



Data Integration and OGSA-DAI (Open Grid Services Architecture – Data Access and Integration)

Neil Chue Hong

Presented by D.Fergusson

Bari 9/3/06



epcc



Project Partners

Powered by



Funded by the Grid Core Programme

OGSA-DAI

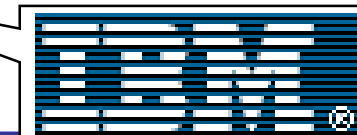
£3 million, 18 months, from Feb 2002
Three major releases, three interim releases

DAIT (DAI -Two)

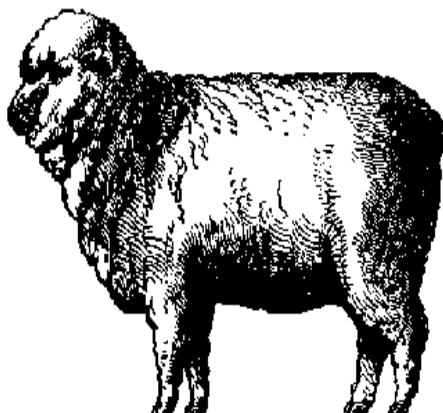
Keep the OGSA-DAI brand name
£1.5 million, 24 months,
from Oct 2003
Four major releases

GGF DAIS WG

Strong involvement.
Standardise the interfaces
OGSA-DAI to be a reference
implementation



OGSA-DAI In One Slide



OGSA-DAI IN A NUTSHELL

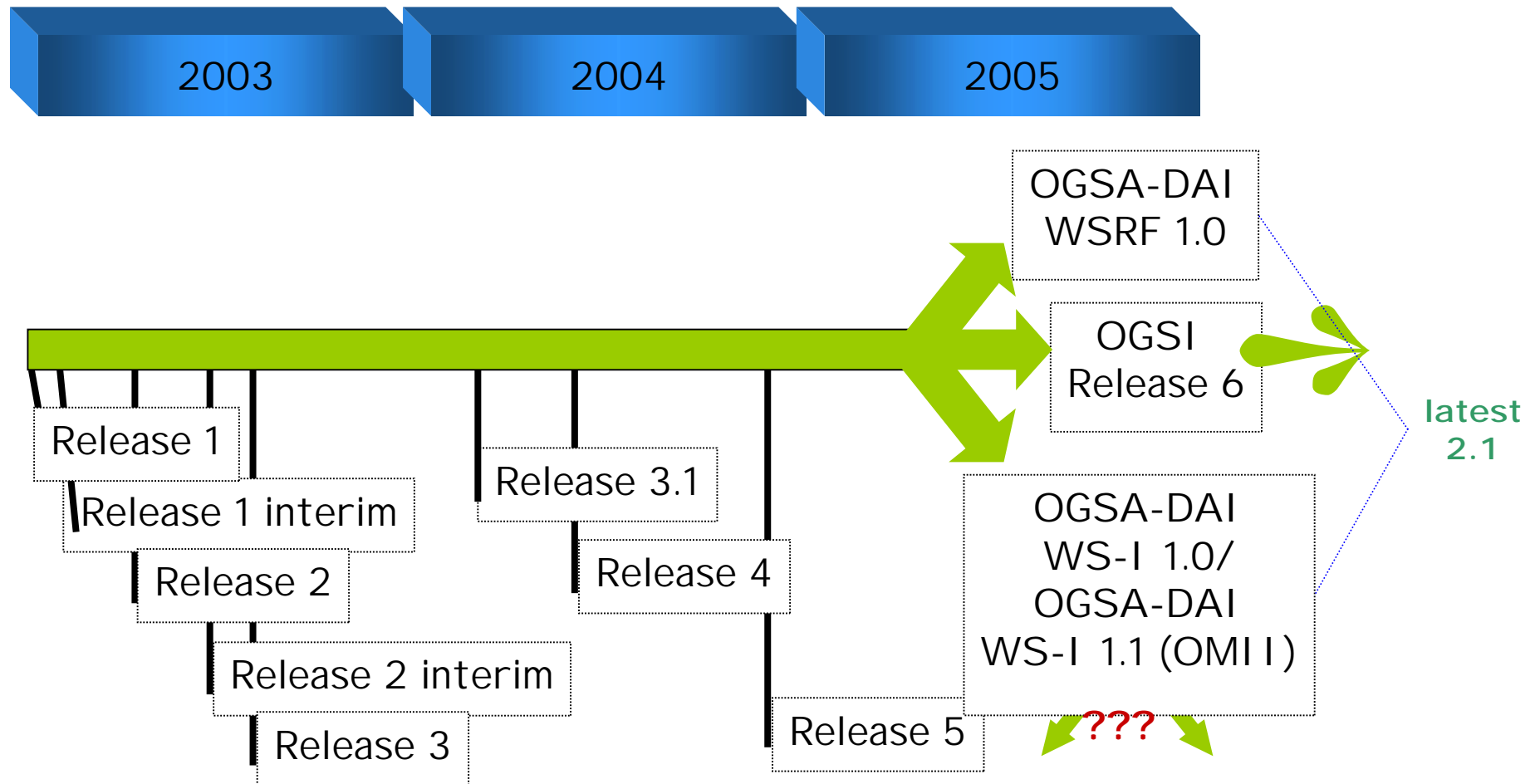
A Desktop Quick Reference

With apologies to
O'REILLY®

Neil Chue Hong

- An *extensible framework* for data access and integration.
- Expose heterogeneous data resources to a grid through web services.
- Interact with data resources:
 - Queries and updates.
 - Data transformation / compression
 - Data delivery.
- Customise for your project using
 - Additional Activities
 - Client Toolkit APIs
 - Data Resource handlers
- A base for higher-level services
 - federation, mining, visualisation,...

OGSA-DAI Timeline

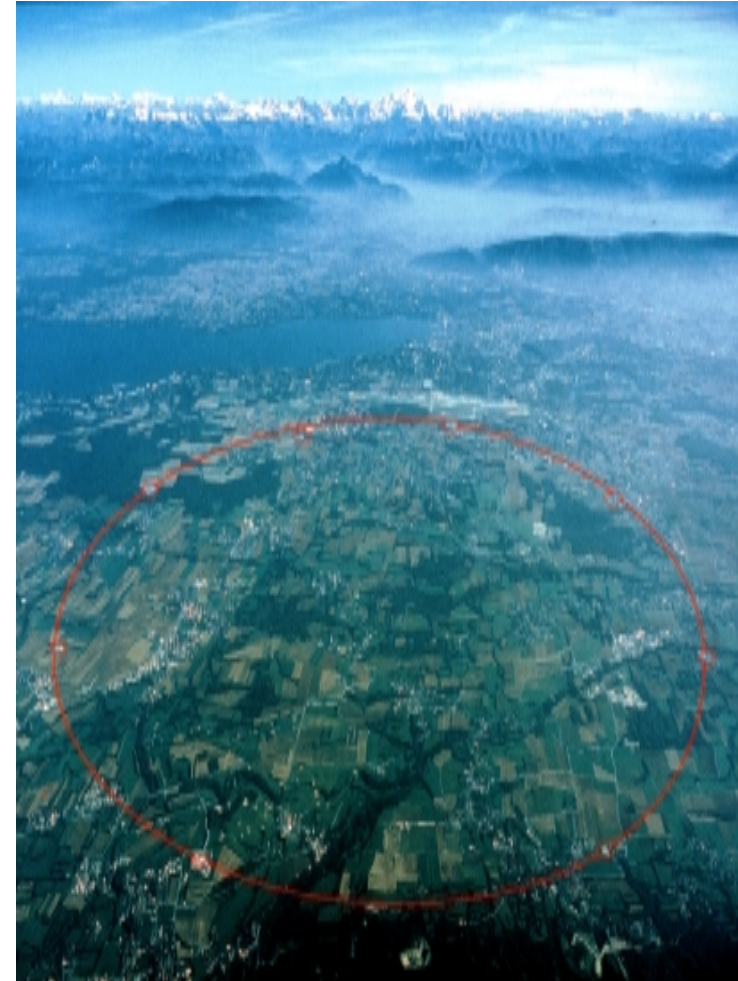


Content

- Data on the Grid – what it's about ●
- DAIS and OGSA-DAI
- Projects OGSA-DAI and its users

