Enabling Grids for E-sciencE

Web Services, WSRF and GT4

Dr. Rüdiger Berlich,
Forschungszentrum Karlsruhe / Germany
Mumbai, 11.02.06

Slides by
Dr. Mike Mineter
National e-Science Centre, Edinburgh

www.eu-egee.org

Enabling Grids for E-sciencE

Goal of presentation

- An orientation to Web Services and to their role in Grid computing
- No prior knowledge assumed

Enabling Grids for E-sciencE

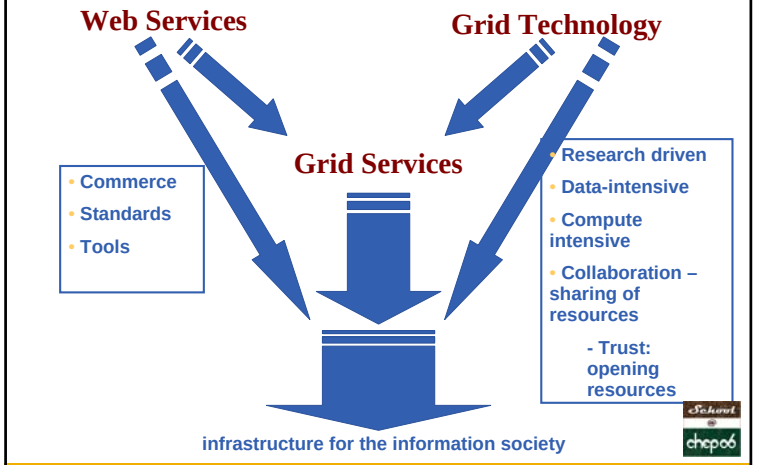
Outline

- " Web Services are *the* way to build Grids"
- Web Services
- Relevance of Web Services to Grids
- Extending WS for grids
- So where are we now ?

Enabling Grids for E-sciencE

Infrastructure for the industrial society: The Forth Bridges

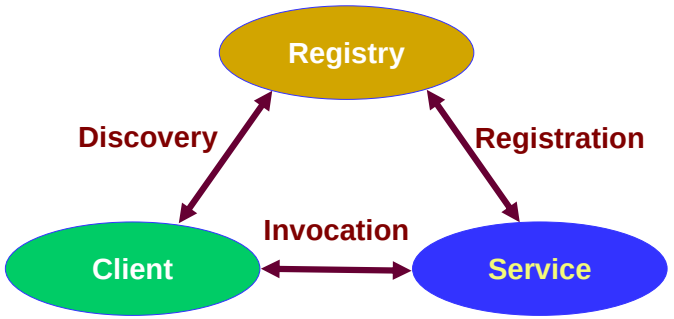
October 2001 View



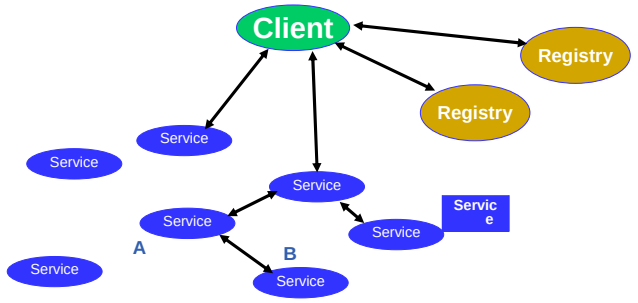
What are "Web Services?"

- **History**
  1. Web browsing
  2. Web pages with content from applications
  3. Applications that are useable by software clients
- **Web Services are software components that are..**
  - Accessible across a network
  - Loosely coupled
    - Defined by the messages they receive / send
    - Modular and self-contained
    - So can change service implementation without changing interfaces
  - Interoperable: each service has a description that is accessible and can be used to create software to invoke that service
- **… and based on standards**
  - Usually built on (extensions of) standards made ubiquitous by the Web: http(s), XML, … and for which tools are already built.
  - Developed in anticipation of new uses – e.g. can compose workflow
  - Encouraging adoption

