Grid Computing:
Standards and Architecture

Martin F. Maldonado, Ph.D.
Technical Architect
IBM Grid Computing, Americas
mfmald@us.ibm.com
Contents

- On Demand Business and Grid Computing
- Grid Standards
- Open Grid Services Architecture
- Grid Services
- Data Access and Integration Services
- Globus Project and Toolkit
- Autonomic Computing
- Additional Information
Computing Evolution

Mainframe
“The Glass House”

Client-Server
“PCs / LANS”

Network-Centric
“The Internet”

On Demand
“Dynamic, Responsive, Integrated”

e-business
On Demand Operating Environment Attributes

Open
Integrated

Virtualized
Autonomic

...an approachable, adaptive, integrated and reliable infrastructure delivering on demand services for on demand business operations ...
Virtualized Storage Applications Processing Operating System Data I/O Distributed Computing Over a Network, Using Open Standards to Enable Heterogeneous Operations
What is a Grid?

• There are three key criteria:
  – Coordinates resources that are not subject to centralized control …
  – Using standard, open, general-purpose protocols and interfaces …
  – to deliver non-trivial qualities of service.

• What is not a Grid?
  – A cluster, a network attached storage device, a scientific instrument, a network, etc.
  – Each is an important component of a Grid, but by itself does not constitute a Grid
  – The web is not (yet) a Grid; its open, general-purpose protocols support access to distributed resource but not the coordinated use of those resources to deliver interesting qualities of service

What is the Grid? A three point checklist, Ian Foster, GRIDToday, July 22, 2002, Vol 1 No. 6
Grid Standards
The Value of Open Standards

Networking:
The Internet (TCP/IP)

Communications:
e-mail (pop3, SMTP, Mime)

Information:
World-wide Web (html, http, j2ee, xml)

Operating System:
Linux

Applications:
Web Services
(SOAP, WSDL, UDDI)

Distributed Computing:
Grid
(Globus -> OGSA)
Open Grid Services Architecture (OGSA)

“The TCP/IP of Grid Computing”
Global Grid Forum

- A community-initiated forum of 5000+ individual researchers and practitioners working on distributed computing, or "grid" technologies.
- Formed in 2001 by a Merger of Grid Organizations
  - European eGrid
  - US Grid Forum
  - Asia Pacific Grid Community
- Primary objective is to promote and support the development, deployment, and implementation of Grid technologies and applications via the creation and documentation of "best practices" - technical specifications, user experiences, and implementation guidelines.
- Participants come from over 400 organizations in over 50 countries, with financial and in-kind support coming from sponsor members including technology producers and consumers, as well as academic and federal research institutions.
- Modeled After IETF and IRTF
  - Meets Three Time Per Year
  - Areas, Working Group and Research Groups
  - Consensus Based
  - Open Membership, Most Work Done on Mailing Lists
- IBM is a Platinum Sponsor Member
  - Member of Steering Committee
  - Member of External Advisory Committee
  - Area Directors
  - Working Group Chairs

