the globus alliance
www.globus.org

# Globus Data Services for Science

**Raj Kettimuthu**
Argonne National Laboratory/Univ. of Chicago

**Ann Chervenak, Rob Schuler**
USC Information Sciences Institute

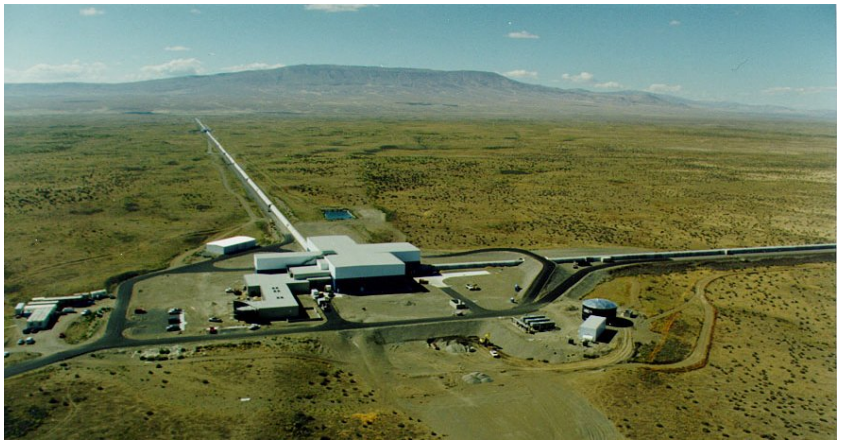
# Globus Services for Data Intensive Science

- Data Movement:
  - GridFTP and Reliable File Transfer Service (RFT)
- Replica management:
  - Replica Location Service (RLS) and Data Replication Service (DRS)
  - New: Policy-based data placement service
- Access to databases and other data sources:
  - OGSA Data Access and Integration (DAI) Service

# Talk Outline

- Examples of production data intensive science projects that use Globus services
- New features:
  - GridFTP and RFT
  - Replica management tools
  - Data placement services
  - Data access and integration services

# The LIGO Project



- Laser Interferometer Gravitational Wave Observatory
- LIGO instruments in Washington State and Louisiana
- During science runs, produce up to 2 terabytes per day
- Published along with metadata at Caltech (archival site)
- Replicated at up to 10 other LIGO sites
  - LIGO scientists typically move data sets near to computational clusters at their sties
- The LIGO Data Grid

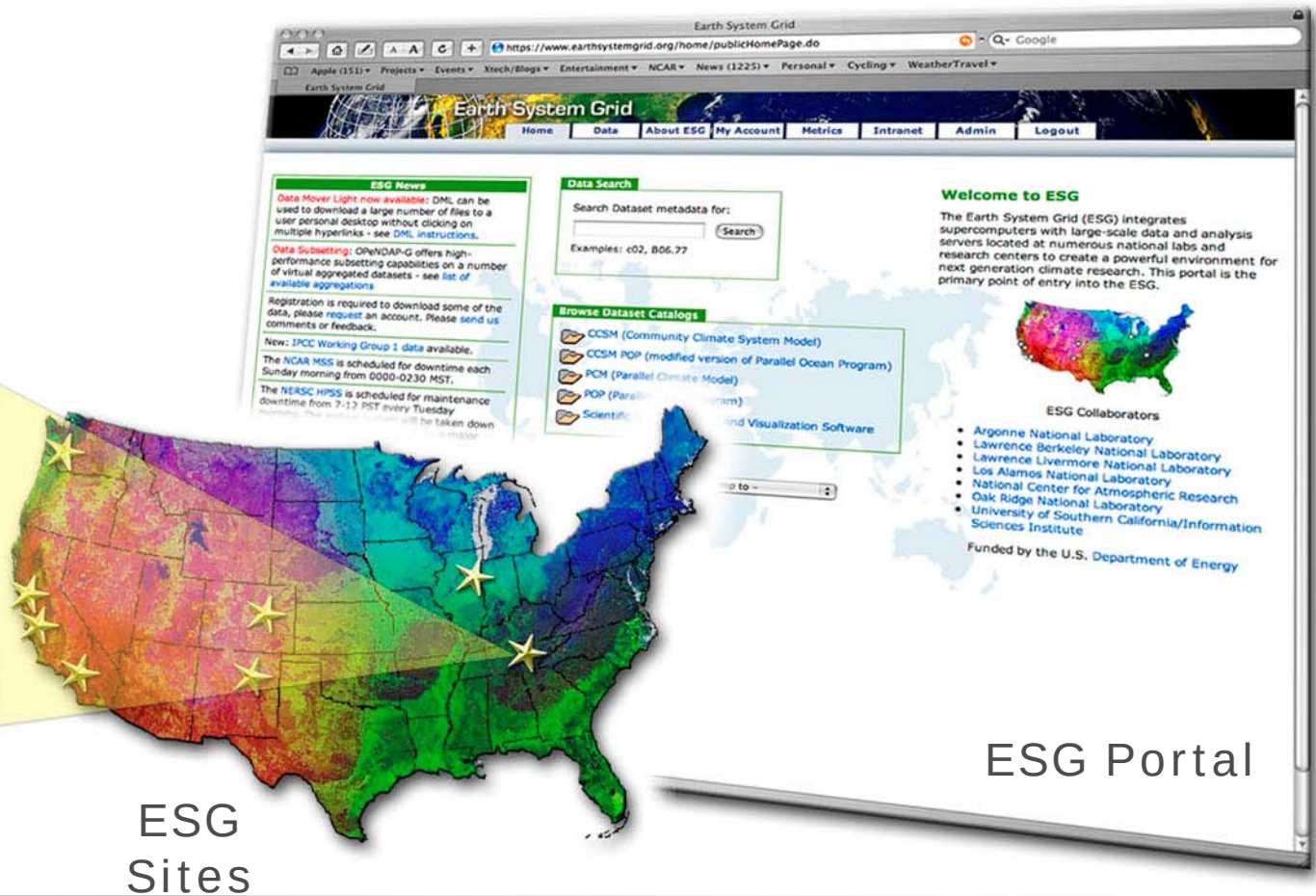
# Globus Services in the LIGO Data Grid

- Lightweight Data Replicator (LDR): data management system developed by LIGO researchers
- Globus data services:
  - GridFTP: used for moving data around the Grid efficiently and securely
  - Replica Location Service: catalogs deployed at all LIGO sites, keep track of locations of over 150 million files
  - Data Replication Service was developed to generalize the functionality in the LDR
- Other Globus services:
  - Globus security
  - Starting to deploy the Globus Monitoring and Discovery Service

# Earth System Grid objectives

**To support the infrastructural needs of the national and international climate community, ESG is providing crucial technology to securely access, monitor, catalog, transport, and distribute data in today's grid computing environment.**

HPC
hardware running
climate models




ESG
Sites

ESG Portal

the globus alliance
www.globus.org

# ESG facts and figures

## Main ESG Portal

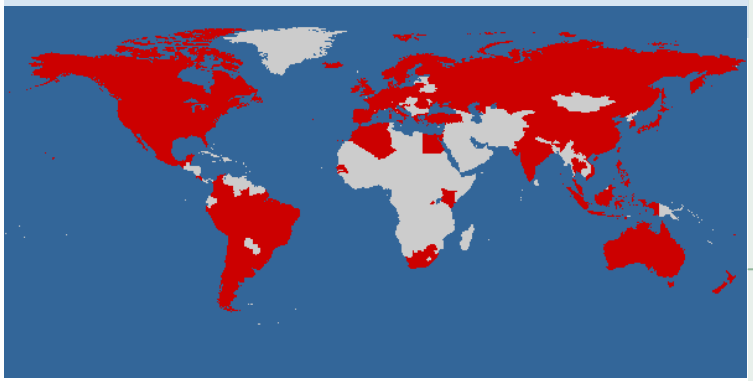
146 TB of data at four locations

- 1,059 datasets
- 958,072 files
- Includes the past 6 years of joint DOE/NSF climate modeling experiments

4,910 registered users

Downloads to date

- 30 TB
- 106,572 files



Worldwide ESG user base

## IPCC AR4 ESG Portal

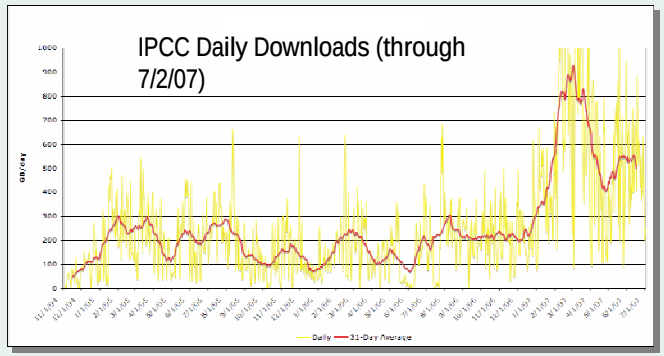
35 TB of data at one location

- 77,400 files
- Generated by a modeling campaign coordinated by the Intergovernmental Panel on Climate Change
- Model data from 13 countries

1,245 registered analysis projects

Downloads to date

- 245 TB
- 914,400 files
- 500 GB/day (average)



