Introduction to Grid
MVP Guest Lecture

Henrik Thostrup Jensen <htj@ndgf.org>
Nordic Data Grid Facility / NorduNET

April 7, 2010

Outline

- What is grid
  - Visions, definitions, and challenges

- Grid Services
  - Grid as services
  - Service types
  - Service interaction

- Summary

A Huge Virtual Distributed Supercomputer

- Most agree that this cannot really work

- Grid does not try to make several systems appear as one
- Provides standard mechanisms for accessing systems

A computational grid is a hardware and software infrastructure that provides dependable, consistent, pervasive, and inexpensive access to high-end computational capabilities.

The Grid - A Blueprint for a new Computing Infrastructure, 1998

The real and specific problem that underlies the Grid concept is coordinated resource sharing and problem solving in dynamic, multiinstitutional virtual organizations. The sharing that we are concerned with is not primarily file exchange but rather direct access to computers, software, data, and other resources, as is required by a range of collaborative problem solving and resource-brokering strategies emerging in industry, science, and engineering. This sharing is, necessarily, highly controlled, with resource providers and consumers defining clearly and carefully just what is shared, who is allowed to share, and the conditions under which sharing occurs. A set of individuals and/or institutions defined by such sharing rules form what we call a virtual organization.

The anatomy of the Grid, 2000

What is the Grid 4/4: The Checklist

- Grid is a system that. . .
  - coordinates resources that are not subject to centralized control
  - using standard, open, general-purpose protocols and interfaces
  - to deliver nontrivial qualities of service.

  - enables distributed collaboration

- Grid is also a buzzword
  - Lots of software have received a "grid" stamp for PR purposes
  - Fortunately the buzz has moved to "cloud" these days

What is Grid Needed For

- Sharing resources → better resource usage
  - Many clusters spend a lot of time idle
  - Standard mechanisms for accessing resources is essential

- E-science
  - Distributed collaboration, huge data sets
    - The LHC is the prime example
  - Many projects span multiple organizations and cannot be carried out by a single organization
  - Many sciences completely depend on computers
    - Bioinformatics, computational chemistry, high-energy physics

- PS. Read the Science 2020 report (google: science 2020)

Challenges in Grid

- Make hardware accessible to persons outside the organization that own the hardware
  - While continuing normal operation

- Authentication and authorization
- Protection of resources
- Control over resources
- Accounting
- Anonymity

Grid as Services

- Resources are accessed through services
  - Note: SOA ≫ Web Services
- A grid is composed of different services
- Grid service types
  - Job Execution, Data Storage, Information System
  - More exist, but these are the foundation

Detour: Cluster LRMSes

- A cluster is basically a bunch of machines+network
- Having persons "just use the resources" $\rightarrow$ poor utilization
- The task of an LRMS is to coordinate resource usage

- LRMS = Local Resource Management System
    - Fancy way to spell "queue system"
    - Submit job to queue
    - When resources are available, the job will be started
- Most LRMSes provide some balancing of users
    - Queue reordering, backfill, balancing
- Several LRMSes exists and are being used

Grid Services 1/4: Job Execution

- Most fundamental grid service
- Submit a job description, service executes job
  - Can be though of as remote LRMS access
- Resource is typically a cluster or supercomputer
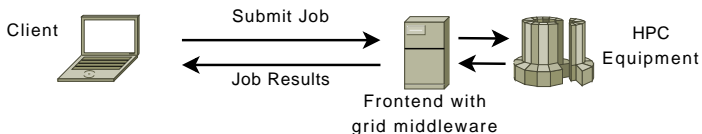
- Underlying details are not abstracted away
  - Grid is not about making resources look the same
- Most grids have their own submission interface
- Several job description languages exist

Grid Services 2/4: Data Storage

- Upload data, retrieve it later
  - Typically done by third parties
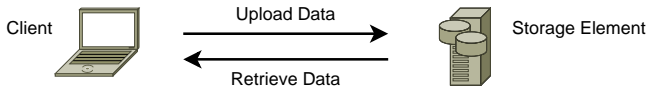
- Data often live for long periods of time
  - Typically shared within a group
  - Surprisingly difficult to manage
- GridFTP is the common transfer protocol
- SRM is the common(only) storage management protocol
- HTTP and others are also used

Grid Services 3/4: Information System

- Describes services - The backbone of a grid

- Aggregated Information Service
  - Provides high level information on other services
  - aka. resource discovery

