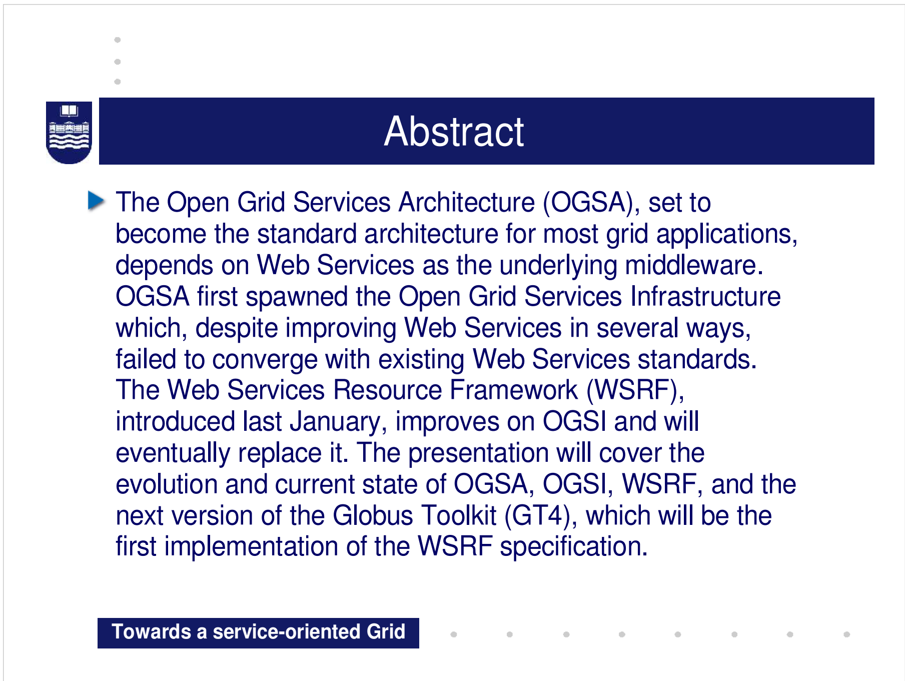
OGSA/OGSI, WSRF & Globus

Towards a service-oriented Grid

Borja Sotomayor

June 16th 2004



Abstract

- The Open Grid Services Architecture (OGSA), set to become the standard architecture for most grid applications, depends on Web Services as the underlying middleware. OGSA first spawned the Open Grid Services Infrastructure which, despite improving Web Services in several ways, failed to converge with existing Web Services standards. The Web Services Resource Framework (WSRF), introduced last January, improves on OGSI and will eventually replace it. The presentation will cover the evolution and current state of OGSA, OGSI, WSRF, and the next version of the Globus Toolkit (GT4), which will be the first implementation of the WSRF specification.

Contents

- OGSA (briefly)
- Web Services and Grid Services
- Where does GT3 fit in?
- WSRF and GT4

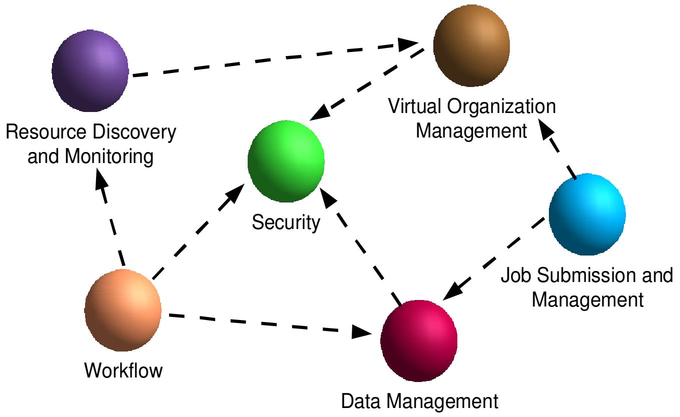
Contents

- OGSA (briefly)
- Web Services and Grid Services
- Where does GT3 fit in?
- WSRF and GT4

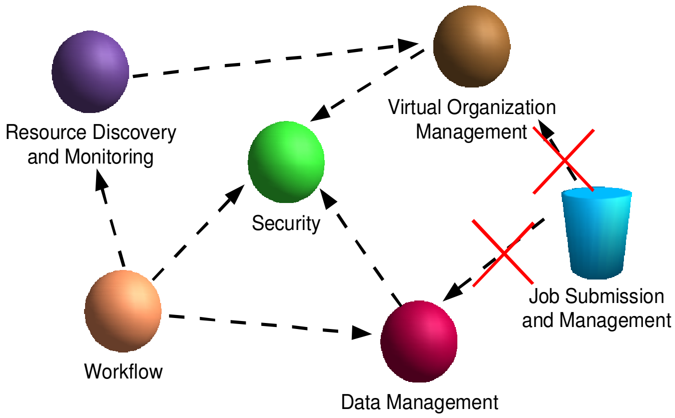
OGSA (I)

A Grid systems is built with several different subsystems.
For example...