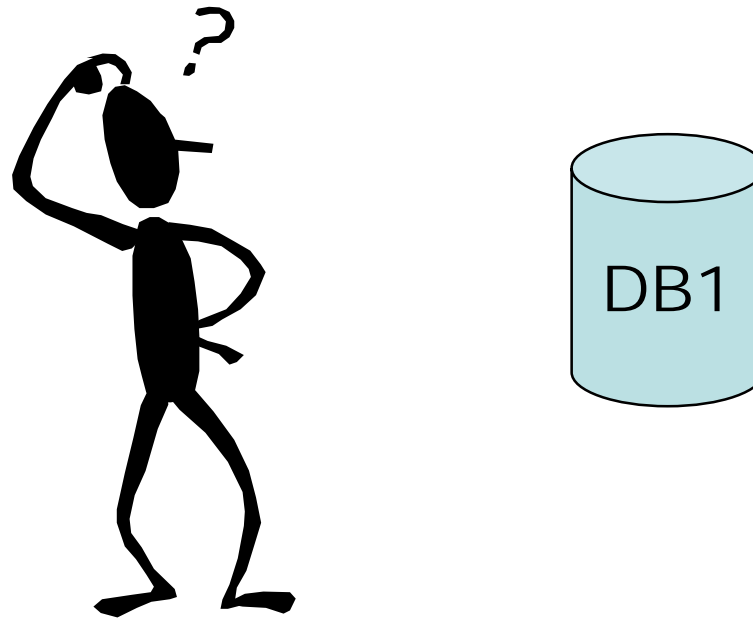
Motivation

- **Entering an age of data**
 - **Data Explosion**
 - ▶ CERN: LHC will generate 1GB/s = 10PB/y
 - ▶ VLBA (NRAO) generates 1GB/s today
 - ▶ Pixar generate 100 TB/Movie
 - **Storage getting cheaper**
- **Data stored in many different ways**
 - **Data resources**
 - ▶ Relational databases
 - ▶ XML databases
 - ▶ Flat files
- **Need ways to facilitate**
 - **Data discovery**
 - **Data access**
 - **Data integration**



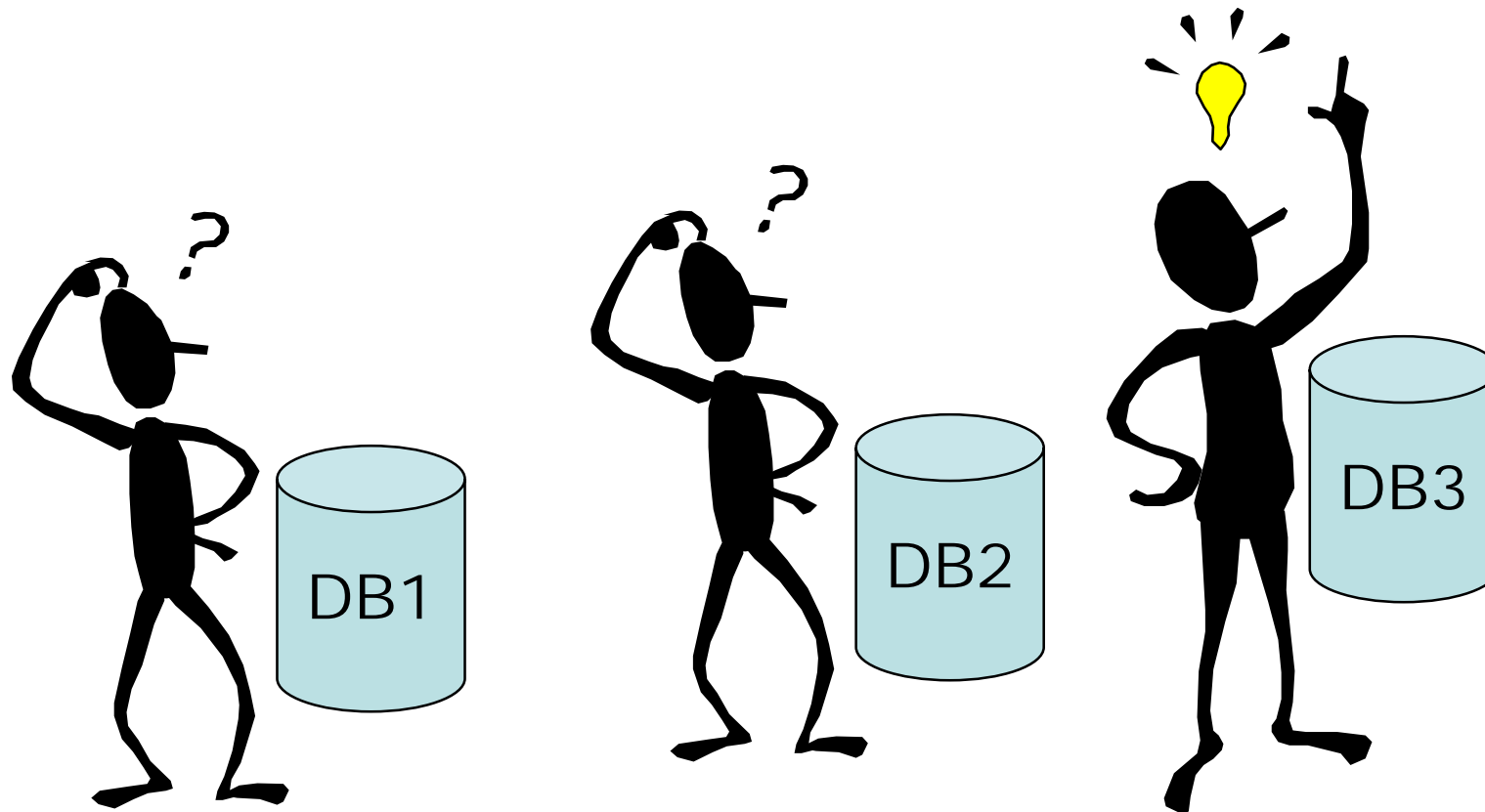
Why is the Grid necessary?

- If I am a researcher with my own database, why do I need the Grid?



It's all about sharing

- You can never have it all...



Scenario: Red Eyed Tree Frogs

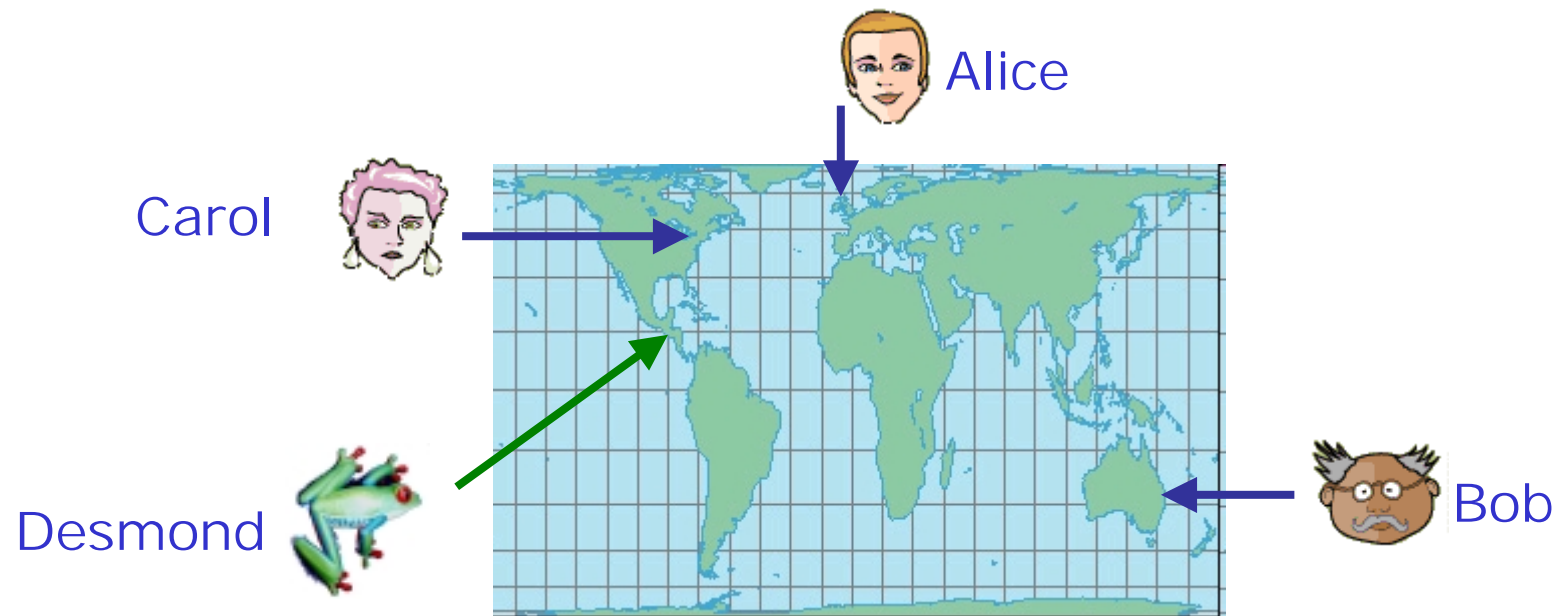


The story of Alice, Bob, Carol
and a frog called Desmond

Thanks to Tom Sugden and Martin Westhead for the original idea

Once upon a time...