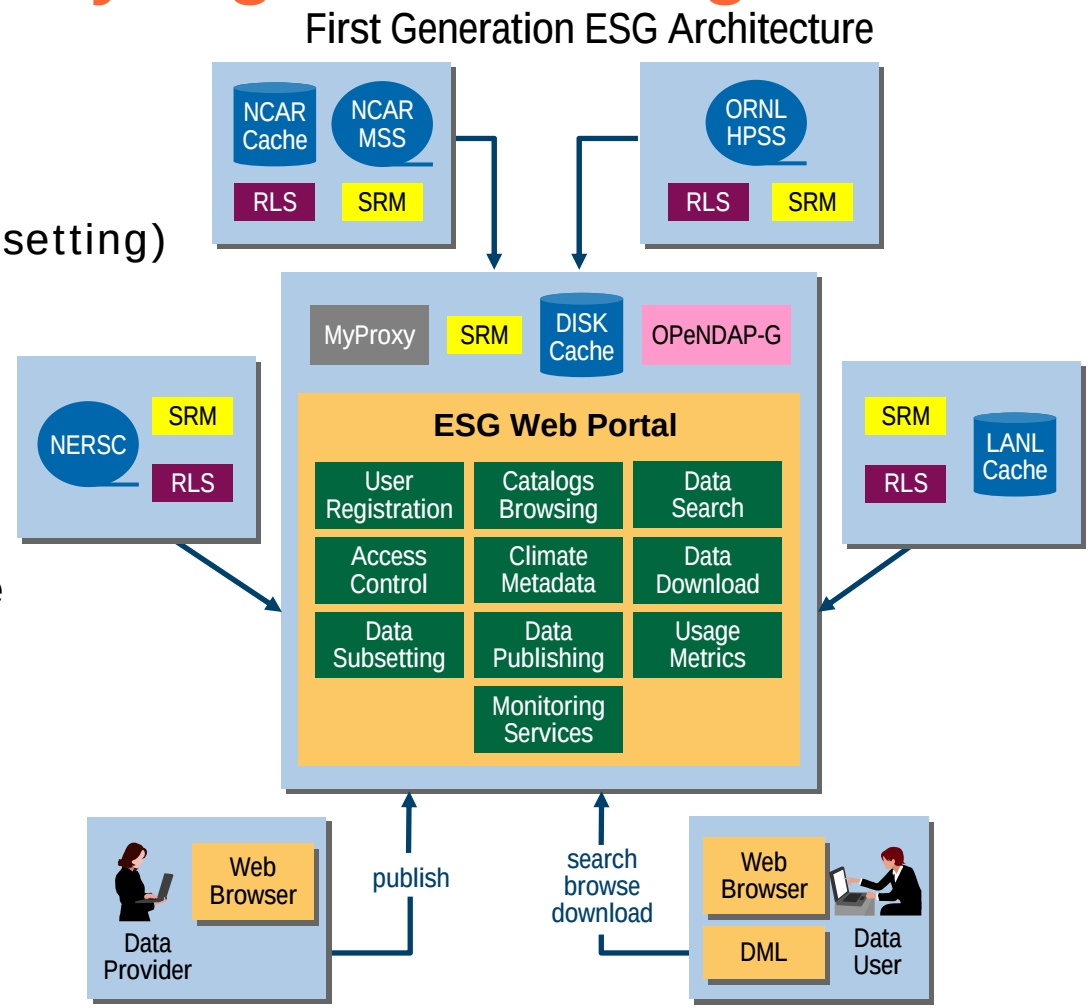

> 300 scientific papers published to date based on analysis of IPCC AR4 data

Slide Courtesy of Dave Bernholdt, ORNL

the globus alliance
www.globus.org

# ESG architecture and underlying technologies

- Climate data tools
  - Metadata catalog
  - NcML (metadata schema)
  - OPenDAP-G (aggregation, subsetting)
- Data management
  - Data Mover Lite
  - Storage Resource Manager
- Globus toolkit
  - Globus Security Infrastructure
  - GridFTP
  - Monitoring and Discovery Services
  - Replica Location Service
- Security
  - Access control
  - MyProxy
  - User registration

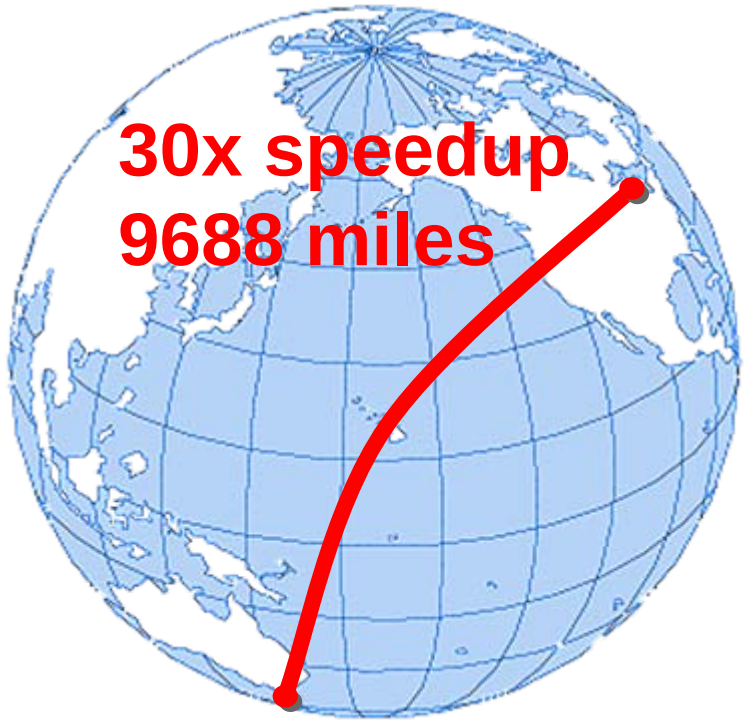



MSS, HPSS: **Tertiary data storage systems**

# GridFTP Data Transfers for the Advanced Photon Source

"One Australian user left nearly 1TB of data on our systems that we had been struggling to transfer via standard FTP for several weeks. The typical data rate using standard FTP was ~200 KB/s. Using GridFTP we are now moving data at 6 MB/s—quite a significant boost in performance!"

**Brian Tieman**
**Advanced Photon Source**



the globus alliance
www.globus.org

# What's New in Globus GridFTP and RFT

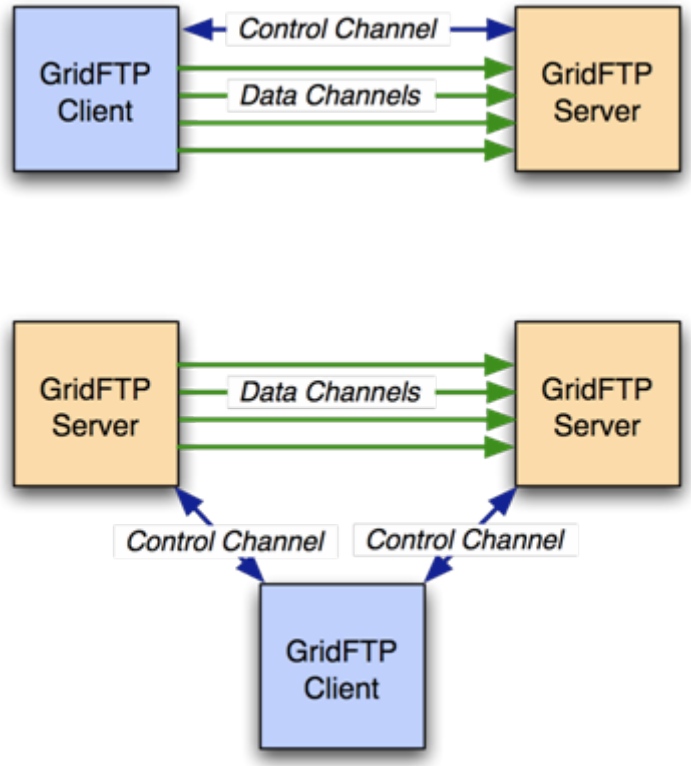
Raj Kettimuthu
Argonne National Laboratory and
The University of Chicago

# What is GridFTP?

- High-performance, reliable data transfer protocol optimized for high-bandwidth wide-area networks
- Based on FTP protocol - defines extensions for high-performance operation and security
- We supply a reference implementation:
  - Server
  - Client tools (globus-url-copy)
  - Development Libraries
- Multiple independent implementations can interoperate
  - Fermi Lab and U. Virginia have home grown servers that work with ours.

# GridFTP

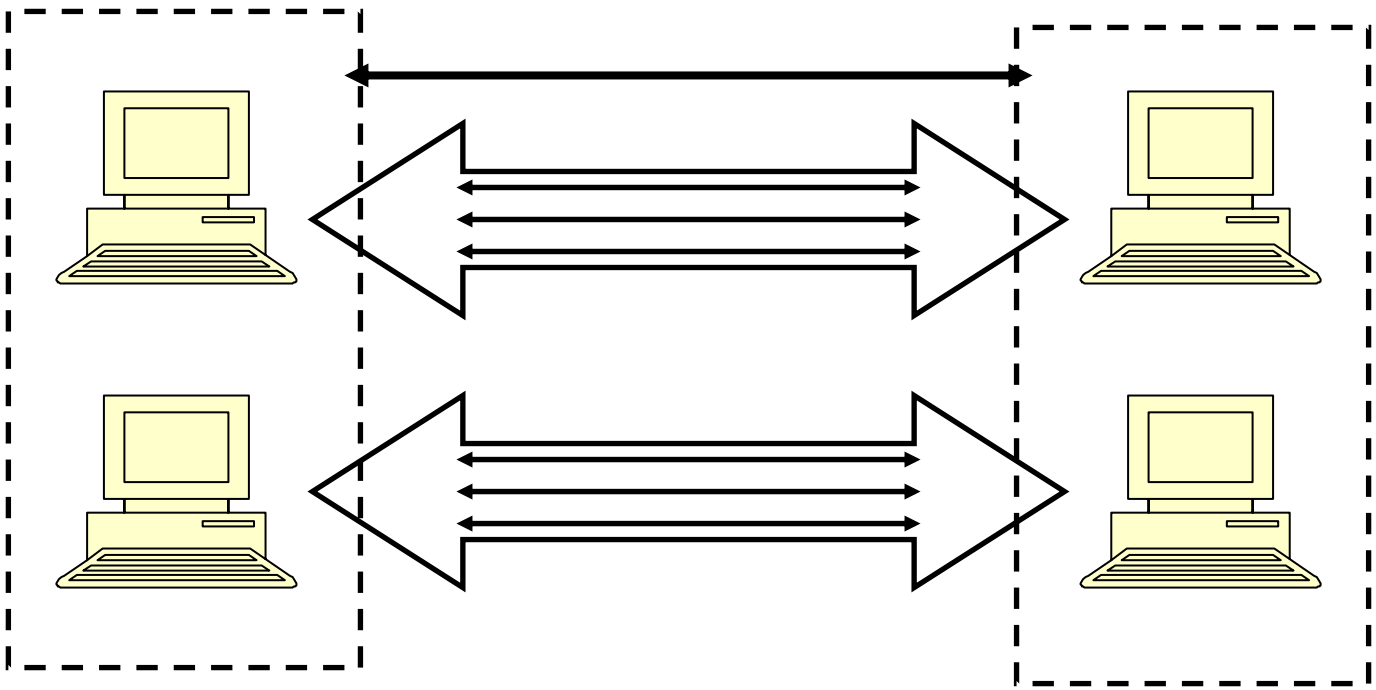
- Two channel protocol like FTP
- Control Channel
  - Communication link (TCP) over which commands and responses flow
  - Low bandwidth; encrypted and integrity protected by default
- Data Channel
  - Communication link(s) over which the actual data of interest flows
  - High Bandwidth; authenticated by default; encryption and integrity protection optional