OGSA (II)


Resource Discovery and Monitoring
Virtual Organization Management
Security
Job Submission and Management
Workflow
Data Management

OGSA (III)


Resource Discovery and Monitoring
Virtual Organization Management
Security
Job Submission and Management
Workflow
Data Management

OGSA (IV)

- The *Open Grid Services Architecture* (OGSA) aims to *standardize* the different *services* that are commonly found in a Grid application.
  - Job management, security, VO management, data management, workflow, deployment, etc.
- OGSA was introduced in the classic paper “Physiology of the Grid” (I.Foster, C.Kesselman, J.Nick, y S.Tuecke)
- The actual standardization process is being carried out by the Global Grid Forum (GGF)
  - http://www.ggf.org/

OGSA (V)

- What exactly does OGSA define?
  - “A set of core interfaces and behaviours that address key concerns in Grid systems.”
    - Each particular service can be accessed in the same fashion, regardless of vendor, organization, internal implementation, etc.
  - Still a work in progress!

OGSA (VI)

- What underlies this?
  - How are the interfaces defined?
  - How does a client request a particular operation from a service?
- OGSA relies on (and aims to improve) *web services* technologies
  - Broader context: Service-Oriented Architectures (SOA)

Contents

- OGSA (briefly)
- Web Services and Grid Services
- Where does GT3 fit in?
- WSRF and GT4

Web Services (I)

- What are Web Services?
  - *Yet another* distributed middleware technology (just like CORBA, RMI, EJBs, etc.)
- What sets it apart:
  - Clean separation between *interface* (*what* the service does) and *implementation* (*how* it does it).
  - Based on standard interoperable languages (XML).
  - Widespread use and abundant software available

Web Services (II)

- Separation of interface and implementation

Interface	A web service 'publishes' the operations it is capable of performing through its interface. The interface is written in a specific language.
Implementation	The implementation and the interface are kept separate. The implementation is done using a programming language (Java, C++, ...)
Running Environment	Both the implementation and the interface are placed in a running environment, which is in charge of handling all incoming calls, crafting the responses, etc.

Web Services (III)

- Separation of interface and implementation
  - Interface is defined in an XML language called WSDL (Web Services Description Language).
  - WSDL is language and platform-neutral, and allows the interface to be defined separately from the particular transport protocol or data encoding used in the actual message passing.
    - e.g. A GNU/Linux client can access a web service in a Windows server using protocol A, while a different client using Solaris might use protocol B.

Web Services (IV)

- We can use web services to build SOAs (*Service-Oriented Architectures*)
  - Architecture in which an application uses several independent services (or *loosely coupled* services) that cooperate to undertake a common task.
    - “loosely coupled”: A change in the implementation of one service doesn't affect the other services.
  - SOAs improve interoperability and reusability.

Web Services (V)

- Why are Web Services ideal for OGSA and, in general, for Grid systems?
  - Can easily cope with heterogeneous systems (different programming languages, platforms, etc.)
  - Common interface language (WSDL) allows:
    - Virtualization: from a pool of services (with the same interface) I can access any service in the same fashion regardless of vendor, platform, etc.
    - Dynamic service indexing and discovery.
    - Dynamic access to services.

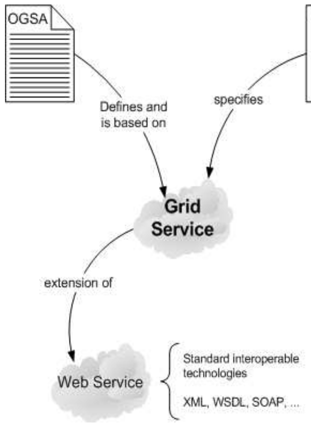
Grid Services (I)

- Despite being the best option for OGSA, Web Services still have important disadvantages:
  - Stateless
  - Non-transient ('Persistent')
  - No supporting services (notifications, lifecycle management, etc.)
- OGSA defines the concept of a *Grid Service* to overcome these limitations.
  - Grid Service = *improved* Web Service
  - Grid Services are compatible with WS

Grid Services (II)

- OGSA is based on Grid Services
  - Grid Services are *specified* in a separate spec spawned by OGSA: Open Grid Services Infrastructure (OGSI).
  - All the services in OGSA (Job management, security, etc.) will be based on (and implemented with) Grid Services.

Grid Services (III)



OGSA

OGSI

Defines and is based on

specifies

Grid Service

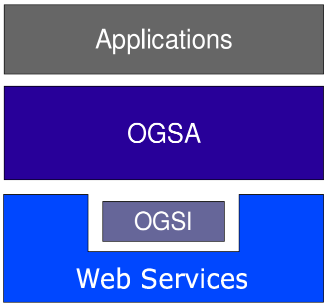
extension of

Web Service

Standard interoperable technologies

XML, WSDL, SOAP, ...

