

**Mesa Redonda sobre Computación Grid
Universidad de Extremadura
5 de noviembre de 2007**



Diseño de Infraestructuras Grid



UNIVERSIDAD COMPLUTENSE
MADRID

Ignacio Martín Llorente
(llorente@dacya.ucm.es)
Grupo de Arquitectura de Sistemas Distribuidos
Dpto. de Arquitectura de Computadores y Automática
Facultad de Informática
Universidad Complutense de Madrid

- Analizar los objetivos y características de un arquitectura Grid
- Identificar los componentes principales de un Grid computacional
- Estudiar las arquitecturas más comunes que se encuentran en las plataformas actuales
- Caso de Estudio: GRIDIMadrid



1. Recursos Computacionales
2. Middleware Grid
3. Arquitectura de Infraestructuras Grid Computacionales
4. GRIDIMadrid

1. Recursos Computacionales

1.1. Computación Paralela y Distribuida

Objetivo de la Computación Paralela y Distribuida

- Ejecución **eficiente** de aplicaciones intensivas en cálculo o datos

Paradigmas de Computación Paralela y Distribuida

High Performance Computing (HPC)

- Reducción del tiempo de ejecución de una aplicación paralela (memoria compartida o distribuida)
- Rendimiento del sistema en FLOPS
- Ejemplos: CFD, modelos meteorológicos

High Throughput Computing (HTC)

- Aumentar el número de ejecuciones por unidad de tiempo
- Rendimiento del sistema en trabajos por segundo, o FLOPY
- Ejemplos: Física de Altas Energías, Bioinformática, Modelos Financieros

1.2. Tipos de Plataformas

Centralizada Acoplada

- Red de Interconexión
- Administración
- Homogeneidad

Descentralizada Desacoplada

SMP (Symmetric
Multi-processors)



MPP (Massive
Parallel Processors)



Clusters



**Network Systems
Intranet/Internet**



High Performance Computing

High Throughput Computing

1. Recursos Computacionales

1.3. Gestores Locales de Recursos

Sistemas Locales para la Gestión de Recursos (LRMS)

- Ofrecen una visión uniforme y única del sistema
- Gestión de los trabajos, optimiza el uso (carga...)
- Habitualmente asumen acceso exclusivo a los recursos
 - ① Sistemas de colas *batch* para servidores HPC
 - ② Gestores de recursos en clusters dedicados
 - ③ Gestores de carga para sistemas en red

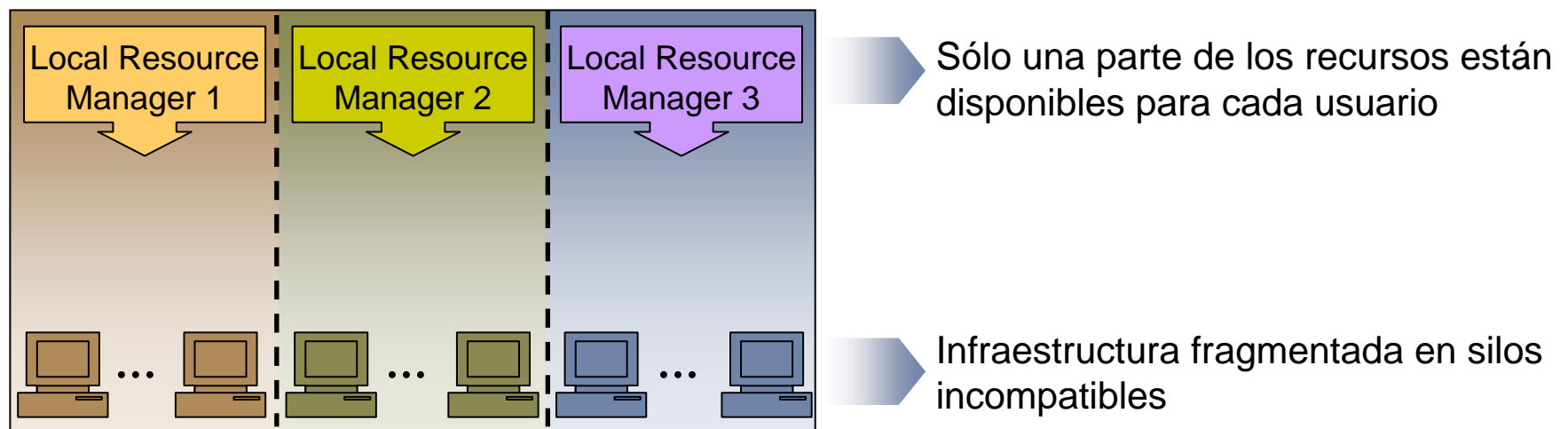
<i>Independent Suppliers</i>	<i>Open Source</i>	<i>OEM Proprietary</i>
② <i>Platform Computing</i> ③ LSF	② <i>Altair</i> Open PBS	① <i>IBM</i> Load Leveler
② <i>Altair</i> PBS Pro	② <i>University of Wisconsin</i> ③ Condor	① <i>Cray</i> NQE
	② <i>Sun Microsystems</i> ③ SGE	

1. Recursos Computacionales

1.3. Gestores Locales de Recursos

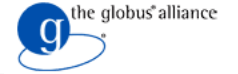
Desventajas de los Gestores Locales

- No disponen de interfaces ni infraestructuras de seguridad comunes
- Basados en protocolos propietarios
- **Silos computacionales incompatibles en una misma organización**
 - Habilidades de administración específicas
 - Aumenta el coste operacional
 - Sobre-aprovisionamiento y desequilibrio global de la carga
- Imposibilidad de construir Infraestructuras computacionales que abarcan varios dominios de administración (políticas de seguridad y LRMS)





2. Middleware Grid



1. Recursos Computacionales
2. Middleware Grid
3. Arquitectura de Infraestructuras Grid Computacionales
4. GRIDIMadrid

2.1. Integración de Diferentes Dominios de Administración

Un Nuevo Problema ...

“The real and specific problem that underlies the Grid concept is coordinated resource **sharing** and problem solving in dynamic, multi-institutional **virtual organizations**.”

*Ian Foster, Carl Kesselman, Steven Tuecke
The Anatomy of the Grid. Enabling Scalable Virtual Organizations (2001)*

sharing ...

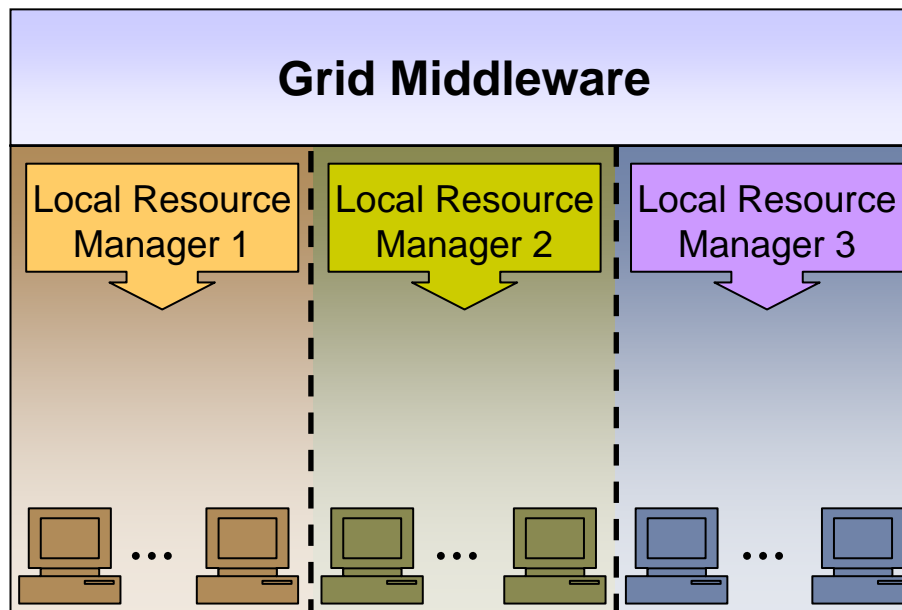
- de **recursos** de naturaleza dispar (software, hardware, datos...)
- entre entidades arbitrarias y dinámicamente (**interoperabilidad**)
- sobre **protocolos** comunes, un Grid es una arquitectura de protocolos
- conjunto de individuos y/o instituciones que colaboran compartiendo los recursos para alcanzar objetivos comunes (VO)

2.1. Integración de Diferentes Dominios de Administración

"Any problem in computer science can be solved with another layer of indirection... But that usually will create another problem." David Wheeler

... Un Nuevo Nivel de Abstracción

“Un grid (*computacional*) ofrece una capa de abstracción (*middleware*) para integrar diferentes dominios de administración (plataformas y políticas heterogéneas) “

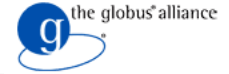


➔ **Interfaces Comunes:** Los usuarios pueden acceder un amplio conjunto (número y tipo) de recursos

➔ **Infraestructura:** Recursos computacionales y de almacenamiento, red y LRMS



2. Middleware Grid



2.1. Integración de Diferentes Dominios de Administración

Características de un Grid

“Un grid es un sistema que ...

1. coordina recursos que no están sujetos a un control centralizado ...
2. usando protocolos e interfaces estándar, abiertos y de propósito general ...
3. para proporcionar calidades de servicio no triviales.”