- In this story, we will learn how Data Access and Integration Services helped:



Use Case: Publishing

- Alice is a molecular biologist
 - ◆ Based at the University of South Edinburgh
 - ◆ Mapped the genetic sequence of the Red-Eyed Tree Frog
- Alice wants to make her work available to the scientific community
 - ◆ Publish a read-only on-line database
 - ◆ Register data resource with a public registry



Use-case: Remote update



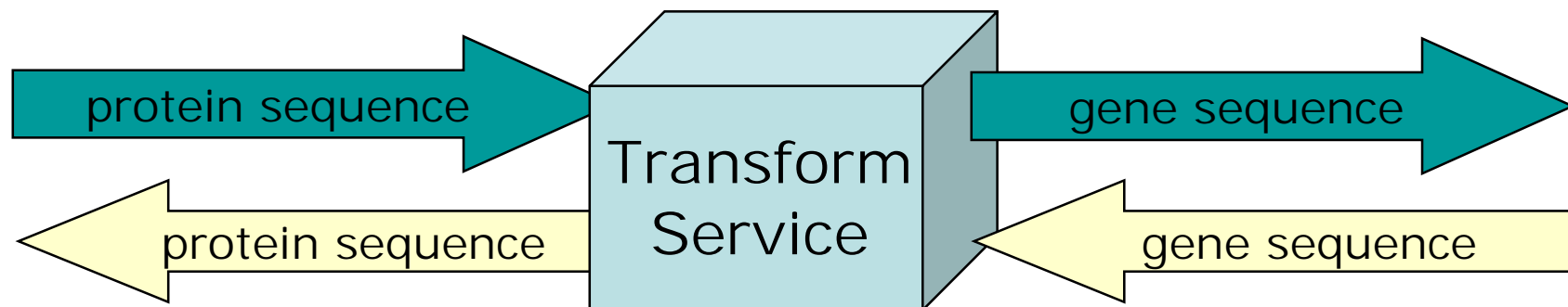
- **Bob is a Professor of Biology**
 - Based at the Organisation for Gene Sequencing in Australia
 - Working in collaboration with Alice on the Red-Eyed Tree Frog genome
 - Alice provides a secure private read/write grid data service
- **Through Alice's services**
 - Bob can contribute new sequences



Use Case: Transformations

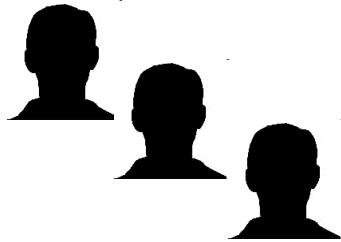


- Carroll is a biochemist
 - Works for a small drugs company called DrugsRUs in Aurora, Illinois.
 - Investigating toxin in saliva of Fire Bellied Toad
- Wants to compare proteins with Red Eyed Tree Frog
 - Carroll has a protein sequence
 - Alice's data is encoded as a gene sequence



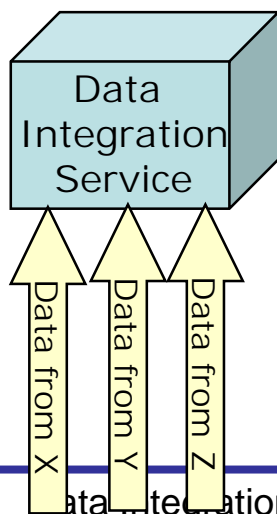
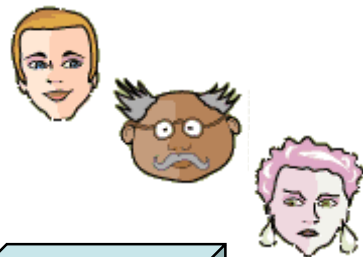
Use Case: Data Integration

- X, Y and Z are other scientists
 - They publish their work as read-only data resources
 - Z only allows specific queries to be run



Alice, Bob and Carol each want to use subsets of data from X, Y, and Z

- Trying to save the nearly extinct variegated red-eyed tree frog
- Alice writes a service which exposes a integrated set of data as another virtual data resource
- Bob and Carol can use this resource as if it were a single data resource

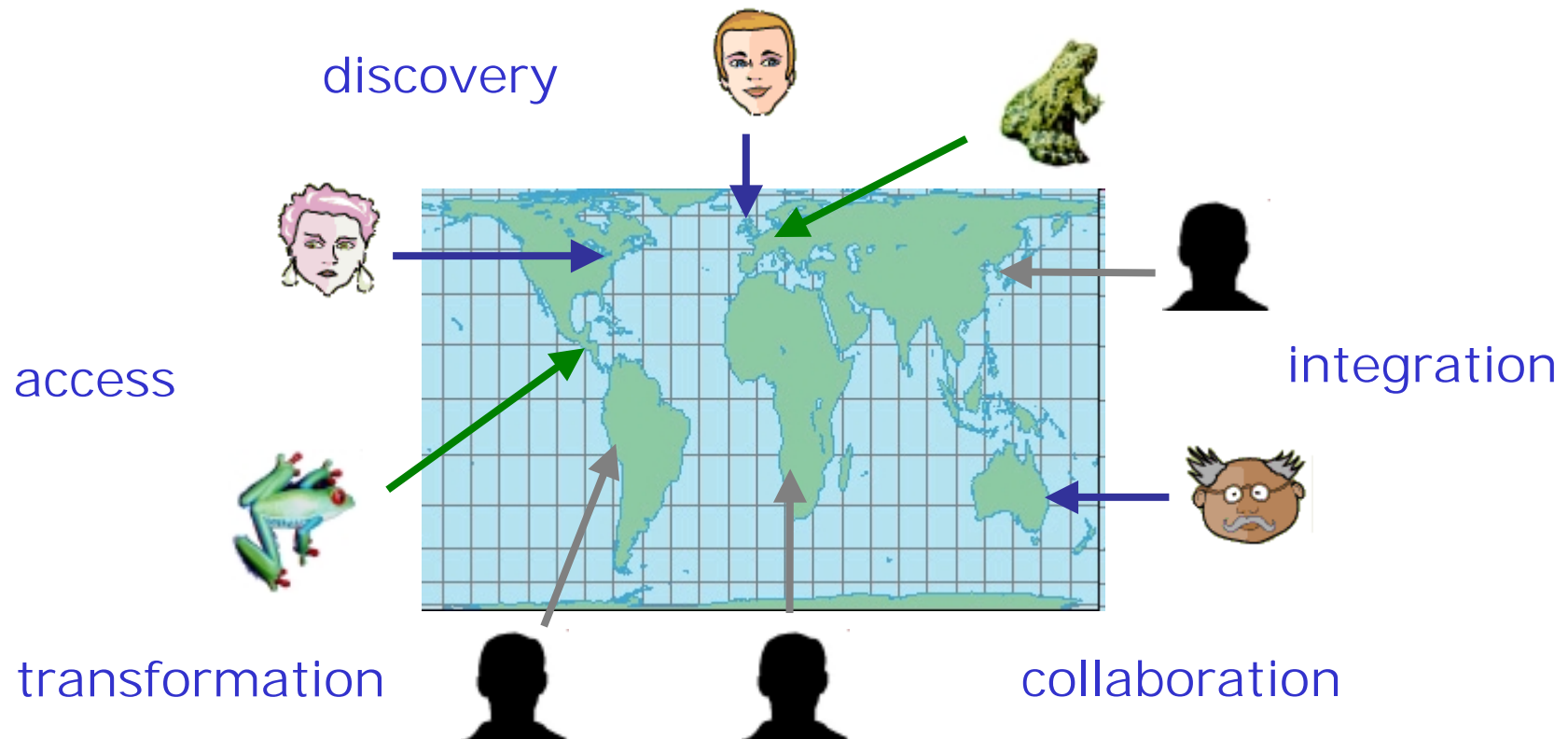


- They find a way to save Desmond



The End

- Use data services to provide the middleware tools to grid-enable existing databases



Share and share alike!

- Many challenges:
 - Scalability, performance, heterogeneity, ownership, economics
 - Common schema, data description and semantics, data formats, process and procedure, provenance
- Can be solved only through collaboration and the sharing of:
 - Ideas
 - Efforts
 - Resources
- Perhaps most importantly: **sharing of data**
 - Beware of data huggers!