Grid Services (IV)

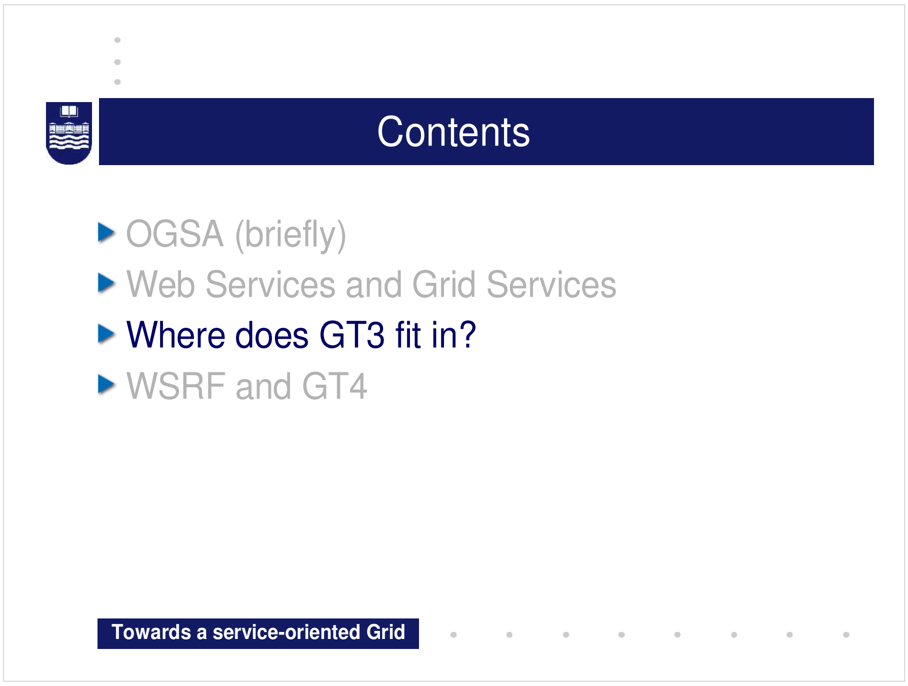


Applications

OGSA

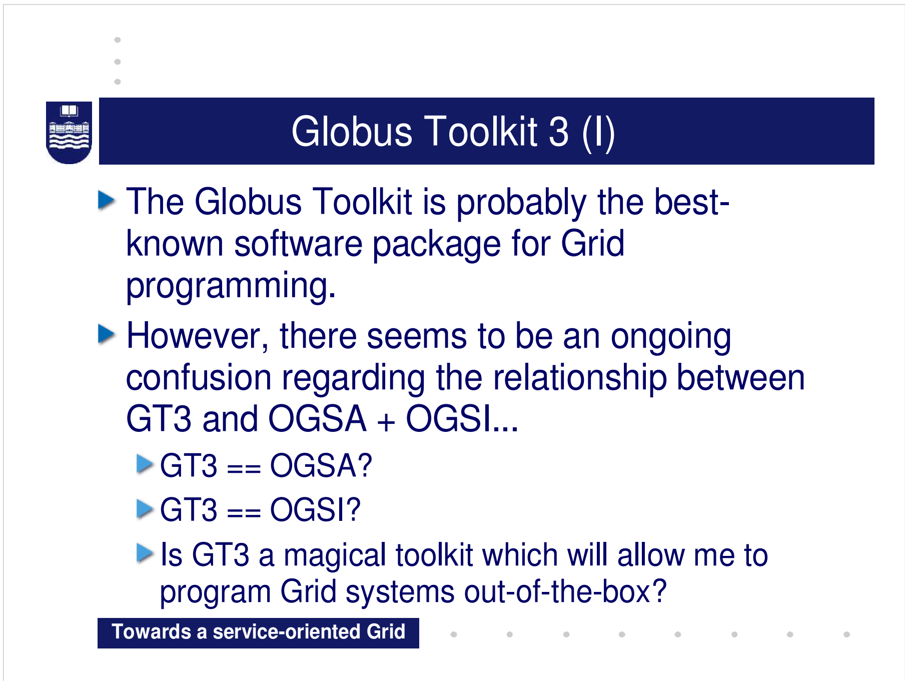
OGSI

Web Services



Contents

- OGSA (briefly)
- Web Services and Grid Services
- Where does GT3 fit in?
- WSRF and GT4



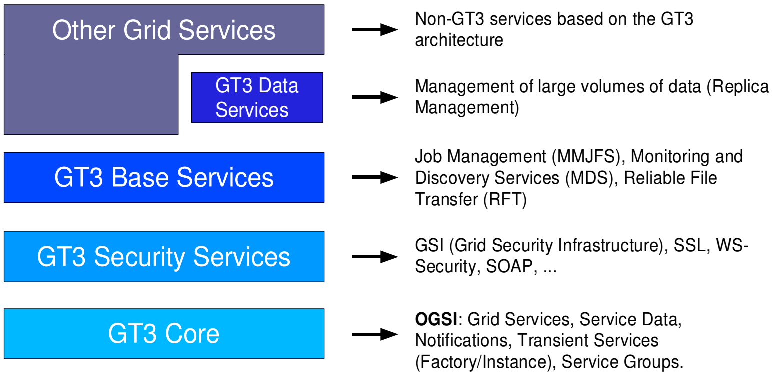
Globus Toolkit 3 (I)

- The Globus Toolkit is probably the best-known software package for Grid programming.
- However, there seems to be an ongoing confusion regarding the relationship between GT3 and OGSA + OGSI...
  - GT3 == OGSA?
  - GT3 == OGSI?
  - Is GT3 a magical toolkit which will allow me to program Grid systems out-of-the-box?

