- Ensure reliable & secure distribution of data from my lab to my partners
- Run 10,000 jobs on whatever computers I can get hold of
- Monitor the status of the different resources to which I have access

Globus Toolkit:
Open Source Grid Infrastructure

Globus Toolkit v4
www.globus.org

Security	Data Mgmt	Execution Mgmt	Info Services	Common Runtime
	Data Replication			
Credential Mgmt	Replica Location	Grid Telecontrol Protocol		
Delegation	Data Access & Integration	Community Scheduling Framework	WebMDS	Python Runtime
Community Authorization	Reliable File Transfer	Workspace Management	Trigger	C Runtime
Authentication Authorization	GridFTP	Grid Resource Allocation & Management	Index	Java Runtime

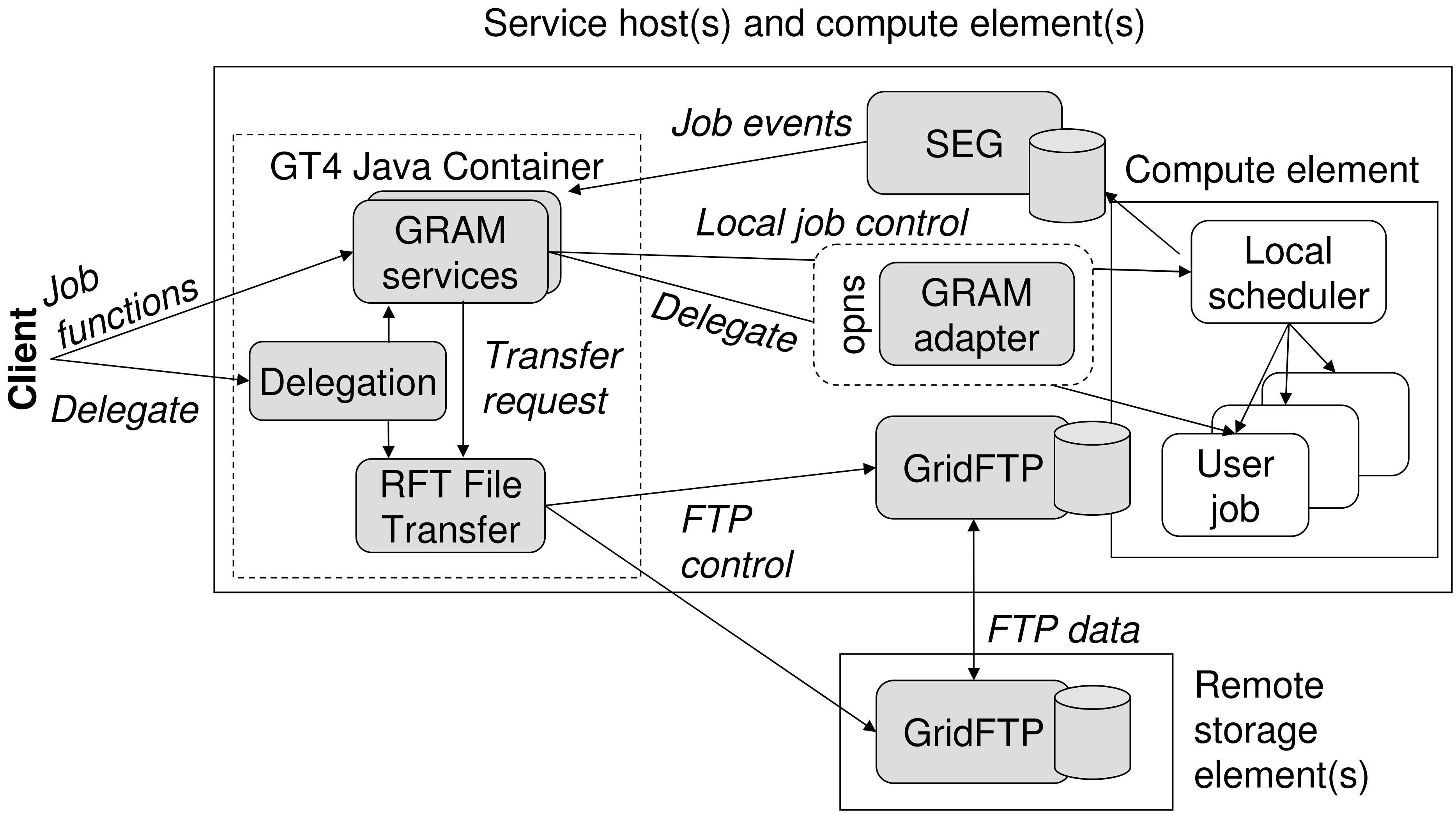
Execution Management (GRAM)

- Common WS interface to schedulers
  - Unix, Condor, LSF, PBS, SGE, …
- More generally: interface for process execution management
  - Lay down execution environment
  - Stage data
  - Monitor & manage lifecycle
  - Kill it, clean up
- A basis for application-driven provisioning

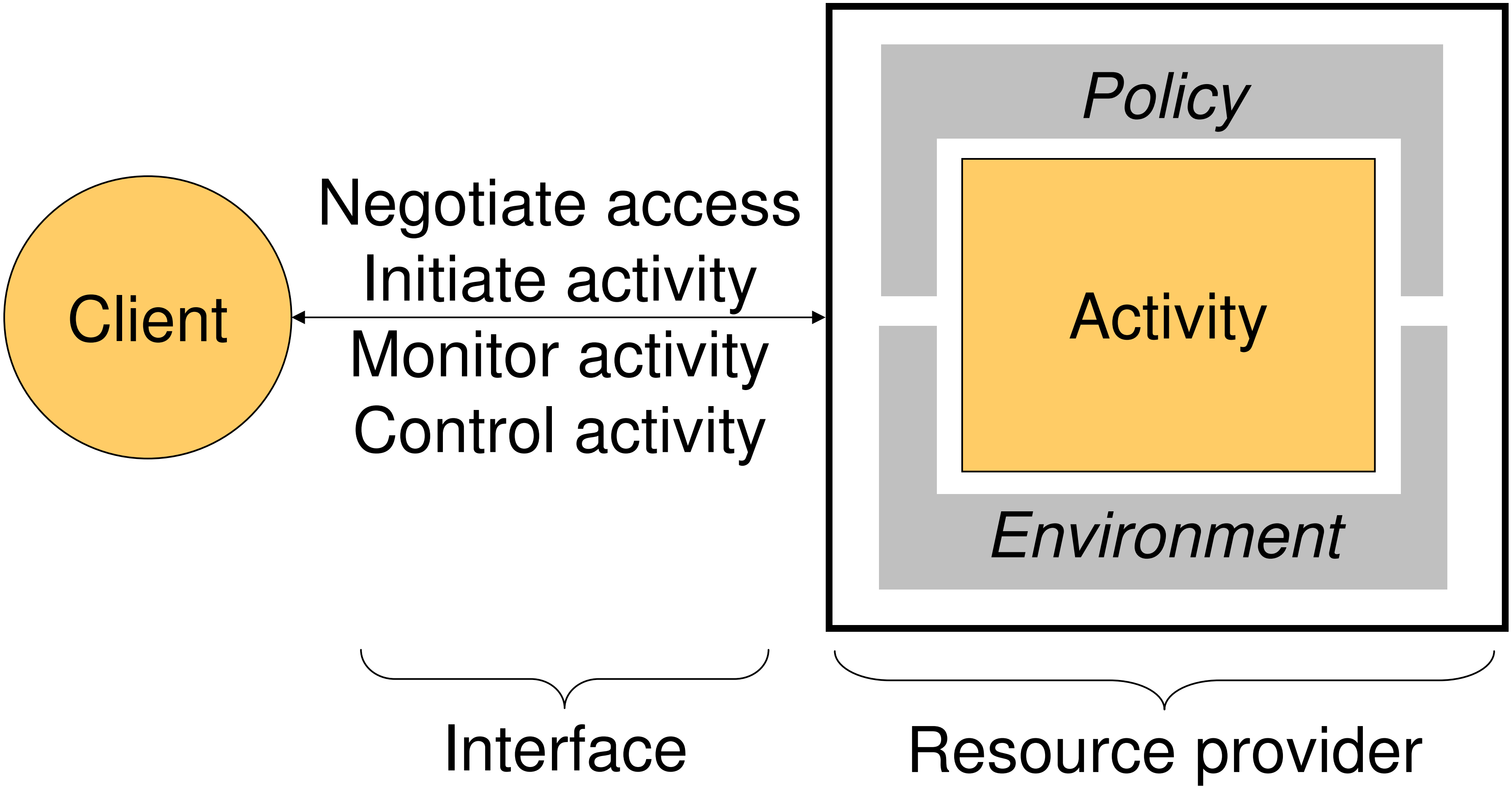
GRAM4:
A Big Advance over GRAM2

- Big scalability/performance improvements
  - 32,000 active jobs (GRAM2 max ~100)
  - Ability to manage load on control node
  - Reuse delegated credentials
- New functionality
  - Flexible authorization
  - Modular LRM interface
  - Notifications
  - JSDL support
  - Advance reservation, BES support (soon)

GT4 WS GRAM Architecture



Workspace Service: The Hosted Activity



Client

Negotiate access
Initiate activity
Monitor activity
Control activity

Policy

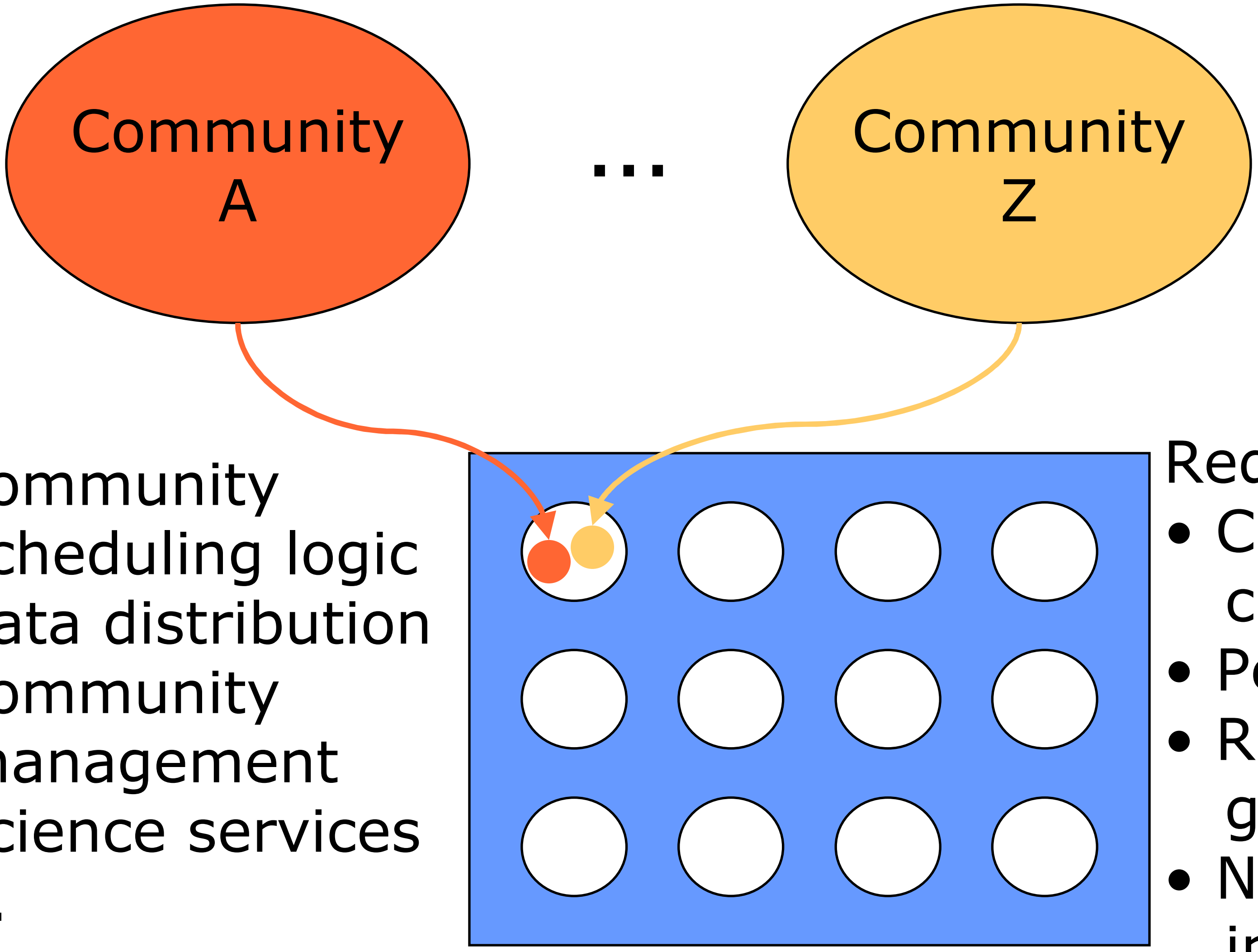
Activity

Environment

Interface

Resource provider

Dynamic Service Deployment



- Community scheduling logic
- Data distribution
- Community management
- Science services
- …

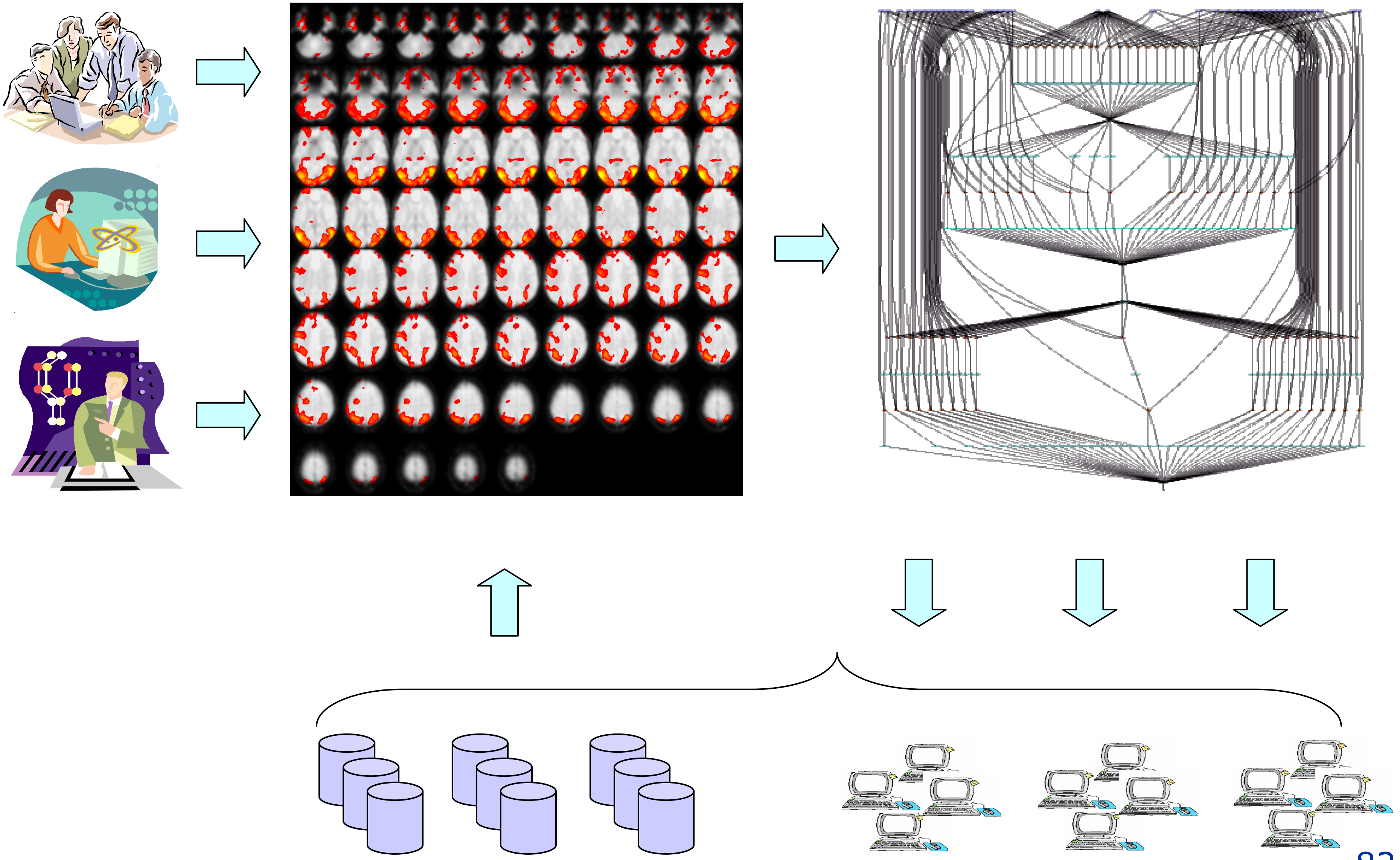
Requirements:

- Community control
- Persistence
- Resource guarantees
- Non-interference

Case Study: Functional MRI (fMRI) Data Center

- An online repository of neuroimaging data
  - A typical study comprises 3 groups, 20 subjects/gp, 5 runs/sub, 300 volumes/run
    - → 90,000 volumes, 60 GB raw data
    - → 1.2 million files processed data
  - 100s of such studies in total
- Many users analyze this data
  - Wide range of complex analysis procedures
  - Testing → production
  - Ensemble: a set of data analyses by parameters, datasets

fMRI: A Broad Picture



Challenges

- Deluge of data: instrumentation, simulation
- Data analysis turns into data integration
- Community-wide collaboration
- Provenance: tracking, query, application
- Scalability: desktop to Grid
- Productivity: throughput, performance

Swift System

- Clean separation of logical/physical concerns
  - **XDTM** specification of logical data structures

+ Concise specification of parallel programs
  - **SwiftScript**, with iteration, etc.

