Progress of Grid Technology Development

A Presentation for the

ESRIN Grid Tutorial

Overview

- Overview the development of Grid standards & technology from inception to current state of art and beyond
  - Pre Globus
  - Globus Toolkit
  - EDG
  - GT3, EGEE and beyond
- Gridification of applications

The Grid Concept

- Distributed collaborative computing, a way to manage a mixed set of distributed computing resources, shared over the extended wide-area
- Principle : collaborative sharing of IT resources and science will deliver a powerful problem solving capability
- Using a widely accessible infrastructure

Grids - Objective

- Develop international standards for information and accessibility, harmonizing the differences between resources managed by different organizations
- In a nutshell :
  - Easy access to computing power
  - Promote intercollaboration
  - Exploitation of resources

Grids - Resources

- Types of available resources :
  - **Processing power** : clusters and supercomputers
  - **Data Storage** : disks and archives
  - **Software** & **tools** : applications and environments
  - **Data** : catalogues, databases
  - **High speed networks**

Grids - How ?

- By mapping the many different local methods, policies, etc. to common *global* standards
- Key is to develop Grid standards, integrating available technologies and innovative solutions
- Grids are highly complementary to the development of research networks (DANTE/ Geant)

Grids - In Contrast

- Similarities with Web services (e.g. service discovery, remote invocation), but aims to assist *intercollaboration* across scientific domains
- More than exploitation of unused CPU cycles e.g. SETI@home, because of the emphasis on large-scale collaborative problem solving

Getting Grids Going

- High-level research is increasingly complex, interdisciplinary and costly
- It requires a constantly increasing critical mass of resources
- Large scale projects can respond directly to these needs by coordinating the efforts of individual organizations and national initiatives

Development - phase 0 (pre Globus)

- Users set up and maintain individual accounts on different machines (labour intensive)
- Manually manage a collection of account names, passwords and application environments on different machines (e.g. master nodes of clusters, supercomputer gateways)

Development - phase 0 (manual)

- login to remote machines to execute commands
- use FTP to transfer files (login again)
- if using different platform architectures develop customized, ‘once off’ solutions to handle local differences
- maintain own database of machines, accounts, passwords, environments installed etc.

Development - phase 1 (Anatomy of the Grid)

- Single login using electronic certificate
- Gatekeeper interface to local systems
- Submit jobs remotely, query status of jobs
- Resource Specification Language
- Grid FTP
- Grid Information System
- Virtual Organization

Development - phase 1 (Anatomy of the Grid)