Ian Foster

What is the Grid? A Three Point Checklist (2002)

Un arquitectura Grid puede definirse de forma funcional mediante el conjunto de servicios que soportan las actividades de una VO (**OGSA**)

Ian Foster, Carl Kesselman, Jeffrey M. Nick y Steven Tuecke

“The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration”



2. Middleware Grid

2.1. Integración de Diferentes Dominios de Administración Grid Middleware (una visión computacional)

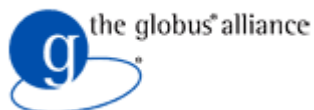
- **Servicios básicos**

- Seguridad
- Información y monitorización
- Gestión de datos
- Ejecución
- **Meta-planificación**

- **Distribuciones “Open-Source” de Middleware Grid**



- **Comunidades “Open-Source”**



The Globus Alliance (dev.globus.org)



2. Middleware Grid

2.2. Globus Toolkit

¿Por qué Globus?...

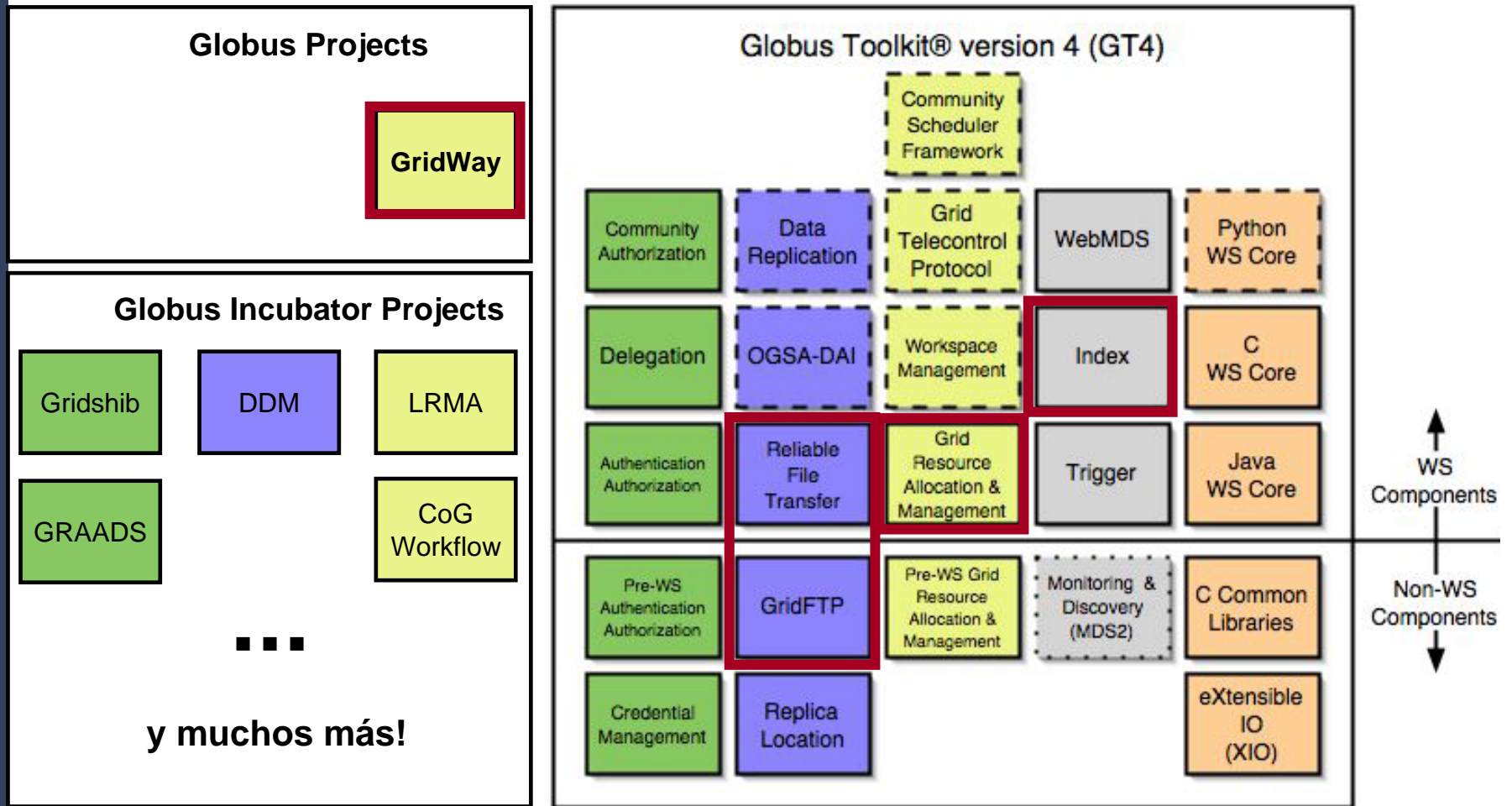
- **Proyecto “Open Community”** basado en el modelo Apache Jakarta:
 - El control de cada proyecto está en manos de los *comitters*
 - Infraestructura pública de desarrollo para cada proyecto: CVS, bugzilla, listas de correo, y Wiki
 - Cada proyecto sigue un proceso de *incubación* antes de ser un proyecto Globus
- **Globus Toolkit (GT)** integra una selección de proyectos Globus
- GT ofrece un conjunto de servicios para acceder de forma segura múltiples recursos en diferentes dominios de administración, cada uno con sus políticas y gestores locales



2. Middleware Grid

2.2. The Globus Toolkit

Componentes Globus



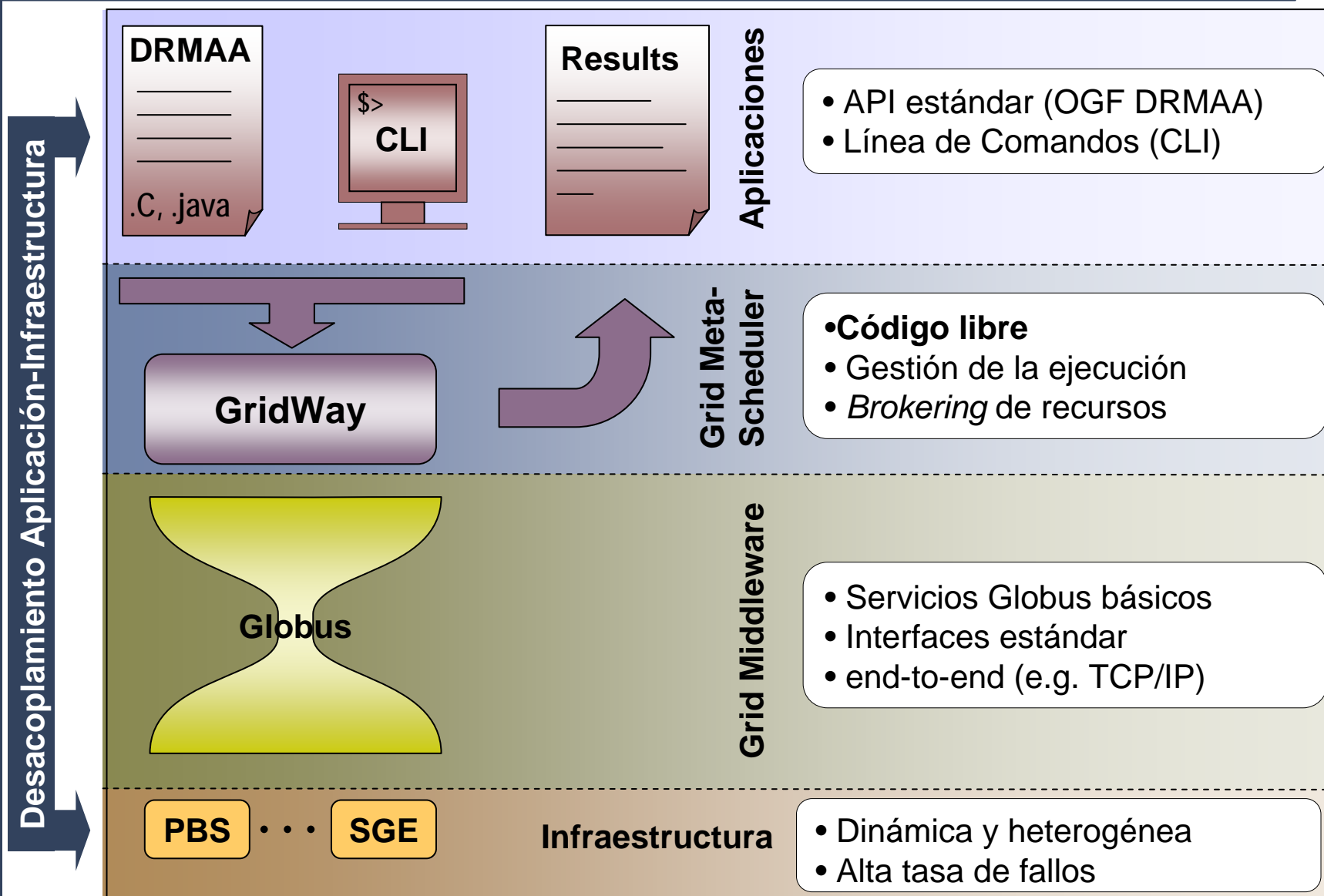
Componentes mínimos para un Grid computacional



2. Middleware Grid

2.3. GridWay Meta-scheduler

Arquitectura de un Grid Computacional





2. Middleware Grid

2.3. GridWay Meta-scheduler

Beneficios

Integración de plataformas no interoperables (Organización)

- Construcción de una única infraestructura flexible y uniforme
- Mayor utilización de recursos

Soporte para plataformas existentes y diferentes LRMS (Sys. Admin.)