Source: www.ggf.org
### GGF Sponsors

<table>
<thead>
<tr>
<th><strong>Charter Sponsor Members</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Argonne National Laboratory</td>
</tr>
<tr>
<td>NASA Information Power Grid</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>2002 Platinum Sponsor Members</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Compaq</td>
</tr>
<tr>
<td>Hewlett Packard</td>
</tr>
<tr>
<td>IBM</td>
</tr>
<tr>
<td>Microsoft</td>
</tr>
<tr>
<td>Platform Computing</td>
</tr>
<tr>
<td>Qwest Communications</td>
</tr>
<tr>
<td>Sun Microsystems</td>
</tr>
<tr>
<td>SGI</td>
</tr>
<tr>
<td>US Department of Energy (DOE), Office of Scientific Computing Research</td>
</tr>
<tr>
<td>US National Science Foundation, Division for Advanced Computational Infrastructure and Research (NSF-ACIR)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>2002 Gold Sponsor Members</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Level 3 Communications</td>
</tr>
<tr>
<td>Intel</td>
</tr>
<tr>
<td>National Computational Science Alliance (NCSA)</td>
</tr>
<tr>
<td>San Diego Supercomputer Center (SDSC)</td>
</tr>
<tr>
<td>National Institute of Advanced Industrial Science and Technology, Japan (AIST)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>2002 Silver Sponsor Members</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Avaki</td>
</tr>
<tr>
<td>Entropia</td>
</tr>
<tr>
<td>Fujitsu America</td>
</tr>
<tr>
<td>Hitachi</td>
</tr>
<tr>
<td>InSORS Integrated Communications</td>
</tr>
<tr>
<td>Johnson &amp;Johnson</td>
</tr>
<tr>
<td>United Devices</td>
</tr>
<tr>
<td>University of Virginia</td>
</tr>
</tbody>
</table>
# 42 GGF Groups as of January 2003

**Applications and Programming Environments**
- Grid Checkpoint/Recovery
- Advanced Programming Models (APM-RG)
- Grid Computing Environments (GCE-RG)
- Life Sciences Grid RG
- Advanced Collaborative Environments (ACE-RG)
- Applications and Test Beds (APPS-RG)
- Grid User Services (GUS-RG)

**Architecture**
- Open Grid Services Infrastructure (OGSI-WG)
- New Productivity Initiative (NPI-WG)
- Accounting Models (ACCT-RG)
- Service Management Frameworks (JINI-RG)
- Open Source Software (OSS-WG)
- Open Grid Services Architecture (OGSA-WG)
- Grid Protocol Architecture (GPA-RG)
- Production Grid Management RG

**Data**
- GridFTP-WG
- Data Replication (REPL-RG)
- Grid High-Performance Networking (GHPN-RG)
- Data Access and Integration Services (DAIS-WG)
- Persistent Archives (PA-RG)
- Data Transport (DT-WG)

**Information Systems and Performance**
- Discovery and Monitoring Event Description (DAMED-WG)
- Grid Information Retrieval (GIR-WG)
- Relational Grid Information Services (RGIS-RG)
- Semantic Grid RG
- Network Measurement (NM-WG)
- CIM based Grid Schema (CGS-WG)
- Grid Benchmarking (GB-RG)

**Peer-to-Peer**
- Appliance Aggregation
- OGSA-P2P-Security

**Scheduling and Resource Management**
- Scheduling Attributes (SA-WG)
- Distributed Resource Management Application API (DRMAA-WG)
- OGSA Resource Usage Service (RUS-WG)
- Usage Record (UR-WG)
- Scheduling Dictionary (SD-WG)
- Grid Resource Allocation Agreement Protocol (GRAAP-WG)
- Grid Economic Services Architecture (GESA-WG)

**Security**
- Grid Security Infrastructure (GSI-WG)
- Open Grid Service Architecture Security (OGSA-SEC-RG)
- Large Site AAA (AAA-WG)
- Grid Certificate Policy (GCP-WG)
- CA Ops (CAO-WG)

*Source: www.ggf.org*
IBM Active Industry Participation in GGF

- APE
  - Boeing

- ARCH
  - Avaki, Fujitsu, IBM, Platform, Sun (JINI only)

- DATA
  - Avaki, IBM

- GIS-PERF
  - Platform, IBM

- SCHED
  - IBM, Intel, Sun

- GS
  - IBM, Verisign
Open Grid Services Architecture
Open Grid Services Architecture Objectives

- Distributed Resource Management across heterogeneous platforms
- Seamless QoS delivery
- Common Base for Autonomic Management Solutions
- Common infrastructure building blocks to avoid "stovepipe solution towers"
- Open and Published Interfaces
- Industry-standard integration technologies
  - web services, soap, xml...
- Seamless integration with existing IT resources
  - Separate interface from implementation
Distributed Computing: A Common Problem