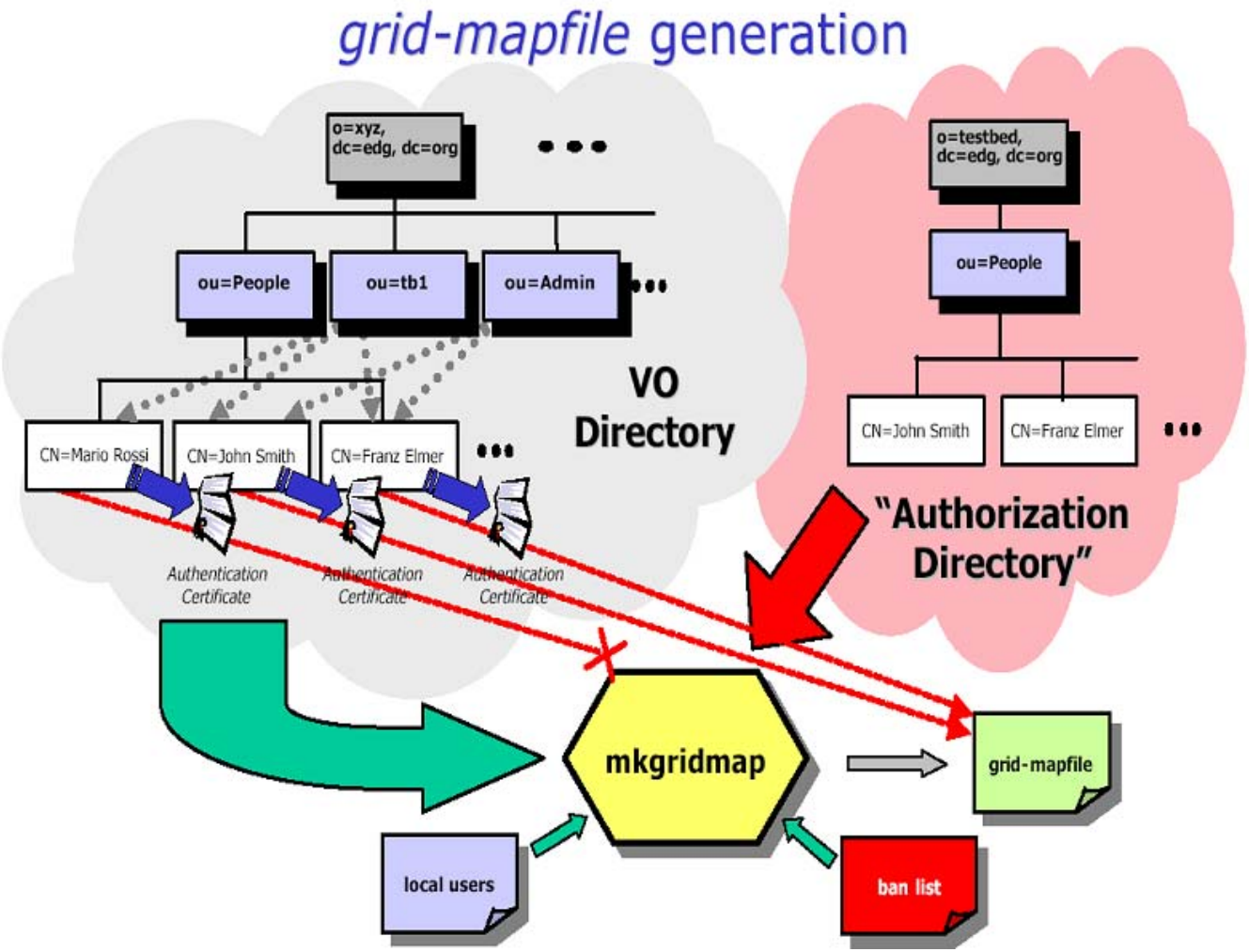
- Global Grid Forum is formed
- Principal Technology drivers are Globus and Condor
  - Grid Security Infrastructure (GSI) provides mutual authentication between user and service
  - Meta Directory Service (MDS) publishes dynamic resources information (GIIS)
  - VO server (LDAP)

• • •

Development - phase 1 (Anatomy of the Grid)

- Globus Gatekeeper
  - GRAM - interfaces to local batch system
  - GRIS - publishes resource capabilities
- List of accepted users (gridmapfile) maps remote credentials onto local accounts
- Condor provides additional resource management and matchmaking (Classads)
- The architecture & components are described in the ‘Anatomy of the Grid’ Paper

Development - phase 1 (Anatomy of the Grid)



Development - phase 2 (Europe concerted effort)

- EU Datagrid develops comprehensive add-on functionality
  - Resource Brokering
  - Job matching based on Condor 'class ads' (JDL)
  - Replica management (Replica Location Service)
  - Storage resource management (SRM)
  - Fine-grained security (VOMS)

. . .

Development - phase 2 (Europe concerted effort)

- Information System
  - GLUE schema
  - Relational Grid Monitoring Architecture
- Fabric management (LCFG)
  - Automated installation & configuration of packaged Grid software distribution
- User requirements input from 3 diverse application groups
- Testbed sites in CERN, UK (RAL), France (IN2P3), Italy (INFN), Germany (FZK), Spain (IFAE)

…and ESRIN

Development - phase 2 (EU DataGrid)

