



Tema 3. Estructura, Ejecución y Monitorización de Trabajos en Grid

Conceptos Básicos de la
Computación en Grid y Cloud



- Definir el Concepto de Trabajo Grid.
- Describir los tipos de Trabajo Grid Existentes.
- Describir los Mecanismos para Crear, Lanzar y Monitorizar los Trabajos en una Infraestructura Grid basada en Globus (GRAM).
- Crear, Lanzar y Monitorizar Trabajos Grid en Infraestructuras Grid basadas en Globus.

- Concepto de Trabajo Grid.
 - Conceptos Básicos
 - Necesidad de los Trabajos Grid
 - Tipos de trabajos
 - Consideraciones
- Soluciones propuestas por Globus Toolkit
 - Conceptos Generales
 - Arquitectura
 - Estados de un trabajo Grid
 - Resource Specification Language (RSL)
 - Integración en globus toolkit
 - Funcionamiento del GRAM



- **Práctica 1:** Autorización de Usuarios en el Grid.
- **Práctica 2:** Lanzamiento de Trabajos Simples en el Grid (Stage).
- **Práctica 3:** Lanzamiento de Trabajos Simples (Entrada/Salida Estándar).

- **Concepto de Trabajo Grid.**
 - Conceptos Básicos
 - Necesidad de los Trabajos Grid
 - Tipos de trabajos
 - Consideraciones
- **Soluciones propuestas por Globus Toolkit**
 - Conceptos Generales
 - Arquitectura
 - Estados de un trabajo Grid
 - Resource Specification Language (RSL)
 - Integración en globus toolkit
 - Funcionamiento del GRAM



Concepto de Trabajo Grid

Conceptos Básicos

- Recursos:
 - Computacionales, Almacenamiento, ...
- Middleware Grid:
 - Proporciona acceso homogéneo a los usuarios a los Recursos Grid.
- Trabajo Grid:
 - Proceso Independiente que Dispone de los Datos de Entrada, el Código Ejecutable, los Argumentos y la Ubicación de los Resultados.

Concepto de Trabajo Grid

Necesidad de los Trabajos Grid (II)

- Los Trabajos Pueden Necesitar Requerimientos Especiales que Condicionen los Recursos Donde Pueden Ejecutarse
 - Propias del Recurso
 - Memoria Disponible y/o Asignable a un Proceso.
 - Procesador Disponible en Términos de Velocidad, Anchura de Palabra o Arquitectura.
 - Equipos Especiales, Como Sistemas de Almacenamiento, Sensores o Instrumental Científico.
 - Relativas a la Arquitectura
 - Sistema Operativo, Muy Sensible si se Utiliza Código Compilado.
 - Versión de Bibliotecas Software, Especialmente Importante en el Caso de Bibliotecas Software Dinámicas, Versiones del Compilador.
 - Relativas a Software Adicional
 - Software Específico Instalado en el Computador.
 - Licencias Disponibles.
- Una Gran Heterogeneidad Puede Conducir a la Necesidad de un Gran Número de Restricciones que Puede Complicar Notablemente la Gestión Reduciendo el Número de Nodos Disponibles.
- Por el Contrario Una Excesiva Homogeneidad Puede Impedir la Introducción en el Grid de Recursos Operativos para Otros Fines o que Necesiten Sistemas Operativos Diferentes (como Multiprocesadores).

Concepto de Trabajo Grid

Tipos de Trabajos – Trabajos Batch (I)

- **Trabajos Batch**

- Trabajos Cuya Entrada y Salida Únicamente Viene Dada por Ficheros y Tras su Puesta en Marcha No Requieren de la Intervención del Usuario.
- Un Trabajo Batch Típicamente Consiste en:
 - Programa Ejecutable.
 - Argumentos de Ejecución.
 - Ficheros de Entrada.
 - Descripción de los Ficheros de Salida (Además de la Salida y el Error Estándar).
- Una Vez Ejecutados se Identifican por un Número y pueden ser monitorizados.

Concepto de Trabajo Grid

Tipos de Trabajos – Trabajos Batch (II)

- Los Datos de Entrada Pueden Estar Disponibles de Diferentes Formas
 - A Través de los Argumentos de Entrada (Línea de Comandos).
 - A Través de Ficheros de Entrada Empaquetados con el Propio Trabajo en forma de script o ejecutable.
 - A Través de Ficheros de Entrada Cuyas Referencias Están Empaquetadas con el Trabajo.
- Por lo General, los Trabajos Batch y Especialmente Aquellos que Tienen Pocas Restricciones, se Ajustan Bien Para su Ejecución Eficiente en el Grid.