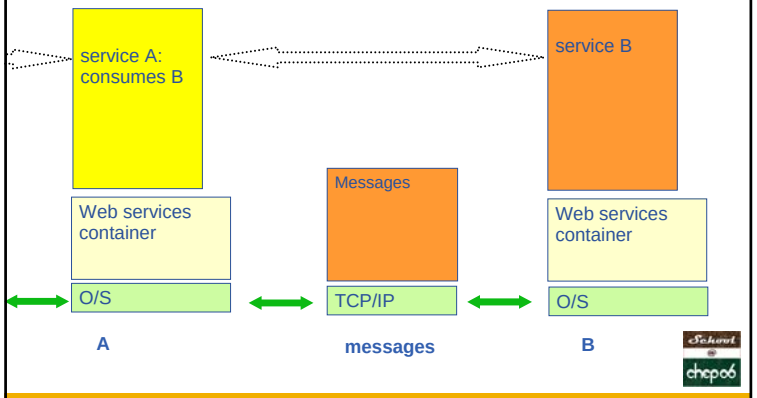
Web Services



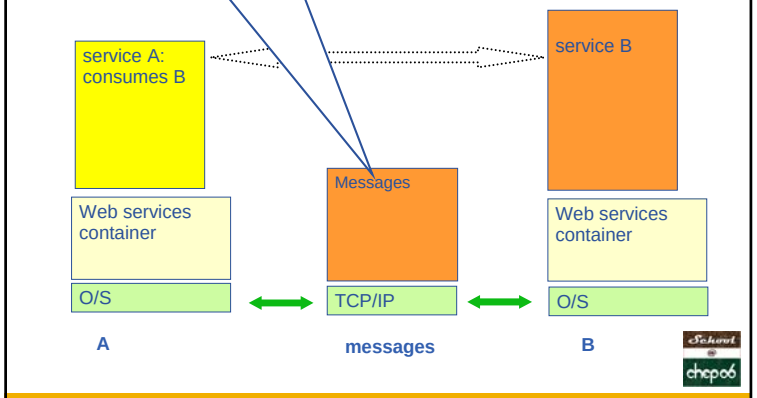
Dynamic composition



Using service B from service A

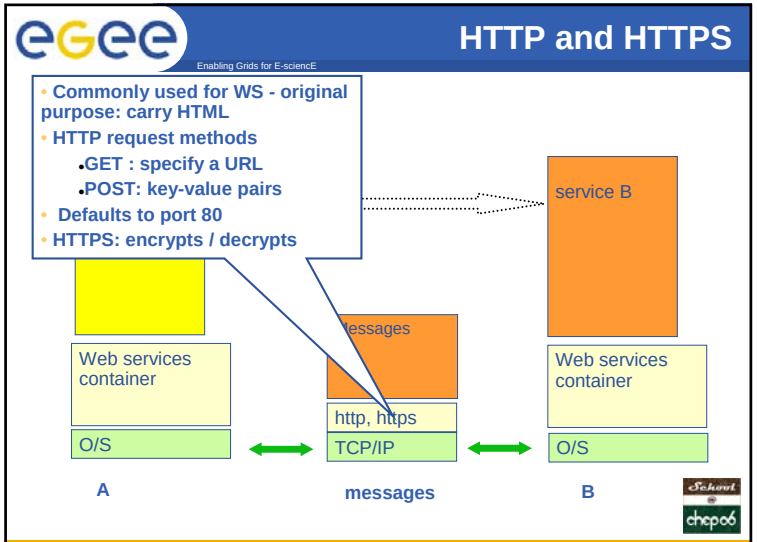


Using service B from service A



HTTP and HTTPS

- Commonly used for WS - original purpose: carry HTML
- HTTP request methods
  - GET : specify a URL
  - POST: key-value pairs
- Defaults to port 80
- HTTPS: encrypts / decrypts