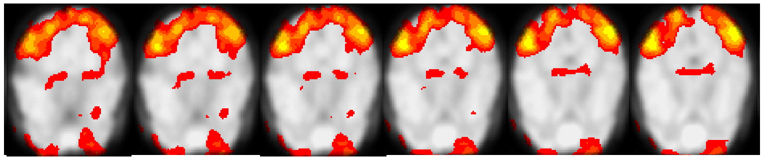
+ Efficient execution on distributed resources
  - Lightweight threading, dynamic provisioning, Grid interfaces, pipelining, load balancing

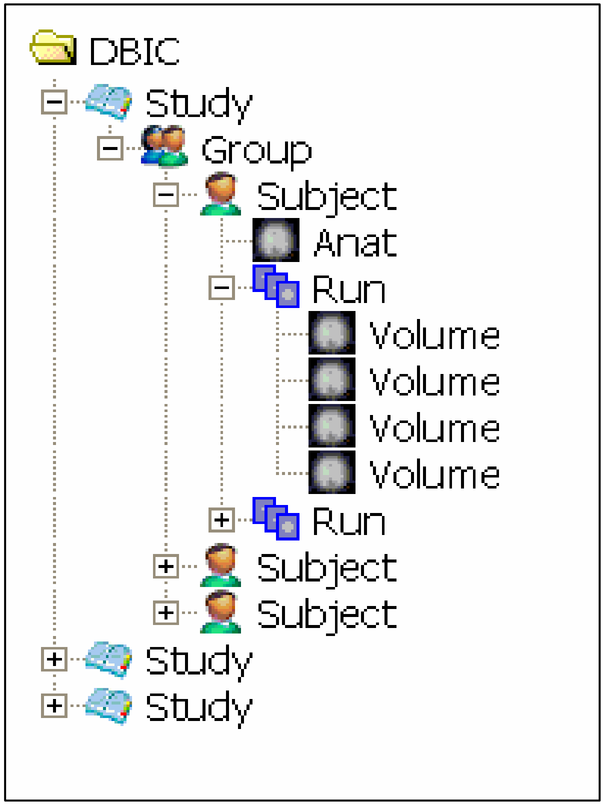
+ Rigorous provenance tracking and query
  - Virtual data schema & automated recording

→ **Improved usability and productivity**
  - Demonstrated in numerous applications

The Messy Data Problem (1)

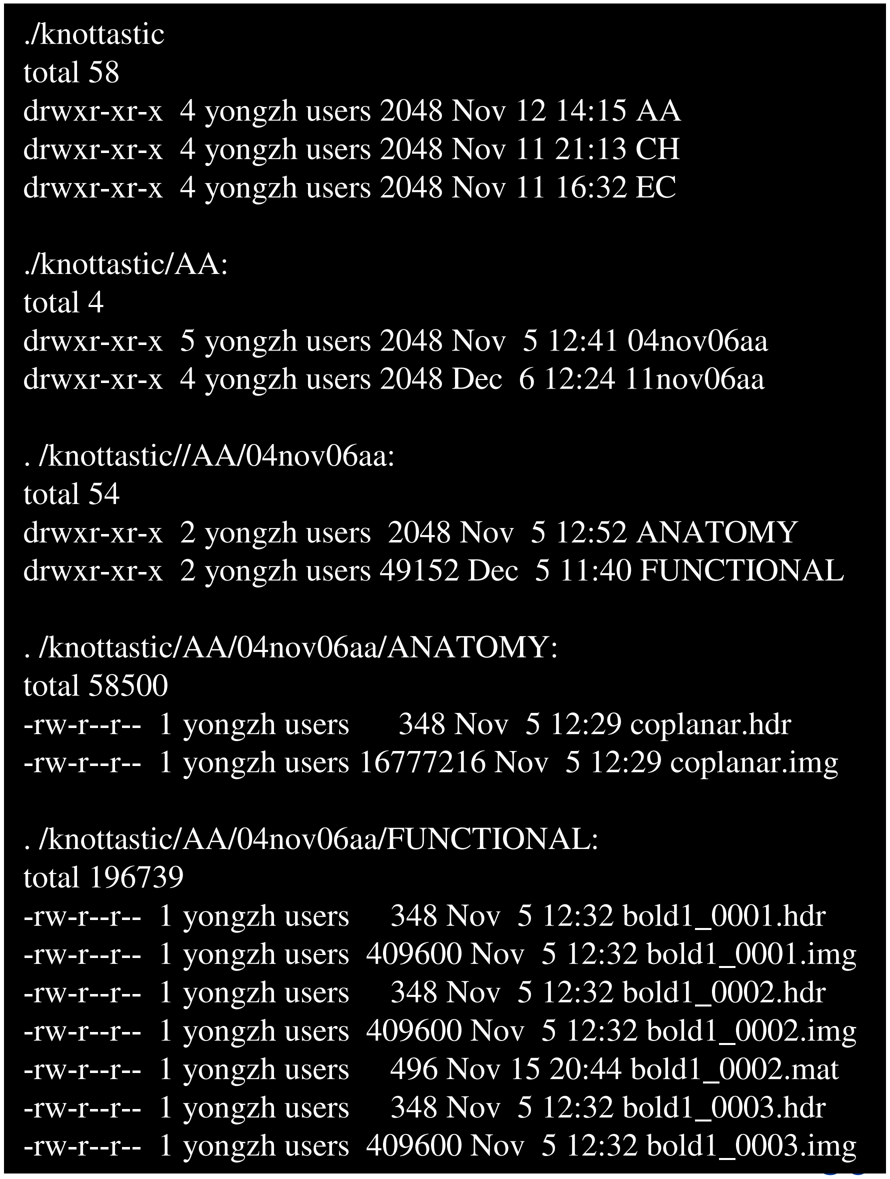
- Scientific data is often logically structured
  - E.g., hierarchical structure
  - Common to map functions over dataset members
  - Nested map operations can scale to millions of objects





The Messy Data Problem (2)

- But physically "messy"
- Heterogeneous storage format and access protocol
    - Logically identical dataset can be stored in textual File (e.g. CSV), spreadsheet, database, …
    - Data available from filesystem, DBMS, HTTP, WebDAV, …
- Metadata encoded in directory and file names
- Hinders program development, composition, execution



```
./knottastic
total 58
drwxr-xr-x  4 yongzh users 2048 Nov 12 14:15 AA
drwxr-xr-x  4 yongzh users 2048 Nov 11 21:13 CH
drwxr-xr-x  4 yongzh users 2048 Nov 11 16:32 EC

./knottastic/AA:
total 4
drwxr-xr-x  5 yongzh users 2048 Nov  5 12:41 04nov06aa
drwxr-xr-x  4 yongzh users 2048 Dec  6 12:24 11nov06aa

. /knottastic//AA/04nov06aa:
total 54
drwxr-xr-x  2 yongzh users  2048 Nov  5 12:52 ANATOMY
drwxr-xr-x  2 yongzh users 49152 Dec  5 11:40 FUNCTIONAL

. /knottastic/AA/04nov06aa/ANATOMY:
total 58500
-rw-r--r--  1 yongzh users      348 Nov  5 12:29 coplanar.hdr
-rw-r--r--  1 yongzh users 16777216 Nov  5 12:29 coplanar.img

. /knottastic/AA/04nov06aa/FUNCTIONAL:
total 196739
-rw-r--r--  1 yongzh users    348 Nov  5 12:32 bold1_0001.hdr
-rw-r--r--  1 yongzh users  409600 Nov  5 12:32 bold1_0001.img
-rw-r--r--  1 yongzh users    348 Nov  5 12:32 bold1_0002.hdr
-rw-r--r--  1 yongzh users  409600 Nov  5 12:32 bold1_0002.img
-rw-r--r--  1 yongzh users    496 Nov 15 20:44 bold1_0002.mat
-rw-r--r--  1 yongzh users    348 Nov  5 12:32 bold1_0003.hdr
-rw-r--r--  1 yongzh users  409600 Nov  5 12:32 bold1_0003.img
```

XML Dataset Typing & Mapping (XDTM)

- Describe logical structure by **XML Schema**
  - ◆ Primitive scalar types: int, float, string, date, …
  - ◆ Complex types (structs and arrays)
- Use **mapping descriptors** for mappings
  - ◆ How dataset elements are mapped to physical representations
  - ◆ External parameters (e. g. location)
- Use **XPath** for dataset selection

XDTM: XML Dataset Typing and Mapping for Specifying Datasets [EGC05]

XDTM: Related Work

- Data format standardization
  - FITS, CDF, HDF-5, DICOM
- Data format description
  - DFDL [Beckerle,Westhead04] embeds annotations with XML Schema
  - PADS [Fisher,Gruber05], PADX [Fernandez,Fisher06], declarative specs of physical layout and semantic properties
- Logical object
  - ADO [Microsoft01], in memory relational model
  - SDO [Beatty,Brodsky03], logical data model for J2EE programming

XDTM: Implementation

- Virtual integration
  - Each data source treated as virtual XML source
  - Data structure defined as XML schema
  - Mapper responsible for accessing source and translating to/from XML representation
  - Bi-directional
- Common mapping interface
  - Data providers implement the interface
    - Responsible for data access details
  - Standard mapper implementations provided
    - String, file system, CSV, …

SwiftScript

- Typed parallel programm [SIGMOD05, Springer06]
  - XDTM as data model and type system
  - Typed dataset and procedure definitions
- Scripting language
  - Implicit data parallelism
  - Program composition from procedures
  - Control constructs (foreach, if, while, …)

Clean application logic
Type checking
Dataset selection, iteration
Discovery by types
Type conversion

A Notation & System for Expressing and Executing Cleanly Typed Workflows on Messy Scientific Data [SIGMOD05]

SwiftScript: Related Work

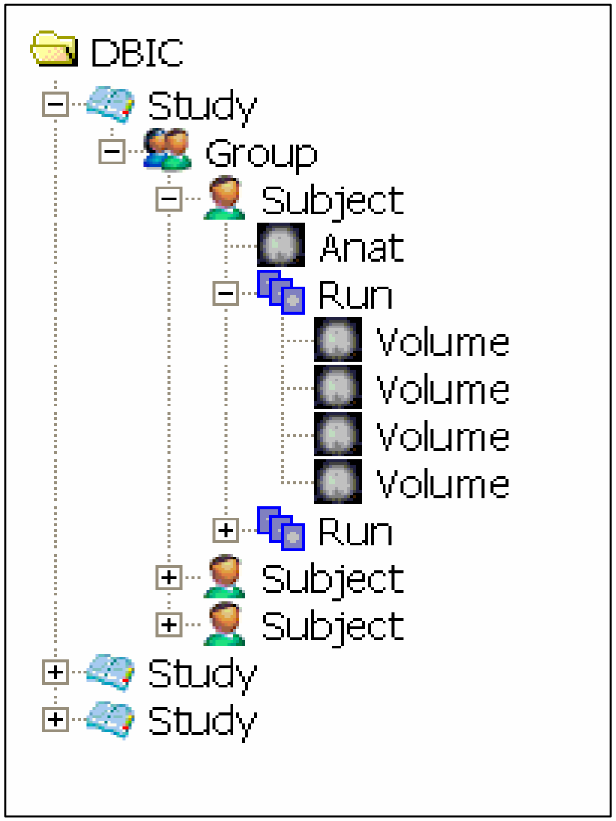
- Coordination language
  - Linda[Ahuja,Carriero86], Strand[Foster,Taylor90], PCN[Foster92]
  - Durra[Barbacci,Wing86], MANIFOLD[Papadopoulos98]
  - Components programmed in specific language (C, FORTRAN) and linked with system
- “Workflow” languages and systems
  - Taverna[Oinn,Addis04], Kepler[Ludäscher,Altintas05], Triana [Churches,Gombas05], Vistrail[Callahan,Freire06], DAGMan, Star-P
  - XPDL[WfMC02], BPEL[Andrews,Curbera03], and BPML[BPML02], YAWL[van de Aalst,Hofstede05], Windows Workflow Foundation [Microsoft05]

Related Work

	SwiftScript	BPEL	XPDL	MW Wflow	DAGMan	Tavena	Triana	Kepler	Vistrail	Star-P
Scales to Grids	++	-	-	-	++	-	-	-	-	+
Typing	++	++	++	++	-	-	-	+	-	+
Iteration	++	-/+	-	+	-	-	-	+	-	+
Scripting	++	-	-	+	+	+	-	-	+	++
Dataset Mapping	+	-	-	-	-	-	-	-	-	-
Service Interop	+	-	+	-	-	-	-	+	-	-
Subflow/comp.	+	-	+	+	-	-	+	+	-	+
Provenance	+	-	-	+	-	+	-	+	+	-
Open source	+	+	+	-	+	+	+	+	+	-

“A 4x200 flow leads to a 5 MB BPEL file … chemists were not able to write in BPEL” [Emmerich,Buchart06]

fMRI Type Definitions in SwiftScript



```
DBIC
  Study
    Group
      Subject
        Anat
        Run
          Volume
          Volume
          Volume
          Volume
        Run
      Subject
      Subject
  Study
  Study
```

Simplified version of fMRI AIRSN Program (Spatial Normalization)

```
type Study {
        Group g[ ];
}

type Group {
        Subject s[ ];
}

type Subject {
        Volume anat;
        Run run[ ];
}

type Run {
        Volume v[ ];
}

type Volume {
        Image img;
        Header hdr;
}
```

```
type Image {};

type Header {};

type Warp {};

type Air {};

type AirVec {
        Air a[ ];
}

type NormAnat {
        Volume anat;
        Warp aWarp;
        Volume nHires;
}
```

Type Definitions in XML Schema

```
<xs:schema targetNamespace="http://www.fmri.org/schema/airsn.xsd"
           xmlns="http://www.fmri.org/schema/airsn.xsd"
           xmlns:xs="http://www.w3.org/2001/XMLSchema">
        <xs:simpleType name="Image">
             <xs:restriction base="xs:string"/>
        </xs:simpleType>
        <xs:simpleType name="Header">
             <xs:restriction base="xs:string"/>
        </xs:simpleType>
        <xs:complexType name="Volume">
            <xs:sequence>
                <xs:element name="img" type="Image"/>
                <xs:element name="hdr" type="Header"/>
            </xs:sequence>
        </xs:complexType>
        <xs:complexType name="Run">
            <xs:sequence minOccurs="0” maxOccurs="unbounded">
                <xs:element name="v" type="Volume"/>
             </xs:sequence>
        </xs:complexType>
</xs:schema>
```

AIRSN Program Definition

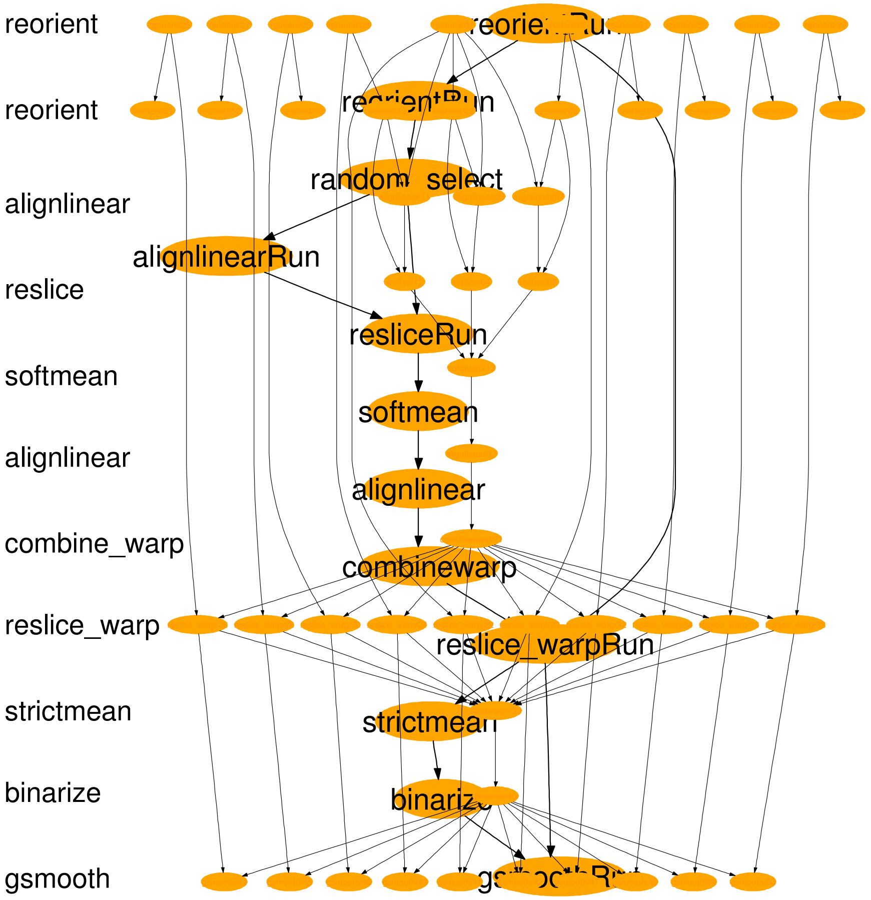
```
(Run snr) functional ( Run r, NormAnat a,
                       Air shrink ) {
    Run yroRun = reorientRun( r , "y" );
    Run roRun = reorientRun( yroRun , "x" );
    Volume std = roRun[0];
    Run rndr = random_select( roRun, 0.1 );
    AirVector rndAirVec = align_linearRun( rndr, std, 12, 1000, 1000, "81 3 3" );
    Run reslicedRndr = resliceRun( rndr, rndAirVec, "o", "k" );
    Volume meanRand = softmean( reslicedRndr, "y", "null" );
    Air mnQAAir = alignlinear( a.nHires, meanRand, 6, 1000, 4, "81 3 3" );
    Warp boldNormWarp = combinewarp( shrink, a.aWarp, mnQAAir );
    Run nr = reslice_warp_run( boldNormWarp, roRun );
    Volume meanAll = strictmean( nr, "y", "null" )
    Volume boldMask = binarize( meanAll, "y" );
    snr = gsmoothRun( nr, boldMask, "6 6 6" );
}
```

```
(Run or) reorientRun (Run ir,
                      string direction) {
    foreach Volume iv, i in ir.v {
        or.v[i] = reorient(iv, direction);
    }
}
```