- Asignación de recursos Grid en función de políticas locales de uso
- Análisis de las tendencias de uso de los recursos
- Monitorización del comportamiento de los usuarios

CLI familiares y APIs Estándar (End Users & Developers)

- Aplicaciones HTC
- Flujos de trabajo (Workflows)



2. Middleware Grid

2.3. GridWay Meta-scheduler

Características

Gestión de Carga de Trabajo

- Políticas de planificación avanzadas (específicas para Grid)
- Detección y recuperación de fallos
- Accounting
- Trabajos en array (barrido de parámetros) y DAG workflows

Interfaz de Usuario

- Estándares OGF: JSDL y DRMAA (C y JAVA)
- Interfaz de comandos similar al disponible en LRMS clásicos

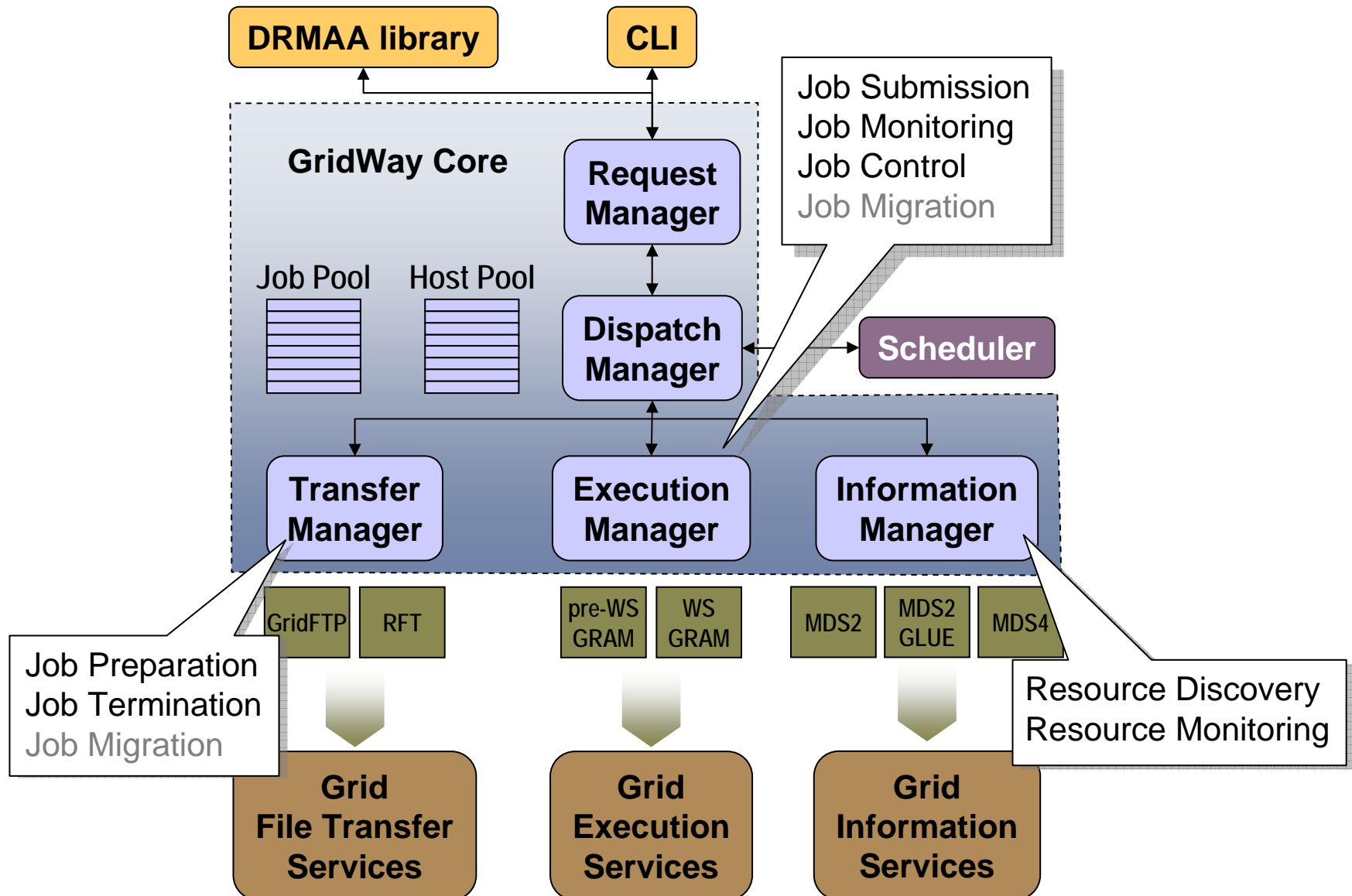
Integración

- Despliegue directo ya que sólo requiere servicios *estándar*
- Interoperabilidad entre diferentes infraestructuras



2. Middleware Grid

2.3. GridWay Meta-scheduler



2.4 Plataformas de Computación

**Centralizada
Acoplada**

- Red de Interconexión
- Administración
- Homogeneidad

**Descentralizada
Desacoplada**

SMP (Symmetric
Multi-processors)

MPP (Massive
Parallel Processors)

Clusters

**Network Systems
Intranet/Internet**

**Grid
Infrastructures**



High Performance Computing

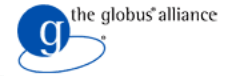
High Throughput Computing

1. Recursos Computacionales
2. Middleware Grid
3. Arquitectura de Infraestructuras Grid Computacionales
4. GRIDIMadrid

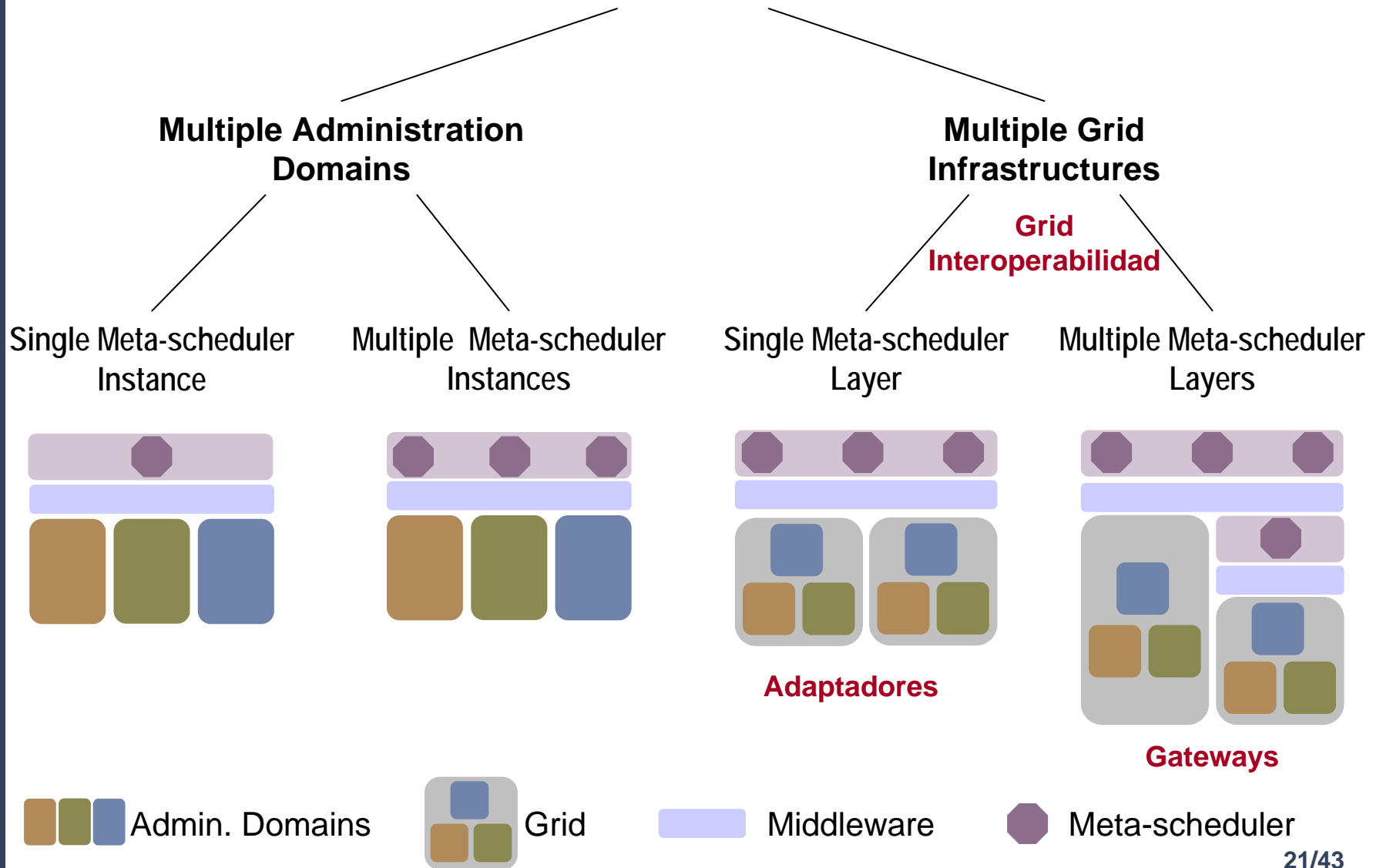


UNIVERSIDAD COMPLUTENSE
MADRID

3. Arquitectura de Infraestructuras Grid



Taxonomía



3.1. Multiple Administration Domains

Grids con Meta-planificador Único

Características

- Un sólo meta-planificador con acceso a varios dominios de administración
- Pequeña escala (campus o enterprise)
- Distribuida geográficamente en diferentes sitios