Concepto de Trabajo Grid

Tipos de Trabajos – Trabajos Paralelos

- **Trabajos Paralelos**

- Un Trabajo Individual Implica Varios Procesos que se Ejecutan de Forma Concurrente en Varias Máquinas.
- Generalmente los Trabajos Paralelos Interaccionan Únicamente con Diferentes Procesadores Dentro de Un mismo Recurso.
- Existen Implementaciones de Bibliotecas Software para la Intercomunicación de Procesos entre Diferentes Recursos Grid.

Concepto de Trabajo Grid

Tipos de Trabajos – Trabajos Interactivos

- **Trabajos Interactivos**
 - Abren un Canal Interactivo con el que se Comunican con el Usuario.
 - Funcionan a Modo de Shell Remoto. Permiten Disponer de un “Computador Virtual”.
 - Utilizados en la Reserva de Recursos.

Concepto de Trabajo Grid

Consideraciones sobre Trabajos Grid (I)

- Consideraciones Sobre los Lenguajes de Programación
 - Portabilidad a Una Variedad de Plataformas
 - Generalmente las Partes Computacionalmente Intensivas Están Escritas en "C" o FORTRAN. Esto Requiere una Compatibilidad a Nivel de Binarios.
 - Alternativas
 - Utilizar Lenguajes Multiplataforma (p.e. Java, python) y/o Lenguajes Interpretados (scripts) en la Medida de lo Posible.
 - Utilizar Código Fuente y Compilar in-situ. Esto Implica que los Recursos Tengan Adecuadamente Instalados los Compiladores
 - La Portabilidad Afecta Tanto a los Programas como a las Bibliotecas Software, por lo que se Recomienda Utilizar un Enlazado Estático.

- Interfaz a la Infraestructura Grid
 - Implementación de Aplicaciones Basadas en Scripts
 - Incluye Comandos y Aplicaciones Autónomas y Scripts que las Coordinan.
 - Utilizan Directamente los Comandos de Acceso al Grid para Ejecutar y Monitorizar Tareas y el Acceso a Datos.
 - Implementación de Aplicaciones Basada en APIs
 - Las Aplicaciones Grid Directamente Enlazan APIs que Implementan Funciones del Grid.
 - Generalmente Permiten Alcanzar un Más Bajo Nivel.

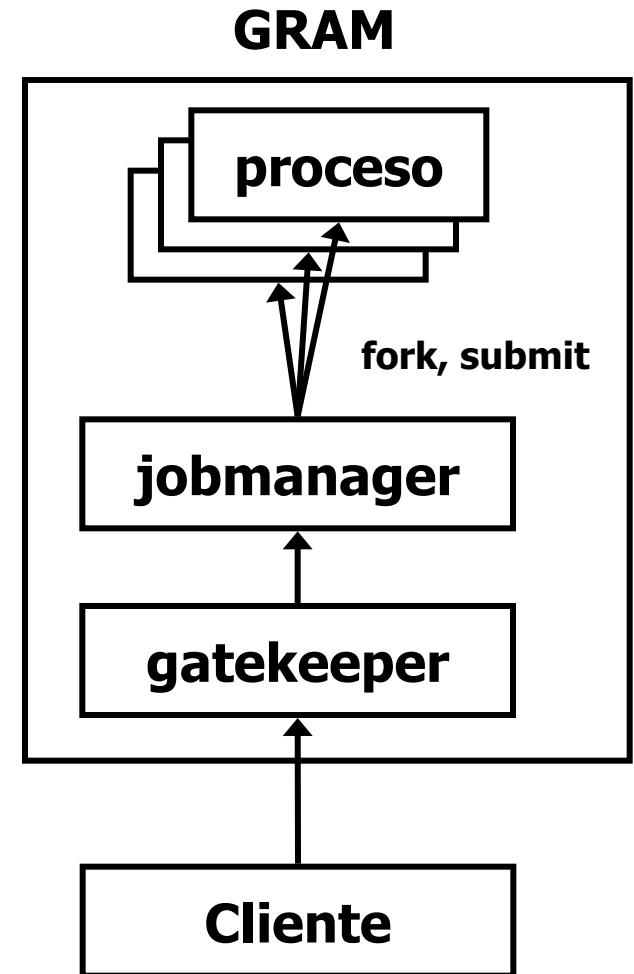
- **Concepto de Trabajo Grid**
 - Conceptos Básicos
 - Necesidad de los Trabajos Grid
 - Tipos de trabajos
 - Consideraciones
- **Soluciones propuestas por Globus Toolkit**
 - Conceptos Generales
 - Arquitectura
 - Estados de un trabajo Grid
 - Resource Specification Language (RSL)
 - Integración en globus toolkit
 - Funcionamiento del GRAM