Globus Toolkit 3 (II)

- The Globus Toolkit 3 (GT3) is a complete implementation, in Java, of the OGSI spec.
  - Not the only implementation out there, but the most complete one.
- Very important: GT3 isn't *only* an OGSI implementation.
  - Includes OGSA-defined services
  - Includes its own non-OGSA services (which have become a *de facto* standard)
  - Also includes non-WS components (GT2.4)

GT3 Architecture



Other Grid Services → Non-GT3 services based on the GT3 architecture

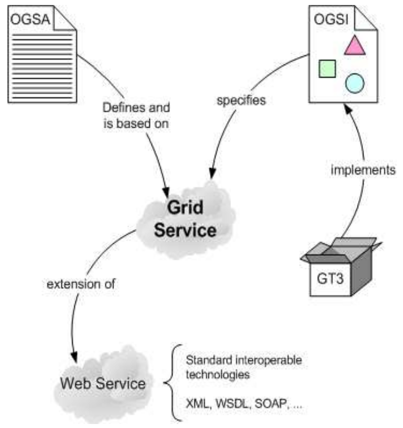
GT3 Data Services → Management of large volumes of data (Replica Management)

GT3 Base Services → Job Management (MMJFS), Monitoring and Discovery Services (MDS), Reliable File Transfer (RFT)

GT3 Security Services → GSI (Grid Security Infrastructure), SSL, WS-Security, SOAP, ...

GT3 Core → **OGSI**: Grid Services, Service Data, Notifications, Transient Services (Factory/Instance), Service Groups.

OGSA, OGSI, and GT3 (I)



OGSA

Defines and is based on

OGSI

specifies

implements

Grid Service

GT3

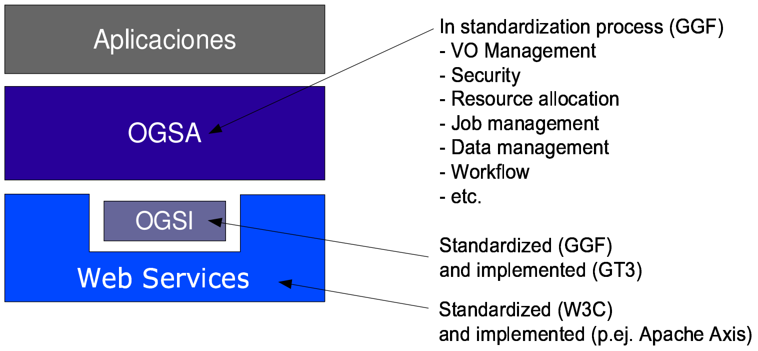
extension of

Web Service

Standard interoperable technologies

XML, WSDL, SOAP, ...

OGSA, OGSI, and GT3 (II)



Aplicaciones

OGSA

OGSI

Web Services

In standardization process (GGF)
- VO Management
- Security
- Resource allocation
- Job management
- Data management
- Workflow
- etc.

Standardized (GGF)
and implemented (GT3)

Standardized (W3C)
and implemented (p.ej. Apache Axis)

Not the panacea...

- Bottom line: GT3 can be an essential part of a Grid system, but not the *only* component.
  - GT3 covers a lot of the work, but each individual project will need to implement its own services, integrate with existing applications, etc.
  - Not a turnkey solution.
  - GT3 is for programmers, not users.
  - Highly recommended:
    - “Putting the Globus Toolkit in its Place” http://www-unix.mcs.anl.gov/~liming/ggf11/

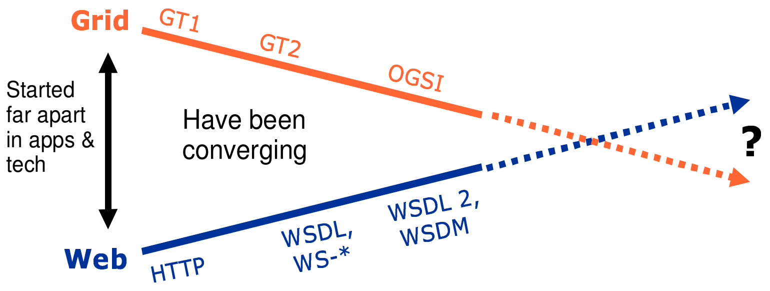
Contents

- OGSA (briefly)
- Web Services and Grid Services
- Where does GT3 fit in?
- WSRF and GT4

WSRF (I)

- Despite being generally accepted as a good change within the Grid community, OGSI has not caught the eye of the Web Service community.
  - GGF had hoped that Web Services standards and OGSI would converge, but that convergence is not happening.

WSRF (II)



Grid

GT1

GT2

OGSI

Started far apart in apps & tech

Have been converging

?