Objetivo y Beneficios

- Integración de múltiples dominios de administración en una infraestructura uniforme
- Aumento del retorno de la inversión en IT: minimización de costes y maximización de uso/rendimiento

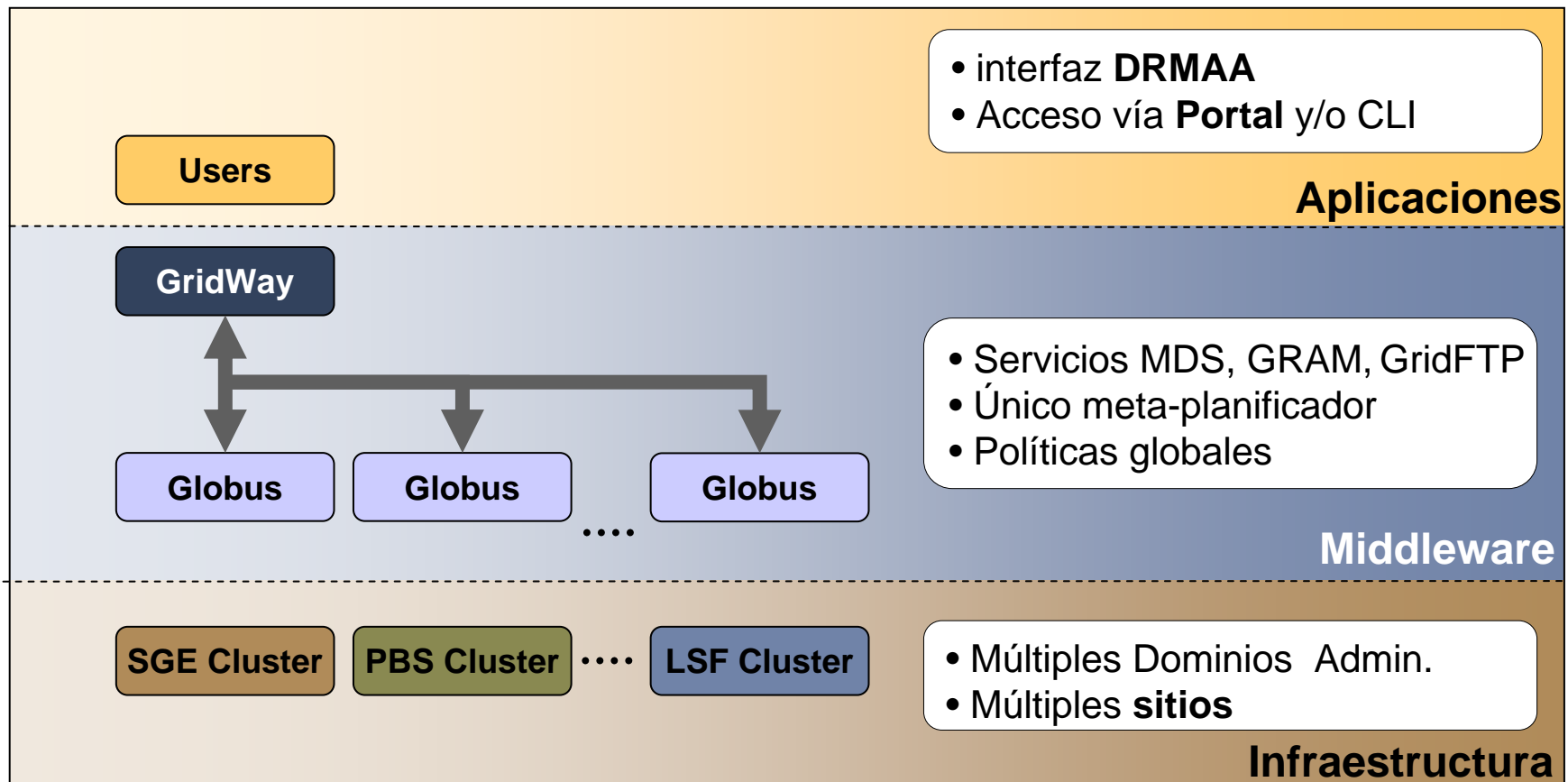
Planificación

- Meta-planificador centralizado
- Establecimiento de políticas para el Grid (e.g. uso de recursos)

3. Arquitectura de Infraestructuras Grid

3.1. Multiple Administration Domains

Grids con Meta-planificador Único con Globus





UNIVERSIDAD COMPLUTENSE
MADRID

3. Arquitectura de Infraestructuras Grid

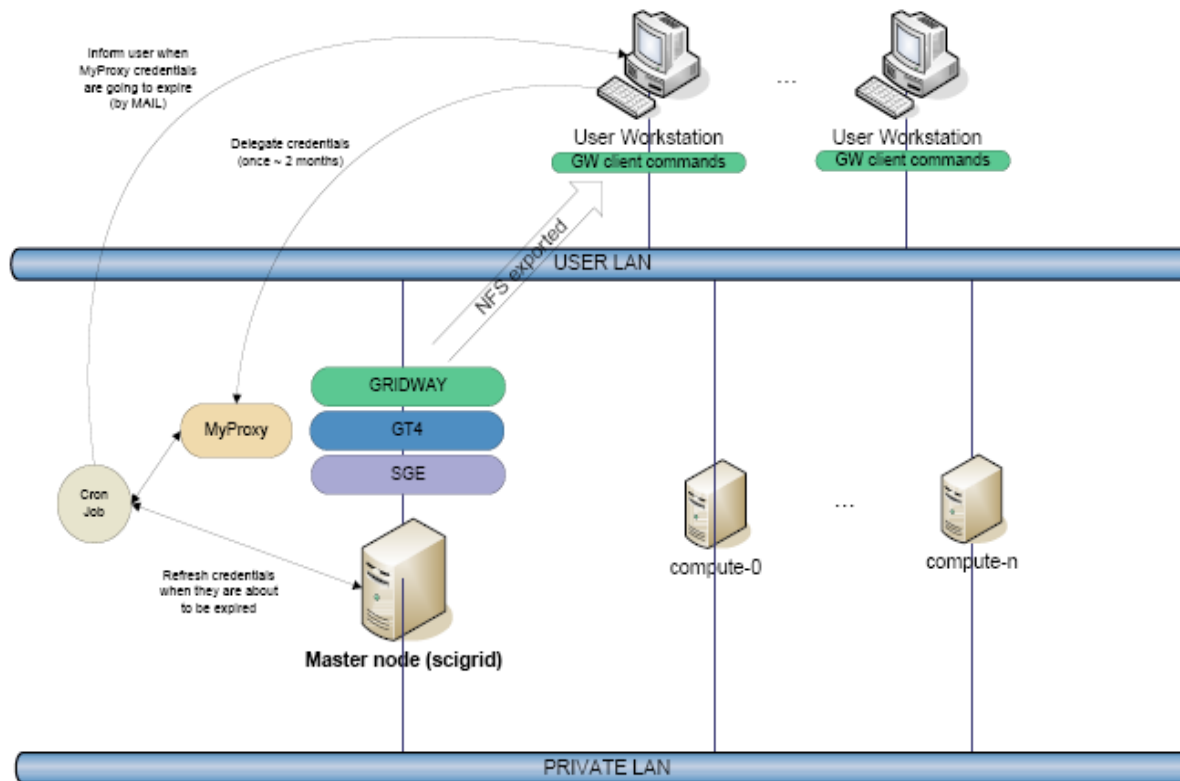
3.1. Multiple Administration Domains

Grids con Meta-planificador Único: Ejemplos

European Space Agency



- Meta-planificador para el sitio
- Un cluster 20 CPUs, 60 GB memoria principal
- Análisis de datos de las misiones espaciales (DRMAA)