- Grid Resource Allocation Manager (GRAM) es el Módulo en globus toolkit que Permite la Ejecución Remota de Trabajos y la Gestión de su Estado.
- GRAM Simplifica el Uso de Sistemas Remotos Proporcionando un Interfaz Único Estándar.
- GRAM Oculta las Peculiaridades de los Diferentes Sistemas de Ejecución de Trabajos Locales (Diferentes Gestores de Colas, Ejecución Directa,...) y se Integra con la Infraestructura de Seguridad de GSI.
- GRAM es Responsable de
 - Verificar y Procesar los Requerimientos del Trabajo Expresados en el Lenguaje de Especificación de Recursos (Resource Specification Language ó RSL).
 - Lanzar los Trabajos en los Recursos Remotos.
 - Permitir la Monitorización Remota de los Trabajos Creados.

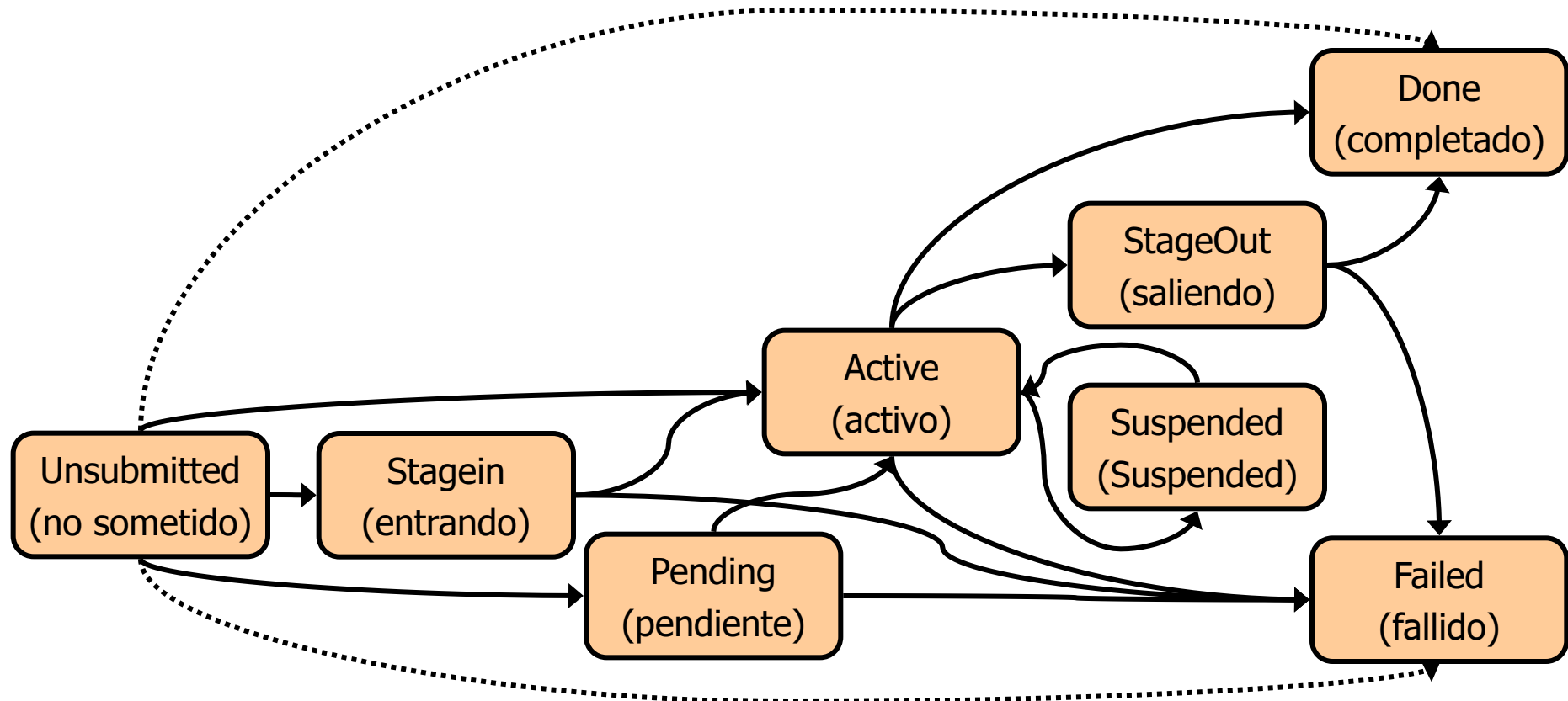
- **GRAM Se Compone de:**

- Gatekeeper. Demonio Instalado en el Host Remoto que se Encarga de Ocultar las Peculiaridades del Sistema y autoriza el usuario.
- Job Manager. Proceso Asociado a un Trabajo que Monitoriza su Estado y que se Destruye a Su Finalización.
- Proceso del Trabajo. Proceso Efectivo del Usuario.



Soluciones propuestas por Globus Toolkit

Estados de un Trabajo Grid (I)



Soluciones propuestas por Globus Toolkit

Estados de un Trabajo Grid (II)

- **Unsubmitted:** El Trabajo Todavía no ha Sido Sometido por el Planificador del Recurso.
- **StageIn:** El job manager Está Enviando el Ejecutable y/o los Ficheros de Entrada.
- **Pending:** El Trabajo ha Sido Sometido Pero Todavía No Hay Recursos Disponibles.
- **Active:** El Trabajo se Encuentra en Ejecución.
- **Suspended:** El Trabajo ha Sido Temporalmente Parado por el Planificador.
- **StageOut:** El job manager se Encuentra Copiando los Ficheros de Salida desde el Host Remoto