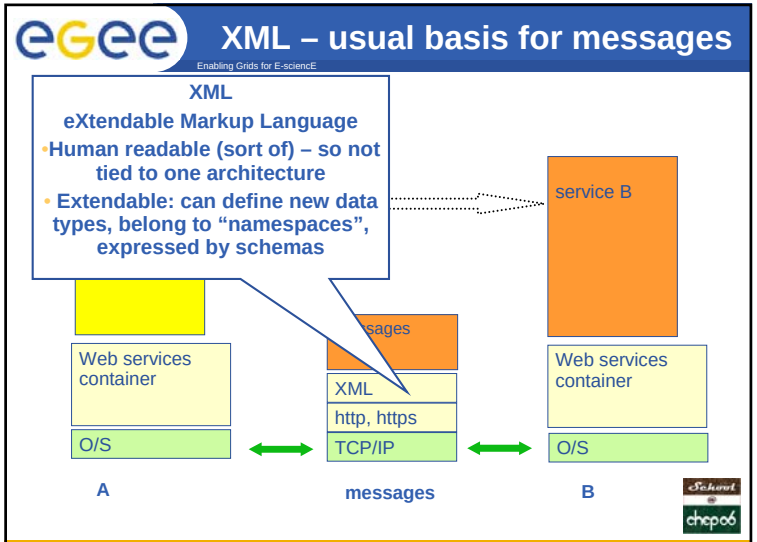


XML – usual basis for messages

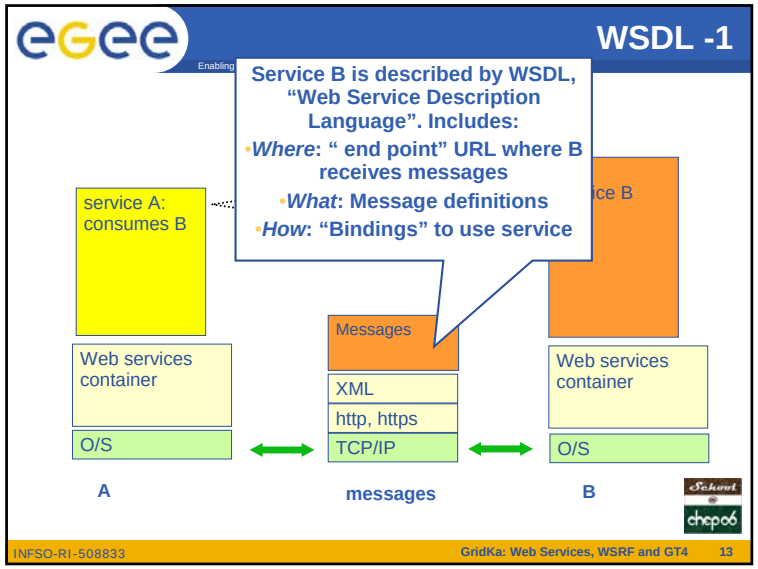
XML

eXtendable Markup Language

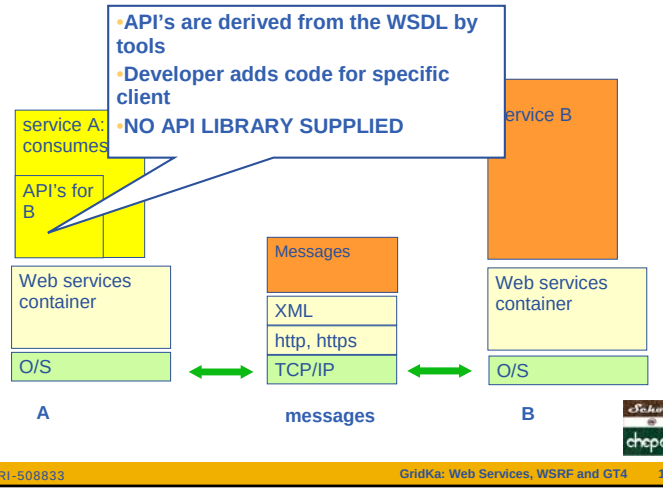
- Human readable (sort of) – so not tied to one architecture
- Extendable: can define new data types, belong to “namespaces”, expressed by schemas