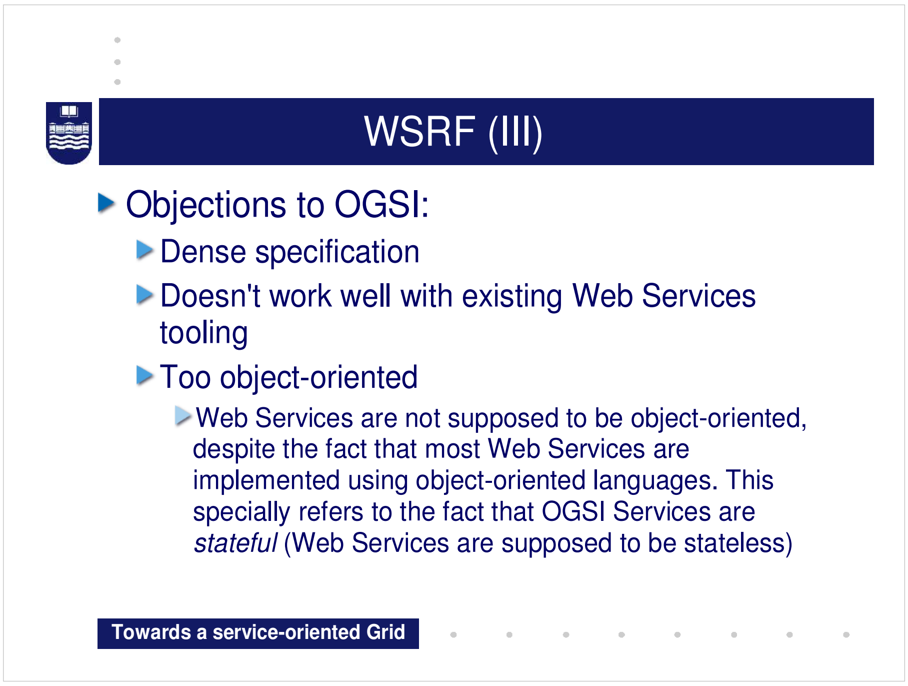
Web

HTTP

WSDL, WS-*

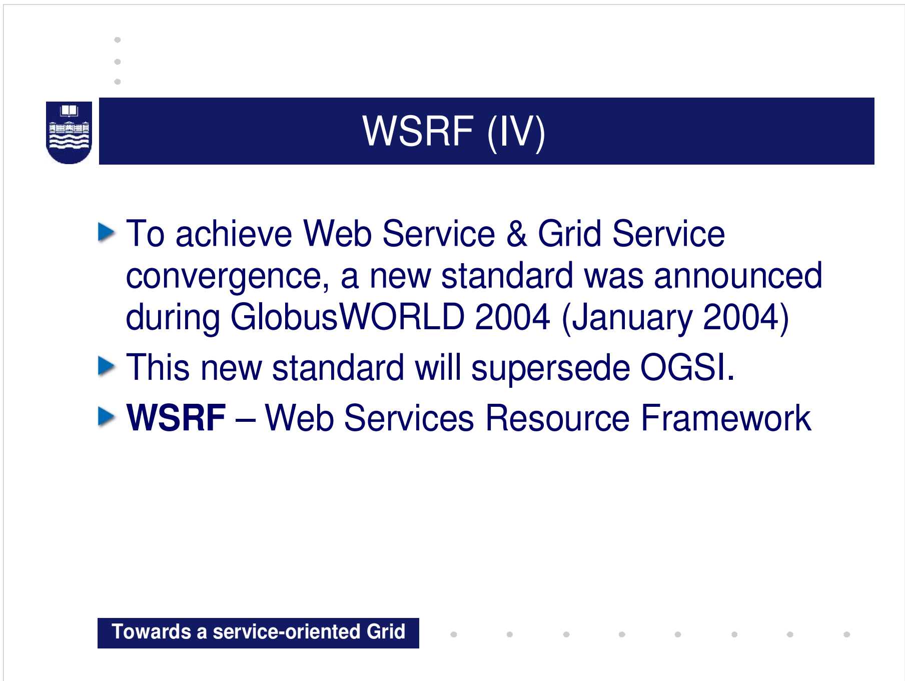
WSDL 2, WSDM

Based on Ian Foster's slides in GlobusWorld 2004



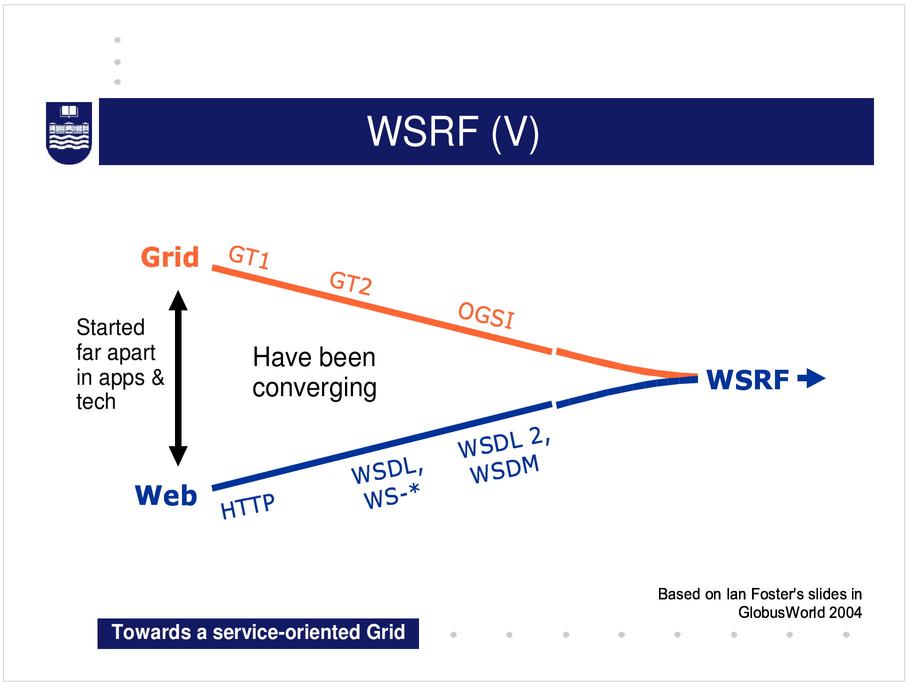
WSRF (III)