- **Done:** El Trabajo se ha Completado Satisfactoriamente.
- **Failed:** El Trabajo ha Terminado de Forma Inesperada, Debido a un Error o una Cancelación por el Sistema o el Usuario.

Soluciones propuestas por Globus Toolkit

Resource Specification Language (RSL) (I)

- El RSL Permite Definir en un Archivo de Texto los Requerimientos y Atributos de un Trabajo.
- Atributos Fundamentales
 - **arguments**
 - Argumentos en Línea de Comandos (Incluir Comillas si hay Espacios).
 - **count**
 - Número de Ejecuciones del Ejecutable (1 por Defecto).
 - **executable**
 - Nombre Absoluto del Ejecutable.
 - **stderr**
 - Nombre del Fichero que se Utilizará Para la Salida del Error Estándar.
 - **stdin**
 - Nombre del Fichero Usado Como Entrada Estándar.
 - **stdout**
 - Nombre del Fichero que se Utilizará Para la Salida Estándar.
- Todos los Nombres y Rutas de los Ficheros Son Relativos a la Máquina Destino.

● Otros Atributos

- **job_type**: Especifica el Tipo de Trabajo
 - *single* para un Trabajo de un Proceso.
 - *multiple* para un Trabajo Multiparamétrico.
 - *mpi* para un trabajo MPI.
- **max_cpu_time**: Fija el Máximo Tiempo de CPU que Utilizará el Programa. Si el Recurso Tiene una Limitación Mayor, Retornará un Error. (Unidades en Minutos).
- **max_memory**: Fija el Máximo Espacio de Memoria que Utilizará el Programa. Si el Recurso Tiene una Limitación Mayor, Retornará un Error. (Unidades en Megabytes).
- **rsl_substitution**: Permite Definir “Constantes” en el Resto del Texto.

- Ejemplo 1
 - Especificación de Proceso que Ejecute una Pausa ("Sleep") de 20 segundos.

```
(* Comentario *)  
& (executable = /bin/sleep (* <-- sin comillas *))  
  (arguments = 20)  
  (count = 1)
```