# Why GridFTP?

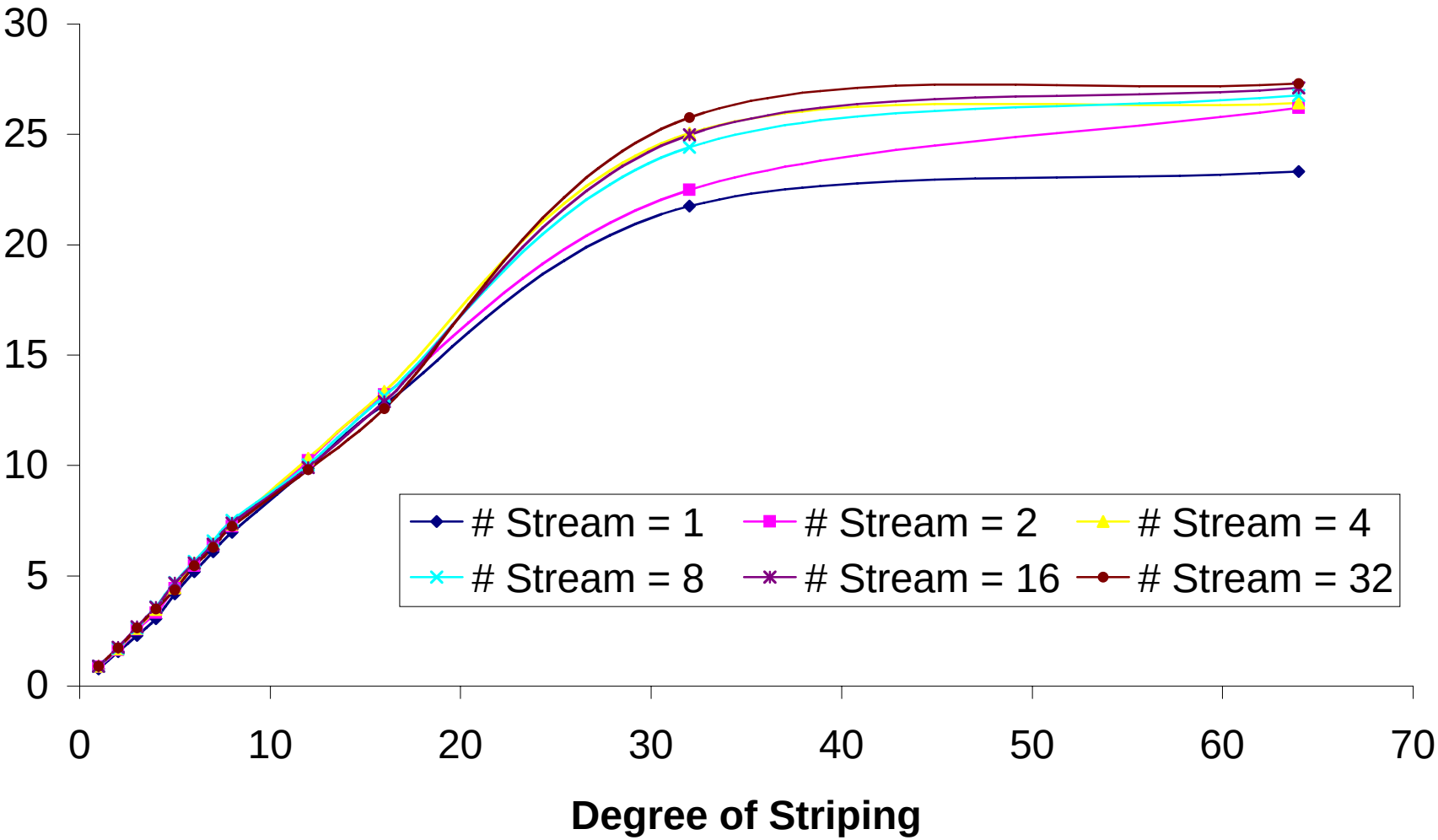
- Performance
  - Parallel TCP streams
  - Non TCP protocol such as UDT
  - Order of magnitude greater
- Cluster-to-cluster data movement
  - Another order of magnitude
- Support for reliable and restartable transfers
- Multiple security options
  - Anonymous, password, SSH, GSI

# Cluster-to-Cluster transfers



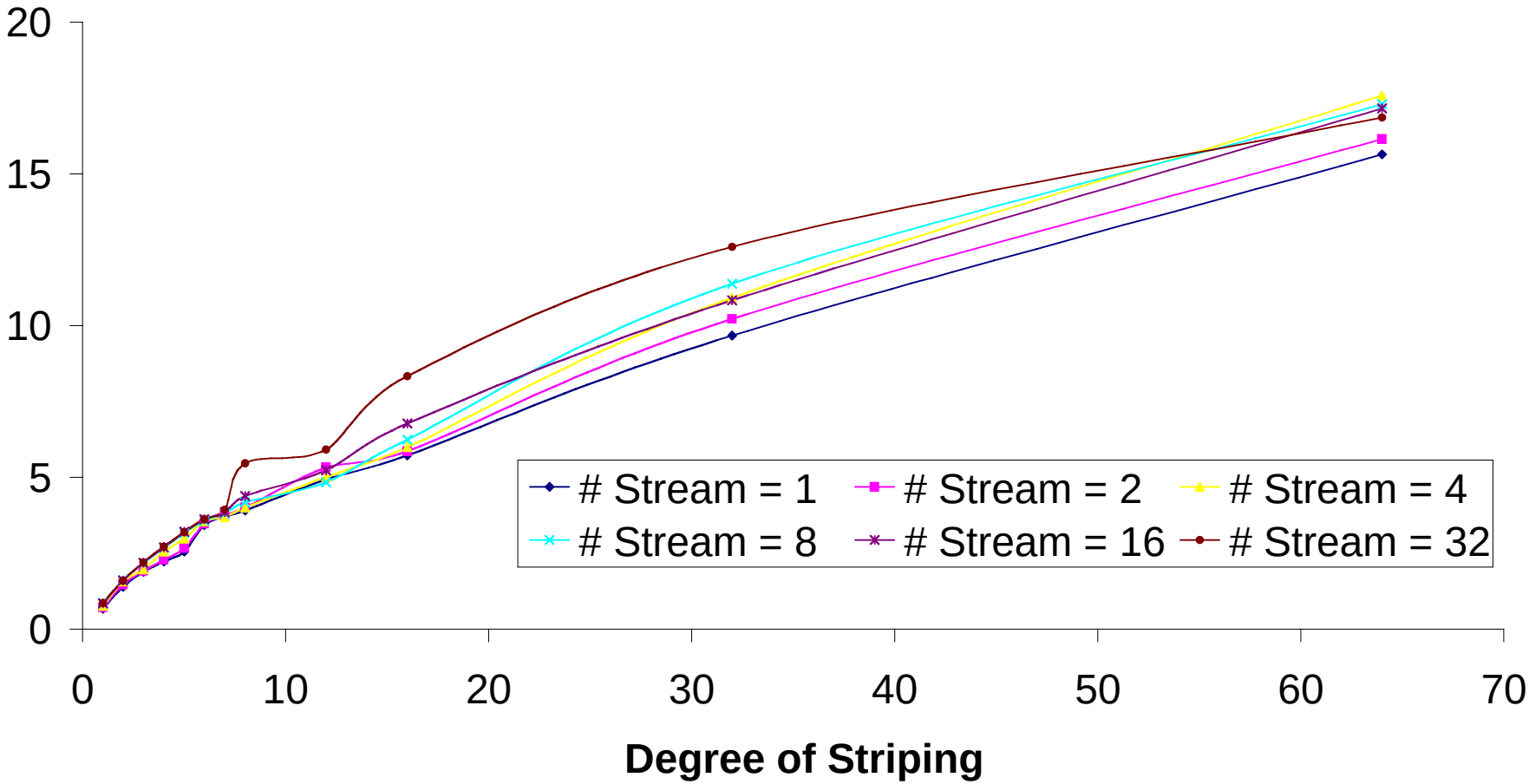
# Performance

- Mem. transfer between Urbana, IL and San Diego, CA



# Performance

- Disk transfer between Urbana, IL and San Diego, CA

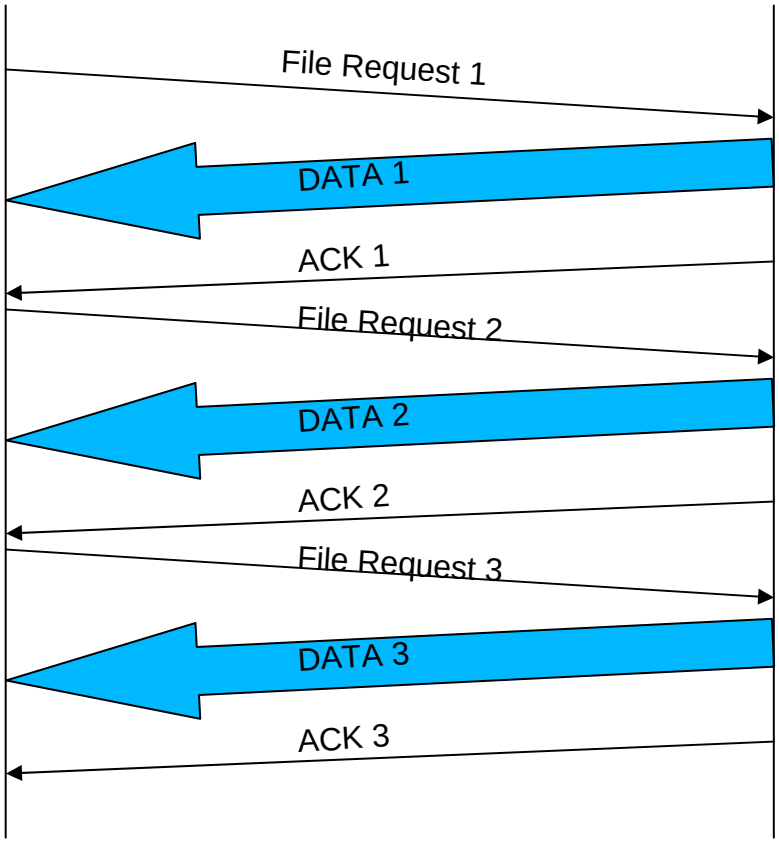


# Users

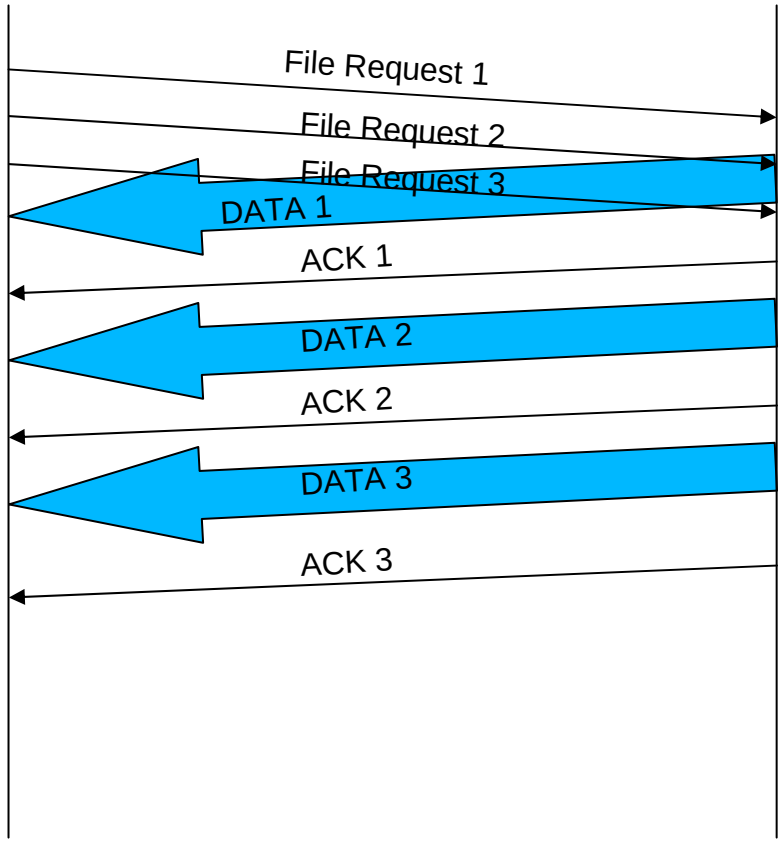
- HEP community is basing its entire tiered data movement infrastructure for the LHC computing Grid on GridFTP
- Southern California Earthquake Center (SCEC), European Space Agency, Disaster Recovery Center in Japan use GridFTP for data movement
- An average of more than 2 million data transfers happen with GridFTP every day