Expressiveness

Lines of code with different encodings

Appln	Script	Generator	Swift Script
ATLAS1	49	72	**6**
ATLAS2	97	135	**10**
FILM1	63	134	**17**
FEAT	84	191	**13**
AIRSN	215	~400	**34**



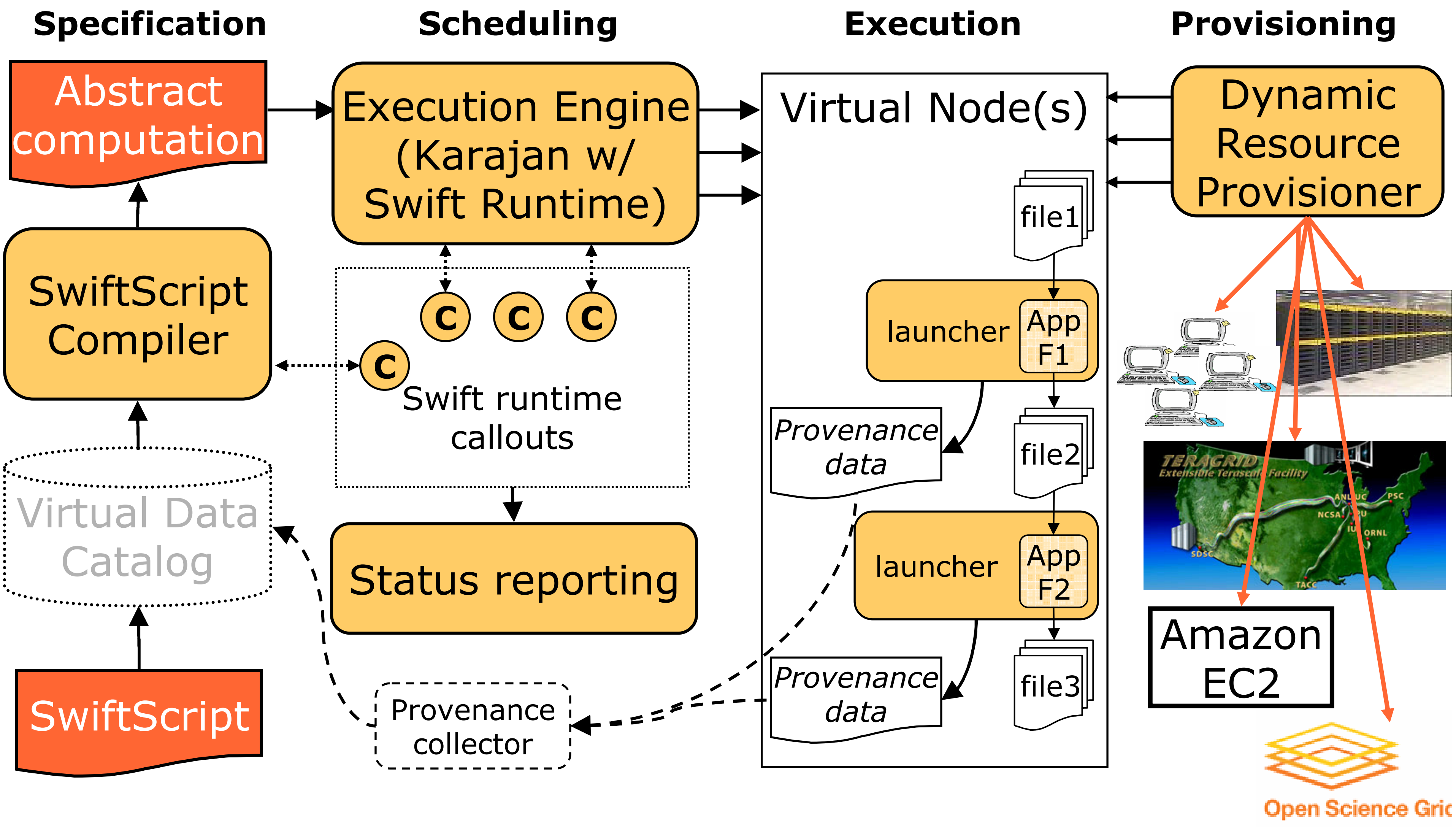
Collaboration with James Dobson, Dartmouth [SIGMOD05]

Swift Runtime System

- Runtime system for SwiftScript [SSDBM02,CIDR03,Springer06]
  - Populate, update, query virtual data products
  - Schedule, monitor, execute resulting computation on distributed Grid resources
  - Annotate virtual data products with customized metadata
  - Trace provenance of virtual data products
- Grid scheduling and optimization
  - Lightweight execution engine: Karajan
  - Dynamic resource provisioning
  - Site selection, data movement, caching
  - Pipelining, clustering, load balancing
  - Fault tolerance, exception handling

A Virtual Data System for Representing, Querying & Automating Data Derivation [SSDBM02]

Swift Architecture



Specification

Scheduling

Execution

Provisioning

Abstract computation

Execution Engine (Karajan w/ Swift Runtime)

Virtual Node(s)

Dynamic Resource Provisioner

SwiftScript Compiler

C C C C

Swift runtime callouts

file1

launcher App F1

Provenance data

file2

launcher App F2

Provenance data

file3

Virtual Data Catalog

Status reporting

SwiftScript

Provenance collector

Amazon EC2

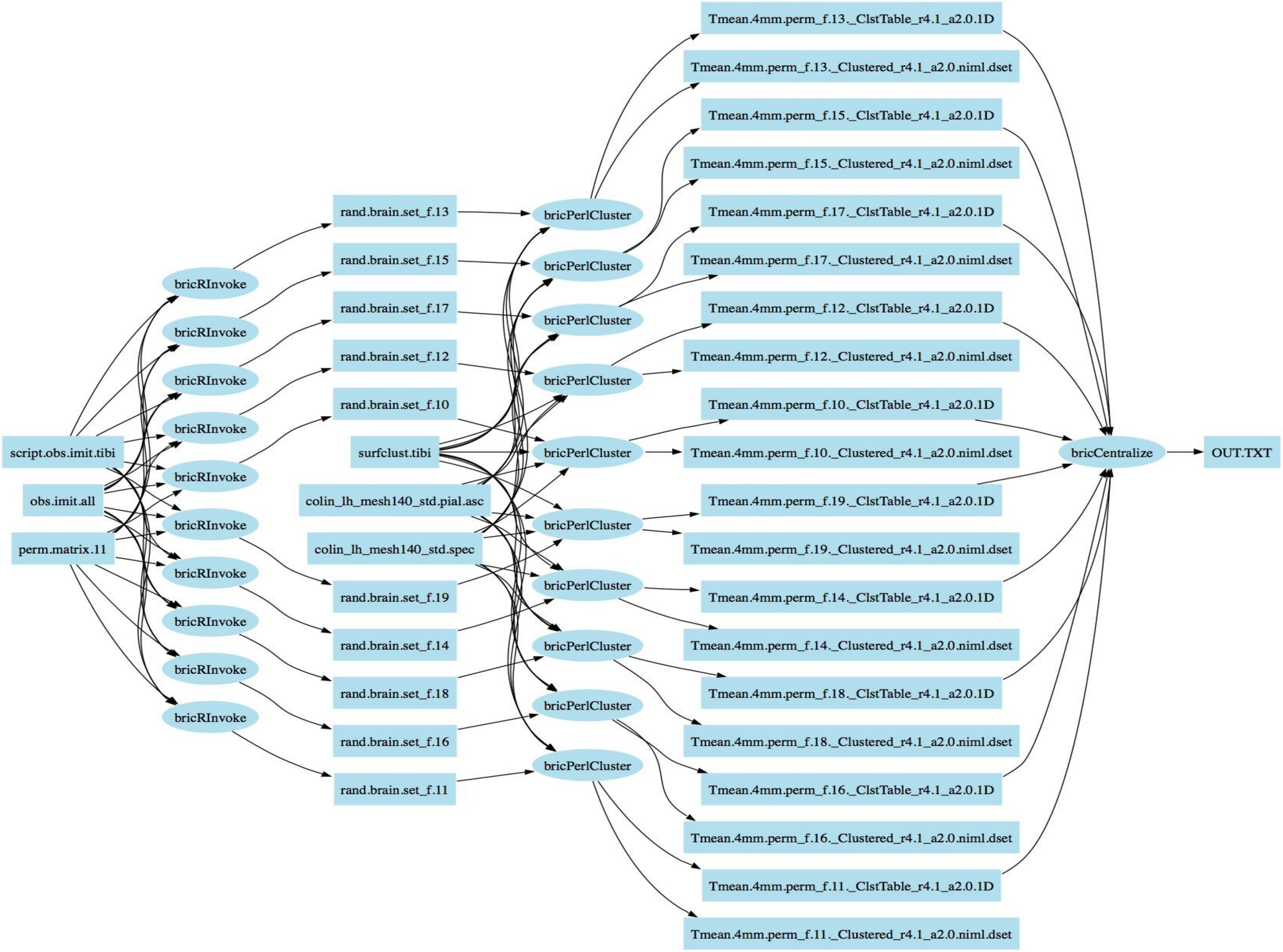
Swift uses Karajan Workflow Engine

- Fast, scalable threading model
- Suitable constructs for control flow
- Flexible task dependency model
  - “Futures” enable pipelining
- Flexible provider model allows for use of different run time environments
  - Job execution and data transfer
  - Flow controlled to avoid resource overload
- Workflow client runs from a Java container

Java CoG Workflow, Gregor von Laszewski et al., *Workflows for Science*, 2007

Application Example
ACTIVAL: Neural Activation Validation

- Identifies clusters of neural activity not likely to be active by chance:
  - switch labels of conditions for 1+ participants;
  - calculate delta values in each voxel;
  - re-calculate reliability of delta in each voxel; &
  - evaluate clusters found
- If clusters in data > majority of clusters found in permutations, then null hypothesis is refuted, indicating that clusters of activity found in experiment are not likely to be found by chance


script.obs.imit.tibi
obs.imit.all
perm.matrix.11
bricRInvoke
bricRInvoke
bricRInvoke
bricRInvoke
bricRInvoke
bricRInvoke
bricRInvoke
bricRInvoke
bricRInvoke
bricRInvoke
rand.brain.set_f.13
rand.brain.set_f.15
rand.brain.set_f.17
rand.brain.set_f.12
rand.brain.set_f.10
surfclust.tibi
colin_lh_mesh140_std.pial.asc
colin_lh_mesh140_std.spec
rand.brain.set_f.19
rand.brain.set_f.14
rand.brain.set_f.18
rand.brain.set_f.16
rand.brain.set_f.11
bricPerlCluster
bricPerlCluster
bricPerlCluster
bricPerlCluster
bricPerlCluster
bricPerlCluster
bricPerlCluster
bricPerlCluster
bricPerlCluster
bricPerlCluster
Tmean.4mm.perm_f.13._ClstTable_r4.1_a2.0.1D
Tmean.4mm.perm_f.13._Clustered_r4.1_a2.0.niml.dset
Tmean.4mm.perm_f.15._ClstTable_r4.1_a2.0.1D
Tmean.4mm.perm_f.15._Clustered_r4.1_a2.0.niml.dset
Tmean.4mm.perm_f.17._ClstTable_r4.1_a2.0.1D
Tmean.4mm.perm_f.17._Clustered_r4.1_a2.0.niml.dset
Tmean.4mm.perm_f.12._ClstTable_r4.1_a2.0.1D
Tmean.4mm.perm_f.12._Clustered_r4.1_a2.0.niml.dset
Tmean.4mm.perm_f.10._ClstTable_r4.1_a2.0.1D
Tmean.4mm.perm_f.10._Clustered_r4.1_a2.0.niml.dset
Tmean.4mm.perm_f.19._ClstTable_r4.1_a2.0.1D
Tmean.4mm.perm_f.19._Clustered_r4.1_a2.0.niml.dset
Tmean.4mm.perm_f.14._ClstTable_r4.1_a2.0.1D
Tmean.4mm.perm_f.14._Clustered_r4.1_a2.0.niml.dset
Tmean.4mm.perm_f.18._ClstTable_r4.1_a2.0.1D
Tmean.4mm.perm_f.18._Clustered_r4.1_a2.0.niml.dset
Tmean.4mm.perm_f.16._ClstTable_r4.1_a2.0.1D
Tmean.4mm.perm_f.16._Clustered_r4.1_a2.0.niml.dset
Tmean.4mm.perm_f.11._ClstTable_r4.1_a2.0.1D
Tmean.4mm.perm_f.11._Clustered_r4.1_a2.0.niml.dset
bricCentralize
OUT.TXT

SwiftScript Program ACTIVAL – Datatypes & Utilities

```
type script {}
type brainMeasurements{}
type precomputedPermutations{}
type brainClusterTable {}
type brainDatasets{ brainDataset b[]; }
type brainClusters{ brainClusterTable c[]; }
type fullBrainData {}
type fullBrainSpecs {}
type brainDataset {}

// Procedure to run "R" statistical package
(brainDataset t) bricRInvoke (script permutationScript, int iterationNo,
    brainMeasurements dataAll, precomputedPermutations dataPerm) {
       app { bricRInvoke @filename(permutationScript) iterationNo
                             @filename(dataAll) @filename(dataPerm); }
}

// Procedure to run AFNI Clustering tool
(brainClusterTable v, brainDataset t) bricCluster (script clusterScript,
  int iterationNo, brainDataset randBrain, fullBrainData brainFile,
  fullBrainSpecs specFile) {
       app { bricPerlCluster @filename(clusterScript) iterationNo
                            @filename(randBrain) @filename(brainFile)
                            @filename(specFile); }
}

// Procedure to merge results based on statistical likelhoods
(brainClusterTable t) bricCentralize ( brainClusterTable bc[]) {
      app { bricCentralize @filenames(bc); }
}
```