- Web services, Autonomic computing, and Grid efforts all try to address aspects of distributed computing:
  - Defining an open distributed computing paradigm.
  - Dealing with heterogeneous platforms, protocols, and applications.
- **GRID** has focused on Scientific / Technical Computing across organizational boundaries
  - Here, secure, distributed Resource Sharing is the key
  - But no standards exist for inter-operability or pluggable components
- **Web Services** initial focus has been on application integration
  - Not resource provisioning or system integration
- **Autonomic computing** is focused on managing commercial IT infrastructures:
  - Here, sharing resources is not the issue: Managing them is!
  - Sharing functions is not the issue: Building solutions on top is!
The Best of Both Worlds

Web Services & Grid Protocols

Open Grid Services Architecture

share

manage

access

Continuous Availability

Applications on demand

Secure and universal access

Business integration

Resources on demand

Global accessibility

Vast resource scalability

Web Services

Grid Protocols

© 2003 IBM Corporation
Architecture Framework

**OGSA Structure**

- Applications
- Open Grid Services Architecture (OGSA)
- Web Services
- Autonomic Capabilities:
  - Security
  - Workflow
  - Database
  - File Systems
  - Directory
  - Messaging
- Servers
- Storage
- Network
Architecture Framework

OGSA Structure

System Management Services
OGSI – Open Grid Services Infrastructure
Web Services
Grid Services
OGSA Structure – OGSI

• Exploits existing web services properties
  – Interface abstraction (WSDL)
  – Protocol, language, hosting platform independence

• Enhancement to web services
  – State Management
  – Event Notification
  – Referenceable Handles
  – Lifecycle Management
  – Service Data Extension

OGSI – Open Grid Services Infrastructure

Web Services

Grid Services

System Management Services
Architecture Framework

OGSA Structure

System Management Services  Grid Services

OGSI – Open Grid Services Infrastructure

Web Services
Architecture Framework

Products and Services for Grids
Grid Services
OGSA Services Model

- Everything is represented by a (Grid) service
- A service is a network-enabled entity that provides some capability
- A service can be a computation resource, storage resource, network, program, database, and so on
- Services can be transient, created dynamically and destroyed when no longer needed
- Separates the definition of the interface and protocols to invoke the interface
- Simplifies virtualization - encapsulation behind a common interface of diverse implementations

Virtualization allows:

- Consistent resource access across multiple heterogeneous platforms with local and remote transparency
- Enable mapping of multiple logical resource instances onto the same physical resource
- Management of resources based on composition from lower-level resources
- Allows the composition of services to form more sophisticated services
Hosting Environment

- OGSA does not address issues of implementation programming model, programming language, implementation tools, or execution environment

- Grid services are instantiated within a specific hosting environment

- Host environment defines how a Grid service meets it obligation to Grid service semantics
  
  - rely on native operating system processes, implementing service in a variety of languages
  
  - implemented on container or component-based hosting environment such as J2EE, Websphere, .NET, and Sun One
Open Grid Infrastructure (OGSI)

Anatomy of a Grid Service

- Service Data Access
- Lifetime Management

- GridService (required)
- Other Interfaces (Optional)

- Service creation (Factory)
- Service discovery (Registry)
- Notification
- Handle Management

- Other functions e.g.
  - Workflow
  - Auditing
  - Resource Management

Hosting Environment
Open Grid Infrastructure (OGSI)

Grid Service Implementation Independence

Abstract service interface remains the same
Open Grid Infrastructure (OGSI)

Grid Service Implementation - Examples

Abstract service interface remains the same
Open Grid Infrastructure (OGSI)

Grid Service Implementation - Examples

Abstract service interface remains the same

Hosting Environment - J2EE
Other Middleware
Operating System
Hardware
File System
Database (DB2)
Storage System (NAS/SAN)
Grid Data Access and Integration
Architectural Principals