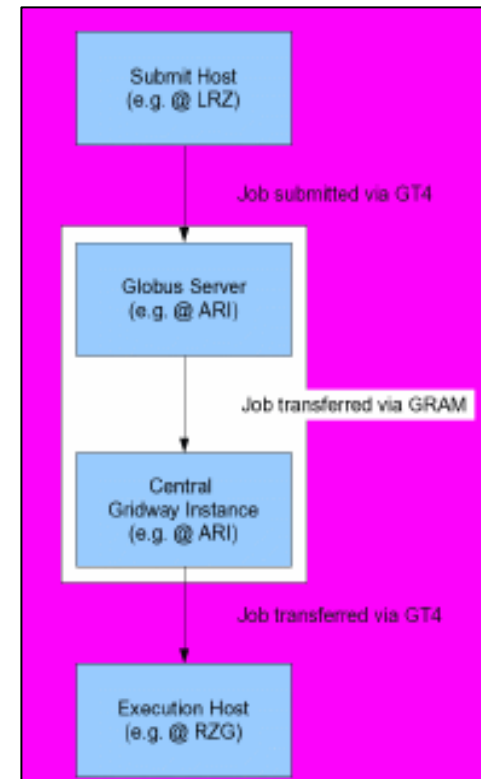
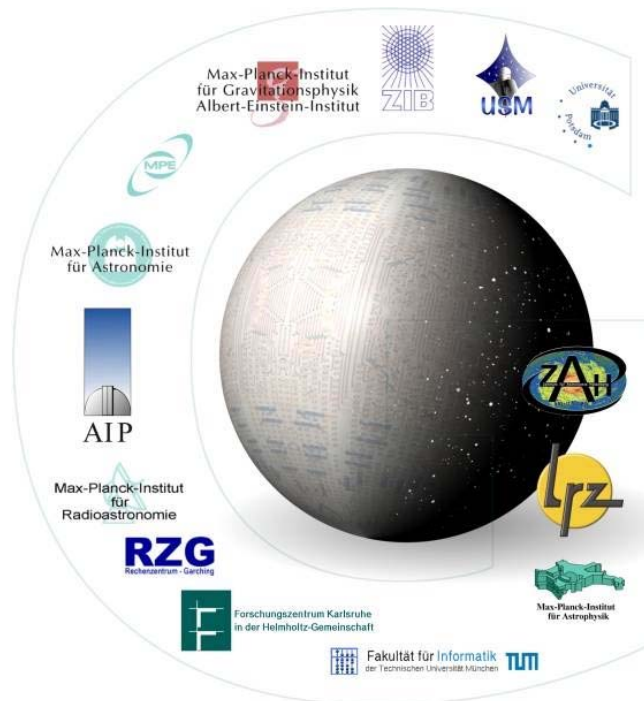


3.1. Multiple Administration Domains

Grids con Meta-planificador Único: Ejemplos

AstroGrid-D, German Astronomy Community Grid

- Meta-planificador para el Grid (interfaz GRAM)
- 22 resources @ 5 sites, 800 CPUs
- Gestión y uso conjunto de recursos computacionales y específicos de astronomía



3.1. Multiple Administration Domains

Grids con Múltiples Meta-planificadores

Características

- Múltiples meta-planificadores con acceso a múltiples dominios de administración (diferentes organizaciones – *partners*)
- Gran escala, débilmente acopladas
- Estructurada en organizaciones virtuales (VO)

Objetivos y Beneficios

- Compartición de recursos a gran escala, segura y fiable
- Soporte a proyectos conjuntos
- Acceso a una mayor potencia computacional para satisfacer demandas pico

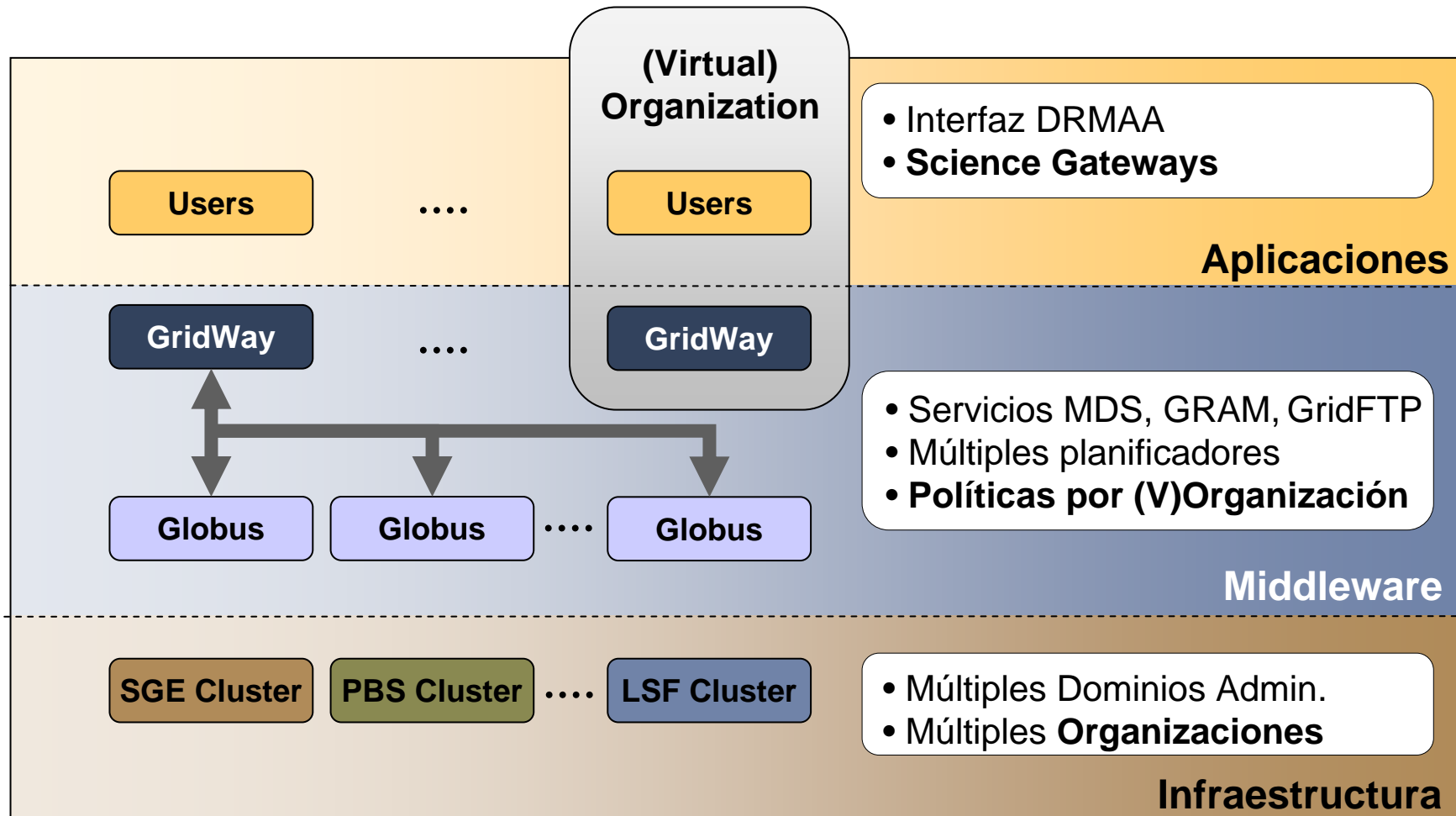
Planificación

- Sistema de planificación descentralizado
- Infraestructura dinámica, heterogénea, con fallos
- Autonomía de los integrantes, políticas para cada organización

3. Arquitectura de Infraestructuras Grid

3.1. Multiple Administration Domains

Grids con Múltiples Meta-planificadores con Globus



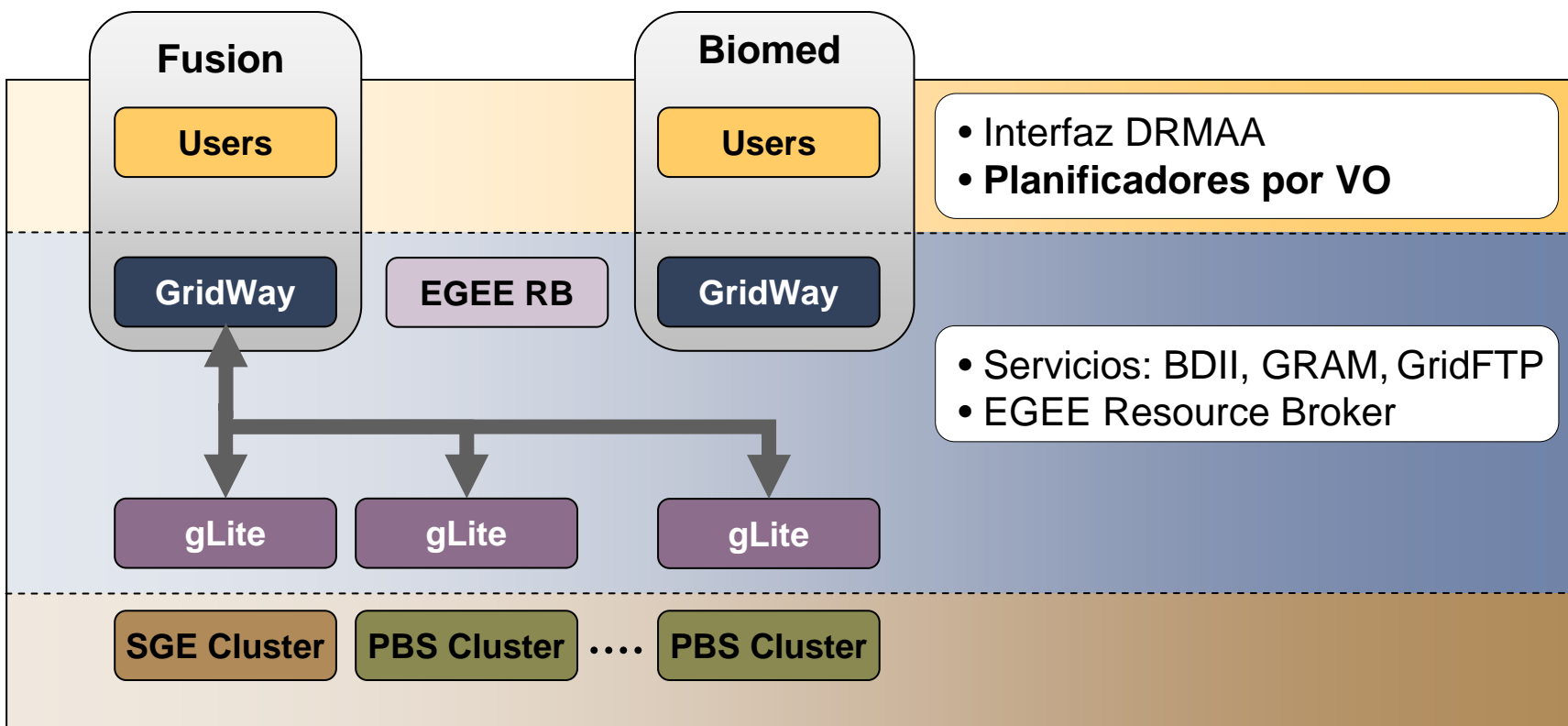
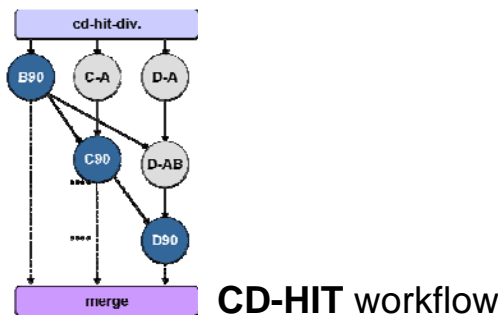
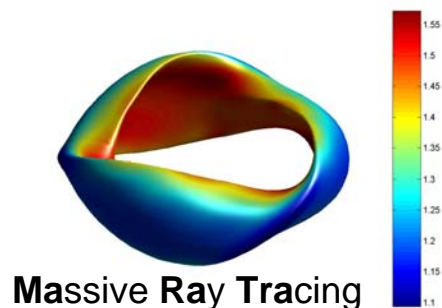