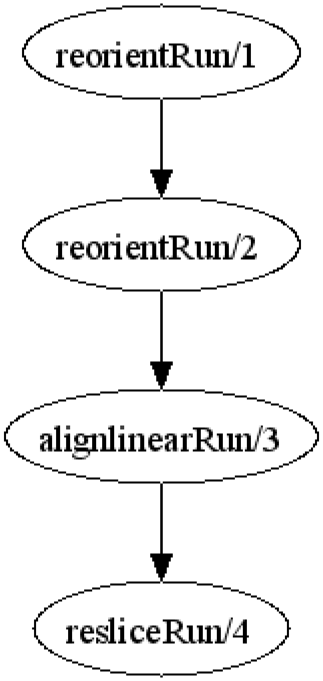
ACTIVAL: Dataset Iteration Procedures

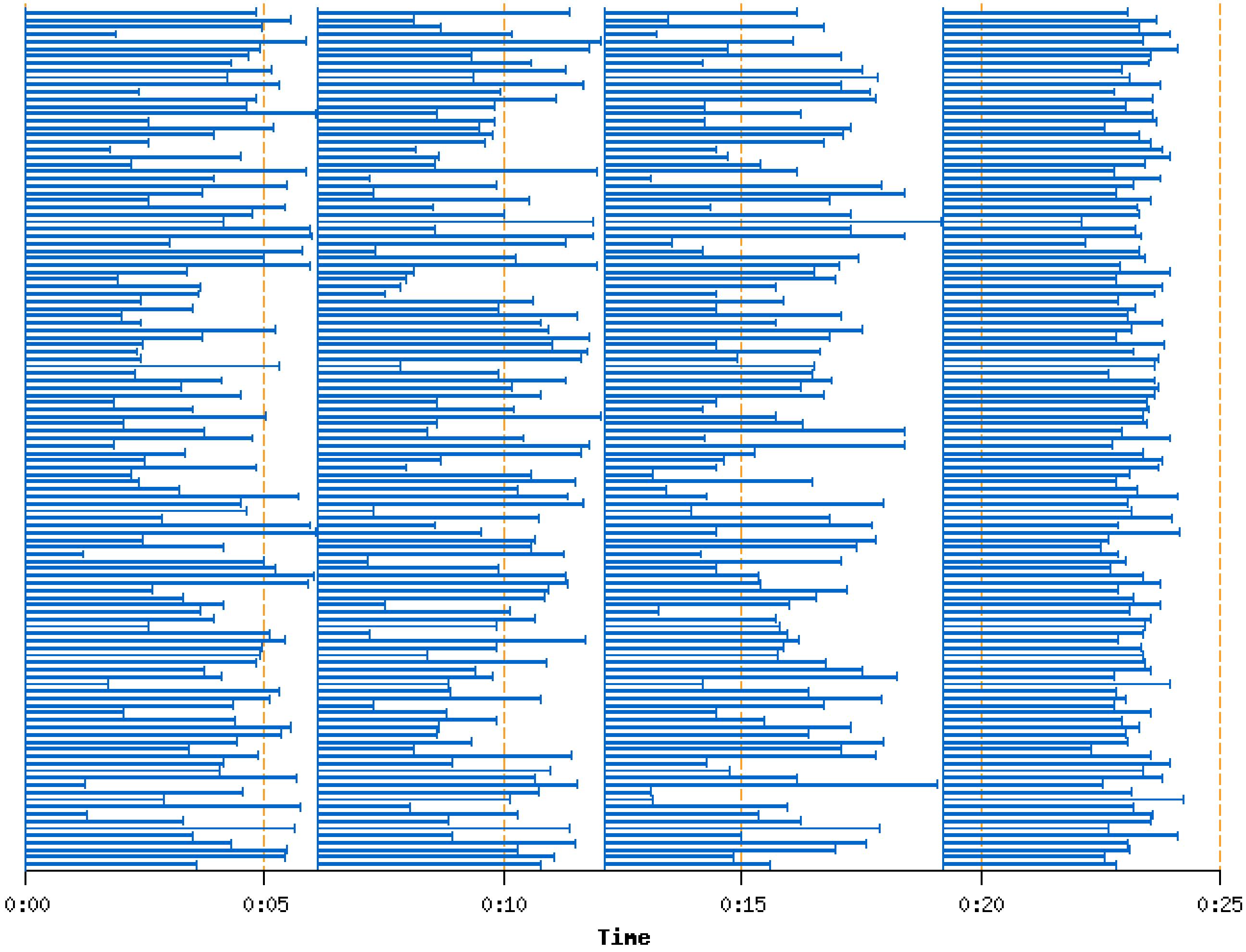
```
// Procedure to iterate over the data collection

(brainClusters randCluster, brainDatasets dsetReturn)
    brain_cluster(fullBrainData brainFile, fullBrainSpecs specFile)
{
  int sequence[]=[1:2000];

  brainMeasurements        dataAll<fixed_mapper; file="obs.imit.all">;
  precomputedPermutations  dataPerm<fixed_mapper; file="perm.matrix.11">;
  script                   randScript<fixed_mapper; file="script.obs.imit.tibi">;
  script                   clusterScript<fixed_mapper; file="surfclust.tibi">;
  brainDatasets            randBrains<simple_mapper; prefix="rand.brain.set">;

  foreach int i in sequence {
     randBrains.b[i] = bricRInvoke(randScript, i, dataAll, dataPerm);
     brainDataset rBrain=randBrains.b[i];
     (randCluster.c[i], dsetReturn.b[i]) =
        bricCluster(clusterScript, i, rBrain, brainFile,specFile);
  }
}
```

ACTIVAL: Main Program

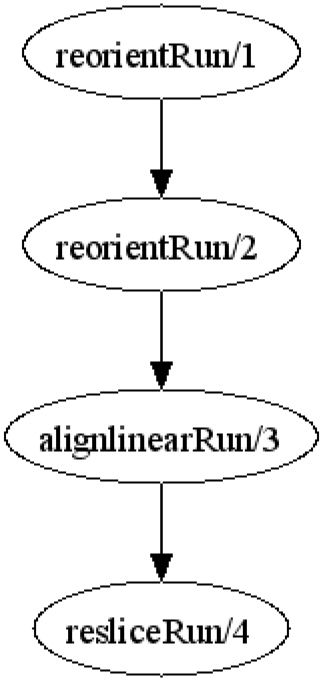
```
// Declare datasets

fullBrainData       brainFile<fixed_mapper; file="colin_lh_mesh140_std.pial.asc">;
fullBrainSpecs      specFile<fixed_mapper; file="colin_lh_mesh140_std.spec">;

brainDatasets       randBrain<simple_mapper; prefix="rand.brain.set">;
brainClusters       randCluster<simple_mapper; prefix="Tmean.4mm.perm",
                                suffix="_ClstTable_r4.1_a2.0.1D">;
brainDatasets       dsetReturn<simple_mapper; prefix="Tmean.4mm.perm",
                                suffix="_Clustered_r4.1_a2.0.niml.dset">;
brainClusterTable   clusterThresholdsTable<fixed_mapper; file="thresholds.table">;
brainDataset        brainResult<fixed_mapper; file="brain.final.dset">;
brainDataset        origBrain<fixed_mapper; file="brain.permutation.1">;


// Main program – executes the entire application

(randCluster, dsetReturn) = brain_cluster(brainFile, specFile);

clusterThresholdsTable = bricCentralize(randCluster.c);

brainResult = makebrain(origBrain, clusterThresholdsTable, brainFile, specFile);
```

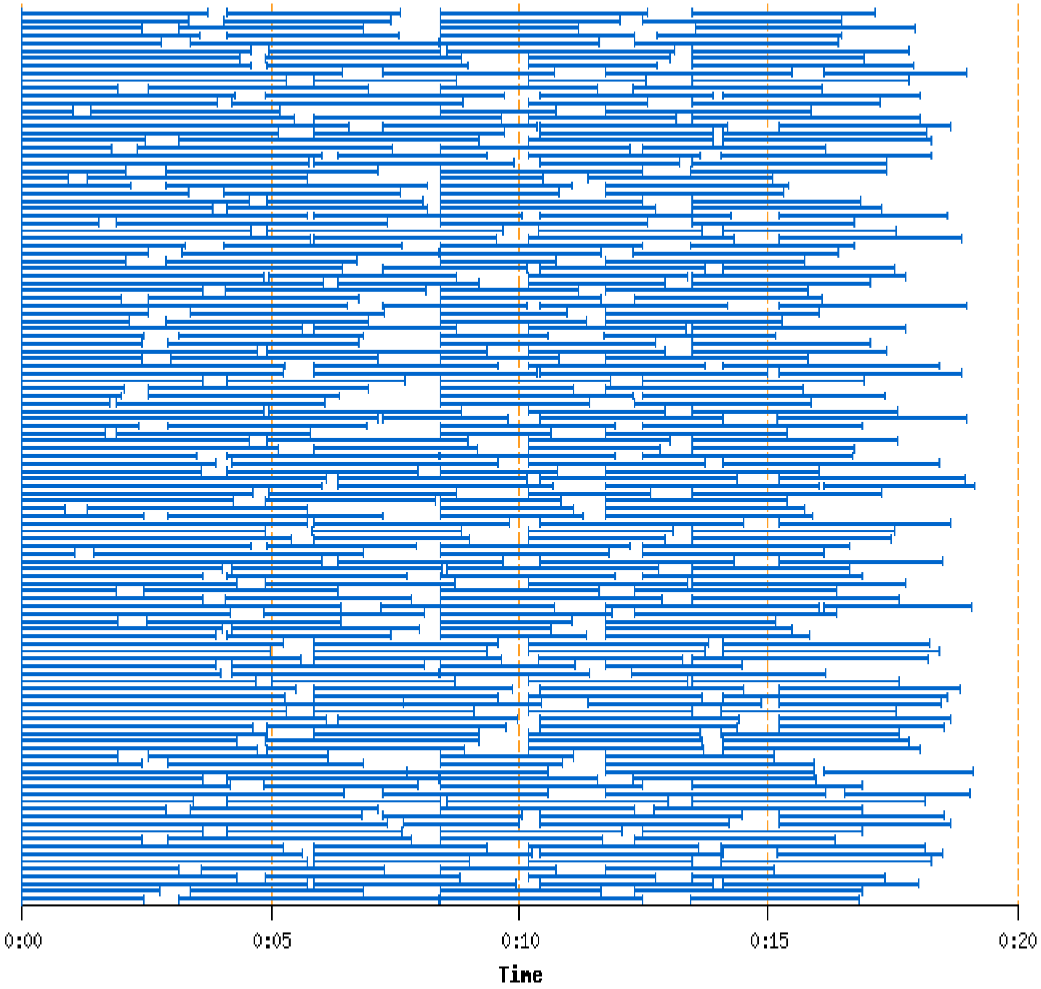
Example Performance Optimizations





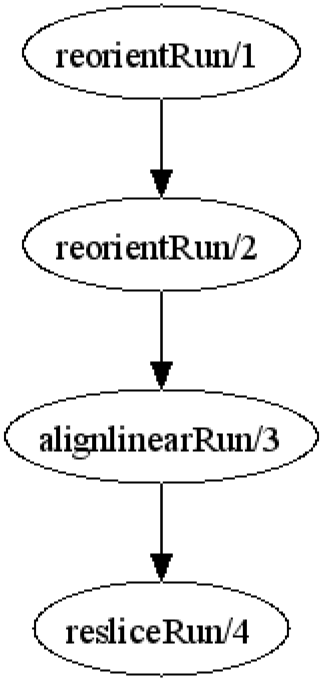
Example Performance Optimizations

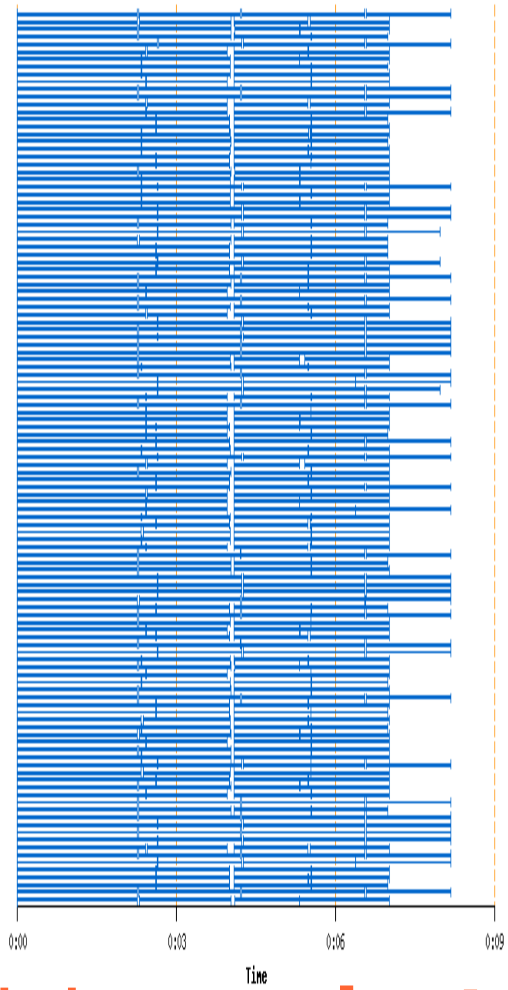




Pipelining

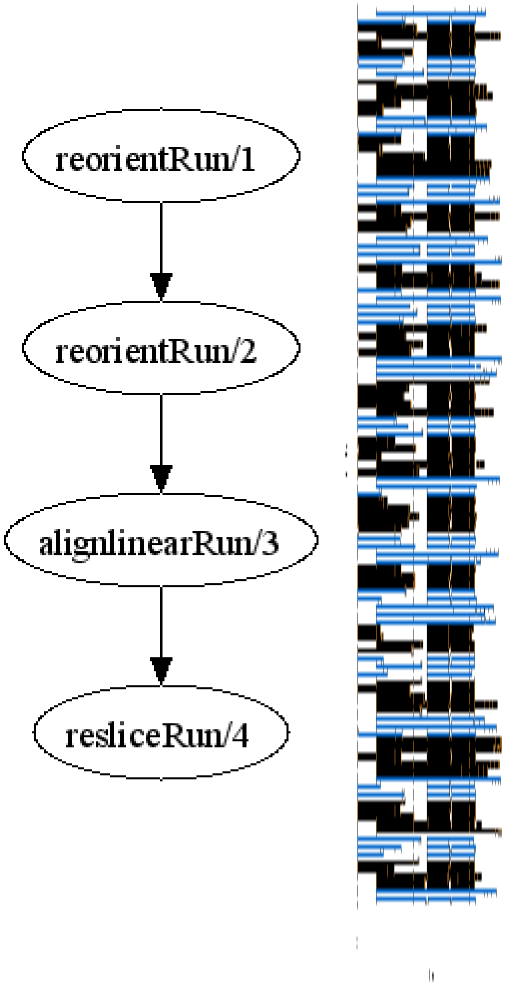
Example Performance Optimizations





Pipelining + **clustering**

Example
Performance Optimizations



Pipelining + **provisioning**

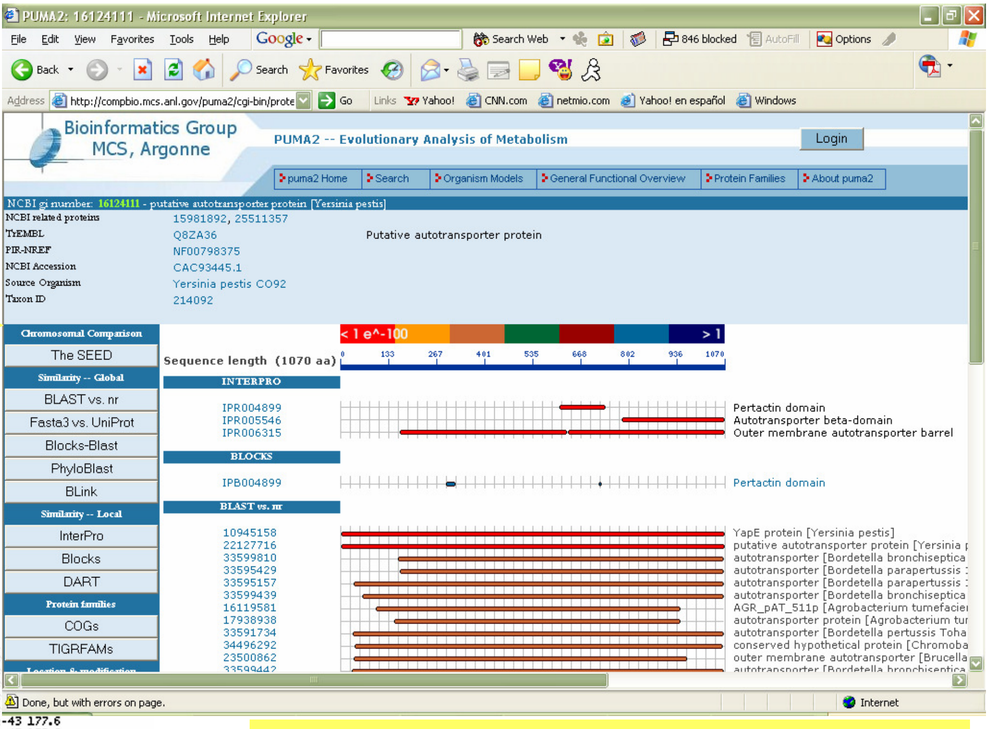
Other Applications

Application	#Jobs/computation	Levels
ATLAS* HEP Event Simulation	500K	1
fMRI DBIC* AIRSN Image Processing	100s	12
FOAM Ocean/Atmosphere Model	2000 (core app runs 250 8-CPU jobs)	3
GADU* Genomics: (14 million seq. analyzed)	40K	4
HNL fMRI Aphasia Study	500	4
NVO/NASA* Photorealistic Montage/Morphology	1000s	16
QuarkNet/I2U2* Physics Science Education	10s	3-6
RadCAD* Radiology Classifier Training	1000s	5
SIDGrid EEG Wavelet Proc, Gaze Analysis, …	100s	20
SDSS* Coadd, Cluster Search	40K, 500K	2, 8

Production Science: Biology

Public PUMA Knowledge Base

Information about proteins analyzed against ~2 million gene sequences


PUMA2: 16124111 - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Google
Search Web
846 blocked
AutoFill
Options
Back
Search
Favorites
Address http://compbio.mcs.anl.gov/puma2/cgi-bin/prote
Go
Links Yahoo! CNN.com netmio.com Yahoo! en español Windows
Login
puma2 Home
Search
Organism Models
General Functional Overview
Protein Families
About puma2
Done, but with errors on page.
Internet

Bioinformatics Group
MCS, Argonne

PUMA2 -- Evolutionary Analysis of Metabolism

NCBI gi number: 16124111 - putative autotransporter protein [Yersinia pestis]

NCBI related proteins	15981892, 25511357	
TrEMBL	Q8ZA36	Putative autotransporter protein
PIR-NREF	NF00798375	
NCBI Accession	CAC93445.1	
Source Organism	Yersinia pestis CO92	
Taxon ID	214092	

Sequence length (1070 aa)

INTERPRO

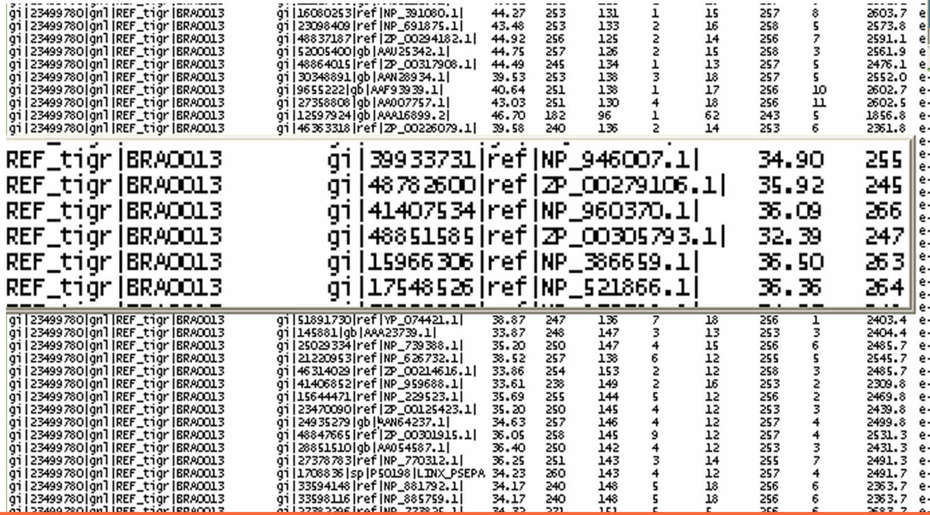
IPR004899	Pertactin domain
IPR005546	Autotransporter beta-domain
IPR006315	Outer membrane autotransporter barrel