• **Heterogeneity Transparency**
  – The access mechanism should be independent of the actual implementation

• **Location Transparency**
  – An application should be able to access data irrespective of its location

• **Name Transparency**
  – An application should be able to access data without knowing its name or location
  – Data access should be via logical domains, qualified by predicates on attributes of the desired object

• **Distribution Transparency**
  – An application should be able to query and update data without being aware that it comes from a set of distributed sources

• **Replication Transparency**
  – Grid data may be replicated or cached in many places for performance and availability

• **Ownership and Costing Transparency**
  – Applications should be spared from separately negotiating for access to individual sources, whether in terms of access authorization, or in terms of access costs.

Source: Grid Database Access and Integration: Requirements and Functionalities
Principal portTypes

• GridDataService
  − Service Data Elements
    • Logical Schema
    • Physical Schema
    • StatementNotificationTypes
    • ResultFormatTypes
    • DatabaseTypes
    • SystemName
    • TransactionCapability
    • preparedStatements
    • resultCollections
  − Operation
    • perform
  − Messages
    • gridDataServiceRequest
    • gridDataServiceResponse

• GridDataTransport
  − Service Data Elements
    • LogicallySupportedTypes
    • PhysicallySupportedTypes
    • activeBlocks
  − Operations
    • perform
  − Messages
    • GridDataTransportStatement
    • GridDataTransportResponse
    • GridDataTransportFault

Source: Grid Database Service Specification
Creating and Using Grid Data Services

Source: Grid Database Service Specification
Requestor Retrieving Data from Grid Data Service

Source: Grid Database Service Specification
Requestor Using Grid Services Ports

Source: Grid Database Service Specification
Query Request with Deliver to Third Parties

Source: Grid Database Service Specification
Sending Data from one GDS to Another

Source: Grid Database Service Specification
Globus Project and Toolkit
Globus Project

- At its core, Globus is a research project. Globus research focuses not only on the issues associated with building computational grid infrastructures, but also on the problems that arise in designing and developing application that use grid services.

- Organized around four main activities.
  - Research: study basic problems in areas such as resource management, security, information services, and data management.
  - Testbed: assist in planning and building large-scale testbeds, both for our own research and for production use by scientists and engineers.
  - Software Tools: We build robust research prototype software that runs on a variety of interesting and important platforms.
  - Applications: develop large-scale grid-enabled applications in collaboration with scientists and engineers.

Source: www.globus.org
Globus Toolkit™

- The Globus Project provides software tools that make it easier to build computational grids and grid-based applications. These tools are collectively called the Globus Toolkit™.
- Is an open architecture, open source software toolkit.
- Is used by many organizations to build computational grids that support their applications.

Source: www.globus.org
Globus Toolkit™ Version 2.2 Layered Grid Architecture

“Coordinating multiple resources”: ubiquitous infrastructure services, app-specific distributed services

“Sharing single resources”: negotiating access, controlling use

“Talking to things”: communication (Internet protocols) & security

“Controlling things locally”: Access to, & control of, resources

Globus Toolkit™ Version 2.2 Key Protocols

• The Globus Toolkit™ v2 (GT2) centers around four key protocols
  – Connectivity layer:
    • Security: Grid Security Infrastructure (GSI)
  – Resource layer:
    • Resource Management: Grid Resource Allocation Management (GRAM)
    • Information: Grid Resource Information Protocol (GRIP/LDAP)
    • Data Transfer: Grid File Transfer Protocol (GridFTP)

• Also key collective layer protocols
  – Monitoring & Discovery, Replication, etc.
Globus Toolkit 2 Layered Grid Architecture

Protocols, Services, and APIs

Grid Protocols
- Applications
  - Collective
  - Resource
  - Connectivity
  - Fabric

Globus Services
- Applications utilize lower Globus services at lower levels
  - GARA, MDS (GRIS, GIIS)
  - GRAM, GRIP, GridFTP, RSL, DUROC, GASS
  - GSS-API