- Objections to OGSI:
  - Dense specification
  - Doesn't work well with existing Web Services tooling
  - Too object-oriented
    - Web Services are not supposed to be object-oriented, despite the fact that most Web Services are implemented using object-oriented languages. This specially refers to the fact that OGSI Services are *stateful* (Web Services are supposed to be stateless)



WSRF (IV)

- To achieve Web Service & Grid Service convergence, a new standard was announced during GlobusWORLD 2004 (January 2004)
- This new standard will supersede OGSI.
- **WSRF** – Web Services Resource Framework



WSRF (V)

Grid

GT1

GT2

OGSI

Started far apart in apps & tech

Have been converging

WSRF

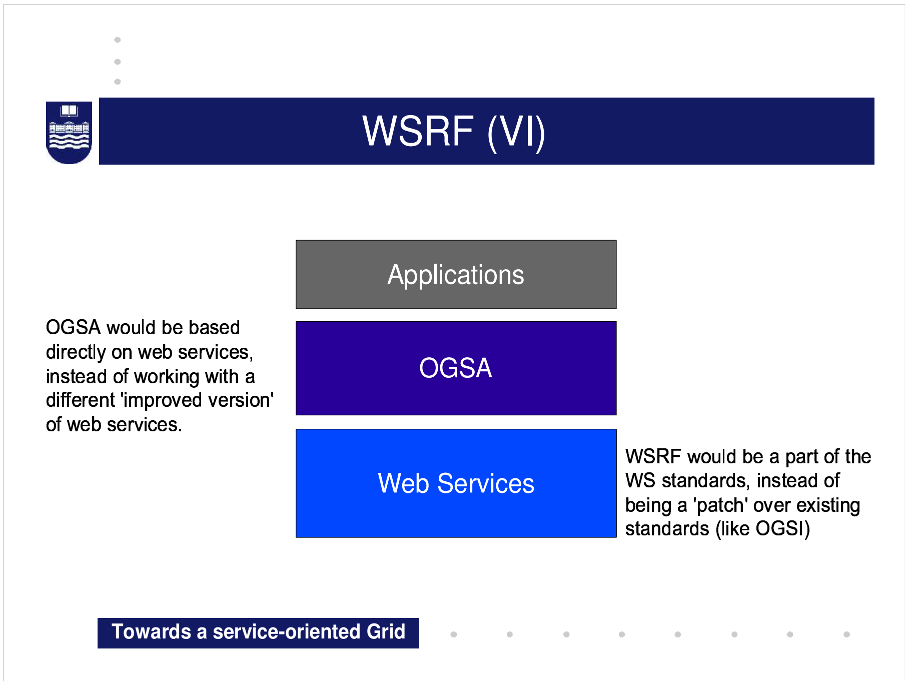
Web

HTTP

WSDL, WS-*

WSDL 2, WSDM

Based on Ian Foster's slides in GlobusWorld 2004



WSRF (VI)

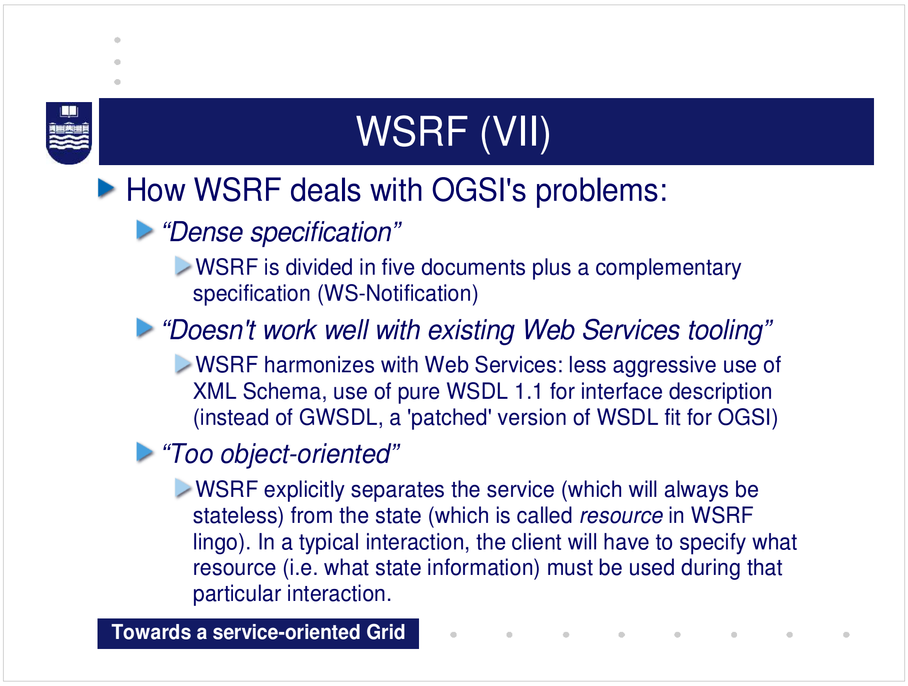
OGSA would be based directly on web services, instead of working with a different 'improved version' of web services.