BLOCKS

IPB004899	Pertactin domain

BLAST vs. nr

10945158	YapE protein [Yersinia pestis]
22127716	putative autotransporter protein [Yersinia p
33599810	autotransporter [Bordetella bronchiseptica
33595429	autotransporter [Bordetella parapertussis 1
33595157	autotransporter [Bordetella parapertussis 1
33599439	autotransporter [Bordetella bronchiseptica
16119581	AGR_pAT_511p [Agrobacterium tumefacie
17938938	autotransporter protein [Agrobacterium tur
33591734	autotransporter [Bordetella pertussis Toha
34496292	conserved hypothetical protein [Chromoba
23500862	outer membrane autotransporter [Brucella
33599442	autotransporter [Bordetella bronchiseptica

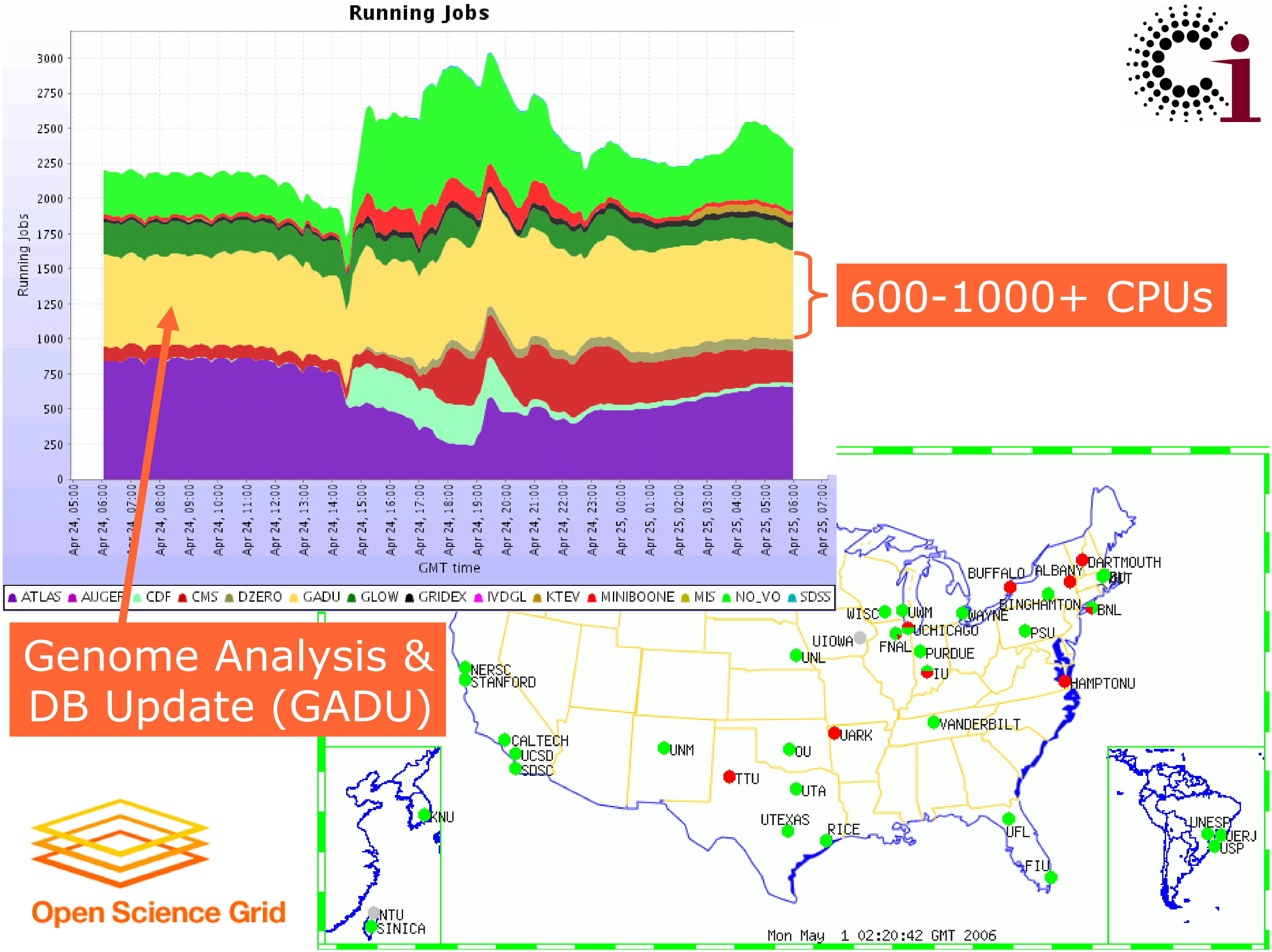


```
gi|23499780|gnl|REF_tigr|BRA0013    gi|16080253|ref|NP_391080.1|     44.27  253  131  1  15  257  8   2603.7  e
gi|23499780|gnl|REF_tigr|BRA0013    gi|23098409|ref|NP_691875.1|     43.48  253  133  2  16  258  5   2573.8  e
gi|23499780|gnl|REF_tigr|BRA0013    gi|48837187|ref|ZP_00294182.1|   44.92  256  125  2  14  256  7   2591.1  e
gi|23499780|gnl|REF_tigr|BRA0013    gi|52005400|gb|AAU25342.1|       44.75  257  126  2  15  258  3   2561.9  e
gi|23499780|gnl|REF_tigr|BRA0013    gi|48864015|ref|ZP_00317908.1|   44.49  245  134  1  13  257  5   2476.1  e
gi|23499780|gnl|REF_tigr|BRA0013    gi|30348891|gb|AAN28934.1|       39.53  253  138  3  18  257  5   2552.0  e-43 177.6
gi|23499780|gnl|REF_tigr|BRA0013    gi|9655222|gb|AAF93939.1|        40.64  251  138  1  17  256  10  2602.7  e-43 177.2
gi|23499780|gnl|REF_tigr|BRA0013    gi|27358808|gb|AAO07757.1|       43.03  251  130  4  18  256  11  2602.5  e-41 170.6
gi|23499780|gnl|REF_tigr|BRA0013    gi|12597924|gb|AAA16899.2|       46.70  182  96   1  62  243  5   1856.8  e-39 162.5
gi|23499780|gnl|REF_tigr|BRA0013    gi|46363318|ref|ZP_00226079.1|   39.58  240  136  2  14  253  6   2361.8  e-36 154.5
                                                                                                   e-33 142.9
REF_tigr|BRA0013        gi|39933731|ref|NP_946007.1|       34.90   255                             e-33 142.5
                                                                                                   e-32 141.4
REF_tigr|BRA0013        gi|48782600|ref|ZP_00279106.1|     35.92   245                             e-32 140.2
                                                                                                   e-32 139.4
REF_tigr|BRA0013        gi|41407534|ref|NP_960370.1|       36.09   266                             e-32 139.0
                                                                                                   e-31 137.9
REF_tigr|BRA0013        gi|48851585|ref|ZP_00305793.1|     32.39   247                             e-31 137.1
                                                                                                   e-30 135.2
REF_tigr|BRA0013        gi|15966306|ref|NP_386659.1|       36.50   263                             e-30 134.8
                                                                                                   e-30 134.8
REF_tigr|BRA0013        gi|17548526|ref|NP_521866.1|       36.36   264                             e-30 134.4
                                                                                                   e-30 134.
                                                                                                   e-30 134.
gi|23499780|gnl|REF_tigr|BRA0013    gi|51891730|ref|YP_074421.1|     38.87  247  136  7  18  256  1   2403.4  e-30 133.7
gi|23499780|gnl|REF_tigr|BRA0013    gi|145881|gb|AAA23739.1|         33.87  248  147  3  13  253  3   2404.4  e-30 133.3
gi|23499780|gnl|REF_tigr|BRA0013    gi|25029334|ref|NP_739388.1|     35.20  250  147  4  15  256  6   2485.7  e-30 132.9
gi|23499780|gnl|REF_tigr|BRA0013    gi|21220953|ref|NP_626732.1|     38.52  257  138  6  12  255  5   2545.7  e-30 132.9
gi|23499780|gnl|REF_tigr|BRA0013    gi|46314029|ref|ZP_00214616.1|   33.86  254  153  2  12  258  3   2485.7  e-30 132.9
gi|23499780|gnl|REF_tigr|BRA0013    gi|41406852|ref|NP_959688.1|     33.61  238  149  2  16  253  2   2309.8  e-30 132.1
gi|23499780|gnl|REF_tigr|BRA0013    gi|15644471|ref|NP_229523.1|     35.69  255  144  5  12  256  2   2469.8  e-30 132.1
gi|23499780|gnl|REF_tigr|BRA0013    gi|23470090|ref|ZP_00125423.1|   35.20  250  145  4  12  253  3   2439.8  e-30 132.1
gi|23499780|gnl|REF_tigr|BRA0013    gi|24935279|gb|AAN64237.1|       34.63  257  146  4  12  257  4   2499.8  e-30 132.1
gi|23499780|gnl|REF_tigr|BRA0013    gi|48847665|ref|ZP_00301915.1|   36.05  258  145  9  12  257  4   2531.3  e-29 131.7
gi|23499780|gnl|REF_tigr|BRA0013    gi|28851510|gb|AAO54587.1|       36.40  250  142  4  12  253  3   2431.3  e-29 131.7
gi|23499780|gnl|REF_tigr|BRA0013    gi|27378783|ref|NP_770312.1|     36.25  251  143  3  14  255  7   2491.3  e-29 131.7
gi|23499780|gnl|REF_tigr|BRA0013    gi|170836|sp|P50198|LINX_PSEPA   34.23  260  143  4  12  257  4   2491.7  e-29 131.3
gi|23499780|gnl|REF_tigr|BRA0013    gi|33594148|ref|NP_881792.1|     34.17  240  148  5  18  256  6   2363.7  e-29 130.2
gi|23499780|gnl|REF_tigr|BRA0013    gi|33598116|ref|NP_885759.1|     34.17  240  148  5  18  256  6   2363.7  e-29 130.2
```

Back Office Analysis on Grid

Millions of BLAST, BLOCKS, etc., on OSG and TeraGrid

Natalia Maltsev et al., http://compbio.mcs.anl.gov/puma2



600-1000+ CPUs

Genome Analysis & DB Update (GADU)

Open Science Grid

Swift Summary

- Clean separation of logical/physical concerns
  - ◆ XDTM specification of logical data structures

+ Concise specification of parallel programs
  - ◆ SwiftScript, with iteration, etc.

+ Efficient execution on distributed resources
  - ◆ Grid interface, pipelining, clustering, load balancing

+ Rigorous provenance tracking and query
  - ◆ Virtual data schema & automated recording

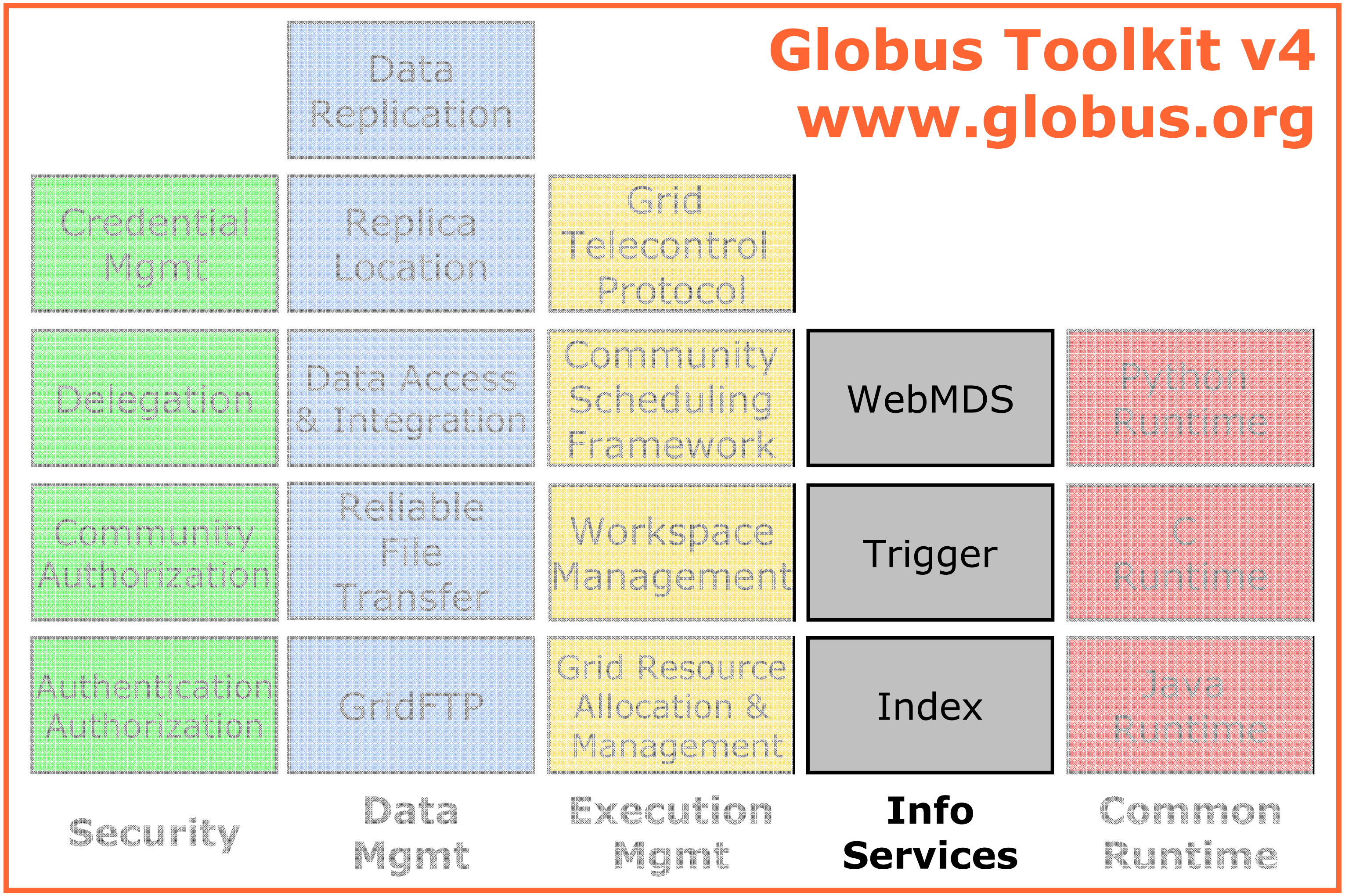
→ **Improved usability and productivity**
  - ◆ Demonstrated in numerous applications