# LOSF and Pipelining

## Traditional



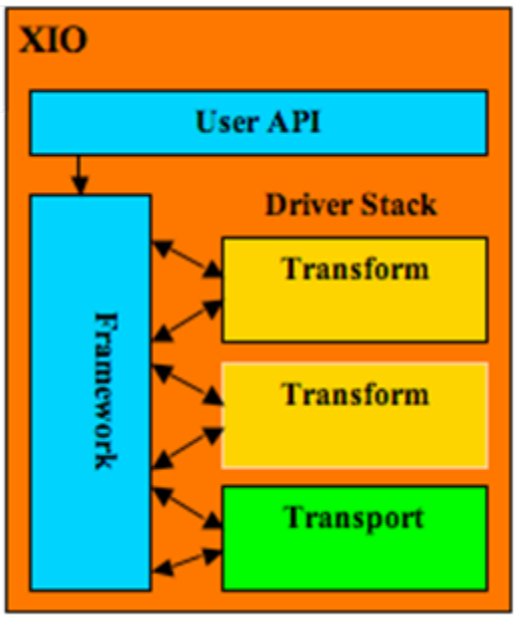

## Pipelining

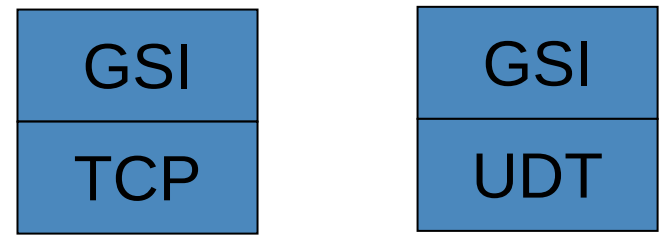



- Significant performance improvement for LOSF

# GridFTP over UDT

- GridFTP uses XIO for network I/O operations
- XIO presents a POSIX-like interface to many different protocol implementations

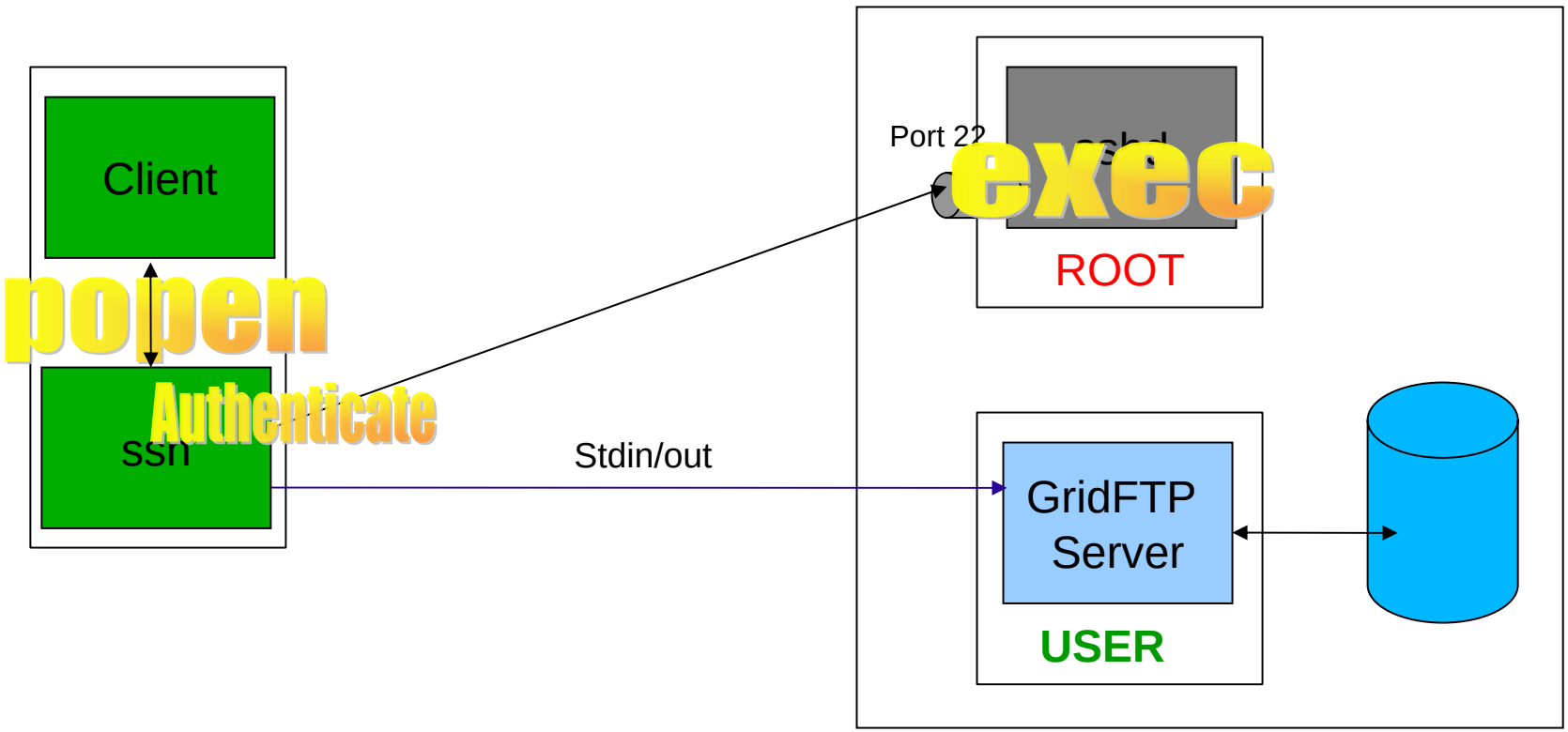


# GridFTP over UDT

| | Argonne to NZ Throughput in Mbit/s | Argonne to LA Throughput in Mbit/s |
|---|---|---|
| Iperf – 1 stream | 19.7 | 74.5 |
| Iperf – 8 streams | 40.3 | 117.0 |
| GridFTP mem TCP – 1 stream | 16.4 | 63.8 |
| GridFTP mem TCP – 8 streams | 40.2 | 112.6 |
| GridFTP disk TCP – 1 stream | 16.3 | 59.6 |
| GridFTP disk TCP – 8 streams | 37.4 | 102.4 |
| GridFTP mem UDT | 179.3 | 396.6 |
| GridFTP disk UDT | 178.6 | 428.3 |
| UDT mem | 201.6 | 432.5 |
| UDT disk | 162.5 | 230.0 |

# SSH Security for GridFTP



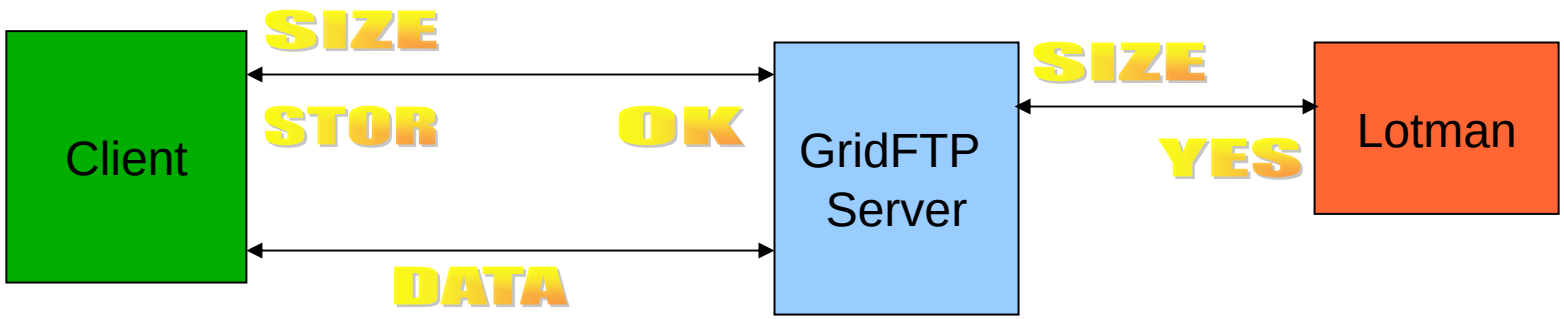
# Multicast / Overlay Routing

- Enable GridFTP to transfer single data set to many locations or act as an intermediate routing node

# Storage Plugin

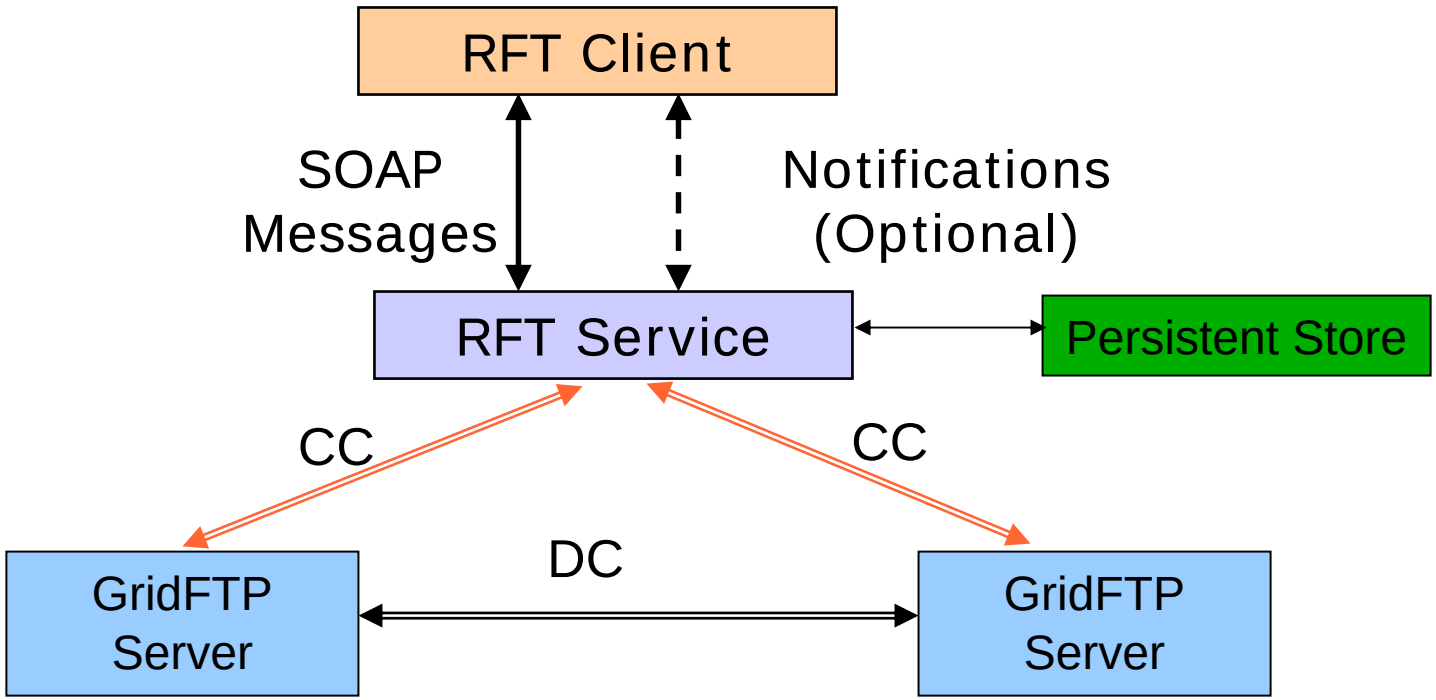
- Destination storage might run out of space in the middle of a GridFTP transfer
- Lotman - tool from univ. of wisconsin that manages storage
- Developed plugin for GridFTP to interact with Lotman
- Space availability (for individual file transfers) determined ahead of transfers to Lotman enabled storage