Data Requirements

- What do we need for effective sharing of data?
 - Structured, organised, annotated & curated data
 - Computable data models
 - Visualisation of data
 - Data provenance
 - Shared distributed systems
 - Networked workplaces, instruments, data sources
 - Metadata, ontologies, standards
 - Authentication, authorisation, accounting, policies

Data Services: motives

- **Key to Integration of Scientific Methods**
 - **Publication and sharing of results**
 - ▶ Primary data from observation, simulation & experiment
 - ▶ Encourages novel uses
 - ▶ Allows validation of methods and derivatives
 - ▶ Enables discovery by combining data collected independently
- **Key to Large-scale Collaboration**
 - **Economies: data production, publication & management**
 - ▶ Sharing cost of storage, management and curation
 - ▶ Many researchers contributing increments of data
 - ▶ Pooling annotation leads to rapid incremental publication
 - **Accommodates global distribution**
 - ▶ Data & code travel faster and more cheaply
 - **Accommodates temporal distribution**
 - ▶ Researchers assemble data
 - ▶ Later (other) researchers access data



Data Services: challenges

- **Scale**
 - Many sites, large collections, many uses
- **Longevity**
 - Research requirements outlive technical decisions
- **Diversity**
 - No “one size fits all” solutions will work
 - ▶ Primary Data, Data Products, Meta Data, Administrative data, ...
- **Many Data Resources**
 - **Independently owned & managed**
 - ▶ No common goals
 - ▶ No common design
 - ▶ Work hard for agreements on foundation types and ontologies
 - ▶ Autonomous decisions change data, structure, policy, ...
 - **Geographically distributed**
- **and I haven't even mentioned security yet!**



Types of data integration

- Well-known databases, well known but complex queries
 - typically read-only
 - efficiency, scalability, performance
 - e.g. astronomy, data mining
- Public repositories, personalised queries
 - metadata, data discovery, education
 - e.g. GIS
- Many sites, loose connectivity
 - query robustness
 - e.g. caBIG, education
- Mixed resources, time variant
 - e.g. meteorology, OLTP

Meta-data: describing data

- Choosing data sources
 - How do you find them?
 - How are they described and advertised?
 - Is the equivalent of Google possible?
- Meta-data is required describing:
 - Structure of data
 - Types of data
 - Operations supported/available
 - Access requirements
 - Quality of service?
- No established standards for heterogeneous data sources

Small problems

- Not just “Grand Challenges”!
 - Also the small problems
- For instance:
 - What happens to data when a researcher leaves a team?
 - How can a research leader point to “popular” data when a new researcher joins?
 - How can you manage your data when you start to run out of local storage space?
 - How do I get my data from one format/database to another?
 - How do I combine *my* data with *your* data?
- You need to manage your data

Content

- Data on the Grid – what it's about
- DAIS and OGSA-DAI ●
- Projects OGSA-DAI and its users

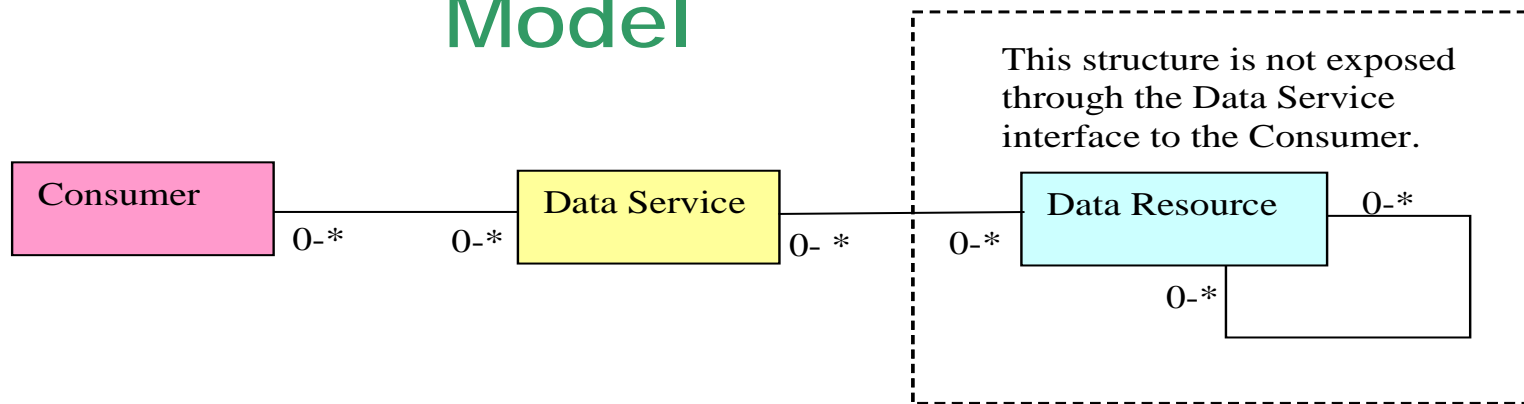
DAIS WG Goals

- Provide service-based access to structured data resources as part of OGSA architecture
- Specify a selection of interfaces tailored to various styles of data access starting with relational and XML
- Interact well with other GGF OGSA specs

DAIS WG Non-Goals

- No new common query language
- No schema integration or common data model
- No common namespace or naming scheme
- No data resource management
- No push based delivery
 - Information Dissemination WG?

DAIS View Of Data Services Model



- A Data Service presents a Consumer with an interface to a Data Resource.
- A Data Resource can have arbitrary complexity, for example, a file on an NFS mounted file system or a federation of relational databases.
- A Consumer is not typically exposed to this complexity and operates within the bounds and semantics of the interface provided by the Data Service

Terminology - Data

- **Data Resource**
 - Any object that can source/sink data
 - Currently databases in scope
- **Data Service**
 - Common interface to a data resource
 - Exposes capabilities of data resource
 - ▶ SQL Queries, X-Path Queries
 - May provide additional capabilities
 - ▶ Data transformations, 3rd party data delivery
- **OGSA-DAI**
 - Open Grid Services Architecture Data Access and Integration

What is a data service?

- An interface to a stored collection of data
 - e.g. Google and Amazon
 - web services
- But the data could be:
 - replicated
 - shared
 - federated
 - virtual
 - incomplete
- Don't care about the underlying representation
 - do care about the information it represents



Why OGSA-DAI?

- **Why use OGSA-DAI over JDBC?**
 - Can embed additional functionality at the service end
 - ▶ Transformations, compressions, third party delivery
 - ▶ The extensible activity framework
 - **Avoiding unnecessary data movement**
 - **Common interface to heterogeneous data resources**
 - ▶ Relational, XML databases, and files
 - **Language independence at the client end**
 - ▶ Do not need to use Java
 - **Platform independence**
 - ▶ Do not have to worry about connection technology, drivers, etc

OGSA-DAI Design Principles – I

- **Efficient client-server communication**
 - Minimise where possible
 - One request specifies multiple operations
- **No unnecessary data movement**
 - Move computation to the data
 - Utilise third-party delivery
 - Apply transforms (e.g., compression)
- **Build on existing standards**
 - Fill-in gaps where necessary

OGSA-DAI Design Principles – II

- Do not hide underlying data model
 - Users must know where to target queries
 - Data virtualisation is hard
- Extensible architecture
 - Modular and customisable
 - e.g., to accommodate stronger security
- Extensible activity framework
 - Cannot anticipate all desired functionality
 - Activity = unit of functionality
 - Allow users to plug-in their own