More Specifically, I May Want To …

- Create a service for use by my colleagues
- Manage who is allowed to access my service (or my experimental data or …)
- Ensure reliable & secure distribution of data from my lab to my partners
- Run 10,000 jobs on whatever computers I can get hold of
- Monitor the status of the different resources to which I have access

the globus alliance
www.globus.org

Globus Toolkit: Open Source Grid Infrastructure



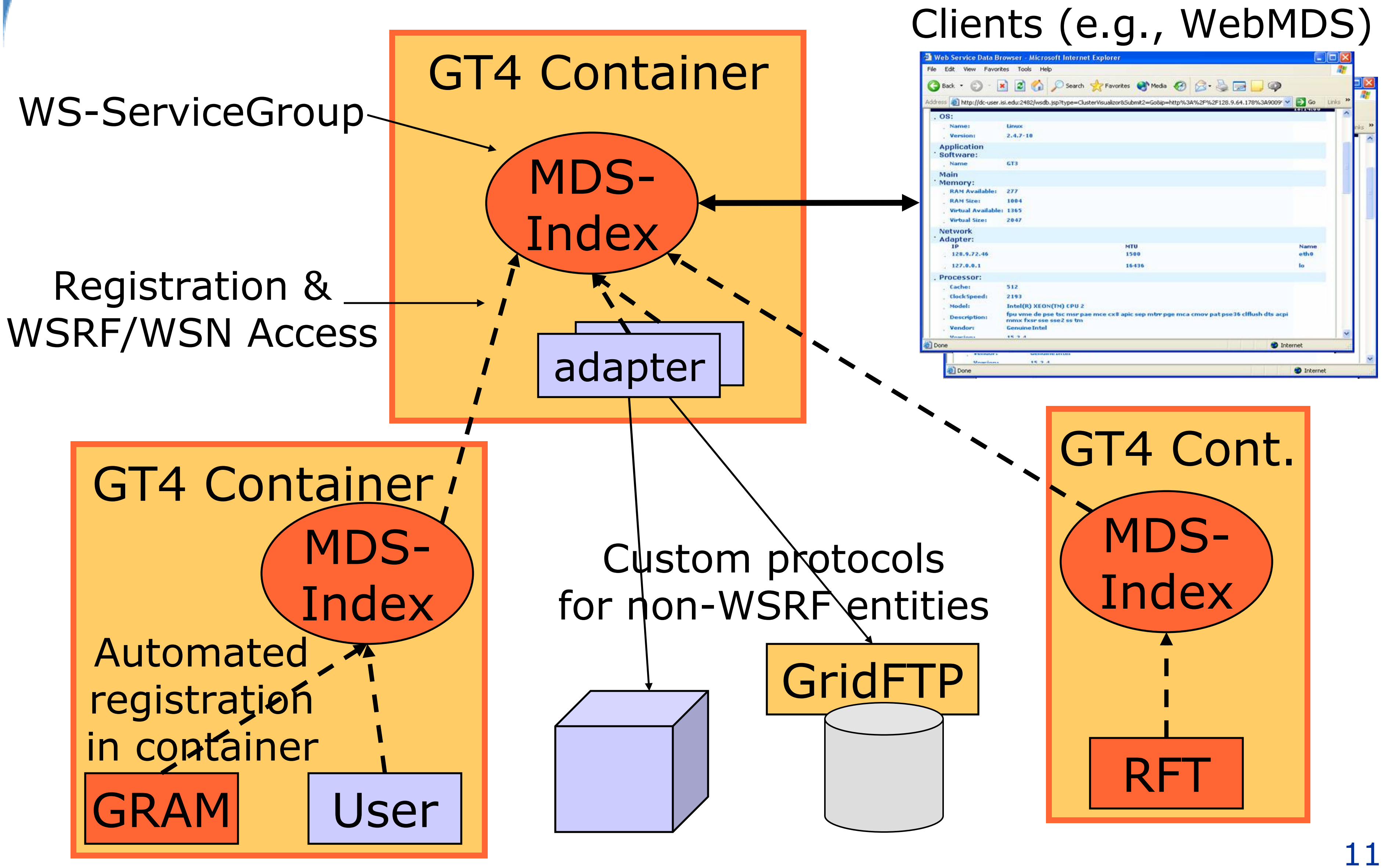
Globus Toolkit v4
www.globus.org

Security	Data Mgmt	Execution Mgmt	Info Services	Common Runtime
	Data Replication			
Credential Mgmt	Replica Location	Grid Telecontrol Protocol		
Delegation	Data Access & Integration	Community Scheduling Framework	WebMDS	Python Runtime
Community Authorization	Reliable File Transfer	Workspace Management	Trigger	C Runtime
Authentication Authorization	GridFTP	Grid Resource Allocation & Management	Index	Java Runtime

Monitoring and Discovery

- "Every service should be monitorable and discoverable using common mechanisms"
  - WSRF/WSN provides those mechanisms
- A common **aggregator** framework for collecting information from services, thus:
  - MDS-Index: Xpath queries, with caching
  - MDS-Trigger: perform action on condition
  - (MDS-Archiver: Xpath on historical data)
- Deep integration with Globus containers & services: every GT4 service is discoverable
  - GRAM, RFT, GridFTP, CAS, …

GT4
Monitoring & Discovery



http://dc-user.isi.edu:2482/wsdb.jsp?type=ClusterVisualizor&Submit2=Go&ip=http%3A%2F%2F128.9.64.178%3A9009

OS:

Name: Linux
Version: 2.4.7-10

Application Software:

Name GT3

Main Memory:

RAM Available: 277
RAM Size: 1004
Virtual Available: 1365
Virtual Size: 2047

Network Adapter:

IP	MTU	Name
128.9.72.46	1500	eth0
127.0.0.1	16436	lo

Processor:

Cache: 512
ClockSpeed: 2193
Model: Intel(R) XEON(TM) CPU 2
Description: fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush dts acpi mmx fxsr sse sse2 ss tm
Vendor: Genuine Intel
Version: 15.2.4

Information Providers

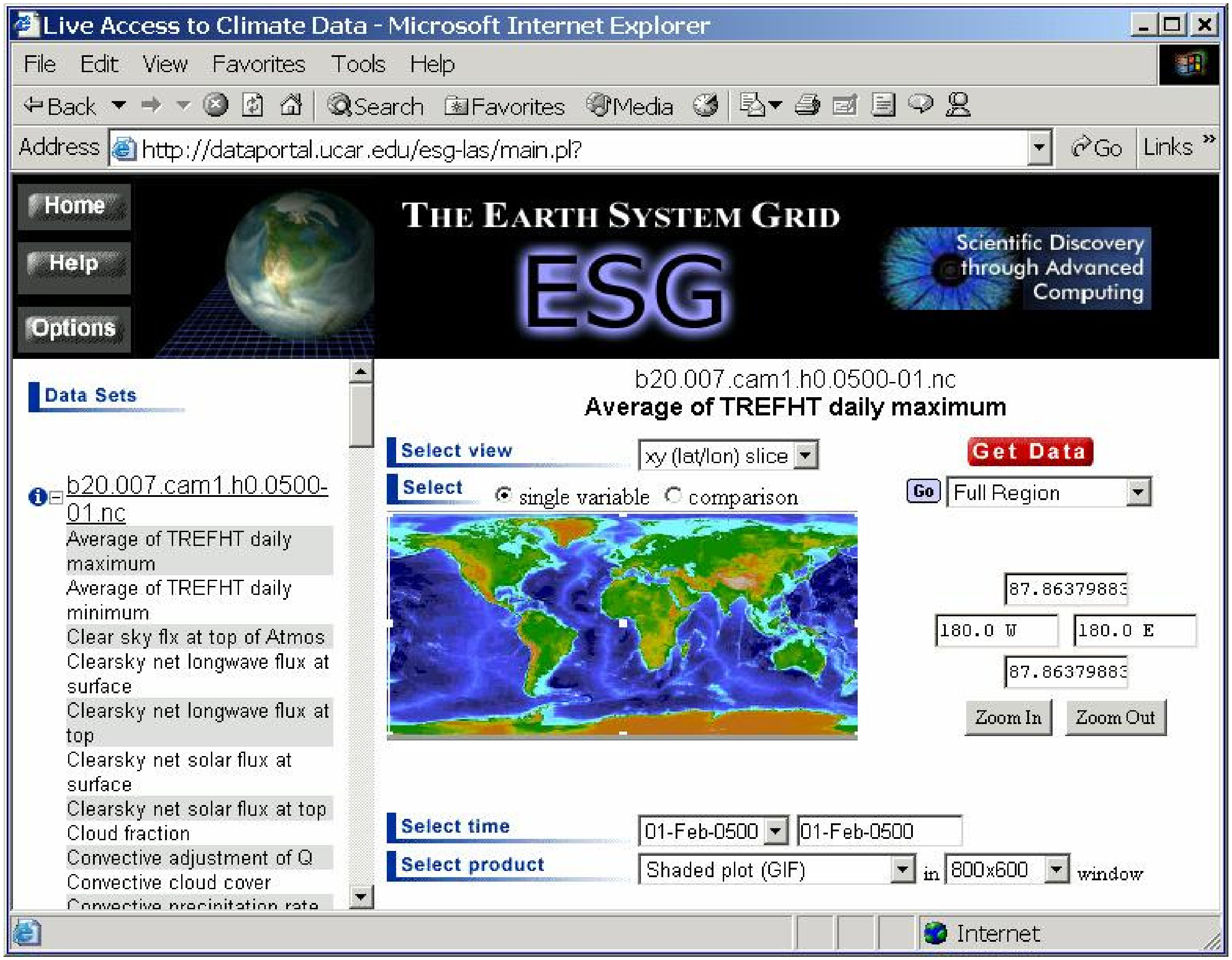
- GT4 **information providers** collect information from some system and make it accessible as WSRF resource properties
- Growing number of information providers
  - Nagios, SGE, LSF, PBS
- Many opportunities to build additional ones
  - E.g., network monitoring, storage systems, various sensors

the globus alliance
www.globus.org

Earth System Grid

DOE Earth System Grid

Goal: Enable sharing & analysis of high-volume data from advanced earth system models


Live Access to Climate Data - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address
Go Links
Home
Help
Options
THE EARTH SYSTEM GRID
ESG
Scientific Discovery through Advanced Computing
Data Sets
Select view
Select
Get Data
Go
Zoom In
Zoom Out
Select time
Select product
Internet

Address: http://dataportal.ucar.edu/esg-las/main.pl?

b20.007.cam1.h0.0500-01.nc
Average of TREFHT daily maximum

Select view: xy (lat/lon) slice

Select: ● single variable ○ comparison

Full Region

87.86379883
180.0 W 180.0 E
87.86379883

Select time: 01-Feb-0500 01-Feb-0500

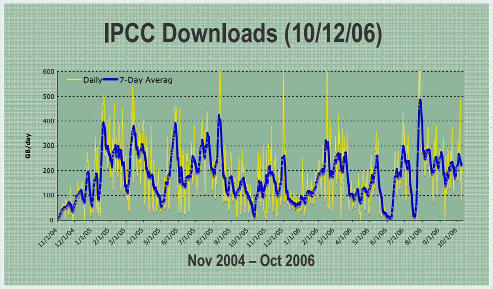
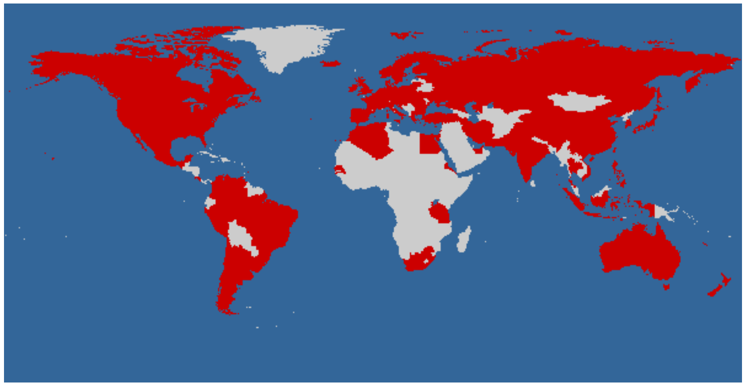
Select product: Shaded plot (GIF) in 800x600 window

Data Sets

b20.007.cam1.h0.0500-01.nc
- Average of TREFHT daily maximum
- Average of TREFHT daily minimum
- Clear sky flx at top of Atmos
- Clearsky net longwave flux at surface
- Clearsky net longwave flux at top
- Clearsky net solar flux at surface
- Clearsky net solar flux at top
- Cloud fraction
- Convective adjustment of Q
- Convective cloud cover
- Convective precipitation rate

www.earthsystemgrid.org

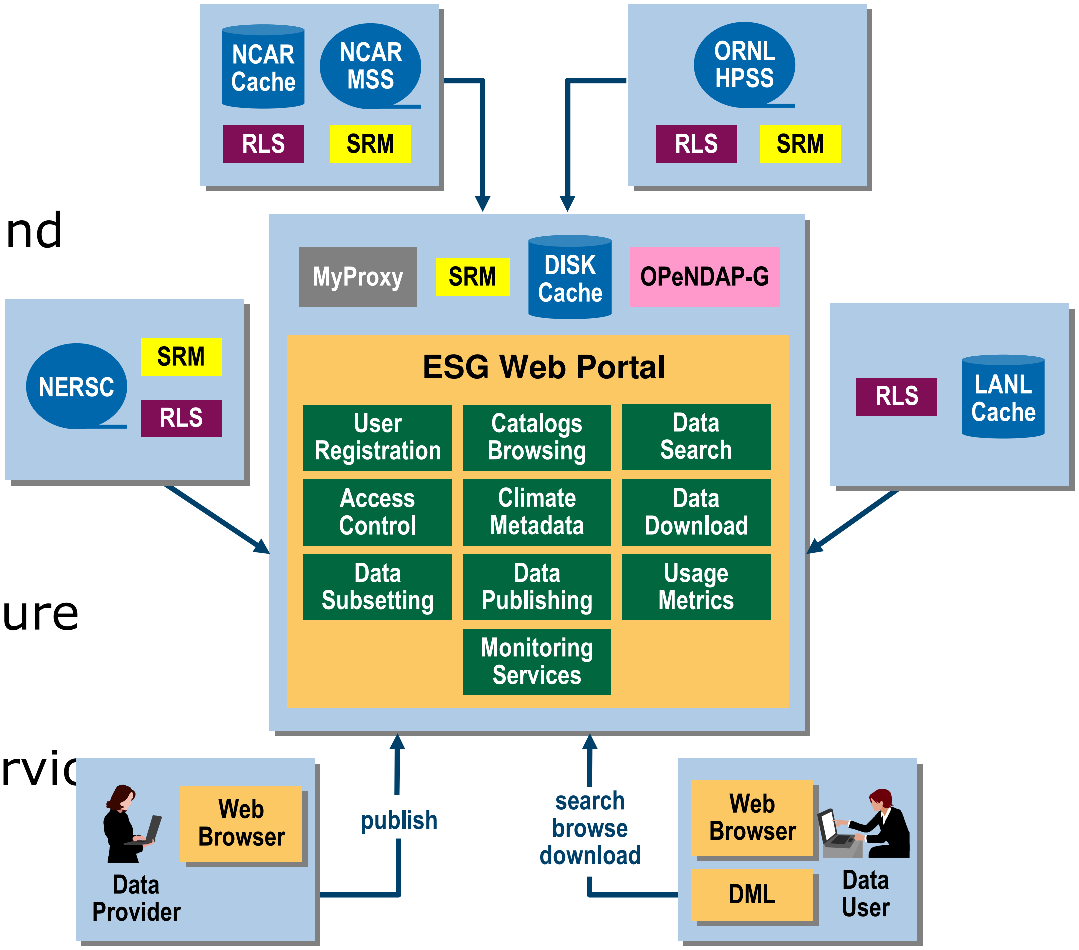
ESG Facts and Figures

ESG Portal at NCAR	IPCC AR4 ESG Portal
130 TB of data at four locations ● 840,331 files ● Includes the past 6 years of joint DOE/NSF climate modeling experiments	28 TB of data at one location ● 68,400 files ● Generated by a modeling campaign coordinated by the Intergovernmental Panel on Climate Change ● Model data from 11 countries
3,200 registered users	818 registered analysis projects
Downloads to date ● 25 TB ● 91,000 files	Downloads to date ● 123 TB ● 543,500 files ● 300 GB/day (average) 
 Worldwide ESG user base	**300 scientific papers published to date based on analysis of IPCC AR4 data**

Slide Courtesy of Dave Bernholdt, ORNL

ESG Architecture and Technologies

- Climate data
  - Metadata catalog
  - OPenDAP-G (aggregation and subsetting)
- Data management
  - Data Mover Lite
  - Storage Resource Manager
  - Globus Security Infrastructure
  - GridFTP
  - Globus Replica Location Servic
- Security services
  - Access control
  - MyProxy
  - PURSE User registration

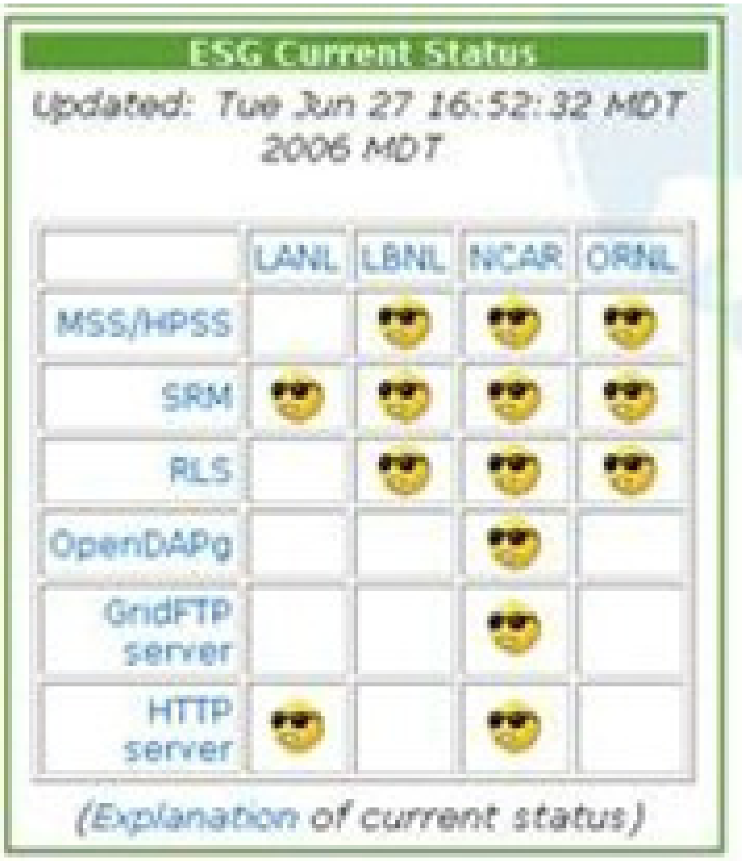


MSS, HPSS: Tertiary data storage systems

Slide Courtesy of Dave Bernholdt, ORNL

Monitoring Overall System Status

- Monitored data are collected in MDS4 Index service
- Information providers check resource status at a configured frequency
  - Currently, every 10 minutes
- Report status to Index Service
- RIformation in Index Service is queried by ESG Web portal
- Used to generate overall picture of state of ESG resources
- Displayed on ESG Web portal page



ESG Current Status

Updated: Tue Jun 27 16:52:32 MDT 2006 MDT

	LANL	LBNL	NCAR	ORNL
MSS/HPSS		😎	😎	😎
SRM	😎	😎	😎	😎
RLS		😎	😎	😎
OpenDAPg			😎	
GridFTP server			😎	
HTTP server	😎		😎	