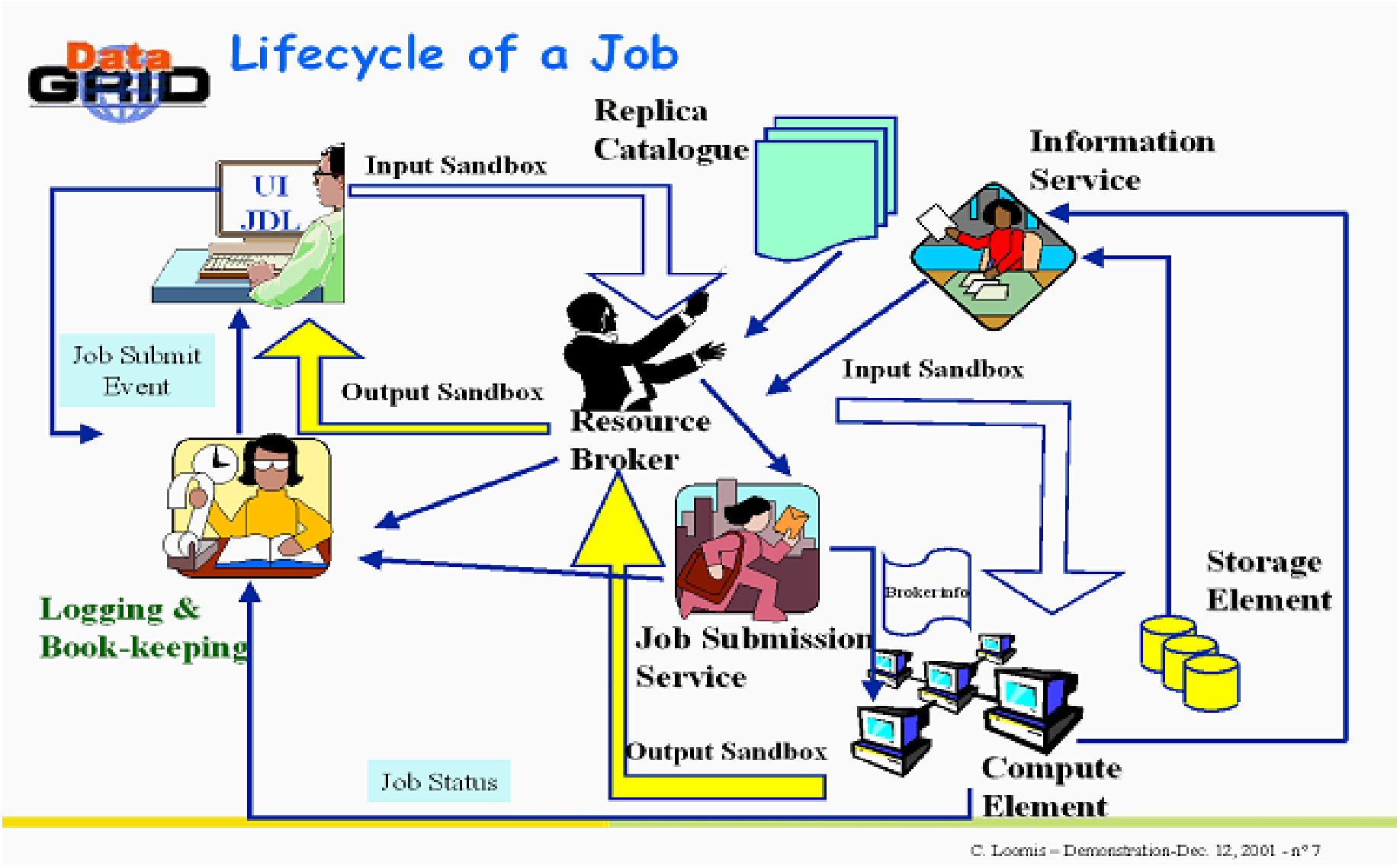
- Tackled problems of operating a large testbed across many countries
  - Middleware architecture & design addresses requirements of large community resource providers & user applications
  - Set up list of recognized Certification Authorities in several countries
  - Develop common set of policies & usage rules

• • •

Development - phase 2 (EU DataGrid)

- Deploy large testbed in operations with user support (sites in six countries)
- Integrate middleware from many different sources
- Testing & user feedback by advance testers and Application groups
- Bug reporting, tracking & voting (Bugzilla)
- Fast turnaround release distribution
- First large scale deployment of Globus in EU

Development - phase 2 (EU DataGrid)



Development - phase 2 ('Physiology' of the Grid)

- Globus technology embraces Web Services and creates 'Grid Services' (GT3)
  - Open Grid Services Architecture (OGSA)
  - OGSI v1.0 specification now 'fixed' as official GGF recommendation
- Focus is on services, rather than resources
  - resources accessed via web services
- Several GT2 components re-writing in Java

Development - phase 2 (GT3 / OGSA)

- The major GT2 components will be kept e.g.
  - GSI, GRAM, GridFTP
- They will be integrated within a common Grid services framework
- MDS/GIIS becomes Index Service and no longer LDAP-based but XML
- No date set yet for end of GT2 support

Development - phase 2 (GT3 / OGSA)

- Aims to provide a standard Grid/Web services framework widely useable by both meta- and system-level services
- Targeted at e-Science and e-Business application domains
- For integration by platform vendors (IBM, etc.)