# GridFTP with Lotman



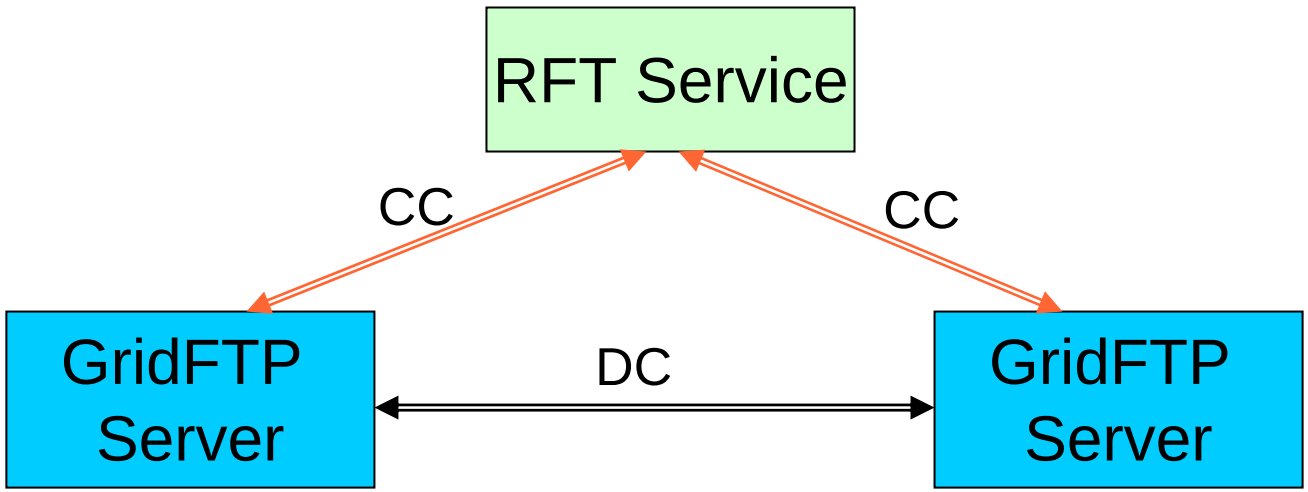
# Reliable File Transfer Service (RFT)

- GridFTP client
- WSRF complaint fault-tolerant service



# RFT - Connection Caching

- Control channel connections (and thus the data channels associated with it) are cached to reuse later (by the same user)



# RFT - Connection Caching

- Reusing connections eliminate authentication overhead on the control and data channels
- Measured performance improvement for jobs submitted using Condor-G
- For 500 jobs - each job requiring file stageIn, stageOut and cleanup (RFT tasks)
  - 30% improvement in overall performance
  - No timeout due to overwhelming connection requests to GridFTP servers

the globus alliance
www.globus.org

# What's new in Data Access and Integration?

Raj Kettimuthu on behalf of OGSA-DAI team

epcc

National e-Science Centre

omii-uk
www.omii.ac.uk

the globus alliance
www.globus.org

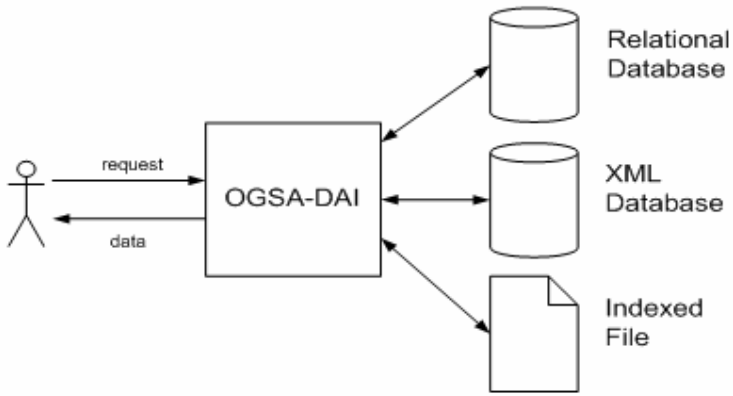
EPSRC Engineering and Physical Sciences Research Council

NextGRID

BEinGRID

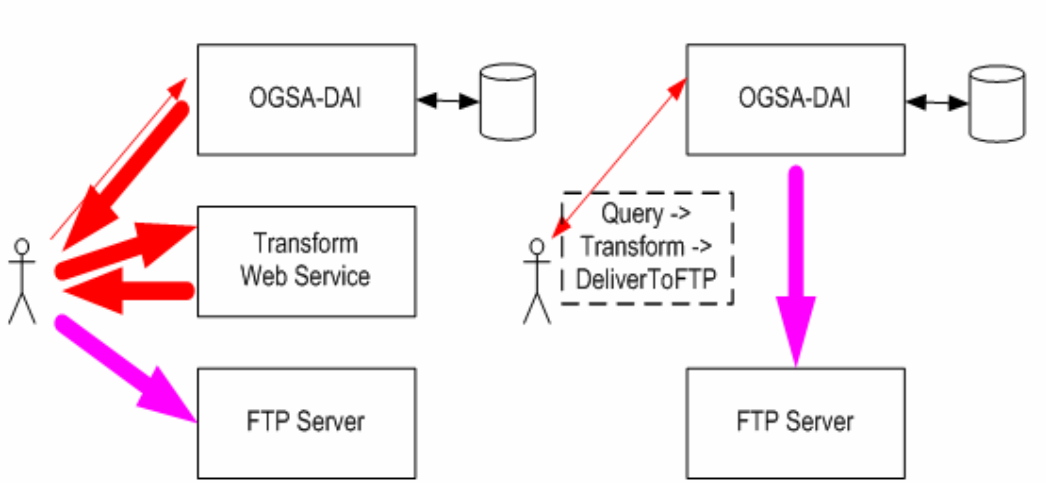
# What is OGSA-DAI?

- Middleware that allows data resources, such as relational or XML databases, to be accessed via web services

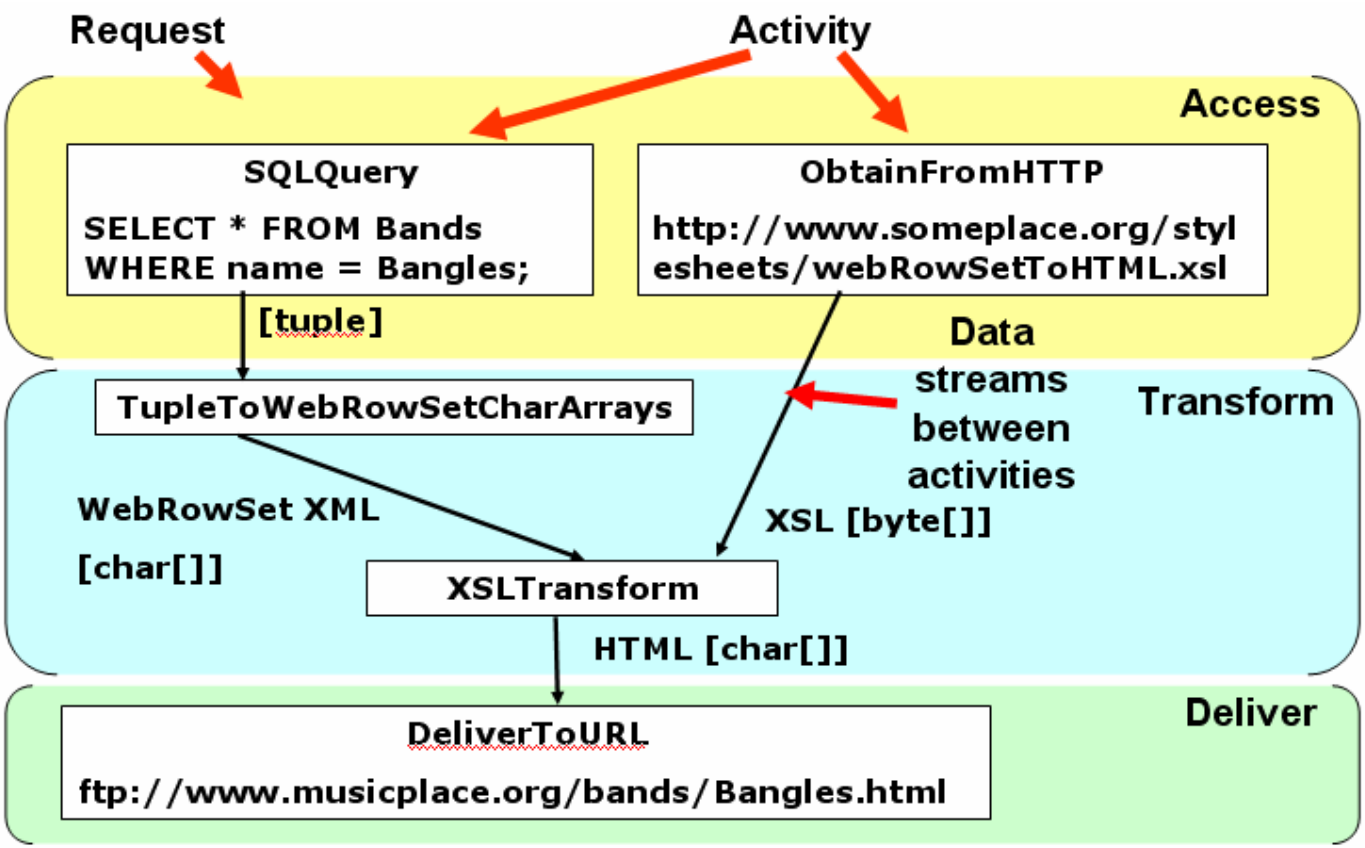


# What is OGSA DAI?

- OGSA-DAI executes **workflows**
- OGSA-DAI is not just for data access, also does data updates, transformations and delivery.

