

The Data Grid: Towards an Architecture for Distributed Management and Analysis of Large Scientific Datasets

Ethar Elsaka

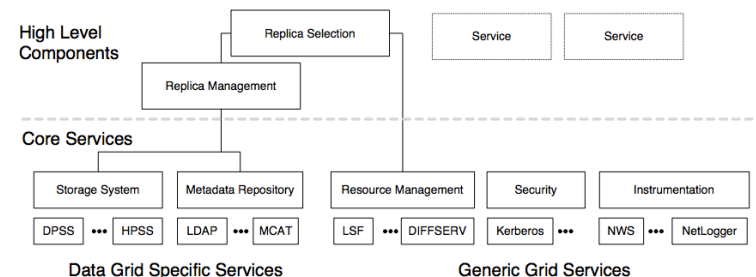
Introduction

- Large data collections appear in many scientific domains like climate studies.
- Users and resources are distributed geographically.
- Numerous solutions to individual issues exist but no overarching architecture.
- Solution: Data Grid
- Specialization and extension of Grid paradigm

Data Grid Design

- Mechanism neutrality
 - DG architecture to be independent of lower-level storage/access systems. Encapsulation of local peculiarities.
- Policy neutrality
 - Expose design decisions with performance implications to the user; ie let user set priorities.
- Compatibility with Grid infrastructure
 - Exploit Grid services like authentication and resource management.
- Uniformity of information infrastructure
 - Use the same data model and interface as required to handle the underlying Grid infrastructure.

Data Grid Design Cont.



Core Data Grid Services

- Data Access
 - It provides mechanisms for accessing, managing, and transferring data stored in storage systems
- Metadata Access
 - It provides mechanisms for accessing and managing information about stored data.
- Separating data and metadata maximizes flexibility.

Storage System and Data Access

- Storage System
 - It provides function for creating, destroying, reading, writing, and manipulating file instances
 - Unit of info: file instance consists of a named, uninterrupted sequence of bytes.
 - File instance can reside in a file system, database, or other storage systems like (SRB).
 - Storage system associates a set of properties with each file instance.

Storage System and Data Access Cont.

- Data Access
 - It is an API that describes the possible operations that can be done on storage systems and file instances.
 - API main operations
 - Remote requests to read/write named file instances.
 - Determining file instance attributes.
 - Supporting optimized implementation of replica.
 - Data grid complications
 - Heterogeneous security environments.
 - Reservation capabilities on storage systems and environments when increased performance is required.
 - Performance monitoring and self-optimization.
 - Error detection and reporting.

The Metadata Service

- It manages the information of the grid, file instances, the contents of file instances, and various storage systems contained in data grid.
- Metadata Types
 - Application metadata
 - Replica metadata
 - System configuration metadata
- Application queries sent to metadata service. Repository consists of references to logical files, which can be mapped to actual replicas.

The Metadata service Cont.

- Metadata complications
 - Difficult to select uniform representation and interface: numerous existing metadata representations reflecting different needs and philosophies (XML, indexing data structures, etc).
 - Large scale metadata issues: scalability, heterogeneity, distributed environment, ownership and local control over data access, robustness in the face of partial failures.
 - Hierarchical and distributed system.
 - Similar systems: LDAP, existing Grid metadata systems.

Other Basic Services

- Authorization and authentication
- Resource reservation and co-allocation
- Performance measurements
- Instrumentation services

Higher Level Data Grid Components

- Replica Management
 - It is responsible of creating, deleting copies of file instances, or replica, within specified storage systems.
 - A replica is created because of achieving better performance.
 - It maintains a repository or catalog.
 - It does not determine when or where replicas are created.
- Replica Selection
 - It is responsible of choosing the replica that will optimize the performance.
 - Grid information service and metadata repository provide it with information about the network and file instances

LDAP Implementation

- Used LDAP to construct catalogs.
- Catalog: tree structure known as Directory Information Tree (DIT).
- Two applications: Climate modeling and data visualizations.
- Climate Modeling:
 - DIT: root node, node for each collection, node for each logical file per collection.
 - Metadata: XML. Data accessed via URLs.
 - Prototype: user manually choose replica.
 - Ability to map a query to the URL using LDAP and move the data using general purpose transfer API

LDAP Implementation Cont.

- Data visualization application
 - Locate files for distance visualization applications.
 - Each file: a timestep in a series
 - Can be of different resolutions.
 - Can be of different data layouts.
 - Several thousand files.
 - Each file has multiple replicas listed in catalog.
 - Their implementation scales poorly: too many objects in replica catalog.
 - Solution: organize logical files into collections.

Lesson Learned

- keep metadata and replica management separate.
 - Replica and Metadata Catalogs should be distinct.
 - Data contents info should be stored in Metadata Catalog.
 - Data location info/mapping should be stored in Replica Catalog.

Data-Intensive Application

- Scientific Applications
Requires transfer, publish, replicate, discover, share, and analyze data.
- Business Applications
Maintain database consistency, manage data replication, facilitate data discovery, and respond dynamically to changes in the load applied to database by users.