Information System

2. Resource discovery

1. Resource registration

Client

3. Query Resource
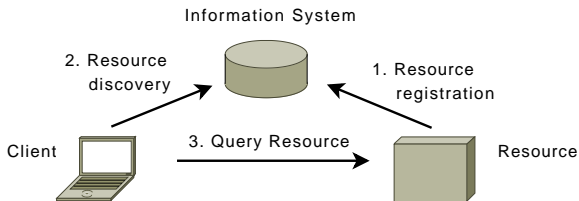
Resource

- After discovery, resources are queried for details
  - Current load, estimated queue, available disk space, etc.
  - Resource query interface runs alongside every service

Grid Services 4/4

- Many service types exist
- Data Catalog
    - Keeping track of file locations
    - Typically contains metadata as well
- Federated services
    - A service providing an abstraction over other service(s)
    - Examples: Work flow, data replication
- Auxiliary services:
    - File transfer, accounting, cache index
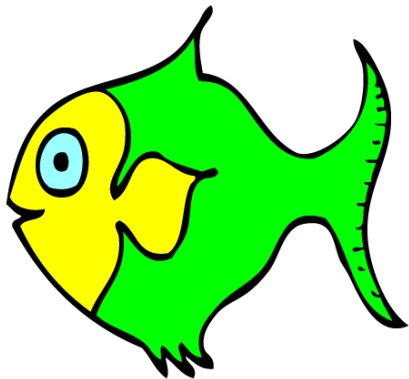
Service Communication

- Services are not islands - they communicate
- Basic example
  - A service registers itself to the information system
  - Makes it possible for clients to discover them

Job Submission Flow

Grid Security
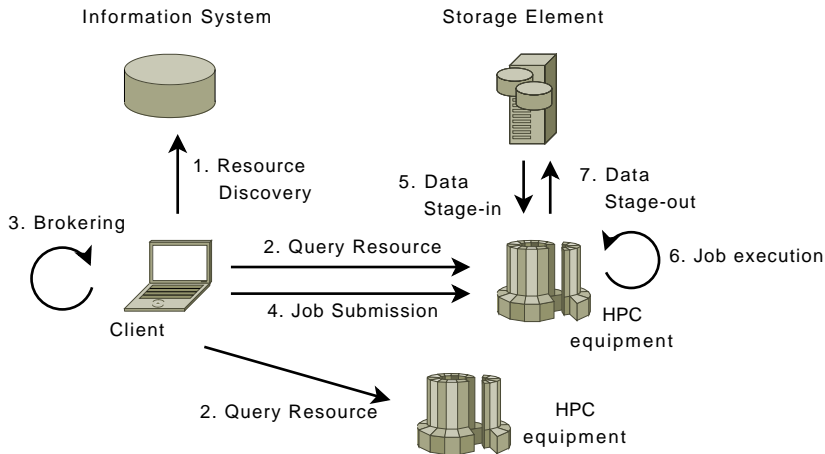
- All these services require secure access
- Requirements
    - Must work between administrative domains
    - Users must be able to delegate rights
- Most existing security solutions are not usable
- A new security infrastructure have been created
    - GSI - Grid Security Infrastructure

Grid Security Infrastructure

- Based on Public Key Infrastructure (PKI)
- Host certificates work as usually in PKI
- Users are equipped with certificates as well
    - Primary certificate is not used directly
    - Proxy certificates are used instead
    - Proxy certificates enable delegation

- Works pretty well

- Typically some user grouping facility is used as well

Grid software and protocols

- Lots of half pieces have been constructed
  - We don't really know how the puzzle should look
- Many grid projects are now case studies in how to not use research money
- Grid software is quite complex
  - Complex behaviour (lots of parallism, failures, etc.)
  - Have to integrate with many heterogenous systems
- Protocols: Many have been made, far fewer is being used
  - Software is typically made before the protocols (good & bad)
  - They often suffer from being "organically" grown
  - Some people like to create standards without real-world experience
- The whole thing is a bit of a mess

ARC: Advanced Resource Connecter

- Middleware developed in the Nordic countries
- Runs a grid infrastructure today
  - About 70 sites, most are in northern Europe.
- Reasonably non-intrusive, relatively simple design
  - Only needs to be installed on the cluster front-end
  - Support for several different LRMSs (many only work with one)
- Somewhat unique view on data staging
- Services:
  - Job execution, data storage, information system
  - Integrates quite well with other services

An outlook on grid infrastructure

- Five years ago we had a lot of separate grids
  - Little or no cross-communication
- Today we share security infrastructure
- Sharing data between grid infrastructure is relatively painless
- Job submission between grids exists - rarely used

- Applications are often build to work on a specific grid
  - Problems when relocating them
  - Typically problems with data staging, scheduling
- The lines between grid middleware are slowly disappearing
- We're building up, sideways and changing the foundation
- Lots of things to do; still a young field

Summary

- Grid is...
  - infrastructure for resource sharing and distributed collaboration

  - composed of services
    - job execution, data storage, information system
    - services interact