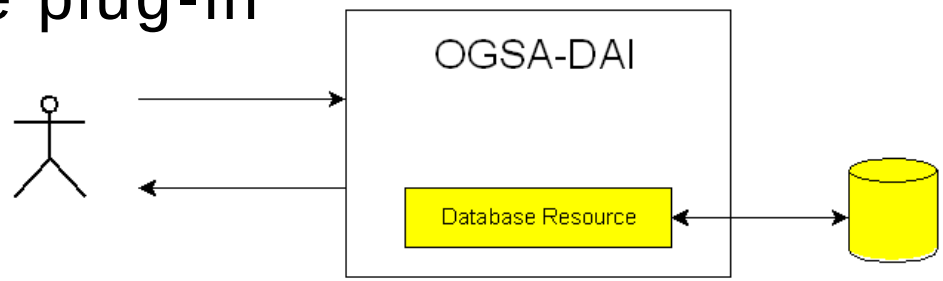


# OGSA DAI Workflow

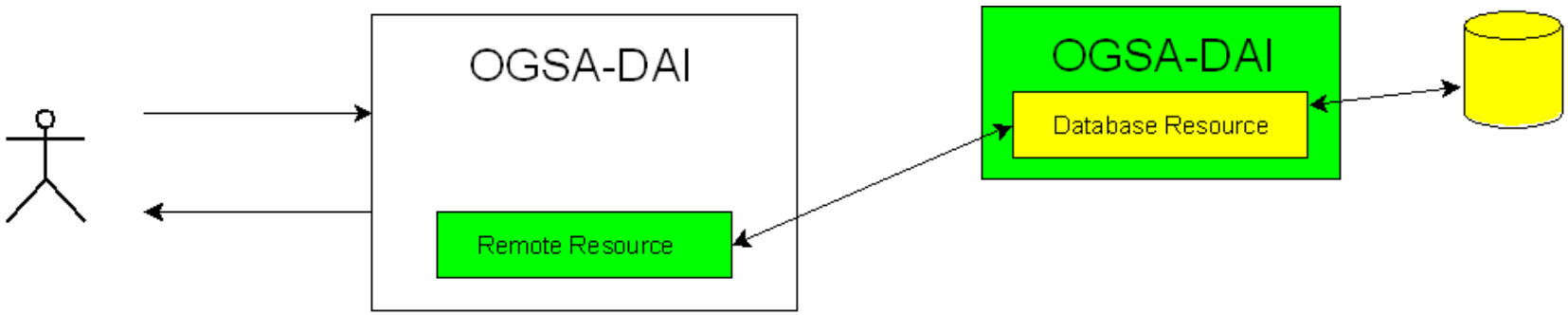


# Remote resource access

- OGSA-DAI ⇔ data resource interaction
  - Via a data resource plug-in




- Remote resource access
  - Access a data resource managed by another OGSA-DAI server

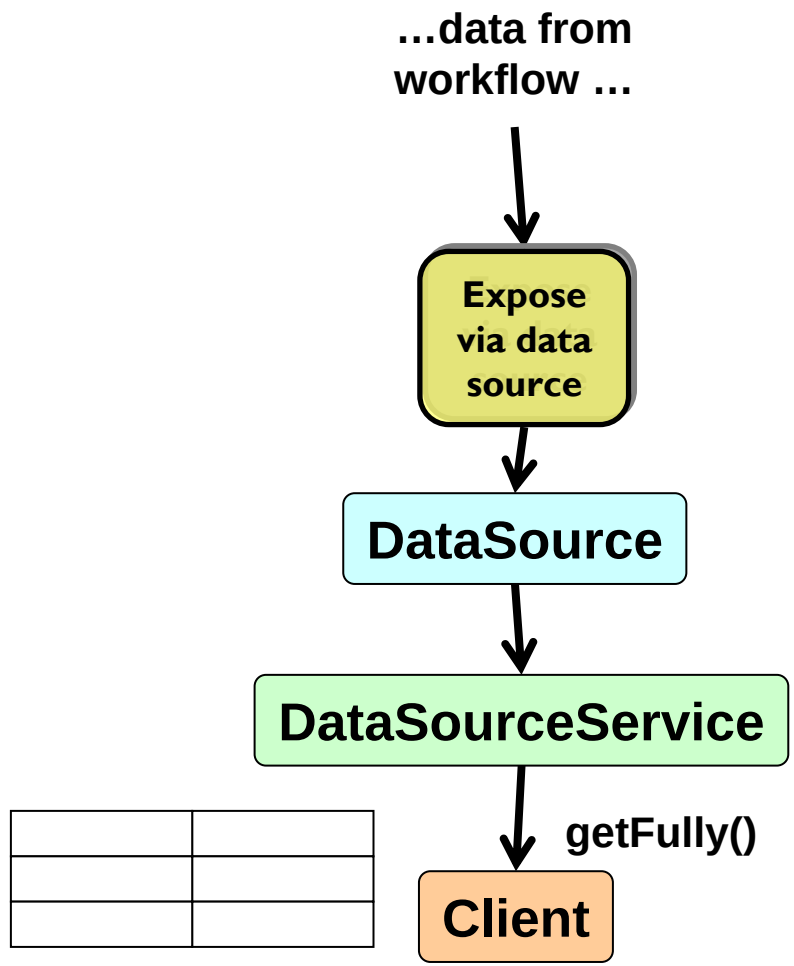


# Remote resource access

- Remote resource plug-in
  - Basically a client to a remote OGSA-DAI server
  - Runs queries via workflow submission
  - Configured with URL of remote server
- Transparent to OGSA-DAI infrastructure
  - Just another data resource plug-in

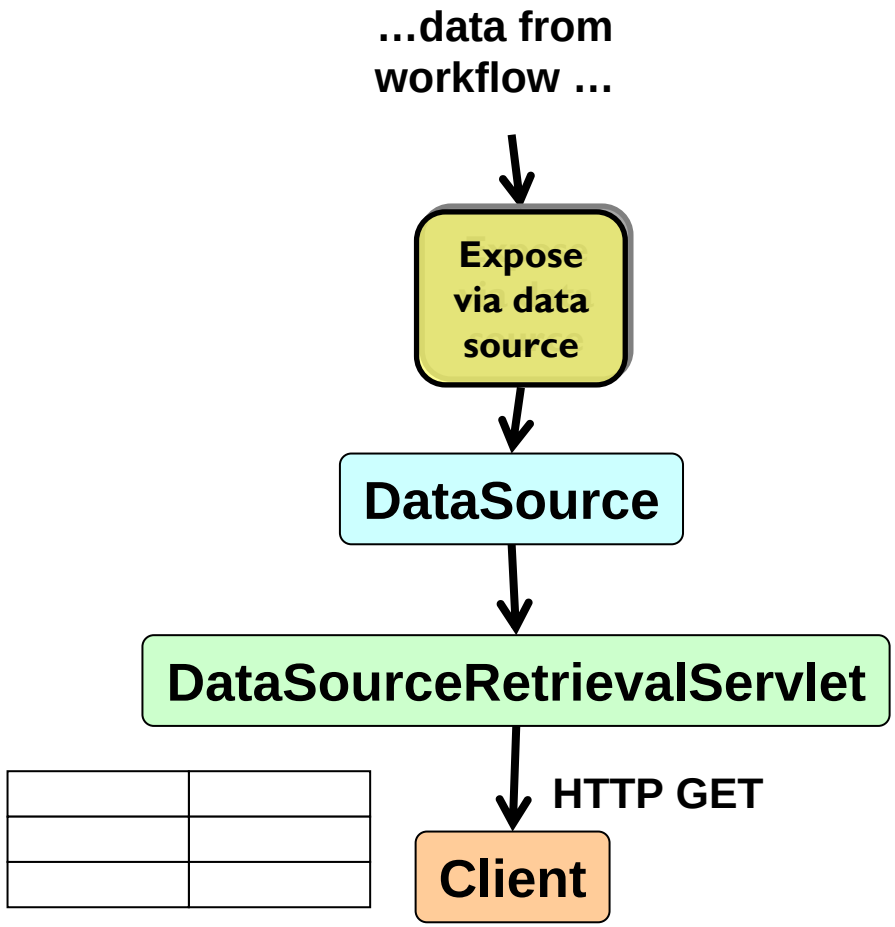
# OGSA-DAI 3.0 data sources

- OGSA-DAI data sources
  - Resource for asynchronous data delivery
- Data source service
  - Web service
  - Invoke GetFully via SOAP/HTTP
  - Use WS-Addressing to specify data source ID



# OGSA-DAI servlet

- Data source servlet
  - Invoke HTTP GET
  - Use URL query string to specify data source ID



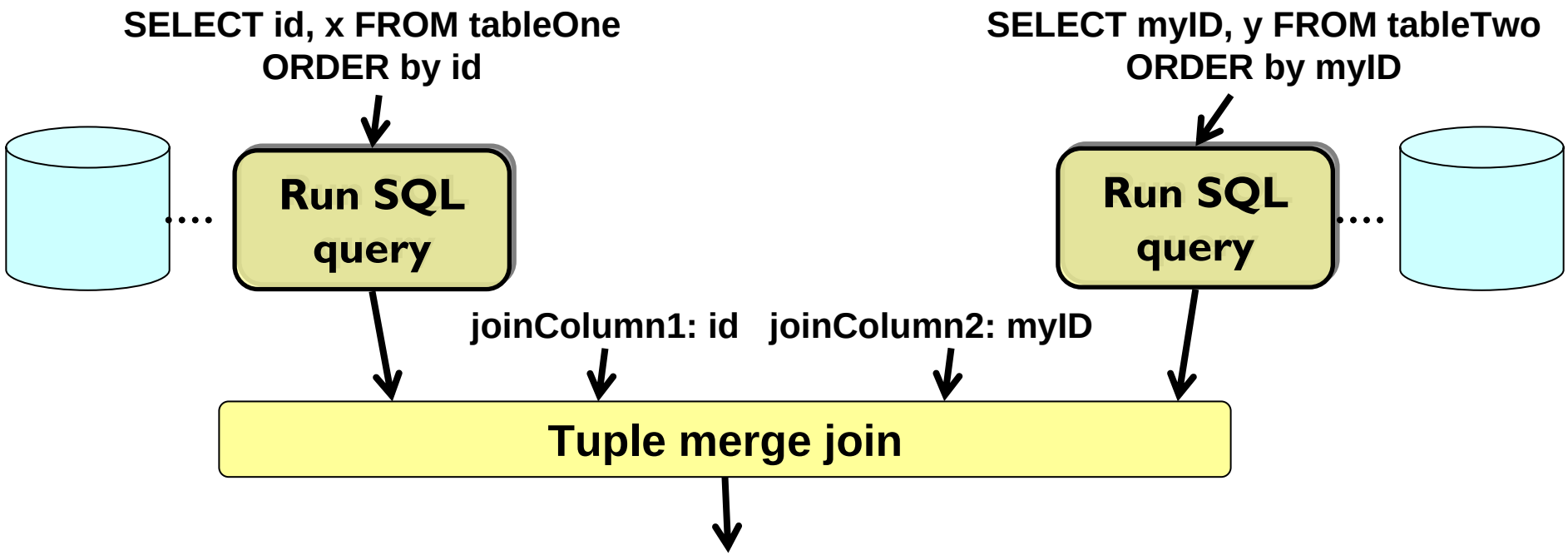
# OGSA-DAI servlet

- Useful for service orchestration and job submission
  - Taverna service-oriented workflow executor
  - Taverna could submit workflow to OGSA-DAI
  - OGSA-DAI returns URL
  - Taverna passes URL as part of job to job submission service
    - e.g. GRAM or GridSAM
  - Data is pulled from the URL when the job is executed
- Advantages
  - Data is only moved when needed i.e. when the job executes
  - Job execution components need no OGSA-DAI-specific components