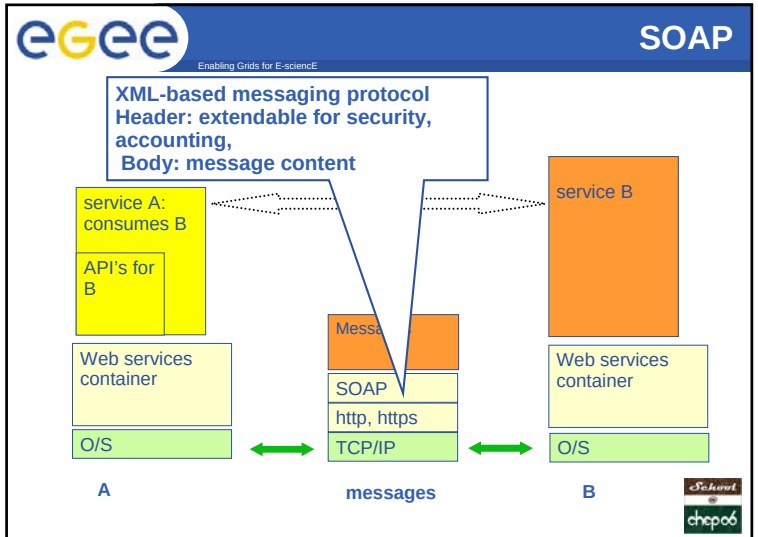
WSDL -1



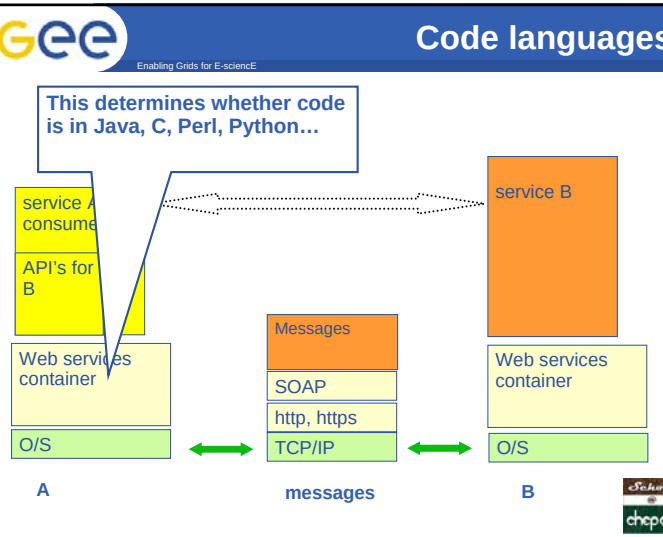
WSDL -2



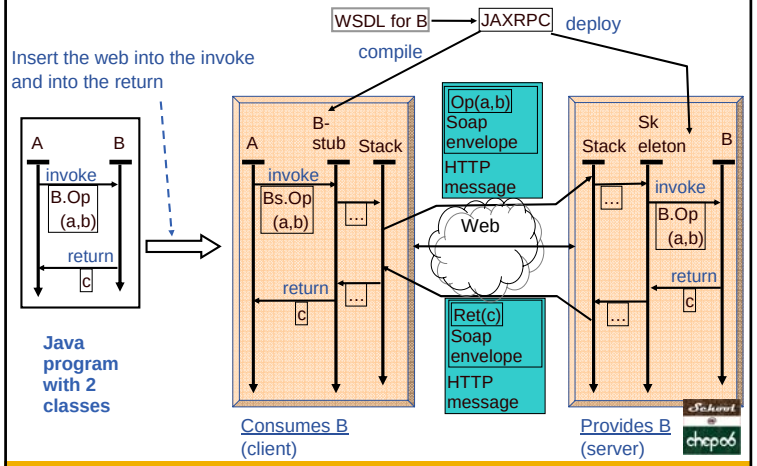
SOAP



Code languages



(JAVA) Web Services



WS-I core of Web Services

- WS-I (Interoperability) delivers practical guidance, best practices and resources for developing interoperable Web services solutions.
- http://www.ws-i.org/

Open standards:

- SOAP: protocol for message passing
- Web Service Description Language: to describe services
- UDDI: Universal Description, Discovery and Integration
- WS-Security: incorporates security

Outline

- **" Web Services are *the* way to build Grids"**
- **Web Services**
- **Relevance of Web Services to Grids**
- **Extending WS for grids**
- **So where are we now ?**

Grids need….

- **Software components that are..**
  - Accessible across a network
  - Loosely coupled
    - Defined by the messages they receive / send
    - Modular and self-contained
    - So can change service implementation without changing interfaces
  - Interoperable: each service has a description that is accessible and can be used to create software to invoke that service
- **… and based on standards**
  - Tools, interoperability, …
  - Developed in anticipation of new uses – e.g. can compose workflow
- **i.e. what web services exist for!**
- **So now building grid architecture based on WS**
- **But there are additional challenges!!!!!**

WS & Grid Goals

Web Services

- **Goals**
  - Computational presentation & access of Enterprise services
  - Marketing integrated large scale software and systems
  - Model for independent development
  - Model for independent operation

Grids

- **Goals**
  - Inter-organisational collaboration
  - Sharing information and resources
  - Framework for collaborative development
  - Framework for collaborative operation

WS & Grid Usage

Web Services

- **Complex services created & delivered persistently by owner organisation**
- **Client interactions short-lived**
- **Multi-organisation integration responsibility of client**
  - Workflow enactment
  - Transaction coordination
  - May be by an intermediate service