Development - phase 2 Standards Development

- Some key components are being developed in common with EDG (via Globus collaborations and GGF interest groups), e.g.
  - Replica Location Service (RLS)
  - Storage Resource Manager (SRM)
  - Proxy certificate extensions using attribute certificates (VOMS)

The situation today ...

- Basic Grid services (i.e. Globus GT2) are mostly stable but miss some 'nice to have' functionality, e.g.
  - Resource brokering
  - Job Submission System
  - Fully scalable Grid Information System
  - Replica Management
  - Storage Resource Management

Today ...

- EU DataGrid (also with some collaboration in GGF) has developed and integrated some of these enhancements, with *caveats* :
  - latest testing shows some stability problems
  - Information System limited scalability
  - VOMS security is being tested
  - a few core EO User Requirements are outstanding, e.g.
    - support for Application Metadata
    - fine-grained access control
    - logical collections

Today ...

- We expect these issues will be solved as EGEE deploys components in a widely distributed testbed
  - 70 leading institutions in 27 countries
  - federated in regional Grids
  - combined capacity of over 20,000 CPUs
- Meanwhile, we can prepare applications using a reduced command-set !

Next Developments

- The standards are emerging
  - OGSI technology is expected to become increasingly integrated by commercial platform vendors
  - Grid Service Specification (GGF)
  - development of standards (IETF)
  - convergence with Web Services (W3C)

Next Developments

- Europe is well advanced in Grid developments, will continue to establish the technology & networking concepts, building on results achieved so far
- EGEE aims to establish a European "*common market in computing*" for e-Science, industry, education, finance, government ...

Summary

- Development of the technology is steadily progressing
- Many new issues are being solved e.g. role-based security, accounting/auditing, QoS, advance reservation, notification
- Different solutions will be tried, with many styles of Grids
- There will be Federations of Grids
- Gradual uptake in operational use

Gridification of Applications

- Types of Applications / Users
  - Collaborations
    - projects composed of participants in different countries / organizations / scientific domains
  - Large dataset production and reprocessing
  - Modelling and Simulation
  - Computing resource intensive, number-crunching

Gridification of Applications

- Port to Linux as a common platform (e.g. DataGrid uses RH)
  - most commonly used tools & languages available (C/C++, Java, PERL, Python, Csh, etc.)
  - IDL environment already installed
- Remove all dependencies on local ‘home environment’
  - applications may execute at distant sites, scheduled on a different machine each time

Gridification of Applications

- Minimize data transfers
  - Small-volume data sent with the job at submission time, but
  - Large-volume programs and data should be pre-installed or replicated in advance
  - Small-volume results sent back to user/client application, but
  - Large-volume data uploaded to 'Close' storage (usually mounted by NFS)
  - Jobs sent to execute close to the data source

Gridification of Applications

- Don’t leave any data on the Worker Node, you can't rely on accessing it later
- Use the Replica Catalogue to register datasets
- Use the Information System to register Application environments, resources, services, etc.

Gridification of Applications

- Exploit possibilities for parallelization
  - think of an application in terms of hundreds of concurrent processes, thousands of data objects
  - possibly widely distributed (across several clusters)
  - may be either loosely coupled :
    - distributed over several sites
  - or tightly coupled :
    - execute on a single cluster
    - compiled with MPI libraries
    - wide-area (i.e. between sites) MPI possible, but not used much yet

Gridification of Applications

- Data is distributed 'somewhere on the grid'
  - the grid knows where it is (Replica Catalogue)
  - the application just refers to a ‘Logical dataset name’
  - JDL matches the jobs requirements with available resources
  - file access is controlled by Grid Security
  - use metadata catalogues (Spitfire, RMC, or other)

Gridification of Applications

- Several application grid interfacing tools and environments are available, e.g.
  - the ESA Grid Portal
  - EO ‘Grid Engine’, ‘Grid Surfer’
  - Grid Assist (OMI)
  - GENIUS
  - others ...

Gridification of Applications

- Getting started
  - Request certificate from one of the recognized CAs
  - Register as Testbed user (if running on EDG)
  - Obtain an account on a User Interface (UI) machine (EDG or Globus installed)
  - Install public and private key on UI machine
  - Ready to submit jobs and transfer files
  - Many people collaborating – join or form a VO
  - Join user mail lists
  - Read the documentation