# A join activity

- Virtual Organisations for Trials and Epidemiological Studies (VOTES)
  - UK Medical Research Council project
  - Relational databases
  - Uses OGSA-DAI
- OGSA-DAI team developed join activities

# A join activity




- This is equivalent to running:

**SELECT id, x, y FROM tableOne, tableTwo where table1.id = table2.myID;**

- Where tableOne and tableTwo are in two different databases

# SQL views

- Imagine we have Patient and Doctor tables

| ID | Name | Age | Sex | ZIP | Dr ID |
|---|---|---|---|---|---|
| 1 | Ken | 42 | M | IL1478305 | 456 |
| 2 | Josie | 25 | F | BN1 7QP | 789 |

| ID | Name | DN |
|---|---|---|
| 123 | Greene | US-Chicago-G |
| 456 | Ross | US-Chicago-R |
| 789 | Fairhead | UK-Holby-F |

- SQL CREATE VIEW command
- Define a DrPatient view to be
  - SELECT p.id, p.name, p.age, p.sex FROM Patient p, Doctor d WHERE p.DrID = d.ID;
- Client runs SELECT * FROM DrPatient;
- Shorthand for complex queries
- Data access control
  - e.g. staff with only access to the DrPatient view will be unable to access a patient's ZIP

# OGSA DAI SQL views

- Layer above the database to implement views
- Define views for databases to which you don't have write access
- Parses query
- Maps view to SQL query over actual database
- e.g if DrPatient was defined as
  - SELECT p.id, p.name, p.age, p.sex FROM Patient p, Doctor d WHERE p.DrID = d.ID AND d.dn = $DN$;
  - Can replace $DN$ by client's DN from their certificate provided using GT4 security components
  - Doctors can only view their own patients
- Factor in the client's security credentials

# OGSA-DQP

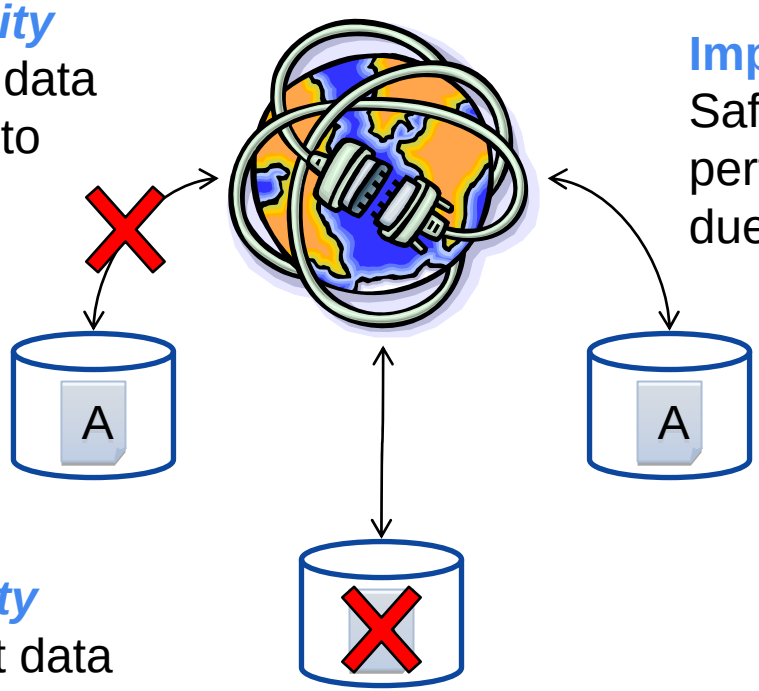
- Distributed query processing
  - Multiple tables on multiple databases are exposed to clients multiple tables in one “virtual database”
  - Client is unaware of the multiple databases
  - Databases can be exposed within one OGSA-DAI server or exposed by remote OGSA-DAI servers
- How it works
  - Query is parsed
  - Query plan is created
  - Query plan is executed – each database has sub-queries executed on it
  - Results are combined
- Good for joins and unions


# What's new in data replication and placement services?

Rob Schuler

# Objectives for Data Replication

**Improve *Availability***
Safeguard against data inaccessibility due to network partition

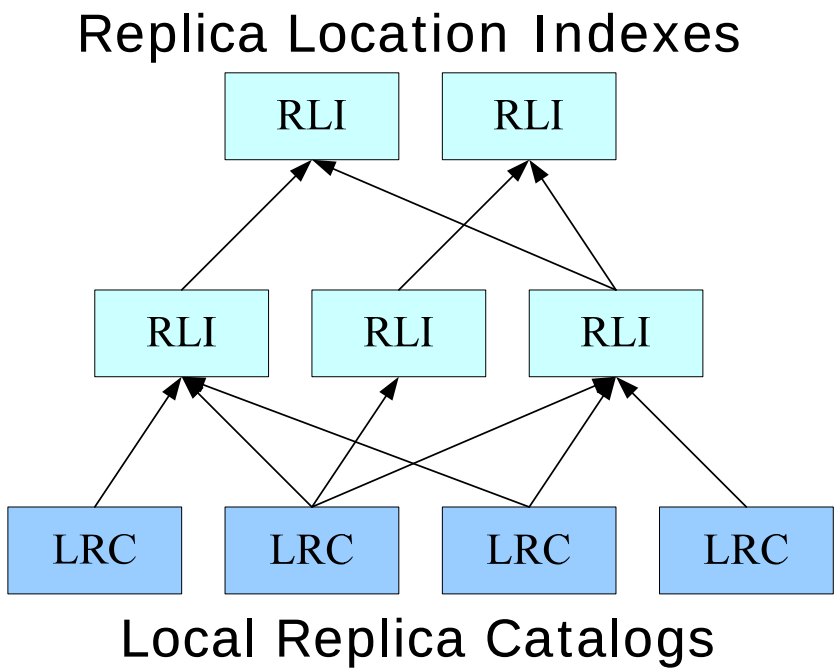



**Improve *Performance***
Safeguard against performance bottlenecks due to resource overload

**Improve *Durability***
Safeguard against data loss due to disk failure

# The Globus Replica Location Service

- Distributed registry
- Records the locations of data copies
- Allows replica discovery
- RLS maintains mappings between logical identifiers and target names
- Must perform and scale well:
  - support hundreds of millions of objects
  - hundreds of clients
- Mature and stable component of the Globus Toolkit



# New Features in RLS

- Embedded SQLite database for easier RLS deployment
  - Open source relational database backends (MySQL, PostgreSQL) depend on ODBC libraries
  - Compatibility problems that have made DB deployment difficult
  - Embedded DB back end now allows easy installation of RLS
    - Allows easier evaluation of RLS by potential users
  - SQLite offers good performance and scalability on queries
  - Does not support multiple simultaneous writers, so not suitable for some high performance environments

# New Features in RLS

- Pure Java client implementation
  - Long-awaited
  - Overcomes problems with JNI-based client, particularly on 64-bit platforms
  - Improves reliability of portals that use RLS Java client
  - Being used by several large applications (ESG, SCEC)
- WS-RLS interface: provides a WS-RF compatible web services interface to RLS
  - Easier integration of RLS services into GT4 Web service environments

# Data Placement Services: Motivation

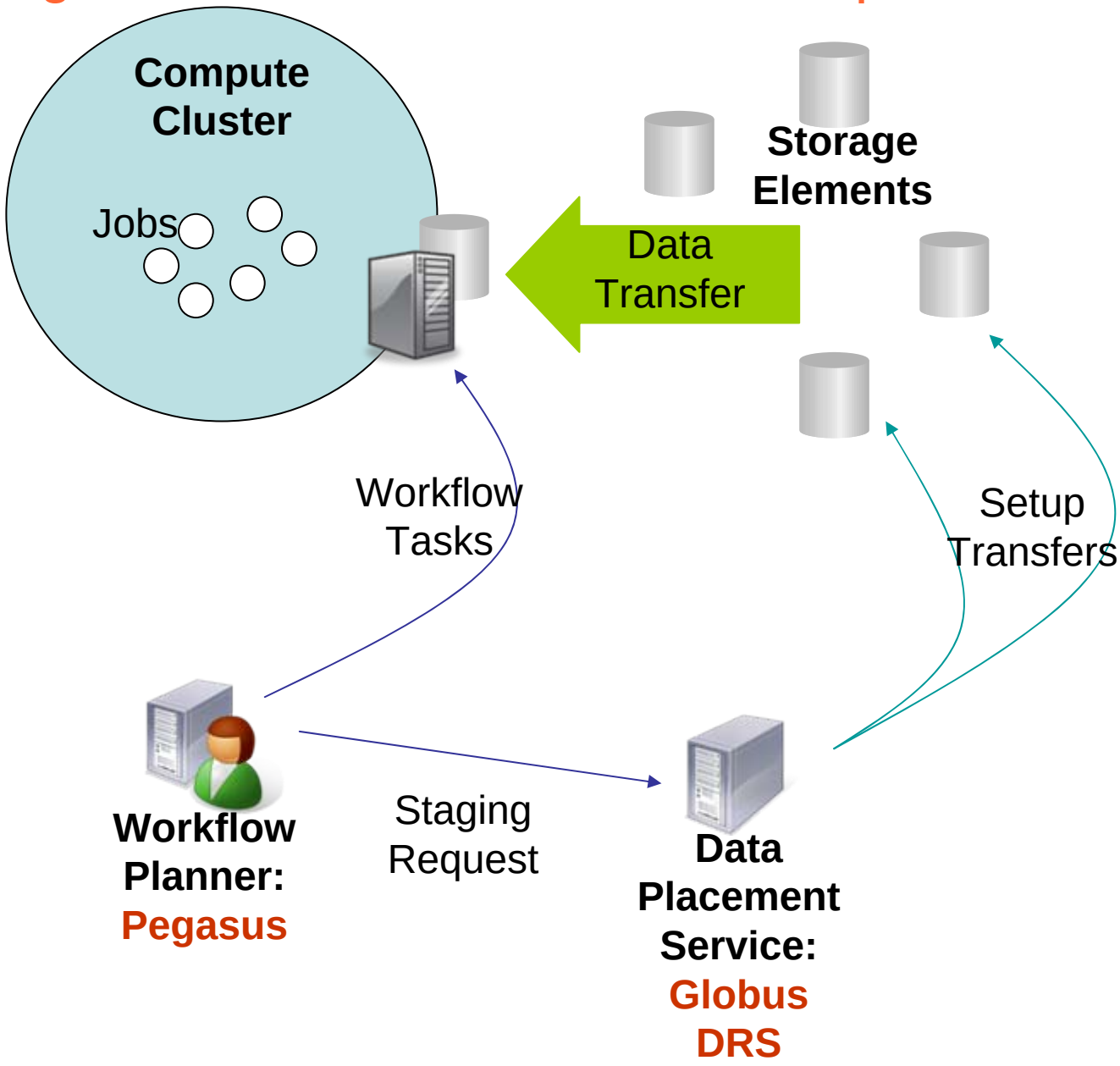
- Scientific applications often perform complex computational analyses that consume and produce large data sets
  - Computational and storage resources distributed in the wide area
- The placement of data onto storage systems can have a significant impact on
  - performance of applications
  - reliability and availability of data sets
- We want to identify data placement policies that distribute data sets so that they can be
  - staged into or out of computations efficiently
  - replicated to improve performance and reliability

