

Introduction to GT3

The Globus Project™
Argonne National Laboratory
USC Information Sciences Institute

Copyright (C) 2003 University of Chicago and The University of Southern California. All Rights Reserved. This presentation is licensed for use under the terms of the Globus Toolkit Public License. See <http://www.globus.org/toolkit/download/license.html> for the full text of this license.

Introduction to GT3

- **Background**
 - The Grid Problem
 - The Globus Approach
 - OGSA & OGSi
 - Globus Toolkit
- **GT3 Architecture and Functionality: The Latest Refinement of the Globus Toolkit**
 - Core
 - Base Services
 - User-Defined Services
 - Future Directions
- **Installation and Administration**
 - Installation
 - Configuration
 - Debugging
 - Support
- **Important Things to Remember**

A Story of Evolution

- Definition of Grid problem has been stable since original Globus Project proposal in 1995
 - Though we've gotten better at articulating it
- **But our approach to its solution has evolved:**
 - From APIs and custom protocols...
 - to standard protocols...
 - to Grid services (OGSA).
- **Driven by experience implementing and deploying the Globus Toolkit, and building real applications with it**

What is a Grid?

- **We believe there are three key criteria:**
 - Coordinates distributed resources ...
 - 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



Challenging Technical Requirements

- Dynamic formation and management of virtual organizations
- Discovery & online negotiation of access to services: who, what, why, when, how
- Configuration of applications and systems able to deliver multiple qualities of service
- Autonomic management of distributed infrastructures, services, and applications
- Management of distributed state
- Open, extensible, evolvable infrastructure



The Globus Project™

Making Grid computing a reality (since 1996)

- Close collaboration with real Grid projects in science and industry
- The Globus Toolkit®: Open source software base for building Grid infrastructure and applications
- Development and promotion of standard Grid protocols to enable interoperability and shared infrastructure
- Development and promotion of standard Grid software APIs to enable portability and code sharing
- Global Grid Forum: We co-founded GGF to foster Grid standardization and community



From APIs & Custom Protocols, To Standard Protocols



API Application Programming Interface

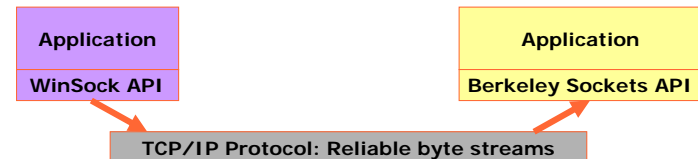
- A specification for a set of routines to facilitate application development
 - Refers to definition, not implementation
- Often language-specific (or IDL)
 - Routine name, number, order and type of arguments; mapping to language constructs
 - Behavior or function of routine
- Examples of APIs
 - GSS-API (security), MPI (message passing)

Network Protocol

- A formal description of message formats and a set of rules for message exchange
 - Rules may define sequence of message exchanges
 - Protocol may define state-change in endpoint, e.g., file system state change
- Good protocols designed to do one thing
 - Protocols can be layered
- Examples of protocols
 - IP, TCP, TLS (was SSL), HTTP, Kerberos

A Protocol can have Multiple APIs

- TCP/IP APIs include BSD sockets, Winsock, System V streams, ...
- The protocol provides **interoperability**: programs using different APIs can exchange information
- I don't need to know remote user's API



An API can have Multiple Protocols

- An API provides **portability**: any correct program compiles & runs on a platform
- Does not provide interoperability: all processes must link against same SDK
 - E.g., MPICH and LAM versions of MPI



Initial Focus On APIs and Custom Protocols

- Primary concern was allowing Grid applications to be built quickly, in order to demonstrate feasibility
- Good development APIs and SDKs mattered most
- Protocols were a means to an end
 - We borrowed and extended standard protocols to make life easier (e.g. LDAP)
 - We defined custom protocols (e.g. GRAM)

But Focus Shifted To Protocols

- As demand grew, customers worried about:
 - compatibility between versions (i.e. Stop changing the protocols!)
 - independent implementations of some components (i.e. What are the protocols?)
- Ubiquitous adoption demands open, standard protocols
 - Internet and Web as guides
 - Enables innovation/competition on end points
 - Avoid product/vendor lock-in

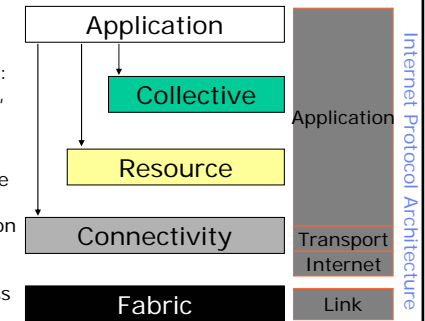
Layered Grid Architecture

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

"Sharing single resources": negotiating access, controlling use

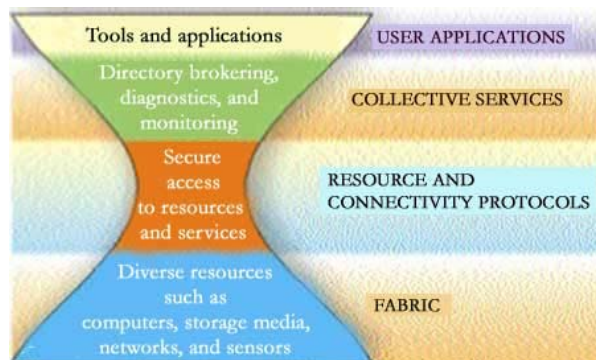
"Talking to things": communication (Internet protocols) & security

"Controlling things locally": Access to, & control of, resources



"The Anatomy of the Grid: Enabling Scalable Virtual Organizations", Foster, Kesselman, Tuecke, Intl Journal of High Performance Computing Applications, 15(3), 2001.