Grid Services

- **All of WS patterns +**
- **Dynamic services / resources**
- **Long-lived interactions**
- **Persistent computational integration**
  - Data management
  - Computation management
- **Persistent operational infrastructures**
  - EGEE managing European-scale grid
- **System organised optimisation**
- **End-to-end security (goal)**
- **Virtual Organisations**
  - Establish multi-organisation security policies

WS & Grid Standards

Web Services

- **WS-I**
  - Core agreed & provided
  - WSDL, SOAP, UDDI, WS-Security
- **Many others under way**
  - WS-* are important
  - Competition & synthesis
- **Commercial battleground**

Grids

- **None**
- **Many exist as proposals**
- **Continuum from requirements & research to well specified standards proposals**
- **Building on & influencing WS**
- **Transfer from OGSI to WS-Resource Framework & WS-Notification**

Operational status

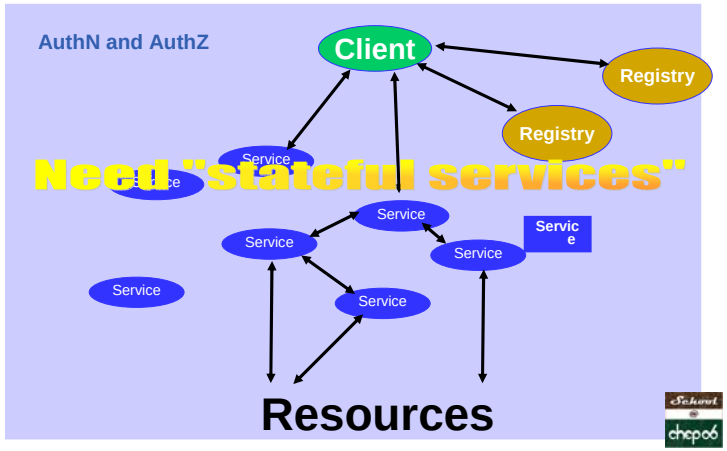
Web Services

- **Commercially successful operational applications**
- **Several good toolsets available**
  - Mostly costly to use outside academia
- **Workflow enactment**
  - BPEL4WS
- **Scale, usability & reliability problems in free-ware**
  - Many fixes were needed to Apache Tomcat
- **Much momentum**
  - Very high levels of investment

Grids

- **Operational research projects and grids**
  - >100 projects use GT2 or GT3
- **No toolsets**
- **Scientific workflow**
  - High-level work-load generators
  - Chimera, Pegasus, Taverna, …
- **Some very robust and well tested technologies**
  - Condor, GT2, VDT, GT3.2, LCG2, EGEE1
- **All free-ware**
- **Performance, usability and reliability problems**
- **Much momentum**
- **High levels of investment**

Grids – and resources

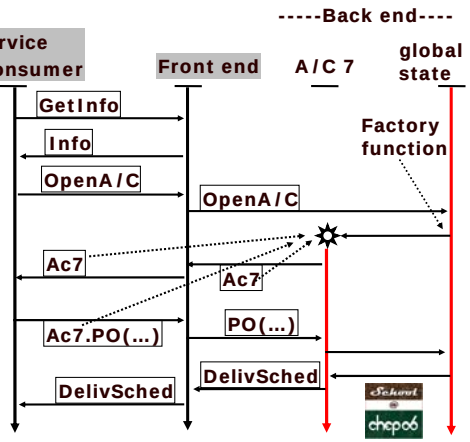


A bit of history

- "Open grid services architecture" OGSA– proposed in 2001
- Open Grid Services Infrastructure
  - Globus Toolkit 3 resulted
  - Specified in 2003
- Then in January 2004
  - OGSI to be replaced by emerging WS-RF (Web Services Resource Framework)
- NOTE:
  - OGSA still under development (GGF)
- Imbalances in OGSI that are addressed by WS-RF(OASIS)
  - Over O-O, megalithic, WS community not engaged
  - Focus onto state and resources

Stateful Resources

- Web service itself is stateless
- Front end Freely to multiple instances of back-end for each resource
- Maintains state in a back-end
- Service request identifies the specific resource



Component Standards

- WSRF builds on
  - WS-Addressing – W3C submission Aug 2004
  - WS-Notification
- WSRF comprises standards
  - **WS-ResourceLifetime**
  - **WS-ResourceProperties**
  - WS-RenewableReferences
  - WS-ServiceGroup
  - WS-BaseFaults
- WSRF supports –
  - **WS-Notification**
    - WS-BaseNotification
    - WS-BrockeredNotification
    - WS-Topics

Where are we now?!