UNIVERSIDAD COMPLUTENSE
MADRID

3. Arquitectura de Infraestructuras Grid

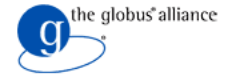
3.1. Multiple Administration Domains

Grids con Múltiples Meta-planificadores: Ejemplos





3. Arquitectura de Infraestructuras Grid



3.2. Multiple Grid Infrastructures

Grids con Capa de Meta-planificación Única

Características

- Única capa de planificación (uno o varios planificadores) con acceso *plano* a las infraestructuras Grid subyacentes
- Compuesto por Grids Múltiples Dominios de Administración
- Espacios de nombres globales (e.g. user DN's)
- Basados en middlewares diferentes

Objetivos y Beneficios

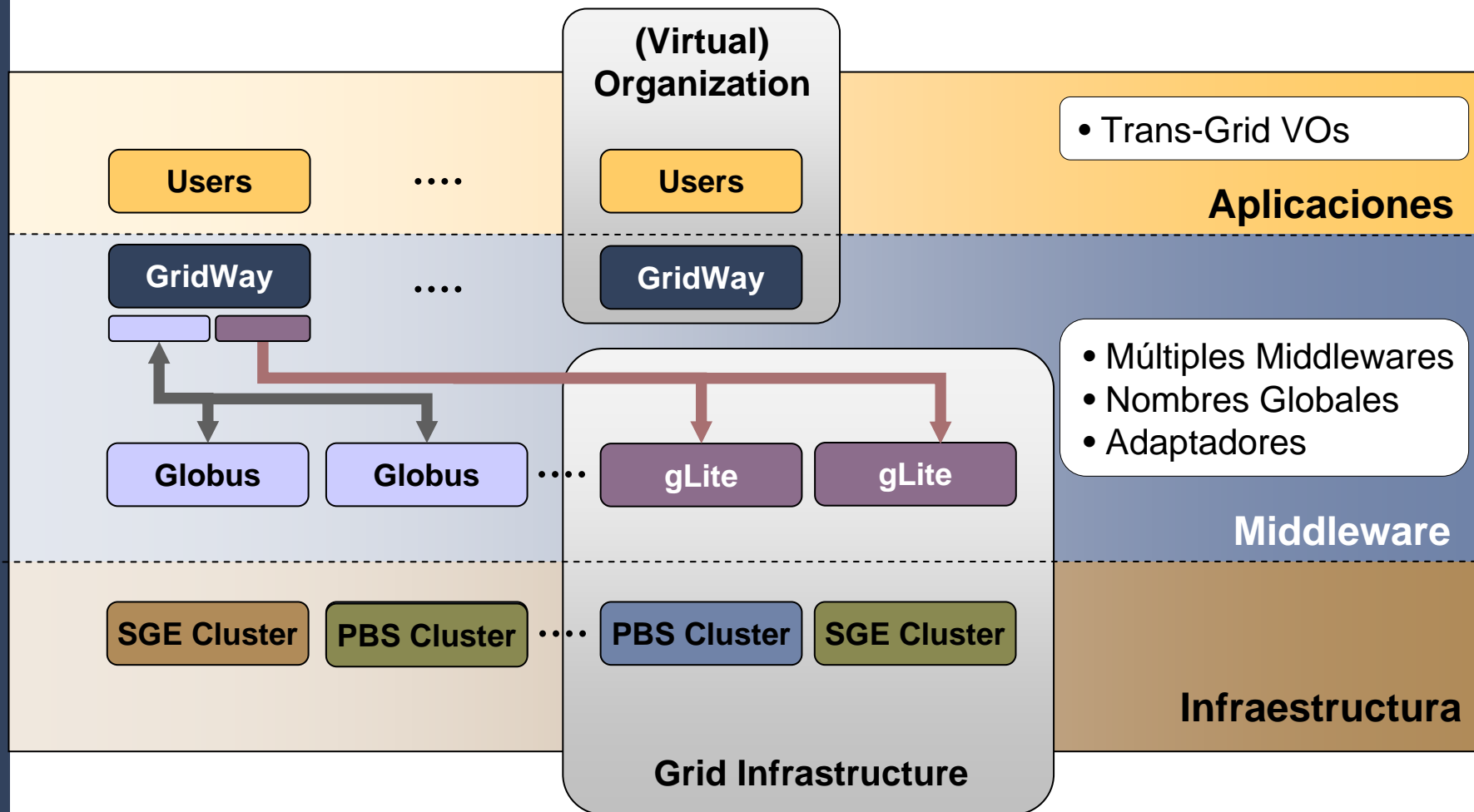
- Integrar múltiples Grids en una única infraestructura
- Colaboración entre trans-grid VOs
- Escalar los procedimientos de gestión, despliegue y administrativos

Planificación

- Políticas de planificación por organización adaptadas al Grid
- Uso de adaptadores para acceder a las distintas infraestructuras