(Explanation of current status)

Slide Courtesy of Ann Chervenak, USC/ISI

Example Monitoring Information

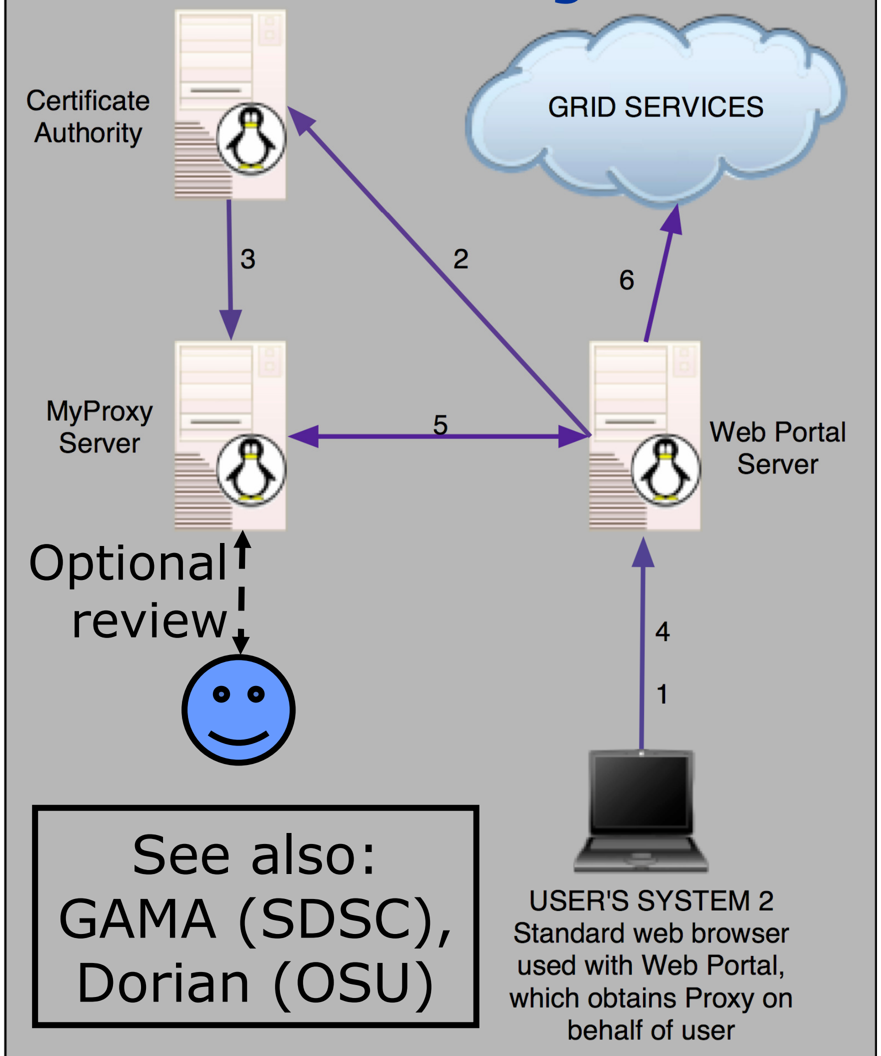
Total error messages for May 2006	**47**
Messages related to certificate and configuration problems at LANL	38
Failure messages due to brief interruption in network service at ORNL on 5/13	2
HTTP data server failure at NCAR 5/17	1
RLS failure at LLNL 5/22	1
Simultaneous error messages for SRM services at NCAR, ORNL, LBNL on 5/23	3
RLS failure at ORNL 5/24	1
RLS failure at LBNL 5/31	1

Slide Courtesy of Ann Chervenak, USC/ISI

Security Needn't Be Hard: PURSe & Earth System Grid

- Purpose
  - Access to large data
- Policies
  - Per-collection control
  - Different user classes
- Implementation (GT)
  - PURSe
  - PKI, SAML assertions
- Experience
  - >4000 users
  - >100 TB downloaded

PURSE User Registration



See also:
GAMA (SDSC),
Dorian (OSU)

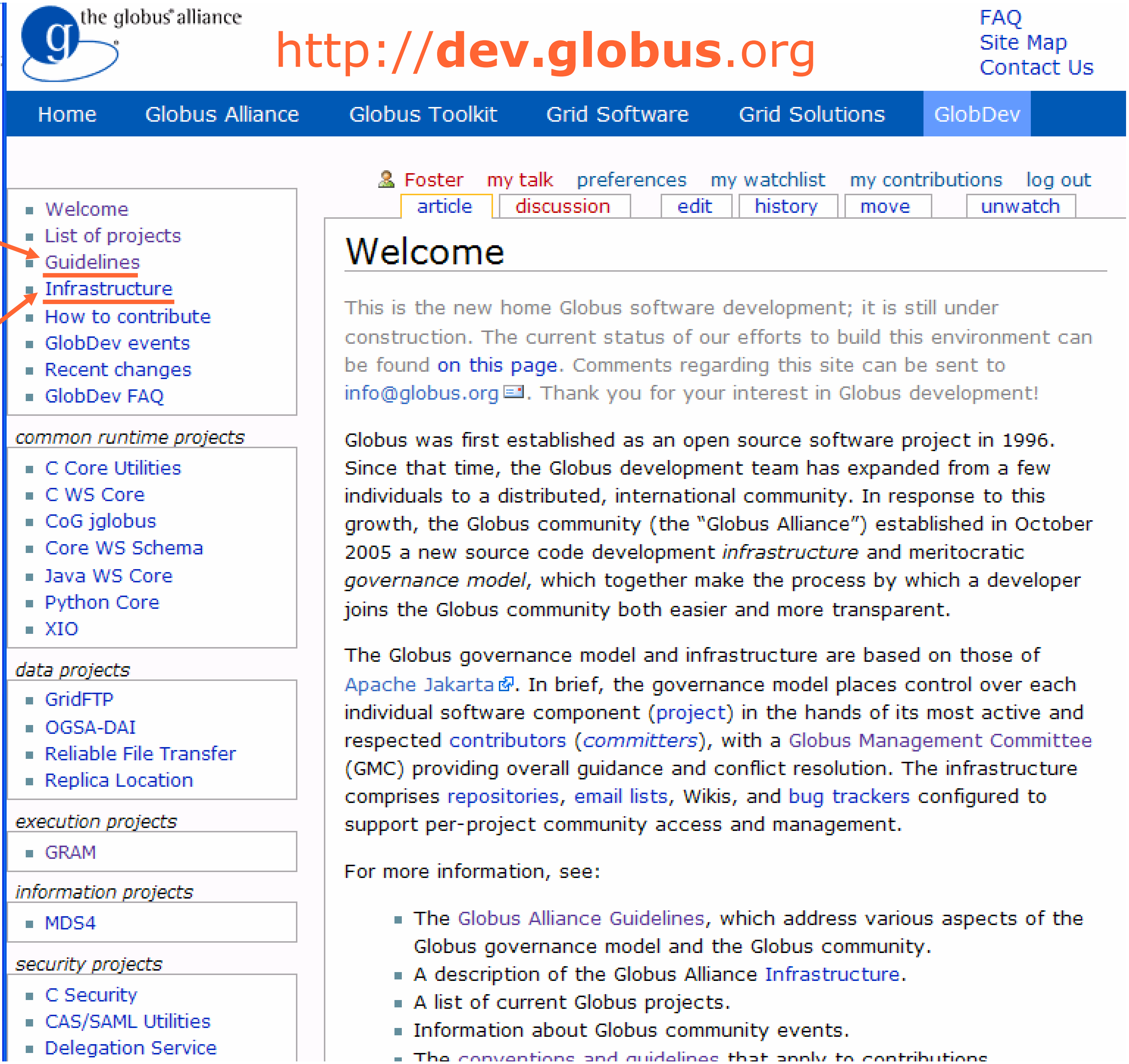
www.earthsystemgrid.org

http://**dev.globus**.org

Guidelines (Apache Jakarta)

Infrastructure (CVS, email, bugzilla, Wiki)

Projects Include …



- Welcome
- List of projects
- Guidelines
- Infrastructure
- How to contribute
- GlobDev events
- Recent changes
- GlobDev FAQ

common runtime projects

- C Core Utilities
- C WS Core
- CoG jglobus
- Core WS Schema
- Java WS Core
- Python Core
- XIO

data projects

- GridFTP
- OGSA-DAI
- Reliable File Transfer
- Replica Location

execution projects

- GRAM

information projects

- MDS4

security projects

- C Security
- CAS/SAML Utilities
- Delegation Service

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 (*committers*), 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.
- Information about Globus community events.
- The conventions and guidelines that apply to contributions

dev.globus

- Globus software is organized as several dozen “Globus Projects”
    - Projects release products
- Each project has its own “Committers”
    - Committers are responsible for governance on matters relating to their products
- A “Globus Management Committee”
    - provides overall guidance and conflict resolution
    - approves the creation of new Globus projects

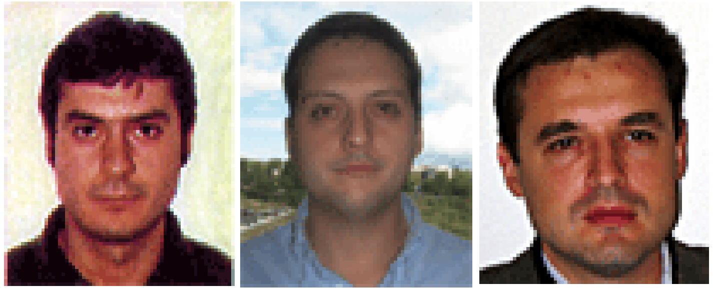
Initial Globus Projects

- Runtime
  - C Core Utilities
  - C WS Core
  - CoG jglobus
  - Core WS Schema
  - Java WS Core
  - Python Core
  - XIO
- Execution
  - GRAM
  - MPICH-G
- Data
  - GridFTP
  - OGSA-DAI
  - Reliable Transfer
  - Replica Location
  - Replication
- Distribution
  - Globus Toolkit
- Documentation
  - Build a Service Tutorial
  - GT Release Manuals
  - GT Programmer's Tutorial
- Security
  - C Security
  - CAS/SAML Utilities
  - Delegation
  - GSI-OpenSSH
  - MyProxy
- Information
  - MDS4

Globus Incubator Projects (Partial List)

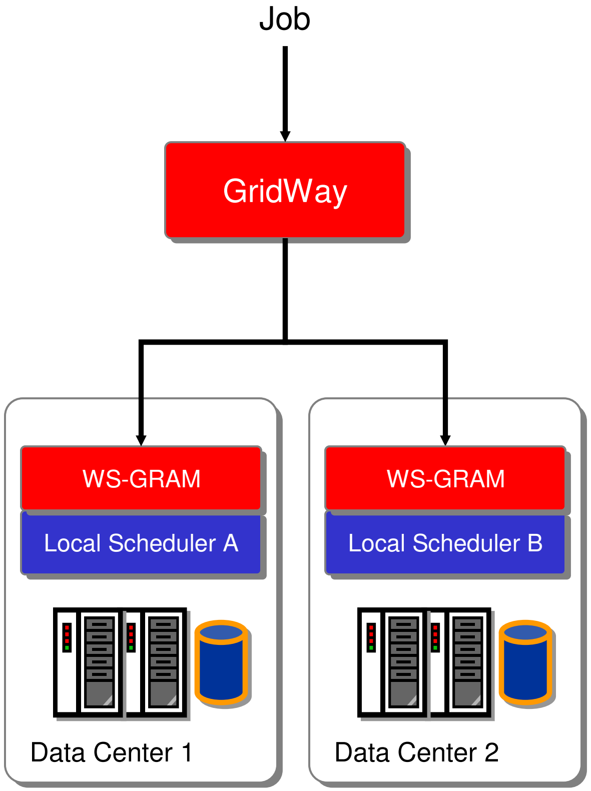
- CoG Workflow — Fine-grained workflow system
- Dynamic Accounts — UNIX account allocation
- GridShib — Integration with Shibolleth
- GridWay — Metascheduler
- gt-hs — Integration of Handle System
- MEDICUS — Medical image management
- Metrics — Infrastructure for usage reporting
- OGCE — Portal toolkit
- PURSe — Portal-based user registration service
- ServMark — Grid service performance tester
- Virtual Workspaces — Virtual machine mgmt

GridWay

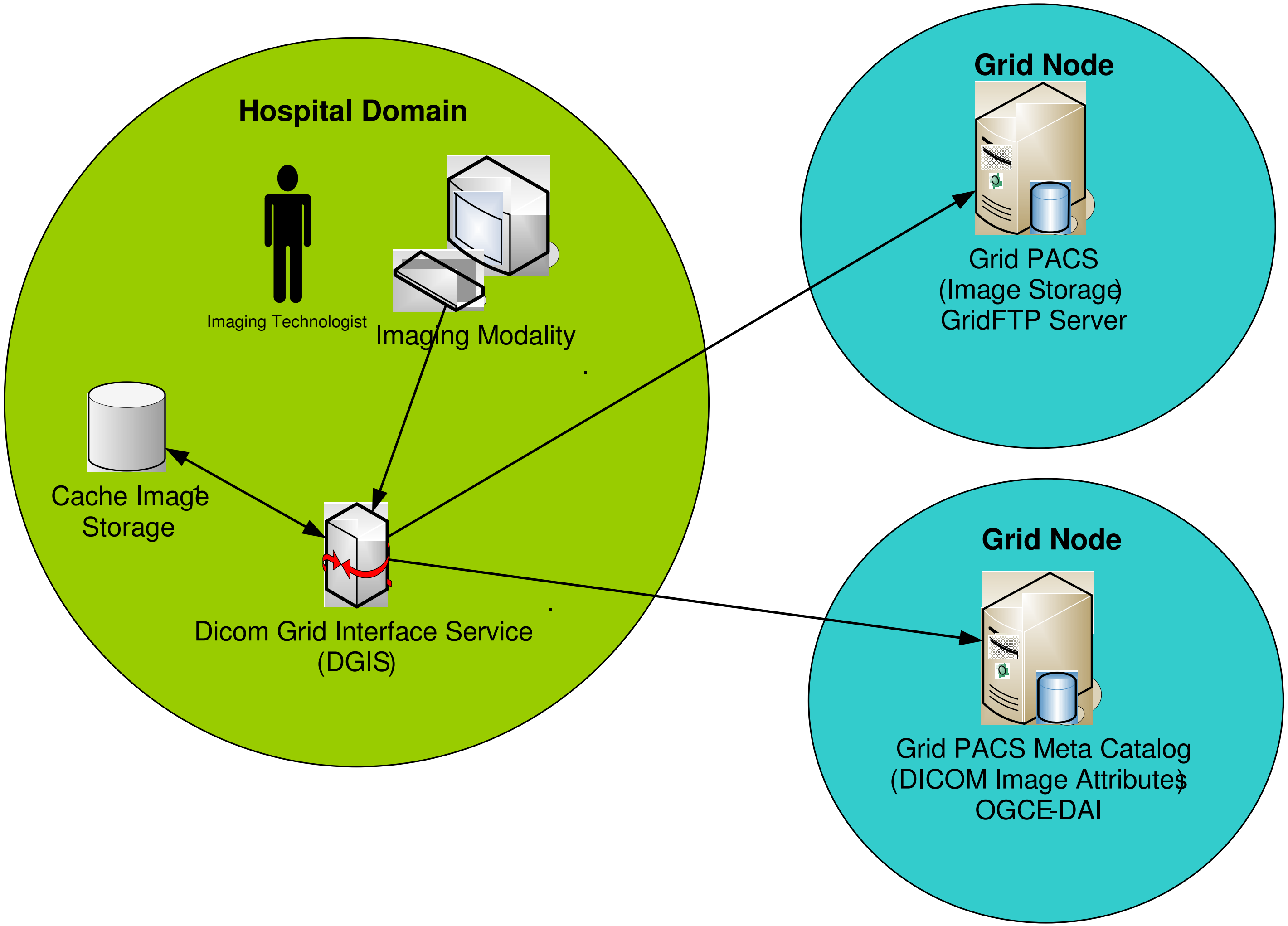


Ignacio M. Llorente,
Ruben S. Montero,
Eduardo Huedo

- Open source meta-scheduler
- dev.globus incubator project
  - Started in 2002, now on v5
- Talks to local scheduler via WS-GRAM
- WS-GRAMs can interface to heterogeneous local schedulers