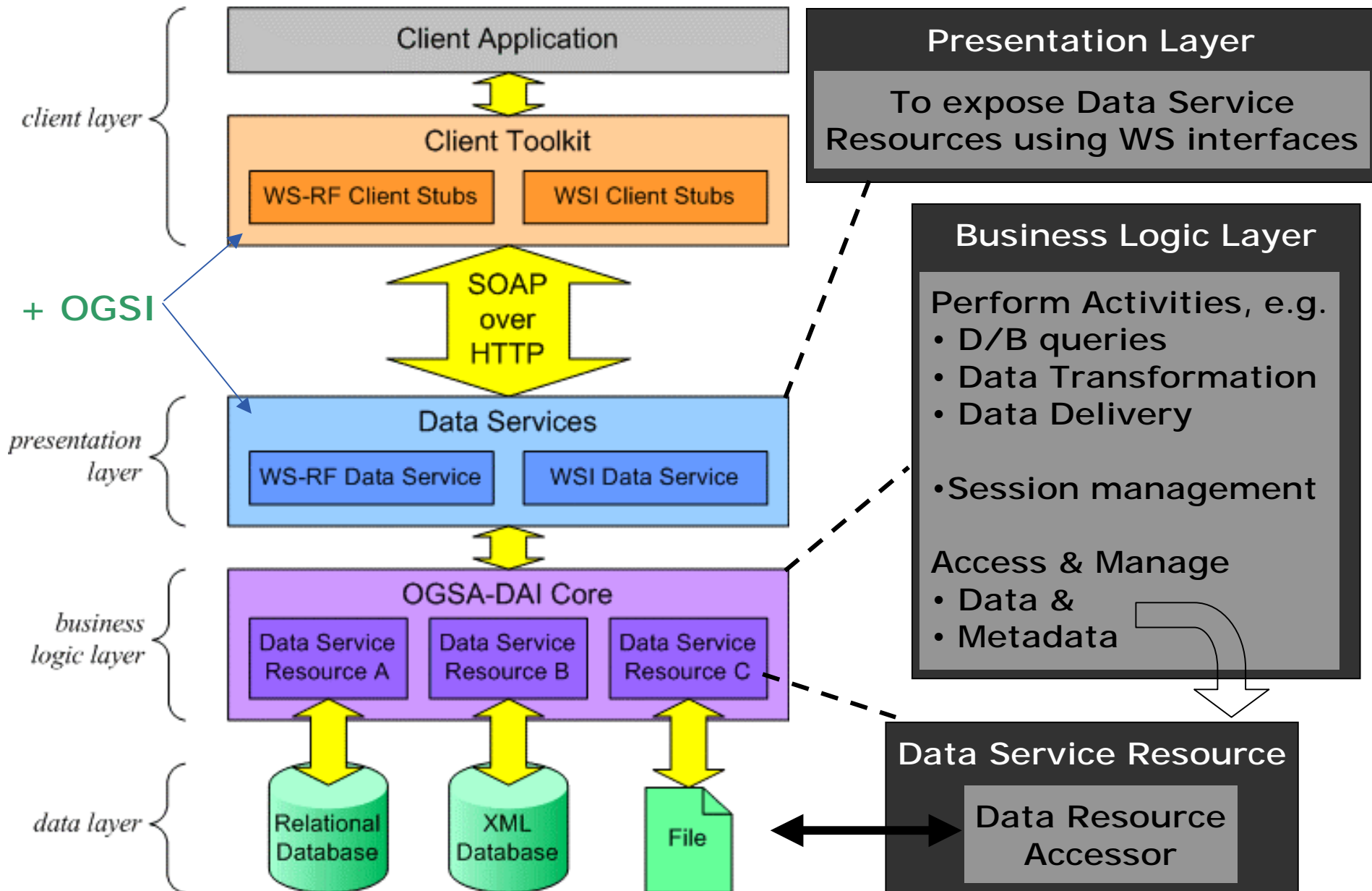
Core features of OGSA-DAI – I

- **A framework for building applications**
 - **Supports data access, insert and update**
 - ▶ Relational: MySQL, Oracle, DB2, SQL Server, Postgres
 - ▶ XML: Xindice, eXist
 - ▶ Files – CSV, BinX, EMBL, OMIM, SWISSPROT,...
 - **Supports data delivery**
 - ▶ SOAP over HTTP
 - ▶ FTP; GridFTP
 - ▶ E-mail
 - ▶ Inter-service
 - **Supports data transformation**
 - ▶ XSLT
 - ▶ ZIP; GZIP
 - **Supports security**
 - ▶ X.509 certificate based security

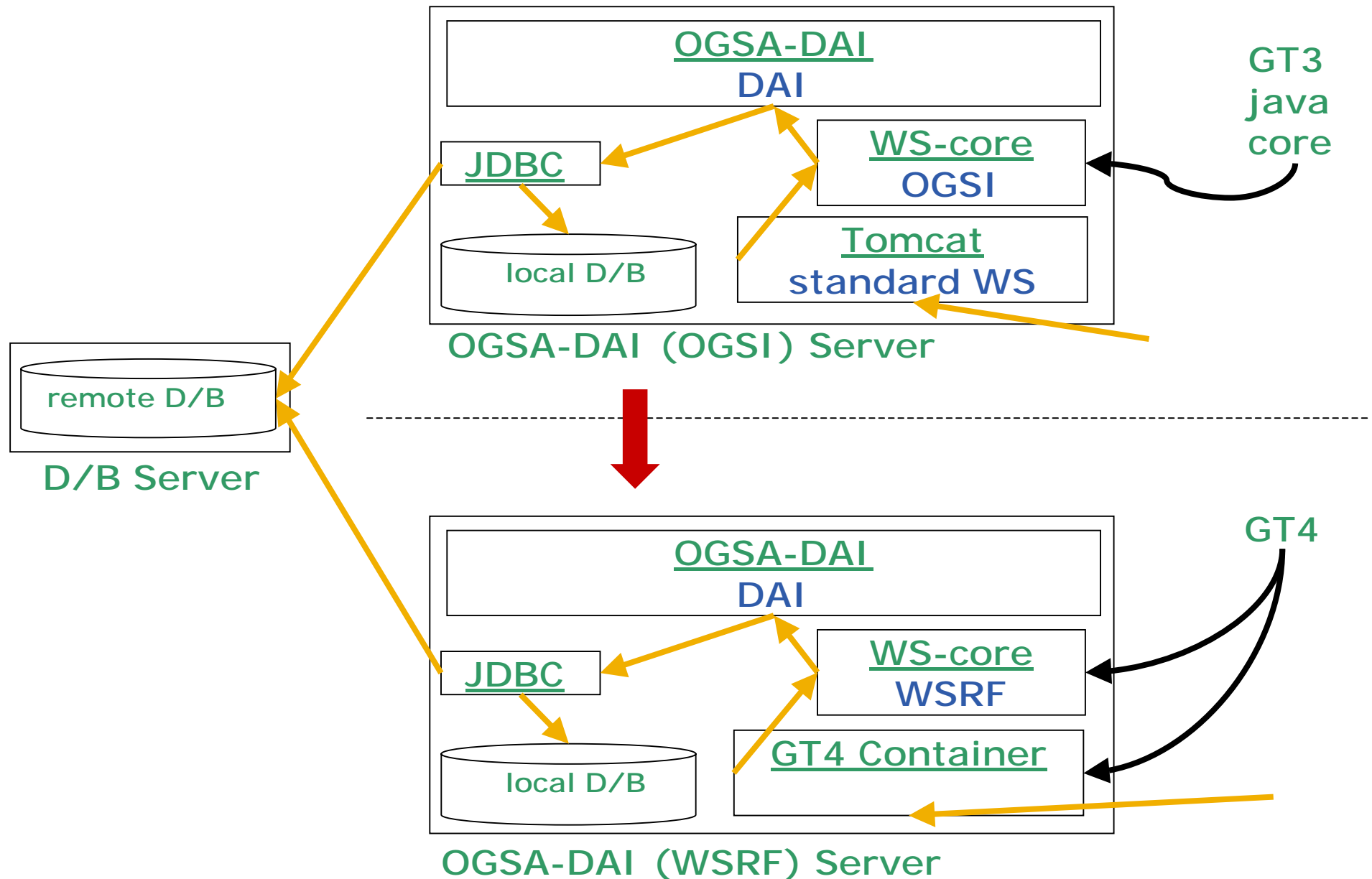
Core features of OGSA-DAI – II

- A framework for building data clients
 - Client toolkit library for application developers
- A framework for developing functionality
 - Extend existing activities, or implement your own
 - Mix and match activities to provide functionality you need
- Highly-extensible
 - Customise our out-of-the-box product
 - Provide your own services, client-side support and data-related functionality
- Comprehensive documentation and tutorials
- Latest release supports GT4.0, and Axis 1.2 / OMII_2 using Java 1.4/1.5