Globus APIs
- RSL, Composite service APIs, application level SDKs/APIs
  - GARA Client API
  - globus_gram_client, globus_rsl, globus_gram_myjob, globus_duroc_control, globus_gss_assist

© 2003 IBM Corporation
GT3 Architecture Overview

- GT3 Core
- GT3 Base Services
- GT3 Security Services
- Other Grid Services
- Workload Management Diagnostics
- File Streaming Service
- Reliable File Transfer Service
- Replica Management
- Managed Job Service
- Index Service
- Secure Conversation Service
- GridService
- NotificationSink
- Registration
- HandleResolver
- NotificationSource
- Factory
- NotificationSubscription
Autonomic Computing
Autonomic Vision

"Intelligent" open systems that...

- Hide complexity
- "Know" themselves
- Adapt to unpredictable conditions
- Continuously tune to meet performance goals
- Recover from failures
- Provide a safe environment

Providing customers with...

- Increased return on IT investment
- Improved resiliency
- Accelerated implementation of new capabilities
Autonomic Computing

**Self-Configuring**
Adapt automatically to the dynamically changing environments

**Self-Healing**
Discover, diagnose, and react to disruptions

**Self-Optimizing**
Monitor and tune resources automatically

**Self-Protecting**
Anticipate, detect, identify, and protect against attacks from anywhere
Autonomic Element
Autonomic Components in a Hierarchy
Self-Configuring Example: DB2 Configuration Advisor

DB2 Configuration Advisor Results

OLTP - 32
OLTP - 64
Cust #1
Cust #2

Performance as Percentage of DBA tuned Solution

0% 50% 100% 150% 200% 250%

OLTP - 32
OLTP - 64
Cust #1
Cust #2

- Default configuration
- DBA tuned
- Advisor as percentage of tuned

Speeds deployment
Improves performance
Frees up resource
# Autonomic Examples

| Systems Management | Access / Identity Managers  
|                   | Storage Resource Manager  
|                   | Service Level Advisor  
| Client            | ImageUltra  
|                   | Rapid Restore PC  
|                   | Embedded Security Subsystem  
| Application       | Prioritization of User Transactions  
|                   | Custom Advisors  
|                   | Problem Analysis and Recovery  
| Database & Collaboration | DB2 Query Patroller  
|                   | Tivoli Analyzer for Domino  
| Servers           | Dynamic Partitioning  
|                   | IBM Director  
|                   | BladeCenter  
| Storage           | Intelligent cache configuration  
|                   | Predictive Failure Analysis  
|                   | Dynamic volume expansion  

© 2003 IBM Corporation
Additional Information
Introduction to Grid Computing Video

• Available at www.ibm.com/grid

• View online or download

- Content:
  - What is Grid Computing
  - Benefits of Grid Computing
  - OGSA
  - Customer Testimonials
ITSO Redbook

• Redbook: Introduction to Grid Computing with Globus

• Available:
  - December 2002
  - Download from www.redbooks.ibm.com

• Content:
  - Presents the architecture and components to design a Grid solution by using the Globus 2.0 Toolkit
  - Explains different Grid types
  - Architecture and security considerations
  - OGSA and Grid middleware
  - Showcases several real-life application examples
Learning Services Class

• **Course**: Introduction to Grid Computing, the Globus Toolkit and OGSA

  ▪ **Content:**
    ▪ 2-day class, lecture-only
    ▪ Based on the Globus tutorial of same name
    ▪ Technical introduction both to Grid computing and the Globus Toolkit incl. descriptions of the core components
    ▪ Usage of the Globus Toolkit in various applications
    ▪ Future directions of Grid computing and the Globus Toolkit

More Courses planned for 2003 (e.g. Globus Developers+Admin Toolkits)
Grid and Autonomic Computing Information

www.ibm.com/grid

www.ibm.com/autonomic
Questions?