Applications

OGSA

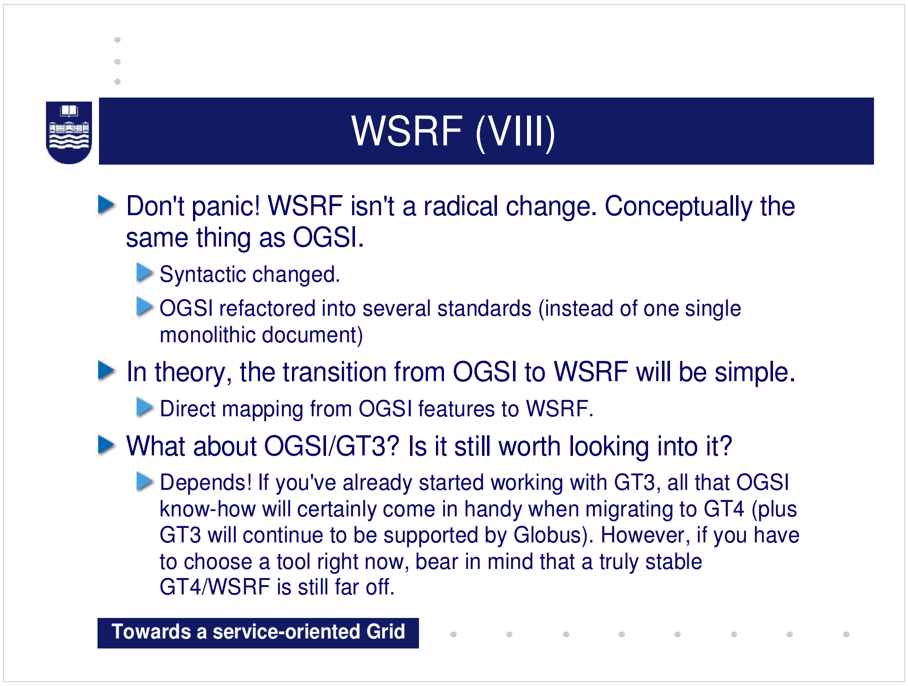
Web Services

WSRF would be a part of the WS standards, instead of being a 'patch' over existing standards (like OGSI)



WSRF (VII)

- How WSRF deals with OGSI's problems:
  - *“Dense specification”*
    - WSRF is divided in five documents plus a complementary specification (WS-Notification)
  - *“Doesn't work well with existing Web Services tooling”*
    - WSRF harmonizes with Web Services: less aggressive use of XML Schema, use of pure WSDL 1.1 for interface description (instead of GWSDL, a 'patched' version of WSDL fit for OGSI)
  - *“Too object-oriented”*
    - WSRF explicitly separates the service (which will always be stateless) from the state (which is called *resource* in WSRF lingo). In a typical interaction, the client will have to specify what resource (i.e. what state information) must be used during that particular interaction.



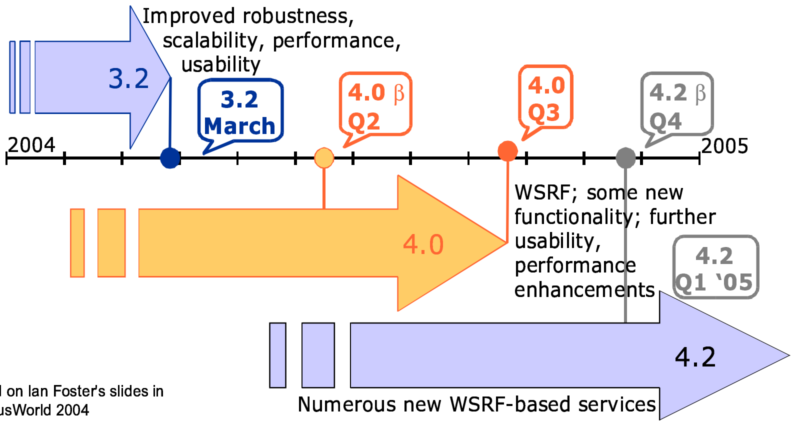
WSRF (VIII)

- Don't panic! WSRF isn't a radical change. Conceptually the same thing as OGSI.
  - Syntactic changed.
  - OGSI refactored into several standards (instead of one single monolithic document)
- In theory, the transition from OGSI to WSRF will be simple.
  - Direct mapping from OGSI features to WSRF.
- What about OGSI/GT3? Is it still worth looking into it?
  - Depends! If you've already started working with GT3, all that OGSI know-how will certainly come in handy when migrating to GT4 (plus GT3 will continue to be supported by Globus). However, if you have to choose a tool right now, bear in mind that a truly stable GT4/WSRF is still far off.

