Metadata Management in the European DataGrid Project

Gavin McCance

University of Glasgow

European DataGrid Project
GridPP Project

DataGrid is a project funded by the European Union
GridPP is funded by PPARC

Outline

- Classes of metadata in EDG
  - Grid internal metadata
  - Application specific metadata
- Products
  - Replica catalogues
  - Spitfire
- Technology details
- Future Work

Types of Metadata

- Two types of metadata used in EDG WP2
- Grid internal metadata
  - Metadata on files (size, checksum, etc)
  - Metadata on logical names (application specific)
- Application specific general metadata
  - Not related on logical filenames
  - Bookkeeping databases
  - Data Catalogues
  - Image metadata
  - etc

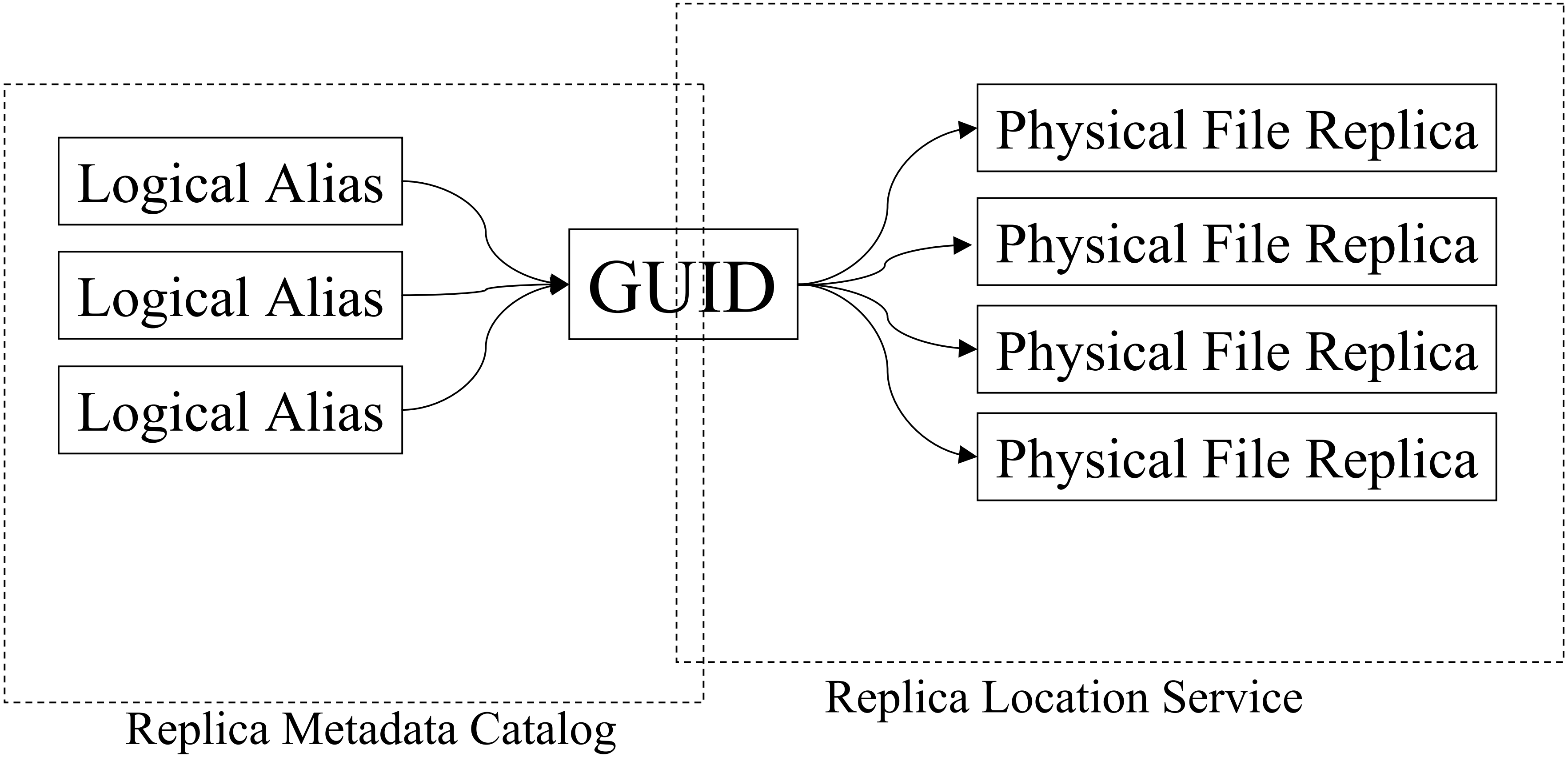
Grid Internal Replication Metadata

Replica Location Problem

- Given a logical file identifier – how do we find all the replicas of that file on the Grid
- Driven by two use-cases:
  - a) Particle physics – multiple replica of the same file so that the data are always near the compute resources - for data hungry applications
  - b) Earth Observation/Medical – convenient mechanism for logical namespace. Don’t need to know the physical location of the files.