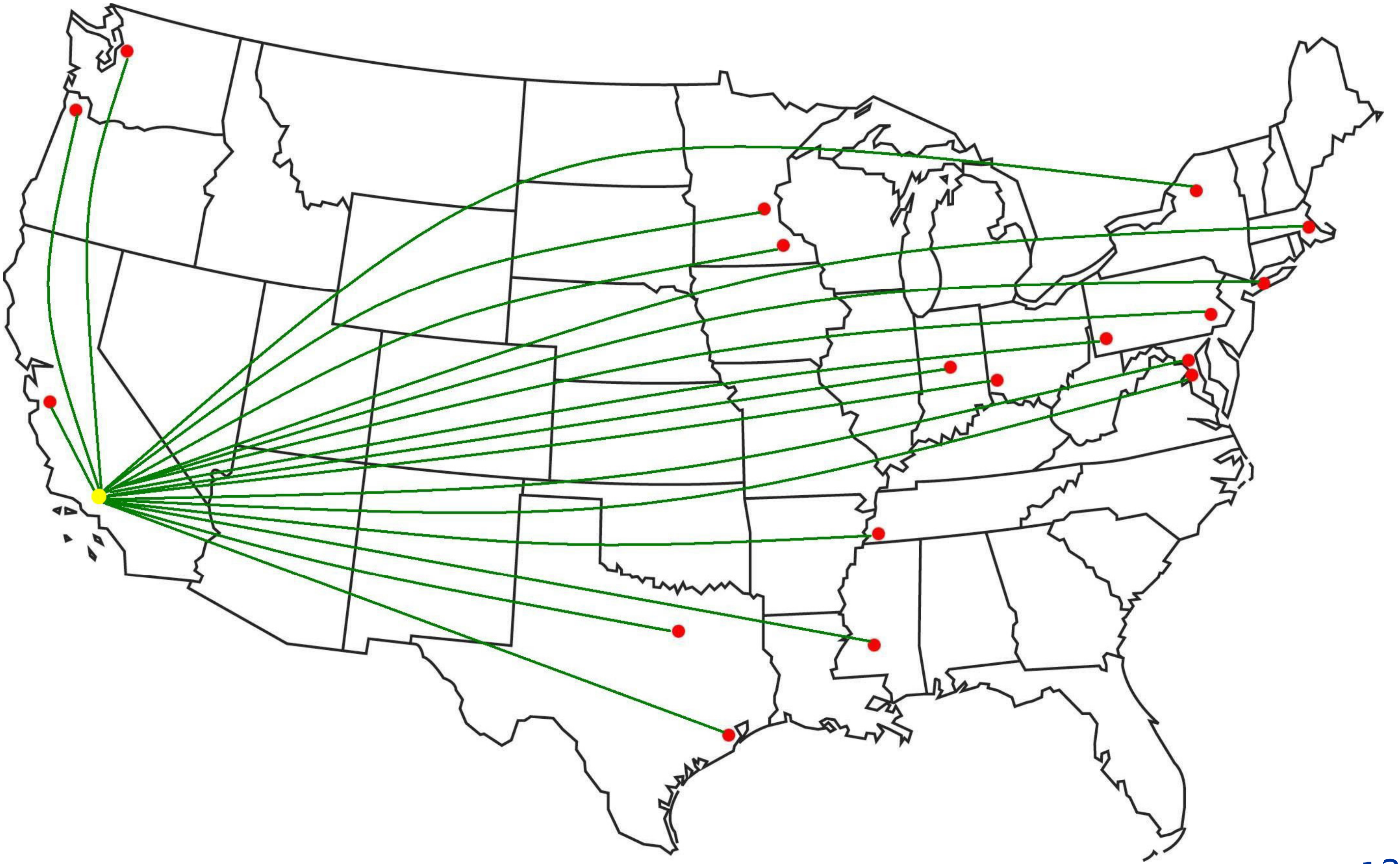


MEDICUS: Management of DICOM Images



Stephan Erberich, Manasee Bhandekar, Ann Chervenak, et al.

Children’s Oncology Grid: A MEDICUS Deployment



MEDICUS Under the Covers

- DICOM images
  - Send (publish)
  - Query/Retrieve (discover)
- Grid Archive
  - Fault tolerant
  - Bandwidth
- Security
  - Authentication
  - Authorization
  - Cryptography
- Access
  - Web portal
- Applications
  - Computing
  - Data Mining

Globus Toolkit Release 4

DICOM Grid Interface Service (DGIS)
+
Meta Catalog Service (OGSA-DAI)

Data Replication Service (DRS)

X.509 Certificates
+
MyProxy Delegation

Grid Web Portal, OGCE / GridSphere

GRAM, OGSA-DAI

Univa

- Provider of commercial support, services, & products around open source Globus
  - Commercial distribution of GT4 & beyond
  - Integration with enterprise systems
  - Committed to open source & open standards
- Univa is contributing to Globus open source
  - Big contributions to GT4 development, testing
  - New functionality: install shields, security configurator, GridFTP extensions
  - Additional contributions expected

Globus User Community

- Large & diverse
  - 10s of national Grids, 100s of applications, 1000s of users; probably much more
  - Every continent except Antarctica
  - Applications ranging across many sciences
  - Dozens (at least) of commercial deployments
- Successful
  - Many production systems doing real work
  - Many applications producing real results
- Smart, energetic, demanding
  - Constant stream of new use cases & tools

Global Community

TERAGRID
Grid.it project
Grid-Ireland
PPDG
iVDgL
Open Science Grid
INFN GRID
EUROGRID
K*GRID
Data GRID
National Grid NG SINGAPORE
D-GRID
Dokumente
Folien
Protokolle
Communities 1 | 2 | 3 | 4 | 5 | 6
Arbeitsk 1 | 2 | 3
ACCESS GRID™
grangenet
GRIP UK Particle Physics
Grid3 2003
DataTAG
NEESgrid
Building the National Virtual Collaboratory for Earthquake Engineering.
LCG
WestGrid
GriPhyN
Data Intensive Science
CERN
openlab for DataGrid applications
Developing Solutions for the Data-Intensive Science of the Large Hadron Collider
NORDUGRID
Grid Solution for Wide Area Computing and Data Handling
Versió en Català
English Version
Versión en Castellano
tota la informació sobre el projecte
presentació
descripció tècnica
els serveis oferts
participants
documentació pública
com puc participar?
GridCat
grid grid grid grid
dintre del grid
accés al Portal
Intranet per a investigadors
estat dels serveis actius
ApGrid
Asia-Pacific Grid
Grid Consortium Japan
Ninf
egee
Enabling Grids for E-science in Europe
NAREGI
超高速コンピュータ網形成プロジェクト
National Research Grid Initiative
Grid Applications
Grid Middleware
Networking
国立情報学研究所グリッド研究開発推進拠点 NII -The National Institute of Informatics
NAREGI
GridLab

Examples of Production Scientific Grids

- APAC (Australia)
- China Grid
- China National Grid
- DGrid (Germany)
- EGEE
- NAREGI (Japan)
- Open Science Grid
- Taiwan Grid
- TeraGrid
- ThaiGrid
- UK Natl Grid Service

Future Directions: Service Oriented Science

People **create** services (data or functions) …

which I **discover** …

& maybe **compose** to create a new function ...

and then **publish** as a new service.

!!

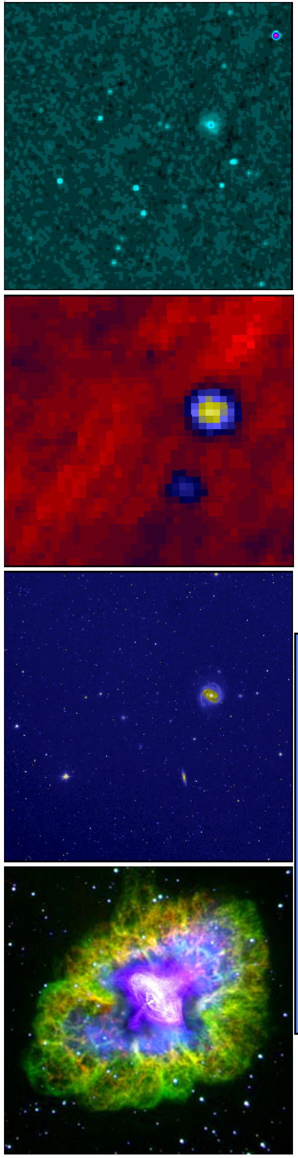
→ *I find "someone else" to* ***host*** *services, so I don't have to become an expert in operating services & computers!*

→ *I hope that this "someone else" can* ***manage*** *security, reliability, scalability, …*

TeraGrid™
EMPOWERING DISCOVERY

"Service-Oriented Science", *Science*, 2005

For Example: Virtual Observatories



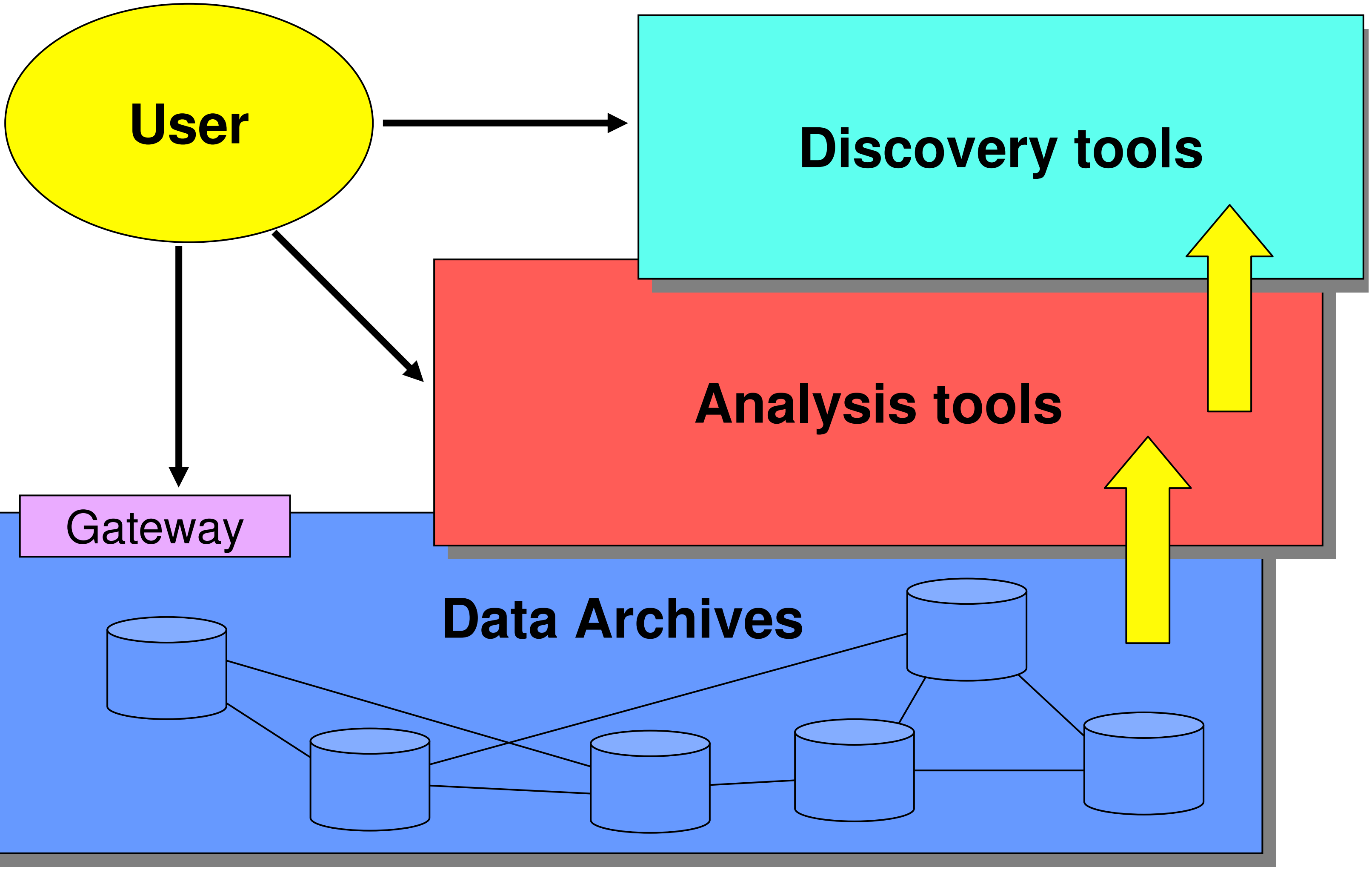
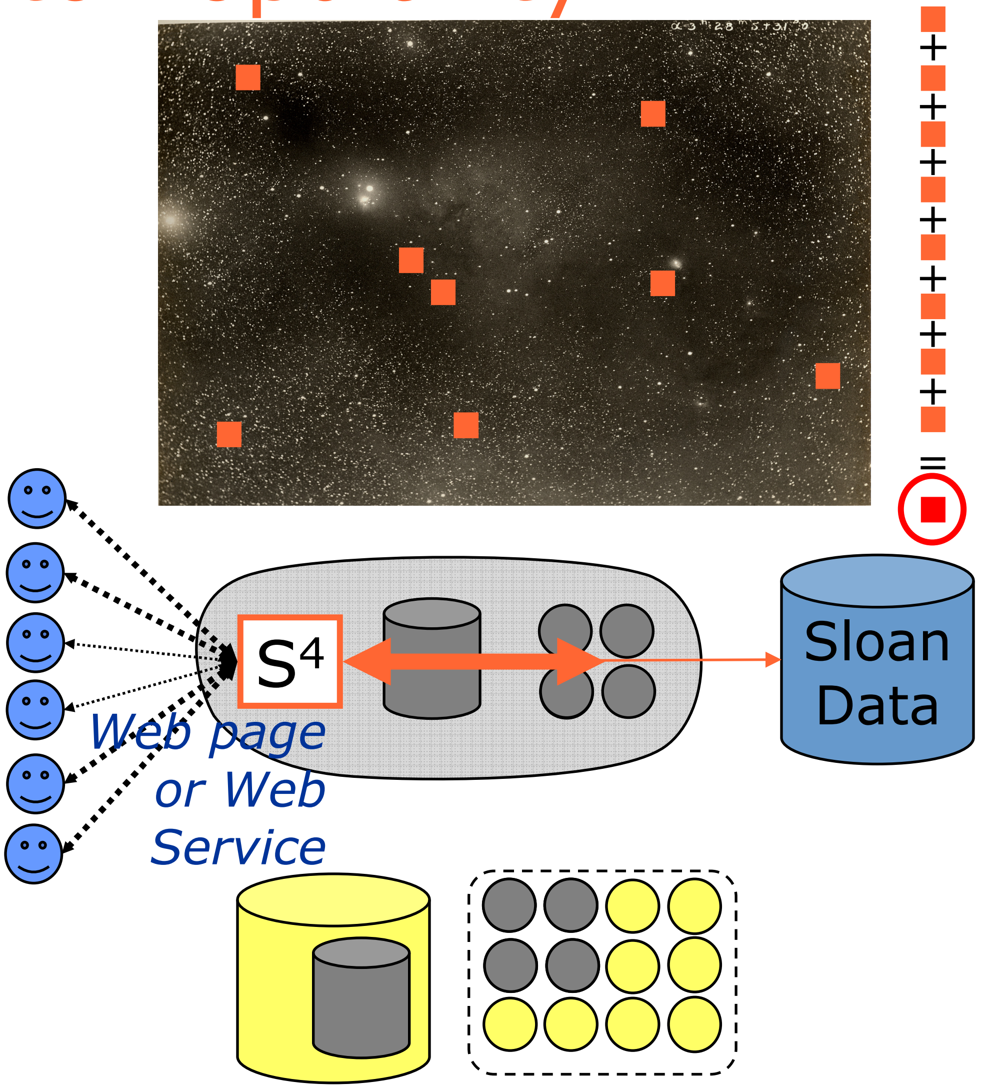


Figure: S. G. Djorgovski

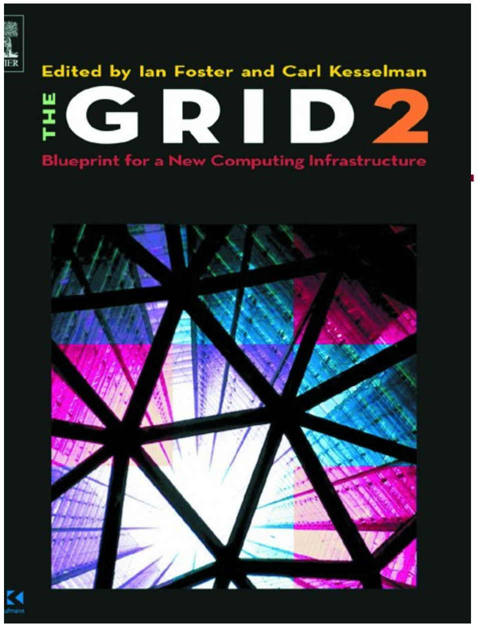
Using Grid Infrastructure to Respond to Popularity

- 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

Summary: Grid is About …


Edited by Ian Foster and Carl Kesselman
THE GRID 2
Blueprint for a New Computing Infrastructure

Enabling *"coordinated resource sharing & problem solving in dynamic, multi-institutional virtual organizations."*

(Source: **"The Anatomy of the Grid"**)

- Access to shared resources
  - → Virtualization, allocation, management
- With predictable behaviors
  - → Provisioning, quality of service
- In dynamic, heterogeneous environments
  - → Standards-based interfaces and protocols

More Specifically, Making it Possible to …

- Create a service for use by my colleagues
- Manage who is allowed to access my service (or my experimental data or …)
- Ensure reliable & secure distribution of data from my lab to my partners
- Run 10,000 jobs on whatever computers I can get hold of
- Monitor the status of the different resources to which I have access
- And so on …

… By Providing Open Infrastructure

- Web services standards
  - State, notification, security, …
- Services that enable access to resources
  - Service-enable new & existing resources
  - E.g., GRAM on computer, GridFTP on storage system, custom application services
  - Uniform abstractions & mechanisms
- Tools to build applications that exploit this infrastructure
  - Registries, security, data management, …
- A rich tool & service ecosystem

For More Information

- Globus
  - www.globus.org: software, documentation
  - dev.globus.org: community development
- Swift
  - www.ci.uchicago.edu/swift
- TeraGrid, Open Science Grid
  - www.teragrid.org, www.opensciencegrid.org
- Random ramblings
  - ianfoster.typepad.com