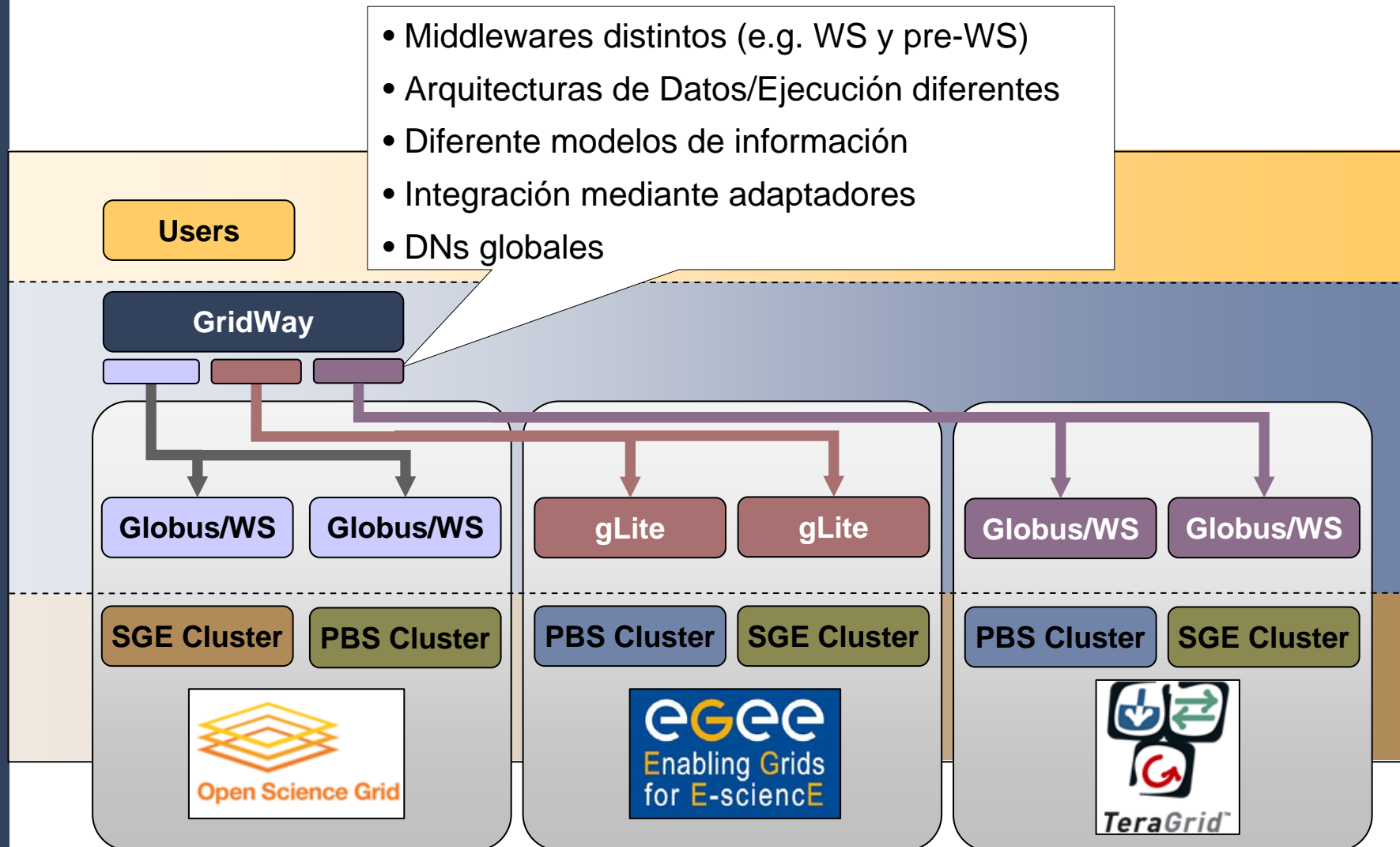
3.2. Multiple Grid Infrastructures

Grids con Capa de Meta-planificación Única con Globus



3.2. Multiple Grid Infrastructures

Grids con Capa de Meta-planificación Única : Ejemplos



3. Arquitectura de Infraestructuras Grid

3.2. Multiple Grid Infrastructures

Grids con Múltiples Capas de Meta-planificación

Características

- Múltiples capas de meta-planificación en una estructura jerárquica
- Provisión de recursos a modo *utility* (proveedor/consumidor)
- Uso de interfaces estándar para virtualizar un Grid

Objetivos y Beneficios

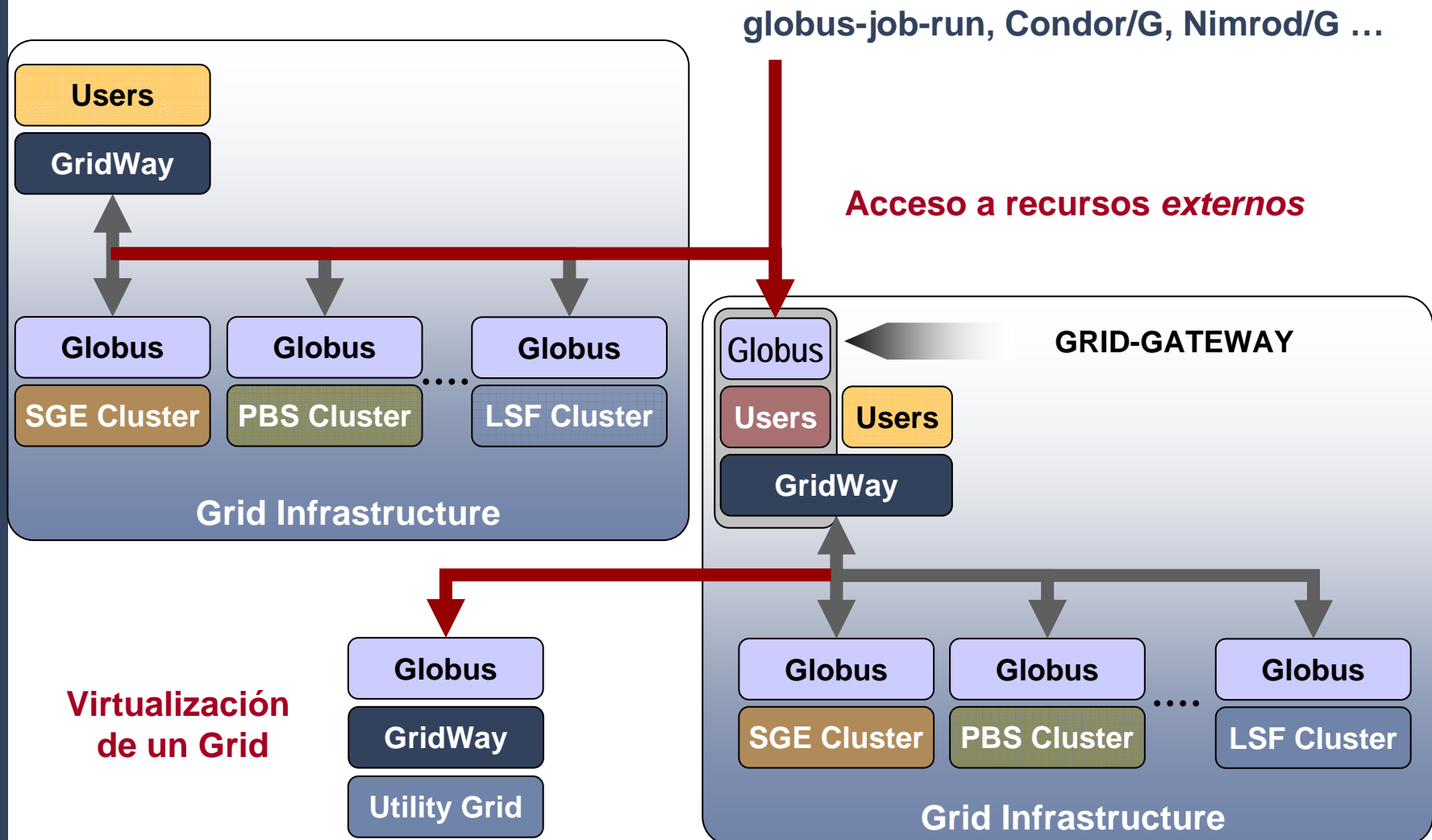
- Provisión de recursos bajo demanda, de forma ágil y adaptable
- Acceso a una potencia computacional *ilimitada*
- Transformar costes de IT fijos en variables
- Integración “*sin costuras*” de diferentes infraestructuras Grids (The Grid)

Planificación

- Gestión de cada Grid como otro recurso más
- Caracterización de un Grid como un recurso genérico