Layers of Grid Architecture



GT2 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 Services: Grid Resource Information Protocol (GRIP)
 - > Data Transfer: Grid File Transfer Protocol (GridFTP)
- Also key collective layer protocols
 - Info Services, Replica Management, etc.

Protocol Standards Efforts

- X.509 Proxy Certificate Profile
 - GGF & IETF
- GridFTP Protocol
 - GGF

From Standard Protocols, To Grid Services

But Along The Way...

- Heterogeneous protocol base was hurting us
- Increasing number of virtual services that needed to be managed
- Web services (WSDL, SOAP) appeared

Web Services

- At the heart of Web services is:
 - WSDL: Language for defining abstract service interfaces
 - SOAP (and friends): Binding from WSDL to bytes on the wire
- Web services appears to offer a fighting chance at ubiquity (unlike CORBA)
- But Web services does not go far enough to serve a common base for the Grid...

Transient Service Instances

- “Web services” address discovery & invocation of persistent services
 - Interface to persistent state of entire enterprise
- In Grids, must also support transient service instances, created/destroyed dynamically
 - Interfaces to the states of distributed activities
 - E.g. workflow, video conf., dist. data analysis, subscription
- Significant implications for how services are managed, named, discovered, and used
 - In fact, much of Grid is concerned with the management of service instances

Standard Interfaces & Behaviors: Four Interrelated Concepts

- Naming and bindings
 - Every service instance has a unique name, from which can discover supported bindings
- Lifecycle
 - Service instances created by factories
 - Destroyed explicitly or via soft state
- Information model
 - Service data associated with Grid service instances, operations for accessing this info
 - Basis for service introspection, monitoring, discovery
- Notification
 - Interfaces for registering existence, and delivering notifications of changes to service data

Grid Evolution: Open Grid Services Architecture

- Refactor Globus protocol suite to enable common base and expose key capabilities
- Service orientation to virtualize resources and unify resources/services/information
- Embrace key Web services technologies for standard IDL, leverage commercial efforts
- Result: standard interfaces & behaviors for distributed system management: the Grid service

OGSA Structure

- A standard substrate: the Grid service
 - OGSi = Open Grid Service Infrastructure
 - Standard interfaces and behaviors that address key distributed system issues
 - Much borrowed from GT abstractions
- ... supports standard service specifications
 - Resource mgt, dbms, workflow, security, ...
 - Target of current & planned GGF efforts
- ... and arbitrary application-specific services based on these & other definitions

OGSI Grid Service Specification

- Defines WSDL conventions and GSDL extensions
 - For describing and structuring services
 - Working with W3C WSDL working group to drive GSDL extensions into WSDL
- Defines fundamental interfaces (using WSDL) and behaviors that define a Grid Service
 - A unifying framework for interoperability & establishment of total system properties

Globus Toolkit (GT)

- A software system addressing key technical problems in the development of Grid-enabled tools, services, and applications
 - Offer a modular set of orthogonal services
 - Middleware for building solutions, not turn-key
 - Enable *incremental* development of Grid-enabled tools and applications
 - Implement and inform Grid standards
 - Available under liberal open source license
 - Large community of developers & users
 - Multiple commercial support providers

Why Open Source is Important

- Leverages large body of code and experience
 - Efforts of a large e-Science community
- Encourages adoption of open standards
 - Reference implementation, community pressure
- Facilitates integration of new platforms
 - Port the implementation
- Allows vendors to focus on value add
 - Platforms, integration, higher-level services, turnkey applications, training, support

OGSA and the Globus Toolkit

- Technically, OGSA enables
 - Refactoring of protocols (GRAM, MDS, GridFTP), *while preserving all GT concepts/features!*
 - Integration with hosting environments: simplifying components, distribution, etc.
 - Greatly expanded standard service set
- Pragmatically, we are proceeding as follows
 - Develop open source OGSA implementation
 - > Globus Toolkit 3.0; supports Globus Toolkit 2.0 APIs
 - Partnerships for service development
 - Also expect commercial value-adds

GT2 Evolution To GT3

- What happened to the GT2 key protocols?
 - Security: Adapting X.509 proxy certs to integrate with emerging WS standards
 - GRIP/LDAP: Abstractions integrated into OGSI as serviceData
 - GRAM: ManagedJobFactory and related service definitions
 - GridFTP: Unchanged in 3.0, but will evolve into OGSI-compliant service in 2004
- Also rendering collective services in terms of OGSI: RFT, RLS, etc.

GT Timeline

- GT 1.0: *1998*
 - GRAM, MDS
- GT 2.0: *2001*
 - GridFTP, packaging, reliability
- GT3 Technology Preview: *Apr-Dec 2002*
 - Tracking OGSI definition
- GT3.0 Alpha: *Jan 2003*
 - OGSI Base, GT2 functionality
- GT3.0 Production: *June 2003*
 - Tested, documented, etc.
- GT3.2.1 Production: *July 2004*
- GT3.9.2 Development: *Aug 2004*
- GT4 Scheduled beta development *Dec 2004*

Summary

- The Grid: Coordinates resources that are not subject to centralized control; using standard, open, general-purpose protocols and interfaces; to deliver non-trivial qualities of service.
- Considerable impact within eScience, growing interest & adoption within eBusiness
- Globus Toolkit an open source, defacto standard source of protocol and API definitions—and reference implementations
- GT3 is evolution of the Globus Toolkit path