Replica Metadata

- Logical filename to storage (physical) filename mapping

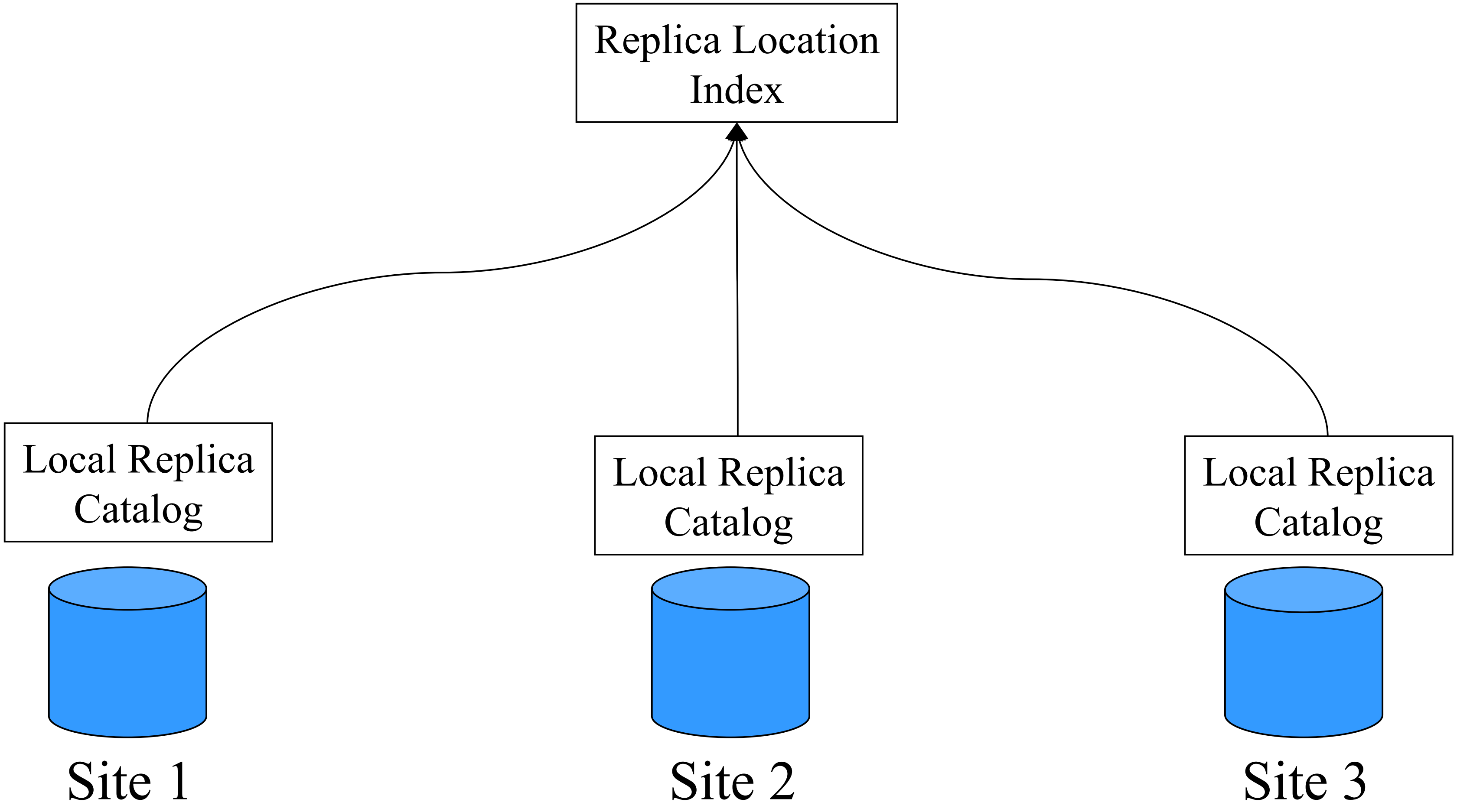


Replica Location Service (RLS)

- Optimised to answer 2 very specific queries:
  *"for a given GUID, give me **all** the replicas"*
  *"for a given GUID give me **all locally available** replicas"*