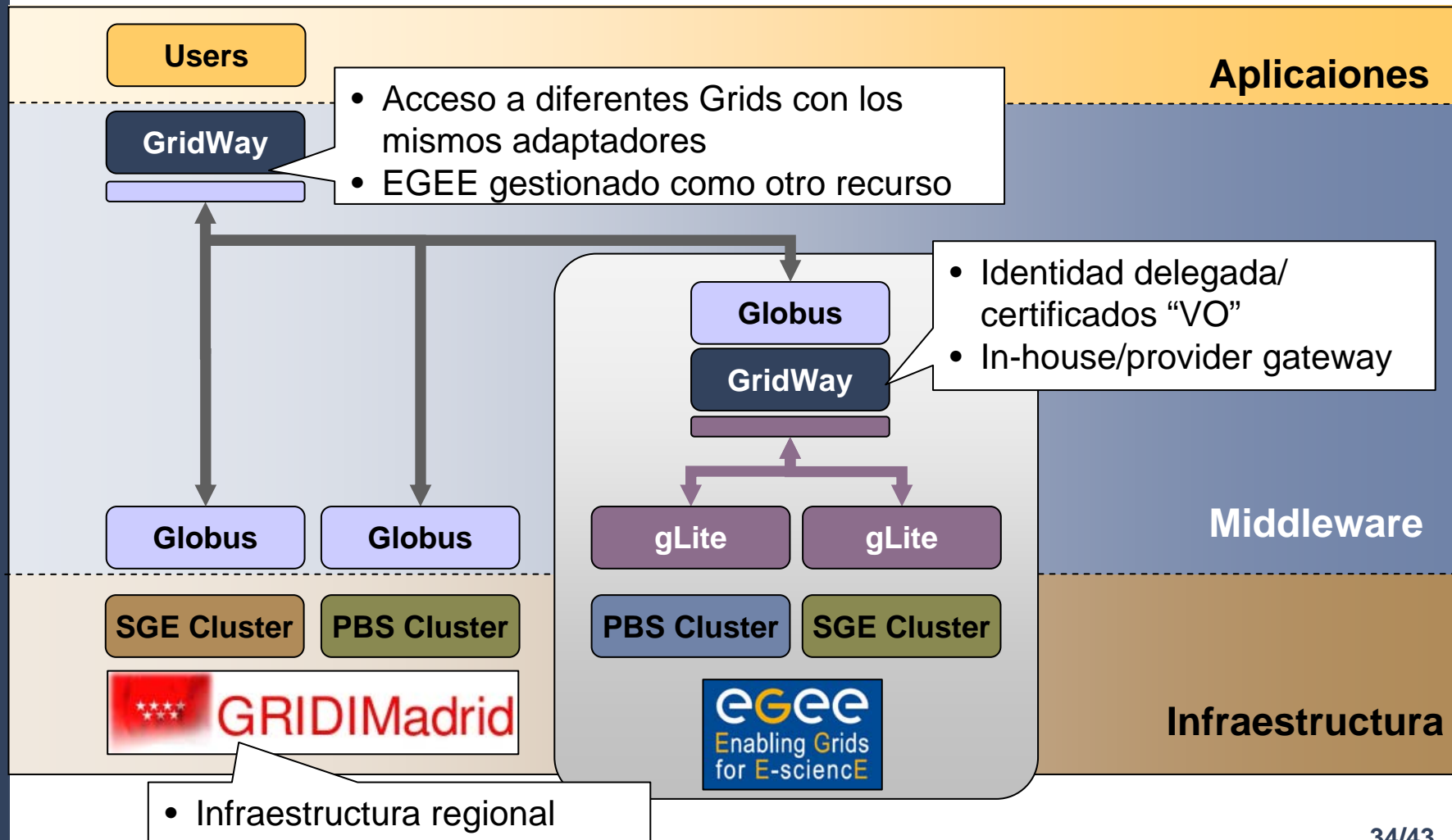
3.2. Multiple Grid Infrastructures

Grids con Múltiples Capas de Meta-planificación con Globus



3.2. Multiple Grid Infrastructures

Grids con Múltiples Capas de Meta-planificación : Ejemplos





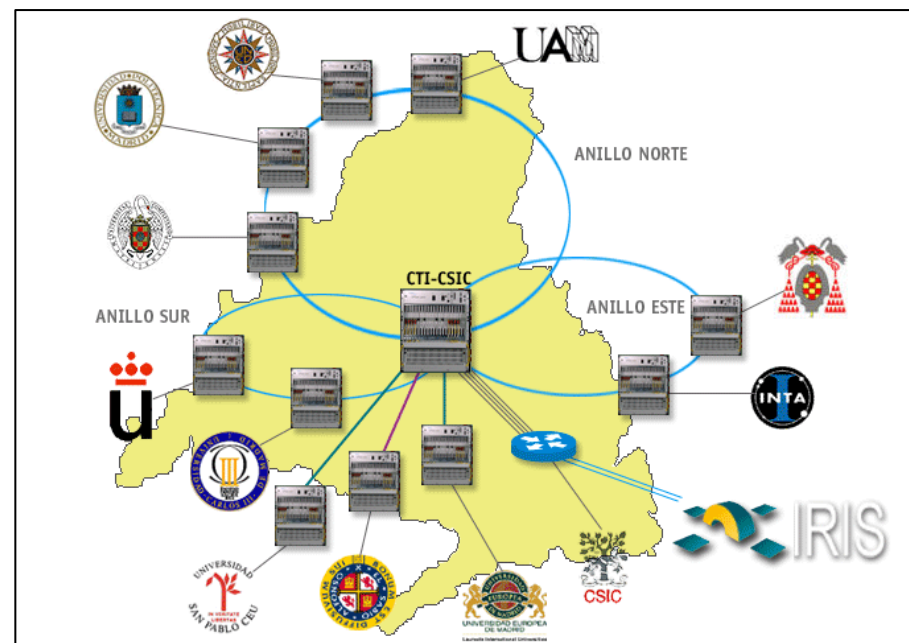
4. GRIDIMadrid



1. Recursos Computacionales
2. Middleware Grid
3. Arquitectura de Infraestructuras Grid Computacionales
4. GRIDIMadrid

Objetivos

- Establecer una infraestructura Grid de investigación en el ámbito de la Comunidad de Madrid, que a su vez proporcione tránsito hacia otras infraestructuras Grid nacionales e internacionales.
- Fomentar la colaboración entre las instituciones, los proyectos y las redes temáticas relacionadas con la investigación en tecnología Grid y migración de aplicaciones.





Estructura de Gestión

- **Coordinador**
- **Coordinador de Operaciones**
- **Representante** de un Centro
- **Responsable de Operaciones** de un Centro
- **Comité Ejecutivo**, compuesto por el Coordinador, el Coordinador de Operaciones y unos Representantes seleccionados

Middleware básico

- Globus Toolkit versión 4.
- Componentes basados en WSRF:
 - Infraestructura de seguridad (GSI)
 - Gestión de recursos (GRAM)
 - Gestión de datos (RFT y GridFTP)
 - Sistema de información (MDS)
- Excepcionalmente, se instalará la versión pre-WS de GRAM.
- Procedimiento para la instalación de otros servicios, previa aprobación del Comité Ejecutivo.

Autorización y Autenticación

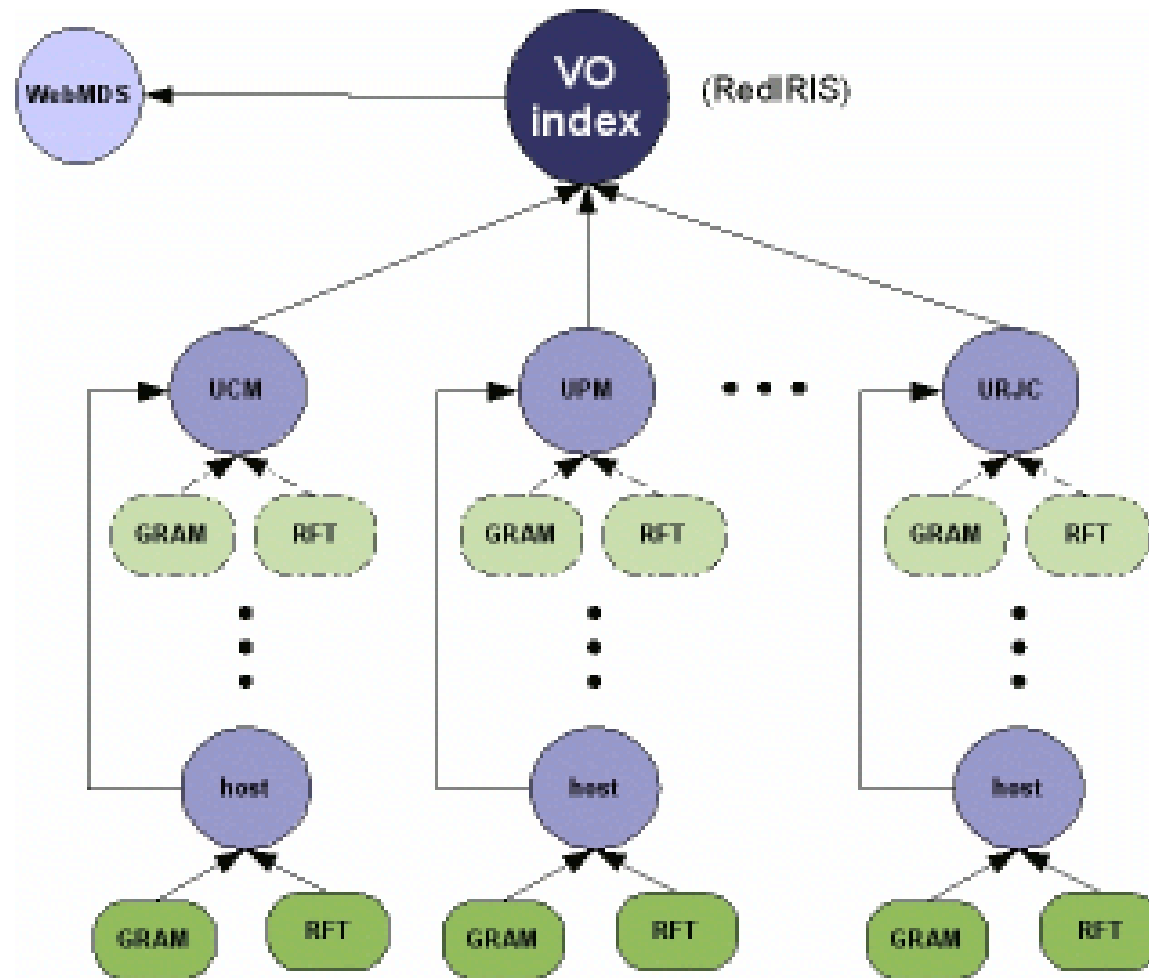
- Autenticación con pkIRISGrid: (<http://www.irisgrid.es/pki>).
- Inicialmente, autorización de usuarios centralizada y en un solo nivel.
- Archivo grid-mapfile público (<http://www.gridimadrid.org/grid-mapfile>).
- Asignación cíclica de los DNS de los certificados a las cuentas grid00, grid01, grid02, grid03.
- En último término, un Centro puede denegar de forma autónoma, temporal o permanentemente, el uso de sus recursos a cualquier usuario.
- Procedimiento para reservar la infraestructura, completa o en parte, para la realización de pruebas.



UNIVERSIDAD COMPLUTENSE
MADRID

4. GRIDIMadrid

Sistema de Información



Miembros Actuales

- Universidad Complutense de Madrid
- Universidad Carlos III de Madrid
- Universidad Rey Juan Carlos
- Universidad Autónoma de Madrid
- RedIRIS/Red.es
- CIEMAT

Requisitos de Adhesión

- Centro público de la Comunidad de Madrid.
- Posibilidad de solicitar certificados a la Autoridad de Certificación de IRISGrid.
- Establecer un Representante del Centro con competencia sobre la infraestructura hardware del mismo.
- Establecer un Responsable de Operaciones del Centro.

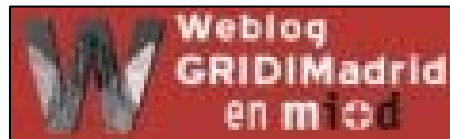
Más Información

- Página web de GRIDIMadrid

`www.gridimadrid.org`

- Weblog de GRIDIMadrid en Madri+D:

`weblogs.madrimasd.org/gridimadrid`





¡Gracias por su atención!