Structured Data

- The data whose structure is explicitly defined in a way that can be exploited by operations.
- Data structured using
 - Data relational model
 - XML documents
- Structured data categories
 - Primary structured data
 - Ancillary data
 - Collaboration data
 - Personal data
 - Service data

Data Management Challenges

- Applications requirements
 - Diverse usage scenarios
 - Heterogeneity at all system levels
 - Performance demand associated with access, manipulation, and analysis of large quantities of data.

Data Management Challenges Cont.

- Data Sources
 - Any facility that may accept or provide data (e.g. database and file systems).
 - Data source may be temporal or permanent.
 - Grid contains diverse data sources with different storage systems, data types, data models, and access mechanisms.

Data Management Challenges Cont.

- Data Discovery, Movement and Replication
 - The ability to discover desired data based on meta data attributes rather than knowledge of name or location.
 - The ability to move the data between storage systems or between programs and data storage.
 - The ability to create replicas to reduce the access latency, maintain local control over necessary data, or improve reliability and load balancing.

Data Management Challenges Cont.

- Data Analysis and Processing
 - Data analysis requires a series of data processing and analysis.
 - These subsequent execution of operations requires planning, scheduling, and monitoring challenges.
 - Once the execution plan is designed, one must map the execution of the operations onto available resources.

Architectural Approaches

- The design of an infrastructure that supports the data-intensive applications various from monolithic integrated architecture to layered protocol architecture.
- The data-oriented services:
 - Resource-level services for data sources.
 - High performance data access and update.
 - Collective services for managing data.
 - Managing data across different data sources like data transfer and data discovery.
 - Collective services that federate data sources
 - Combining data from different data sources
 - Domain-specific services.
 - Specialized data processing and analysis operations

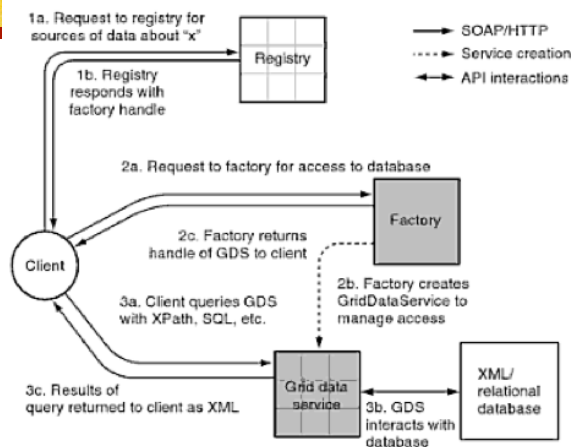
Data Sources Services

■ Data Access

Responsible of the interfaces provided to the user to access the data on the devices and performance characteristics.

- GridFTP as a File Access Service
 - Designed as a fundamental data access and data transport service
 - Third party transfers
 - Parallel and stripped transfer
 - Partial file transfer
 - Fault tolerance and restart
- Data Access and Integration (enterprise applications)
 - Standard interface
 - Establish a session
 - Submit a series of queries
 - Each submission receives a response
 - OGSA DAI components are suitable for Grid environment.

OGSA Architecture



Data Sources Services

■ Managing Data Sources

Responsible of managing the storage systems holding the data collections being accessed.

- NeSt
 - Software-only grid storage appliance.
- **Lots** provides a specific capacity to a particular user or group
- **Lots** logically groups sets of files to facilitate sharing, discovery, and removal.
- **Lots** can be either Best-effort or guaranteed.

Data Sources Services Cont.

- NeST main components:
 - Storage manager
 - Transfer manager
 - Dispatcher
 - Protocol layer
- Storage Resource Manager
 - It manages associated storage via a range of techniques includes cache management, preallocation and advance reservation of space
 - It manages two types of shared resources: files and space

Collective Data Management Services

- Data Transport Services
 - Responsible of managing the transport of data between data storage services.
 - Multiple data object transfer services or reliable data transfer services.
- General Data Discovery Services
 - Responsible of identifying data items that match specified characteristics or metadata.
 - MCAT & MCS provide a general metadata schema.
- Workflow Management, Planning, and Scheduling
 - Workflow manager is responsible of initiating the execution of individual tasks in the specified order.

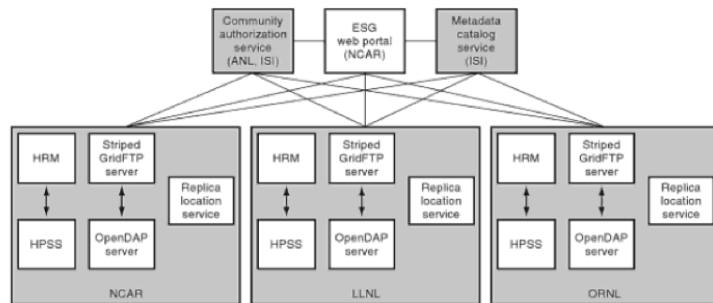
Federation Services

- Many databases contribute data and resources to a multi-database federation, but each participant has full local autonomy.
- Loosely coupled federated database and tightly coupled federated database
- The goal is to provide access to sophisticated analysis and simulation computations using integrated data from multiple resources.

Federation Services Cont.

- Data Mediation Services
 - They provide transparency with respect to data models
- Replication Services for Location Transparency
 - They provide for replication transparency.
 - Types
 - Replica management services
 - Replica location services
 - Consistency services

Case Study: The Earth System Grid



Thank You