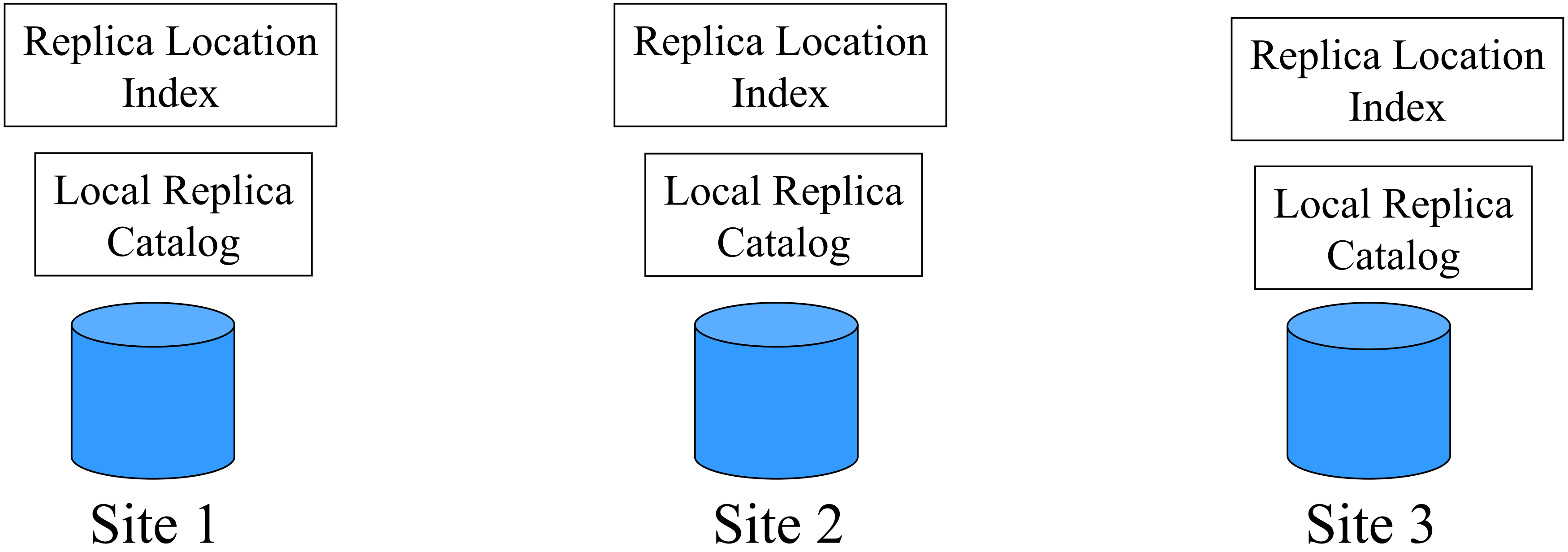
- Scalability achieved by:
  - Each site has a Local Replica Catalog LRC containing mappings for files located at the given site
  - Each site runs a Replica Location Index RLI which contains a bloom-filter hashmap for all GUIDs in all LRCs

Architecture…



Architecture…

- Each LRC updates the RLI on every other site.



Sequence to answer the query

- *for a given GUID, give me **all locally available** replicas*
  - simply contact the Local Replica Catalog.

- *for a given GUID, give me **all** the replicas*
  - contact Replica Location Index to retrieve all LRCs potentially having a mapping for the given GUID:

    GUID → List of LRCs

  - contact each LRC in the list to retrieve all replicas

Bloom Filter Indexing

- Advantages:
  - High level of scalability
  - Fast
  - Not a memory intensive hash

- Disadvantages:
  - Only fulfills “EQUALTY” type queries, i.e. no wildcards
  - Non-deterministic, i.e. there are a small number of false positives to be dealt with

Replica Metadata Catalog (RMC)

- Stores GUID metadata:
  - logical file names (human readable)
  - small number of user-defined attributes ~O(10)

- Attributes are natively typed:
  - string, float, int, date

RMC

- Used to do GUID selection based on application-specific metadata
  - Subsequently use the RLS to find the physical replica based on the GUID

- Currently a centralised catalog
  - though work ongoing with Oracle Streams for replicated architectures
  - Work on clustering and replication for high availability solutions

Application Specific General Metadata

Spitfire: Technology Demonstrator

- **Capabilities**:
  - Simple Grid-enabled front-end for any remote RDBMS through secure Web Services (SOAP-RPC)
  - Provides sample generic RDBMS methods that may easily be customized with little additional development
  - WSDL interfaces
  - Web Browser integration (data browser servlet)
  - GSI authentication
  - Local authorization module
  - Not suitable for the retrieval of LARGE result sets
- **Status**: current version 2.1
  - Used by EU DataGrid Earth Observation and Biomedical applications.

Spitfire Sample API

- Spitfire Sample API based upon common SQL operations. Use the Spitfire Grid service where you might have used JDBC before.
- Provides DB query operations, update operations, and schema update operations.
- Provides browser servlet to expose specific views of the data to web based clients.