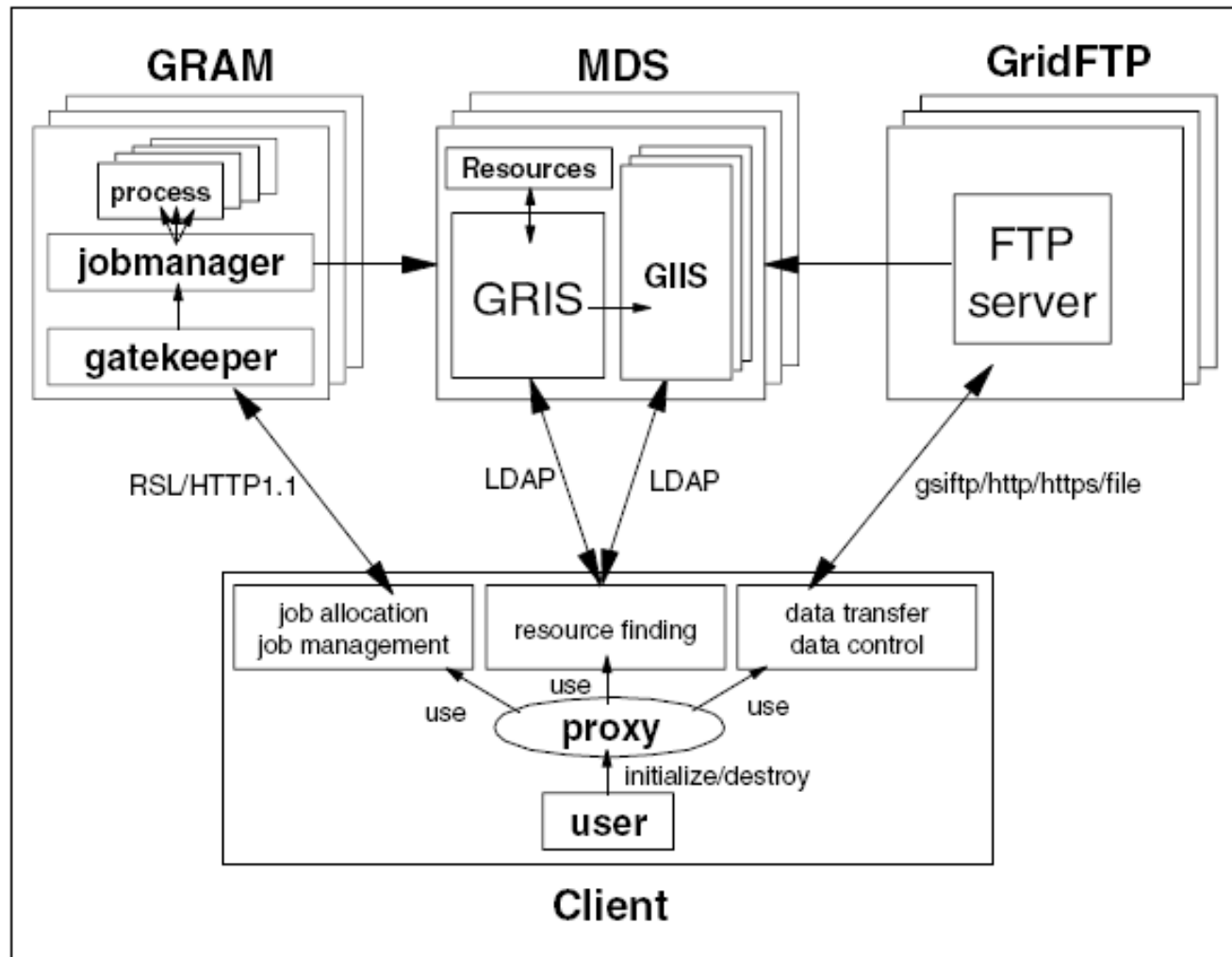
● Ejemplo 2

- Especificación de Proceso que Concatena dos Ficheros en la Salida Estándar.

```
& (rsl_substitution = ("DATADIR" "/home/dsegrelles/data/")  
                          ("EXECDIR" "/bin/") )  
  
(executable = $(EXECDIR) # "cat")  
(arguments = $(DATADIR) # "fich1"  
              $(DATADIR) # "fich2" )  
  
(count = 1)
```

Soluciones propuestas por Globus Toolkit

Integración en Globus Toolkit



- **El Comando globusrun**
 - El Comando globusrun Somete y Gestiona los Trabajos Remotos
 - Comprueba la Autenticación Mutua Entre los Hosts.
 - Verifica los Permisos del Usuario Para Ejecutar el Trabajo (Comprobados por el gatekeeper).
 - Transfiere el Ejecutable y los Ficheros de Entrada Asociados.
- **Gatekeeper**
 - El Demonio del gatekeeper Verifica los Permisos del Usuario.
 - Tras la Verificación Crea un jobmanager Delegando en él la Comunicación con el Usuario.
- **Jobmanager**
 - Cuando es Creado por el gatekeeper Verifica la Sintaxis del RSL y si es Válido Lanza el Ejecutable en los Recursos Locales.
 - El jobmanager Monitoriza el Trabajo a Través de los Recursos Locales e Informa al Cliente.