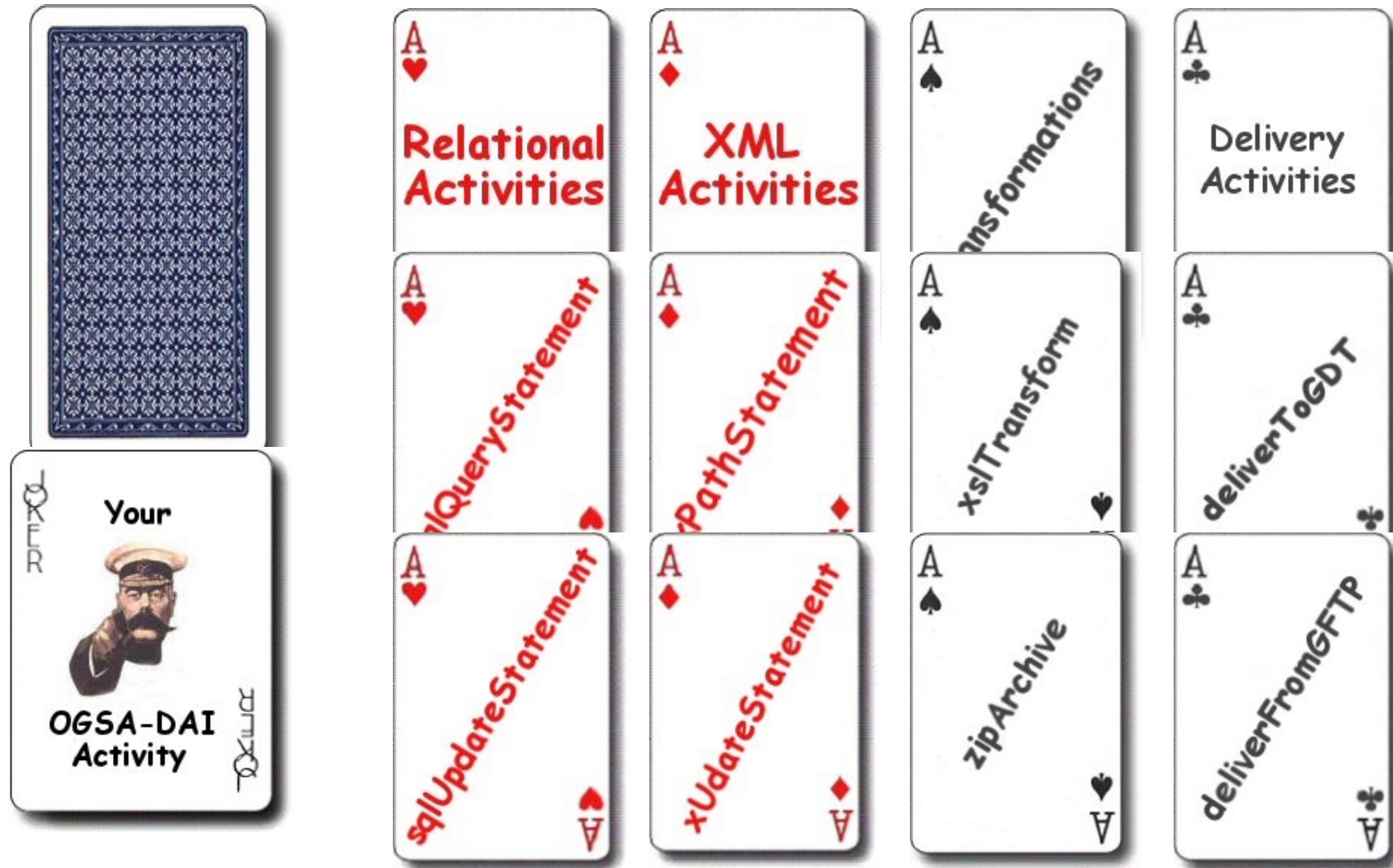
Layering



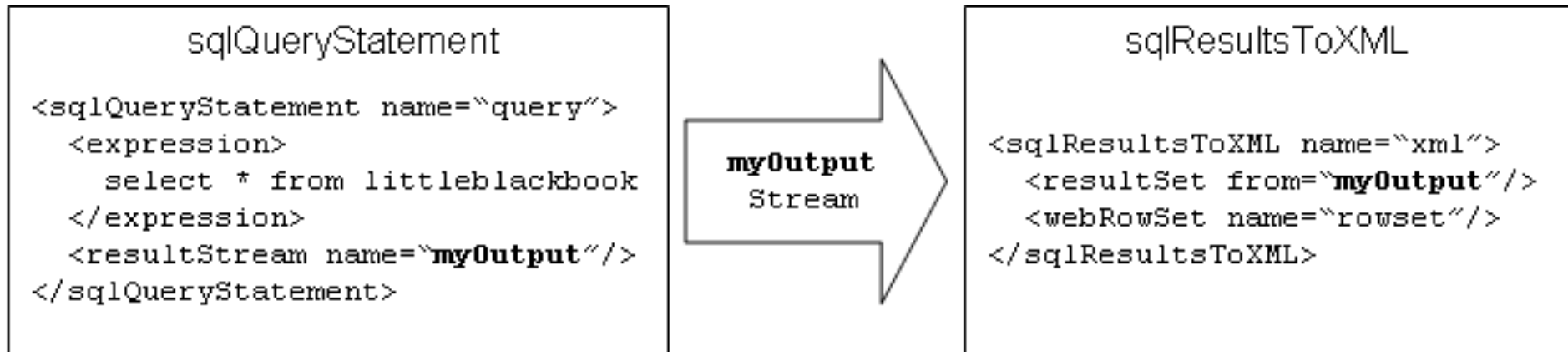
The OGSA-DAI stack



OGSA-DAI Deck of Activities



Activities Architecture



- activity is an operation that a data service can perform
- a request comprises zero or more activities
- within one request activities communicate data using named streams – pipeline / workflow model
- one stream can be input to multiple activities
- a stream is a sequence of blocks of data
 - written and read sequentially
- A block is a Java Object

Examples of built-in activity types

- Relational
 - [sqlQueryStatement](#) Run an SQL query statement **
 - [sqlUpdateStatement](#) Run an SQL update statement
 - [sqlStoredProcedure](#) Invoke an SQL stored procedure **
 - [sqlBulkLoadRowSet](#) Bulk load data into a table
 - [sqlResultsToXML](#) Convert ** results to XML
- XML
 - [XPathStatement](#) Run an XPath statement
 - [XUpdateStatement](#) Run an XUpdate statement
 - [xmlCollectionManagement](#)
Create or remove collections within an XML database.
 - [xmlResourceManagement](#)
Create or remove resources within an XML database.
 - [xmlBulkLoad](#) Bulk load resources into a collection.
- Transformation – [gzipCompression](#), [gzipDecompression](#)

Examples of built-in activity types

- **Delivery**

- ▶ `deliverToStream` – to make data available via http / https

- `deliverFromX` / `deliverToX` where $X =$

- ▶ *URL*

- ▶ *GFTP* – a GridFTP server

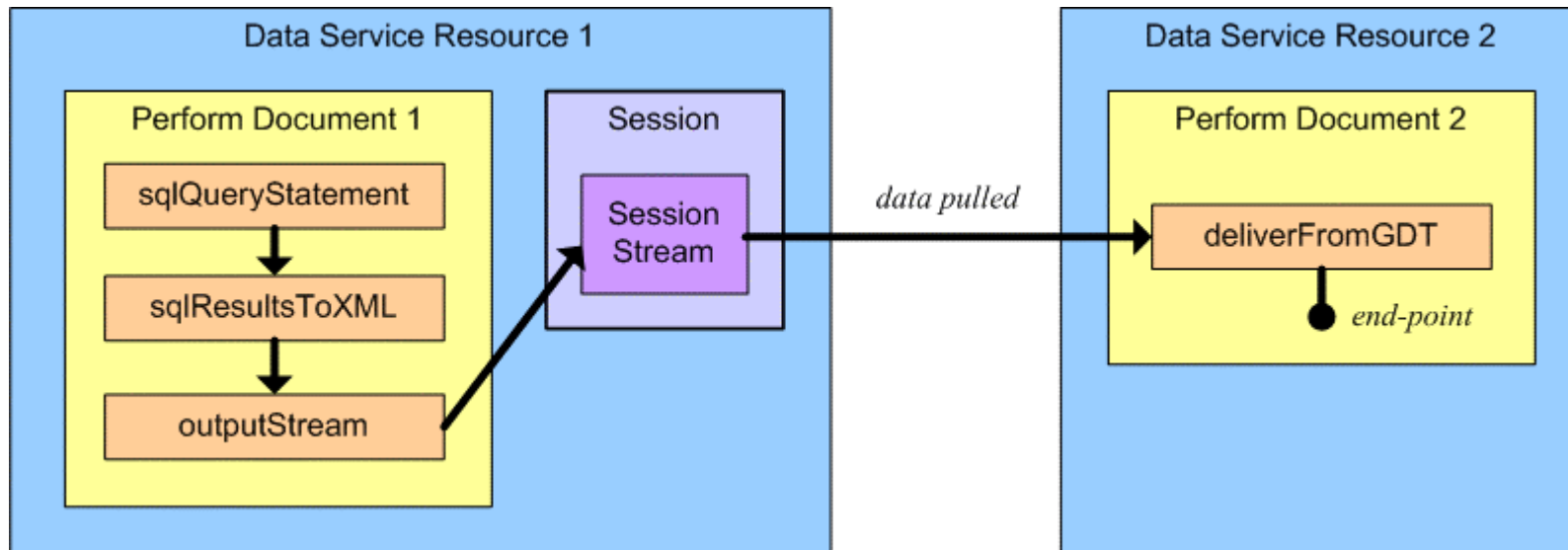
- ▶ (*File* – the container's local file system (insecure))

- ▶ *GDT* – for session stream of another session (another service)

- **Stream I/O**

- ▶ `InputStream` / `OutputStream` – for session stream of this session

Inter-session communication



- **PULL** : `outputStream` → `DeliverFromGDT`
 - stream exists in producer session
 - producer is demand-driven by consumer
- **PUSH** : `DeliverToGDT` → `inputStream`
 - stream exists in consumer session
 - consumer is availability-driven by producer
- Lifetime of stream bounded by lifetime of session where it exists
- OGSi omits session concept – lifetime problem

Content

- Data on the Grid – what it's about
- DAIS and OGSA-DAI
- Projects OGSA-DAI and its users ●