- **Standards are emerging… some near acceptance and some being discarded**
  - Standards bodies:
    - W3C http://www.w3c.org/
    - GGF http://www.ggf.org/
    - OASIS http://www.oasis-open.org/home/index.php
    - IETF http://www.ietf.org/
  - For a summary see http://www.innoq.com/soa/ws-standards/poster/
- **Production grids are based on de-facto standards at present**
  - Inevitably!
  - GT2 especially
  - But locks a grid into one middleware stack unable to benefit from the diverse developments of new services
- **Some confusion remains after the OGSI era**
  - Many projects sidestepped this by using "pure" WS
- **Globus Toolkit 4 has been released**

GT4-view of OGSA and WSRF -1

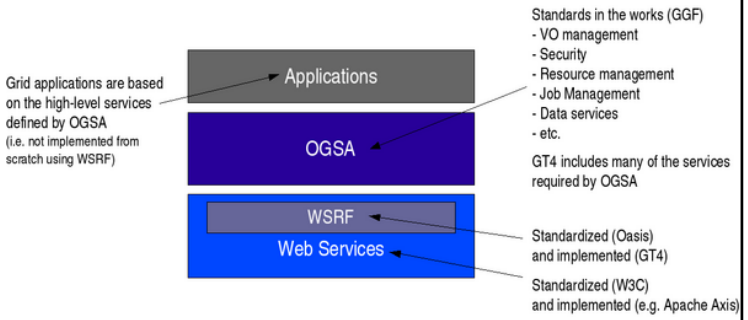


Diagram from Globus Alliance

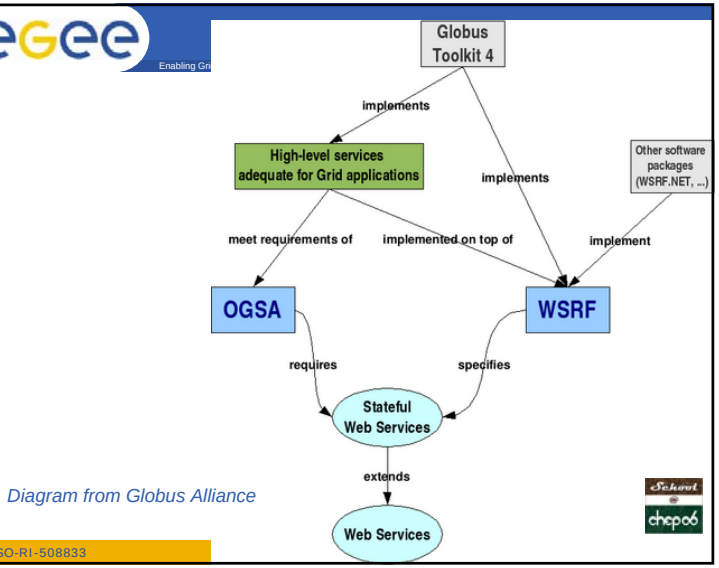


Diagram from Globus Alliance

GT4 and VDT

- **VDT: Virtual Data Toolkit**
  - ensemble of grid middleware that can be easily installed and configured
  - Been used by LCG and EGEE with GT2, Condor, MyProxy,…
- **Pre-requisite for using GT4 in gLite and other production grids has been achieved:**
- **"VDT 1.3.7 introduces the Globus Toolkit 4.0 (GT4) series – both pre-web services and some web services."**

http://vdt.cs.wisc.edu/globus_3.2_vs_4.0.html

Further reading

- The Grid Core Technologies, Maozhen Li and Mark Baker, Wiley, 2005
- The Globus Toolkit 4 Programmer's Tutorial
  Borja Sotomayor, Globus Alliance,
  http://gdp.globus.org/gt4-tutorial/multiplehtml/index.html
- The Web Services Grid Architecture (WSGA)
  www.nesc.ac.uk/technical_papers/UKeS-2004-05.pdf
- http://java.sun.com/xml/webservices.pdf

Summary

- Current way people try to create grid middleware is using Service Oriented Architectures based on WS
- An abundance of standards is en route
  - Extensions to manage resources are in WS-RF framework
  - Workflow – service composition
  - Also portals/portlets to expose services
- Initial implementation based on WS-RF and OGSA is in Globus Toolkit 4
- We'll soon have experience to test the perception that this is the way to go!