# Data Placement and Workflow Management

- Studied relationship between asynchronous data placement services and workflow management systems
  - Workflow system can provide hints r.e. grouping of files, expected order of access, dependencies, etc.
- Contrasts with many existing workflow systems
  - Explicitly stage data onto computational nodes before execution
- Some explicit data staging may still be required
- Data placement has potential to
  - Significantly reduce need for on-demand data staging
  - Improve workflow execution time
- Experimental evaluation demonstrates that good placement can significantly improve workflow execution performance

"Data Placement for Scientific Applications in Distributed Environments," Ann Chervenak, Ewa Deelman, Miron Livny, Mei-Hui Su, Rob Schuler, Shishir Bharathi, Gaurang Mehta, Karan Vahi, in Proceedings of Grid 2007 Conference, Austin, TX, September 2007.

# Approach: Combine Pegasus Workflow Management with Globus Data Replication Service



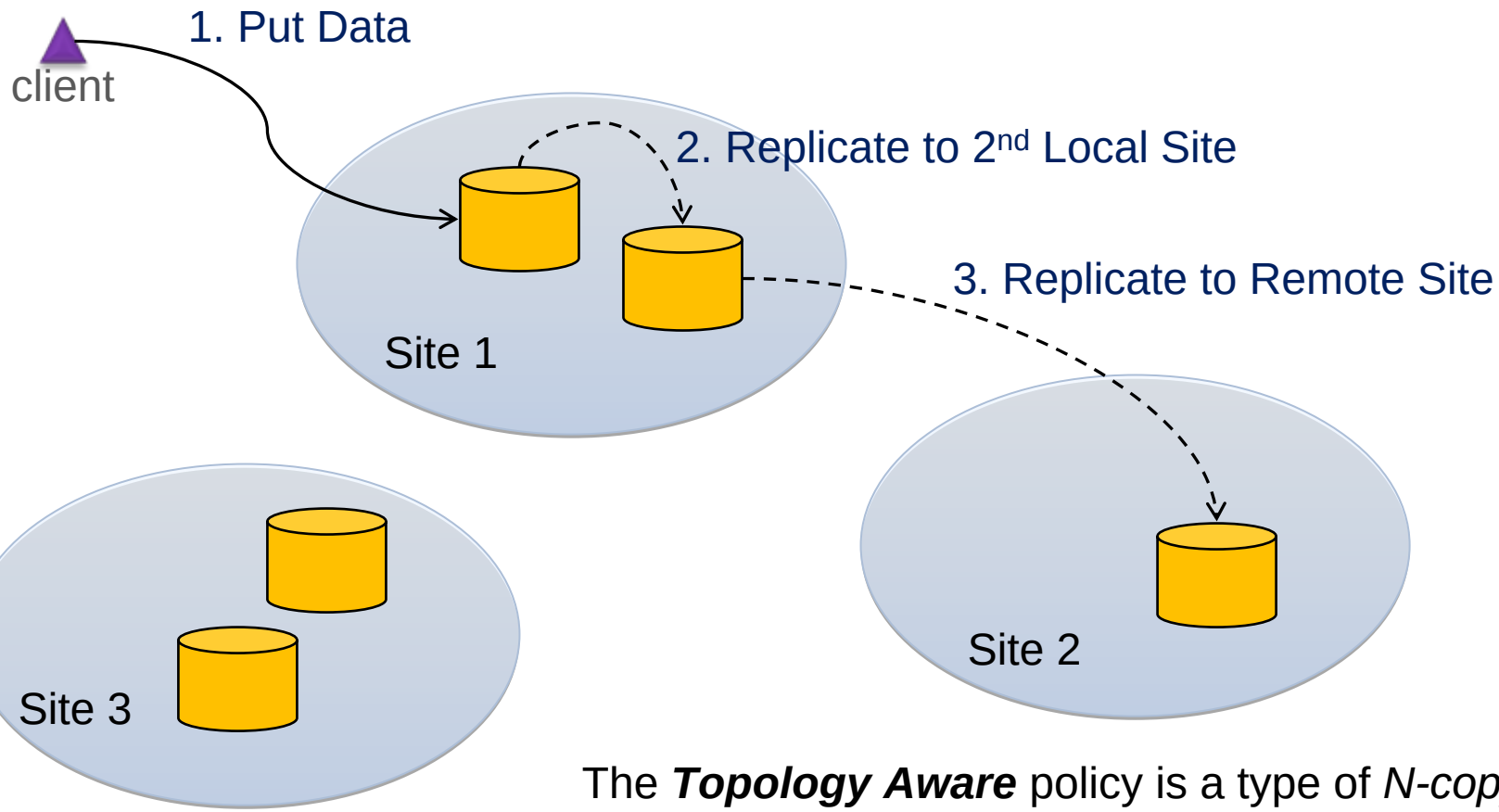
# Replication occurs when…

- Replica *Placement*
  - I want replica X at sites A, B, and C
  - I want N replicas of each file
  - I want replicas near my compute clusters
- Replica *Repair*
  - Due to *replica failure*: lost or corrupted
  - *But* it can be hard to tell the difference between permanent and temporary failure*!*

# Examples of Placement Policies

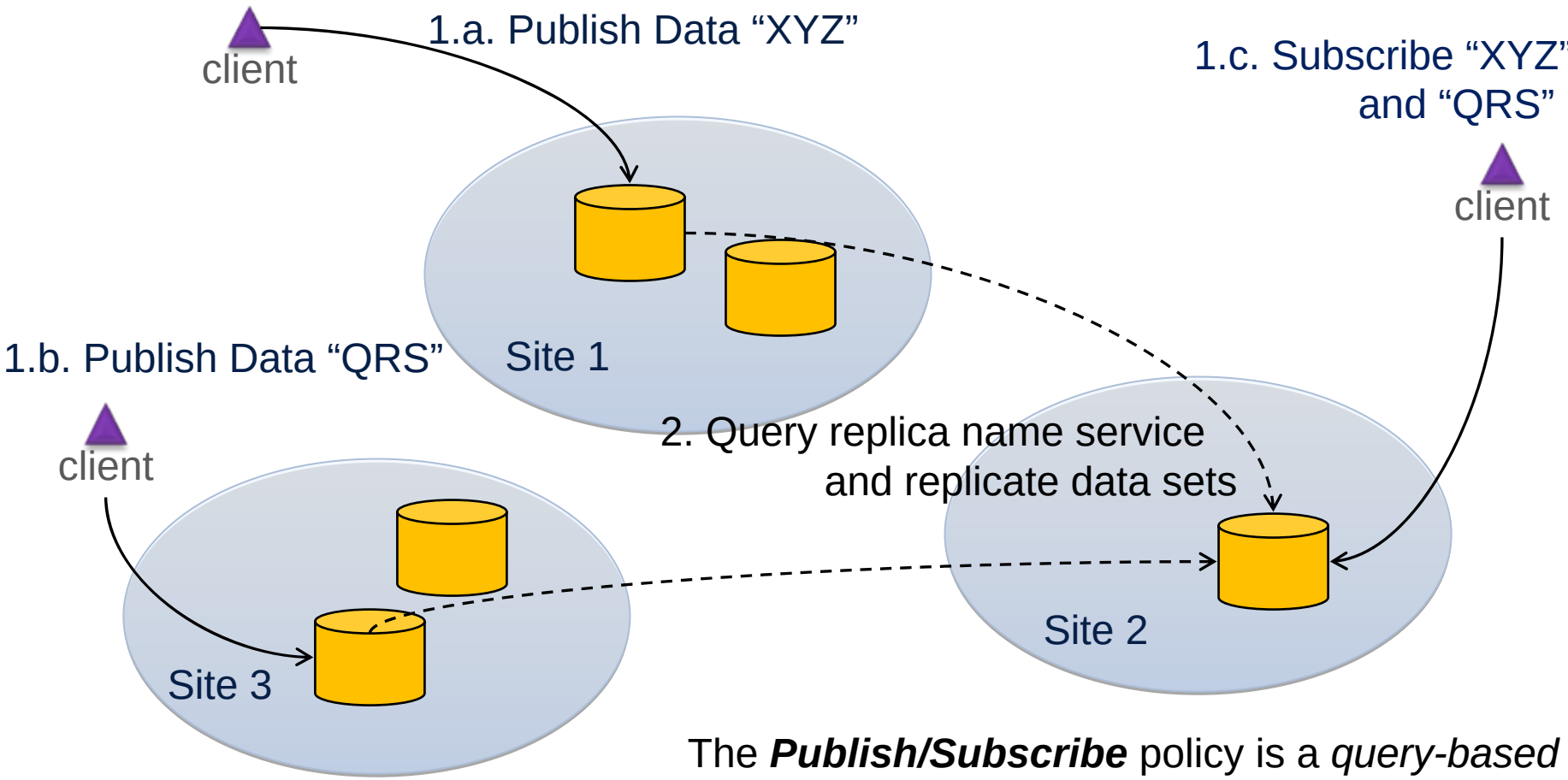
| | |
|---|---|
| Random | • Make N copies placed randomly on different sites |
| Topology-aware | • One on my server, one on the same rack, one on another rack |
| Publish/Subscribe | • Query-based replication requests to push or pull data to make new replicas |
| Tree-based dissemination | • Push replicas toward the "leaf" nodes (or access points) of the tree |
| Pervasive | • Exploit locality of reference by creating replicas at any site where they are accessed |
| QoS Aware | • Place replicas at sites in order to optimize Quality-of-Service (QoS) criteria |

# Topology-Aware Placement




The ***Topology Aware*** policy is a type of *N-copy* policy that (in this 3-copy example) ensures that replicas are distributed within and between sites

# Publish/Subscribe Placement




The ***Publish/Subscribe*** policy is a *query-based* policy that identifies desired replicas based on a query and replicates them to the desired site

# Reactive vs. Proactive Replication

- Reactive Replication
  - When a replica failure occurs, replicate
  - Difficult to tell the difference between a permanent replica failure and a temporary loss – e.g., temporary network partition
- Proactive replication
  - Continually replicate files beyond the minimum required
  - Avoid bursts of network traffic to repair failures; limit bandwidth for repairs
  - Need creation rate >= failure rate