OGSA-DAI Project

- **Develop a component library**
 - Access and manipulate data in a grid
- **Aims to provide**
 - Common interface to data resources
 - Simple integration of distributed queries to multiple data resources
- **Contribute to standardisation efforts**
 - Input into GGF DAIS WG and other groups
- **Based on Open Grid Services Architecture (OGSA)**
 - Started with Globus Toolkit 3 (GT3)
 - Moved to WS-RF (GT4) and WS-I + (OMII)

Project Status

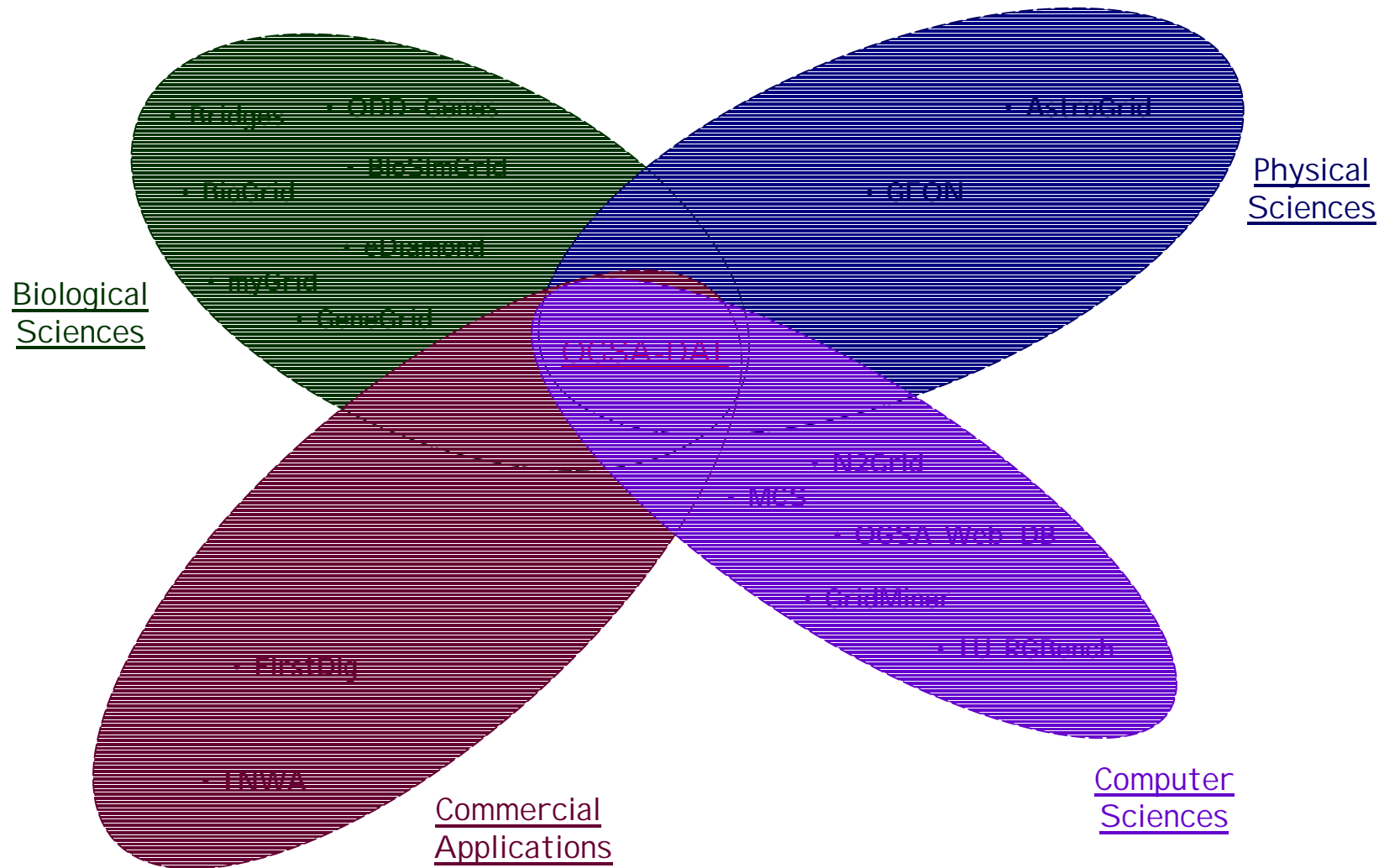
- **Current release 7.0**
 - GT4.0, OMII_2, Axis 1.2
 - Platform and language independent
 - ▶ Java 1.4
 - ▶ Document model
- **Work concentrated on data access**
 - Wraps data resources without hiding underlying data model
 - Provide base for higher-level services
 - ▶ Distributed Query Processing (DQP)
 - ▶ Data federation services

Platforms and Projects

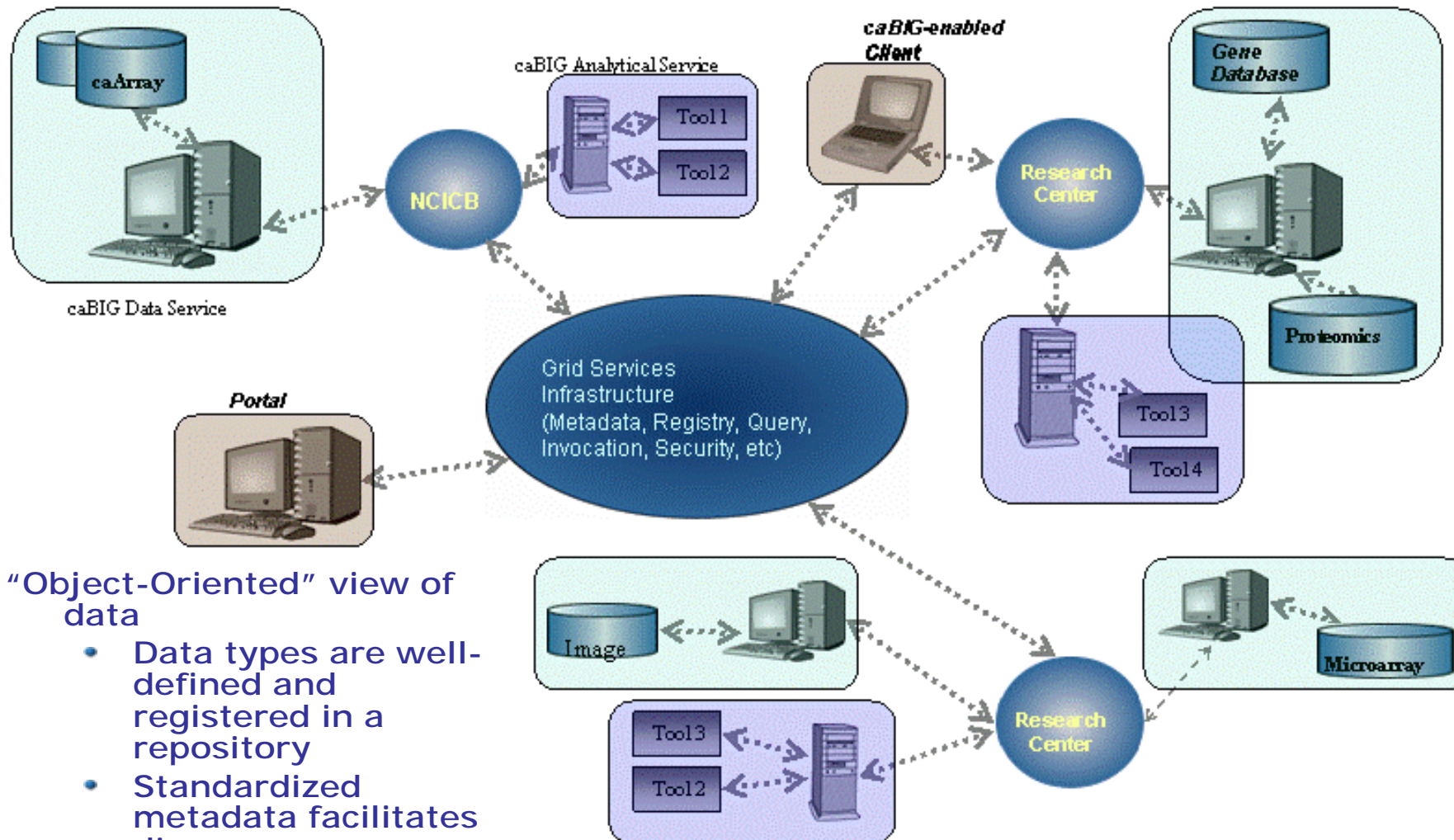
- **OMII**
 - Current version of OGSA-DAI WS-I distribution runs on OMII_2
- **Globus**
 - WSRF 0.9.6 distribution bundled with GT4.0
 - WSRF 1.0 distribution bundled with GT4.0.1
 - WSRF 2.0 works with GT4.0.1
- **Projects**
 - Number of projects have used/use/will use OGSA-DAI