WSRF/GT4

- Why is WSRF more likely to succeed?
  - Harmonizes with Web Services
  - Backed by IBM and HP
- First, full, free implementation: GT4
  - IBM will include support for WSRF in some of its tools.
- More WSRF details: http://www.globus.org/wsrf

GT3/GT4 Evolution



Improved robustness, scalability, performance, usability

3.2

3.2 March

4.0 β Q2

4.0 Q3

4.2 β Q4

2004

2005

4.0

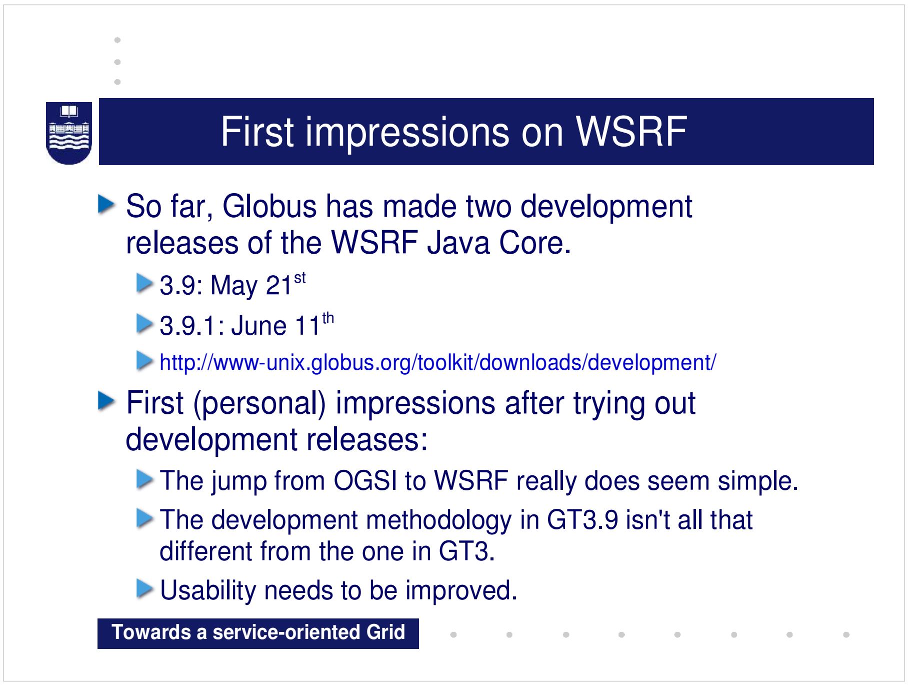
WSRF; some new functionality; further usability, performance enhancements

4.2 Q1 '05

4.2

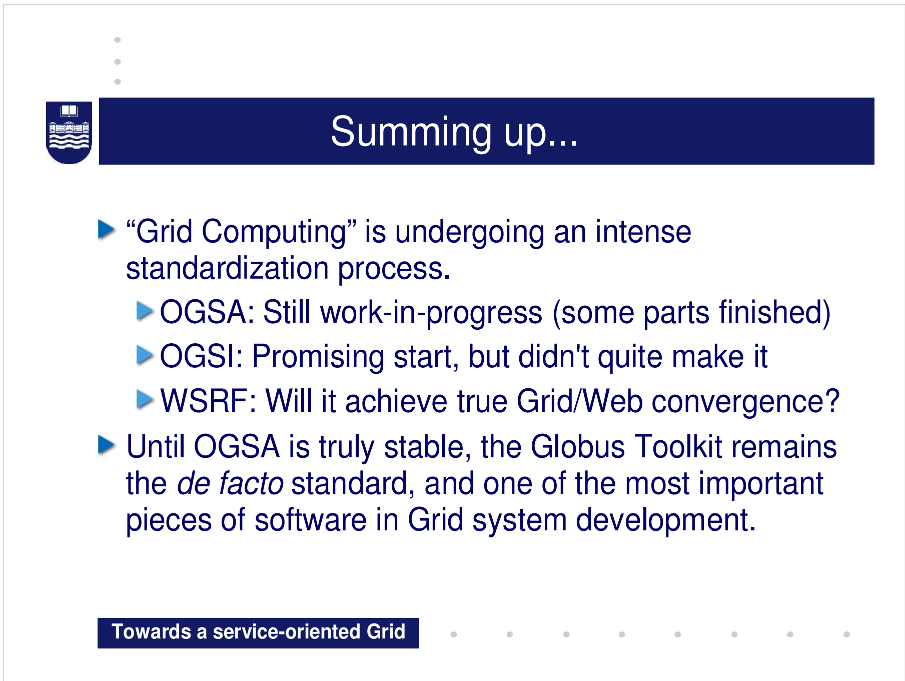
Numerous new WSRF-based services

Based on Ian Foster's slides in GlobusWorld 2004



First impressions on WSRF

- So far, Globus has made two development releases of the WSRF Java Core.
  - 3.9: May 21st
  - 3.9.1: June 11th
  - http://www-unix.globus.org/toolkit/downloads/development/
- First (personal) impressions after trying out development releases:
  - The jump from OGSI to WSRF really does seem simple.
  - The development methodology in GT3.9 isn't all that different from the one in GT3.
  - Usability needs to be improved.



Summing up...

- "Grid Computing" is undergoing an intense standardization process.
  - OGSA: Still work-in-progress (some parts finished)
  - OGSI: Promising start, but didn't quite make it
  - WSRF: Will it achieve true Grid/Web convergence?
- Until OGSA is truly stable, the Globus Toolkit remains the *de facto* standard, and one of the most important pieces of software in Grid system development.

Questions?

Borja Sotomayor

Faculty of Engineering - ESIDE

University of Deusto

bsotomay@eside.deusto.es