Service-oriented Architectures

B. Ramamurthy

Introduction

- We looked at grid "core" services such as life cycle, notification, service data etc.
- In this discussion we will look at high level services:
  - Security
  - Data Management
  - Resource Management
  - Information Services
- Service-oriented application models

OGSA Services

- Core services
- Data and information services:
  - Data naming and access: access and federate diverse sources
  - Replication
  - Metadata and provenance: describing and tracking how data are created and recreating steps required to regenerate data on demand.

OGSA Services (contd.)

- Management of services
  - Provisioning and resource management: required for negotiating service-level agreement (SLA) between consumers and providers, dynamic reallocation and distribution policy consistent with SLA.
  - Service orchestration: managing the choreography of interacting services.
  - Transactions
  - Administration: change management, identity management for deployment etc.

Service-oriented Arhitectures

- Service is an entity that provides some capability to its clients by exchanging messages. Operations are defined in terms of message exchanges.
- A service oriented architecture (SOA) is one in which all entities are services and any operation visible to the architecture is the result of message exchange.
- We will look into architecture and operation of SOA.

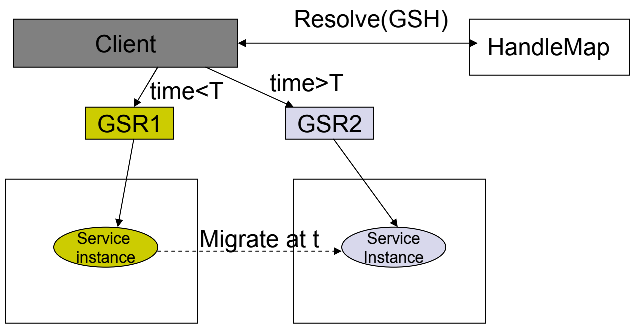
Examples of Services

- Storage service
- Data transfer
- Troubleshooting service
- Common theme is monitoring service, storage services and query services.

Virtualization

- Encapsulating service operations behind a common message-oriented service interface is called service virtualization.
- Isolates users from details of service implementation and location.
- Assumes support of a standard architecture.
- Webservices (WS) can do this, however grid life cycle management, fault handling and other features we have seen in the GT3 tutorial are not available with WS.
- OGSI specification addresses these issues using a core set of standard services.

GSH, GSR and HandleMap



Security

- Grid Security Infrastructure (GSI)
- Features:
  - Single sign-on
  - Mutual authentication
  - Secure communication
- Based on:
  - Public key encryption
  - Secure shell layer (SSL)
  - X.509 certificate
- Community Authorization Service (CAS) is new from GT3.2 (we have GT3.0.2 on CSE Linux grid)
  - resource providers to specify course-grained access control policies in terms of communities as a whole, delegating fine-grained access control policy management to the community itself.

Data Management

- GridFTP:
  - high-performance, secure, reliable data transfer protocol optimized for high-bandwidth wide-area networks.
- RFT (Reliable File Transfer Protocol):
  - The Reliable File Transfer Service (RFT) is an OGSA based service that provides interfaces for controling and monitoring 3rd party file transfers using GridFTP servers.
- Replica Location Service (RLS):
  - The Replica Location Service (RLS) maintains and provides access to mapping information from logical names for data items to target names.
  - These target names may represent physical locations of data items.

Information Services

- Monitoring and Discovery System (MDS):
  - is the information services component of the Globus Toolkit and
  - provides information about the available resources on the Grid and their status.
  - Has an Index service that aggregates the services available.
  - Service Data Providers
  - Tools for browsing and querying service data
- MDS4 has advanced features such as trigger service.

Resources

- GRAM does not provide accounting or billing.
- Opportunity for research:
  - Resource metering
- Resource Definition Framework (RDF)
- WSRF (Web Services Resource Framework)

Resource and Service Mangement

- Central theme of grid is the ability to discover, allocate and negotiate use of network-accessible capabilities.
- Resource management describes all aspects of the processes: locating a capability, arranging for its use, utilizing it, and monitoring its state.
- Resources refers to both traditional resources and virtualized services.
- We could use “resource management” and “service management” interchangeably.

Issues

- Managed resources span administrative domains.
- Heterogeneous configurations.
- Different access policies
- Establish a mutual agreement between a resource provider and a resource consumer.
- Provider of the resource agrees to supply a capability that can be used to perform some task on behalf of the consumer.

Requirements

- Task submission
- Workload management
- On-demand access
- Co-scheduling
- Resource brokering
- Quality of service
- Provisioning capabilities; Ex: database service can be provisioned to service 10 request at a time; file space can be provisioned to allow submission of jobs requiring up to 200 processors.
- Service level agreements

Resource Management Framework

- Resource management operations: submit, acquire, bind are applicable to any resource.
- Available resources are discovered
- Capabilities are obtained by acquire operation
- Resources are utilized by associating submitted tasks with acquired resources using bind operations.

Service Level Agreements

- A resource consumer needs to affect the resource behavior, often requiring guarantees concerning the level and type of service being provided by the resource.
- Conversely, provider of resource may want maintain control over how the resource is used.
- A common means of reconciling these two demands is to negotiate a Service Level Agreement (SLA).
- SLA is a means by which resource provider contracts with a client to provide some measurable capability or perform a task.
- Task SLA
- Resource SLA
- Binding SLA

Policy and Security Resource Agreement

- Resource policy controls by whom and how its resources may be used.
- This policy will govern SLAs to which resource provider is willing to agree.
- Provider will publish only policies necessary to enable discovery and later use other private policies during negotiations.
- Resource brokers are often used to carry out the negotiations.

Task Management

- Resources are selected and action initiated.
- Then task monitoring during execution: monitoring SLAs is specially important.
  - Terminate an SLA
  - Extend an SLA's lifetime
  - Renegotiate an SLA terms
  - Create a new SLA

Resource discovery and selection

- RDF (Resource Definition Framework) is used for establishing SLAs.
- Description used for purposes of discovery is different from description used for SLAs.
- Resource selection is the process of choosing from a set of candidates provided by resource discovery.
- My opinion: this can be unified.

Resource Management

- Grid Resource Allocation and Management Service (GRAM):
  - provides a single interface for requesting and using remote system resources for the execution of "jobs".
  - The most common use of GRAM is remote job submission and control.
  - It is designed to provide a uniform, flexible interface to job scheduling systems.

Grid Resource Management Systems

- GRAM a first generation tool
- GRAM defines resource-layer protocols and APIs that enable clients to securely instantiate a computational task.
- GRAM itself does not implement any resource management functionality but instead relies on local resource management interfaces to provide the functions.
- Wrapper on local resource management system.
- Opportunities: Resource management in grid

General-purpose Architecture for Reservation and Allocation (GARA)

- GARA generalizes GRAM to provide for advanced reservations and end-to-end management of quality of service on different types of resources.
- Generalizes GRAM's API and protocol to provide more generic resource management.
- An interesting example, reserve and provide access to network bandwidth. (bandwidth broker, slot manager..)
- Opportunity: This is a evolving area.

Future Directions

- Tools: new tools are needed to facilitate integration of grids into different application environment
- Implementation: Lightweight implementations and effective sandboxing
- Semantics: development of mechanisms for analyzing and reasoning about the behavior of service compositions.
- Scalability: Need technologies that can scale to increasingly complex communities and interactions including service economies.