Technology details

- All services implemented as secure web services
- WSDL exposed allowing auto-client generation
  - Supplied clients: Java, C++
  - Others have successfully used perl, python clients using our WSDL
- SSL secure authentication using Grid Proxy certificates (GSI, but NOT **httpg**)
- ‘Medium-grained’ authorization including web-based administration tool:
  - ‘medium-grained’ meaning each method can be allowed/denied based on patterns of distinguished names, VOMS capabilities.
  - can interpret grid-map files
  - can interpret VOMS credentials and capabilities contained therein

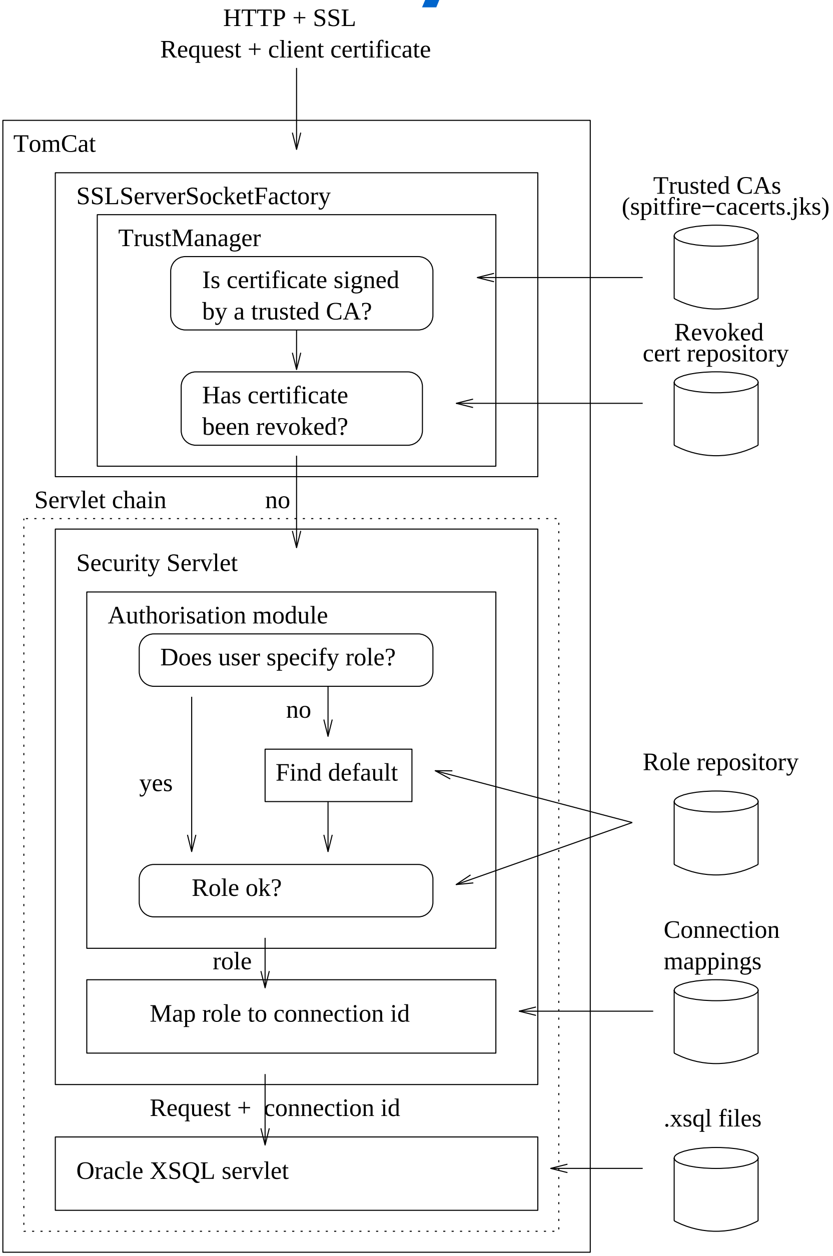
Deployment

- Tested and deployed on
  - Tomcat/MySQL,
  - Tomcat/Oracle9i
  - Oracle9iAS/Oracle 9i.
- Testing ongoing for Tomcat/DB2.

Future Work

- Plan to work together with DAIS working group of GGF to ensure that our services can be re-factored into DAIS-compliant services.
  - Should be fairly easy since we are starting from web services.
- Plan to work more closely with applications in order to refine the metadata interface, or just to enable their existing metadata applications to be ‘on the Grid’.

Security modules



- Authentication using standard GSI certs or proxies
  - Trustmanager checks validity and revocation
- Role based Authorisation
  - Specific and default roles