AstroGrid	Biogrid	BioSimGrid	Bridges	caGrid	DataMining Grid
eDiamond	FirstDig	GEDDM	GeneGrid	GEON	GridMiner
INWA	IU RGRBench	LEAD	MCS	myGrid	N2Grid
ODD-Genes	OGSA- WebDB	SIMDAT	GOLD		

Project classification



caBIG

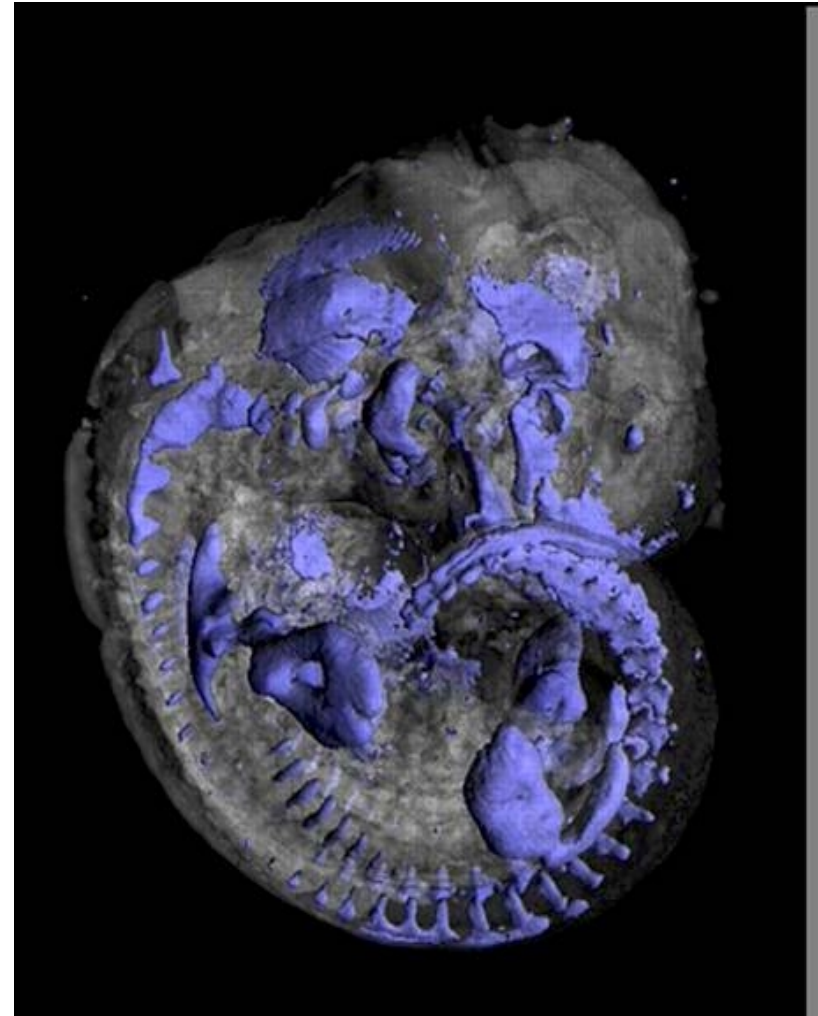


"Object-Oriented" view of data

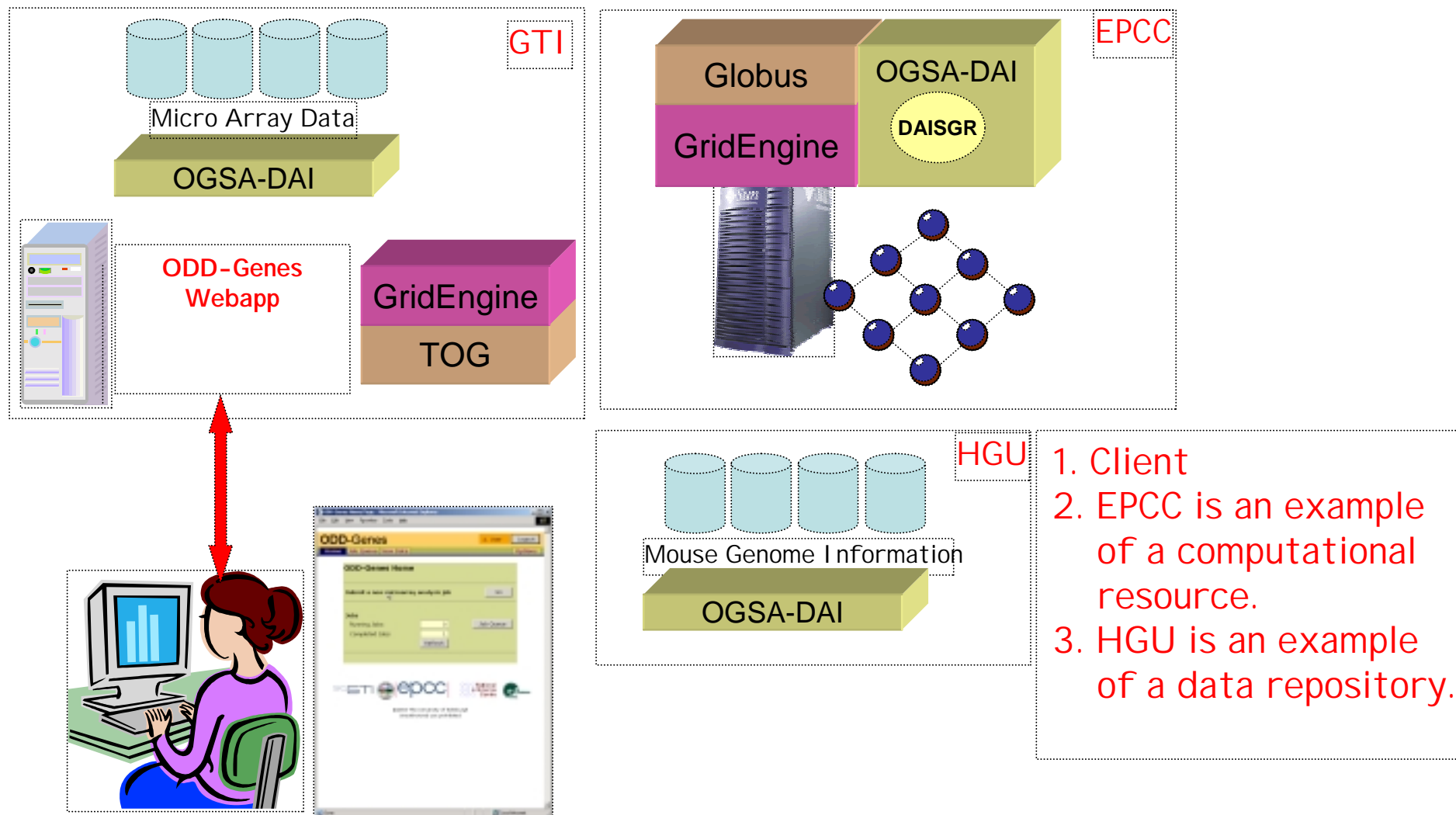
- Data types are well-defined and registered in a repository
- Standardized metadata facilitates discovery
- custom query language implemented as an activity

ODD-Genes

- OGSA-DAI Demo for Genetics
- Collaboration between
 - EPCC
 - Scottish Centre for Genomic Technology and Informatics (GTI)
 - Human Genetics Unit (HGU)
- ODD-Genes demonstrates:
 - Perform high-speed batch analysis of microarray data on the Grid
 - Browse the results of previous analyses stored in a database
 - View data from arbitrary databases as HTML
 - Discover related databases on the Grid
 - Perform coupled queries on newly-discovered databases to provide a richer analysis of gene data



ODD-Genes Actors



1. Client
2. EPCC is an example of a computational resource.
3. HGU is an example of a data repository.

ODD-Genes Findings

- Data discovery perceived to be very important
 - Map data views: time -> spatial locations
 - Discovery of new resources
- Transparency to data access
 - @HGU had an XML database
 - @GTI had a relational database
 - Deploy OGSA-DAI and not worry about databases
- Issues
 - Registry maintenance policy
 - Semantics of the discovery process
 - Groups working the same area but different schemas, no generic metadata (schemas were the effective metadata)
- Provides an additional tool for researchers

Further information

- **The OGSA-DAI Project Site:**
 - <http://www.ogsadai.org.uk>
- **The DAIS-WG site:**
 - <http://forge.gridforum.org/projects/dais-wg/>
- **OGSA-DAI Users Mailing list**
 - users@ogsadai.org.uk
 - General discussion on grid DAI matters
- **Formal support for OGSA-DAI releases**
 - <http://www.ogsadai.org.uk/support>
 - support@ogsadai.org.uk
- **OGSA-DAI training courses**

OGSA-DAI Project Webpage

- <http://www.ogsadai.org.uk>



- Background
- News & Events
- Software Releases
- Documentation
- On-line Tutorials
- Support
- Training Courses
- Links