Evaluación del Uso Coordinado de Infraestructuras Grid

J.L. Vázquez-Poletti
DACYA
Universidad Complutense
28040 Madrid
jlvazquez@fdi.ucm.es

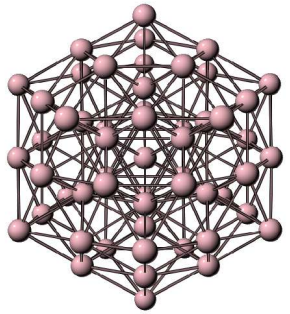
A. Fuentes
RedIRIS
28040 Madrid
afuentes@rediris.es

E. Huedo
LCASAT
Centro de Astrobiología
28850 Torrejón de Ardoz
Madrid
huedoce@inta.es

R.S. Montero e I.M. Llorente
DACYA
Universidad Complutense
28040 Madrid
rubensm@dacya.ucm.es
llorente@dacya.ucm.es

¿Qué vamos a ver?

- **Características ideales de un Grid**
  - Principio “extremo a extremo”
  - Nuestra aproximación
- **Infraestructuras Grid**
  - Ejemplos usados
  - Testbed conjunto
- **Experimentos**
  - Métricas
  - Resultados
- **Conclusiones**



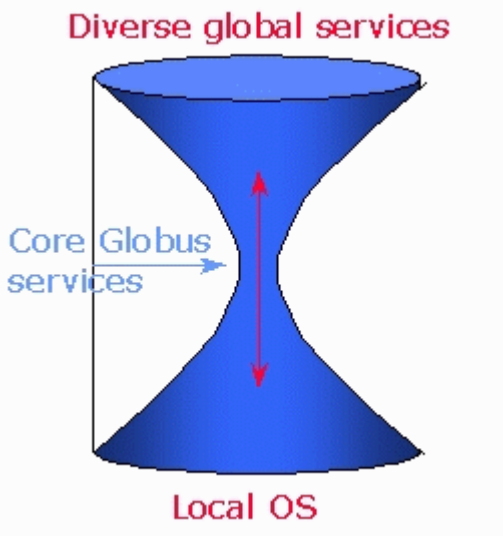
Grid es...

- **Según Ian Foster**
  - Control descentralizado
  - Basado en protocolos e interfaces estándar
    - Abiertos
    - Propósito General
  - Proporcionar calidad de servicio
    - Seguridad
    - Productividad
    - Tiempo de respuesta
    - Uso coordinado de recursos heterogéneos

- **Según nuestro subconsciente (o no)**
  - ¿Quién no cambiaría los 2 primeros requisitos por más calidad de servicio?
  - Optimización Vs. Uso general

Principio “extremo a extremo”

- Filosofía Grid conduce a entornos *desacoplados*
  - Múltiples dominios de administración y autonomía
  - Escalabilidad
  - Heterogeneidad
  - Dinamismo (adaptabilidad)
- El principio en palabras:
  - *Las funciones finales solo deberían ser realizadas por sistemas finales*
- El principio en una imagen:



- Similar al de Internet
- Globus sigue este principio

the globus toolkit

http://www.globus.org/

Nuestra aproximación

- Desarrollo de herramientas cliente
  - Adaptables a distintas implantaciones de *middleware*
  - Usar simultáneamente recursos asignados a diferentes proyectos
    - Mayoría usa Globus como *middleware* básico
- Desplazamiento de funcionalidad
  - Desde recursos
  - A intermediarios o clientes
- Compartición de recursos entre proyectos

Infraestructura Grid: Ideas Básicas

- Capas
  - Aplicaciones y portales Grid
  - *Middleware* Grid de usuario **(1)** EJ.: Condor-G
  - *Middleware* Grid básico **(2)** Ej.: Globus
  - Recursos Grid
- **(1)** y **(2)** son denominadas el *"middleware"*
  - Conectan aplicaciones con recursos
  - Grid *desacoplado*
    - Importante mantenerlas separadas e independientes
    - Conjunto de protocolos e interfaces
      - Limitado
      - Bien definido

Infraestructura Grid: EGEE

- *Enabling Grids for EsciencE*
- Nivel de producción
  - Requisitos muy restrictivos
- Define:
  - *Middleware* Grid de usuario
  - *Middleware* Grid básico
  - Recursos Grid (estrechamente interrelacionados)
- Usa *middleware* LCG (*LHC Computing Grid*): LCG-2
  - Limitaciones en términos de heterogeneidad
    - Intel, Sci-Linux, configuración concreta en *clusters*
  - Limitaciones en escalabilidad
  - Limitaciones en complejidad de despliegue
    - *Middleware* en todos los nodos, conexión externa
  - Se centra en aplicaciones de Física de Partículas
  - Dependiente de un solo centro: CERN
- ¿gLite superará estas limitaciones?

Infraestructura Grid: IRISGrid

- Resultado de Acción especial “Preparación de Proyectos GRID en el marco de iniciativas de e-ciencia en Europa”
- Objetivo:
  - Creación de infraestructura Grid estable en España
  - Unión de recursos distribuidos geográficamente
- En fase de definición de protocolos, procedimientos y directrices
- Define:
  - Solo *middleware* Grid básico
    - Puede funcionar sobre diferentes recursos Grid
- 1ª Versión: Basado únicamente en Globus
  - Pruebas con herramienta Grid*W*ay
  - Pruebas con herramienta GRID Superescalar (CEPBA)

Infraestructura Grid: Grid*Way*

Metaplanificador ligero

Usa Globus como *middleware* Grid básico sobre cualquier tipo de recurso Grid

Capaz de usar cualquier Infraestructura Grid

- Basada en Globus
- Sin hacer modificaciones

Testbed: Consideraciones iniciales

- EGEE
  - Comportamiento de Globus ha sido modificado
  - No pierde principales protocolos e interfaces
  - Transferencias de ficheros
    - Iniciadas por envoltorio de trabajos enviado a nodos de cálculo
- Grid*W*ay
  - Ejecución en 3 pasos:
    - *Prólogo*: Prepara sistema remoto
    - *Envoltorio*: Ejecuta trabajo real
    - *Epílogo*: Finaliza sistema remoto
  - No depende del *middleware* subyacente
  - No requiere
    - Instalación nueva
    - Conexión externa de red en nodos de cálculo

Testbed: Características

- Infraestructura: Recursos heterogéneos
  - Estaciones de trabajo y servidores SMP
  - *Clusters* con diferentes arquitecturas (Alpha, Intel)
- Gestores locales: Diferentes sistemas de gestión de recursos distribuidos (DRMS)
  - PBS, SGE, Fork
- Red de interconexión: Red pública no dedicada
  - Latencias y anchos de banda variables
- Middleware: Grids basados en Globus Toolkit
  - Diferentes Arquitecturas y servicios ofrecidos

Testbed: Cifras

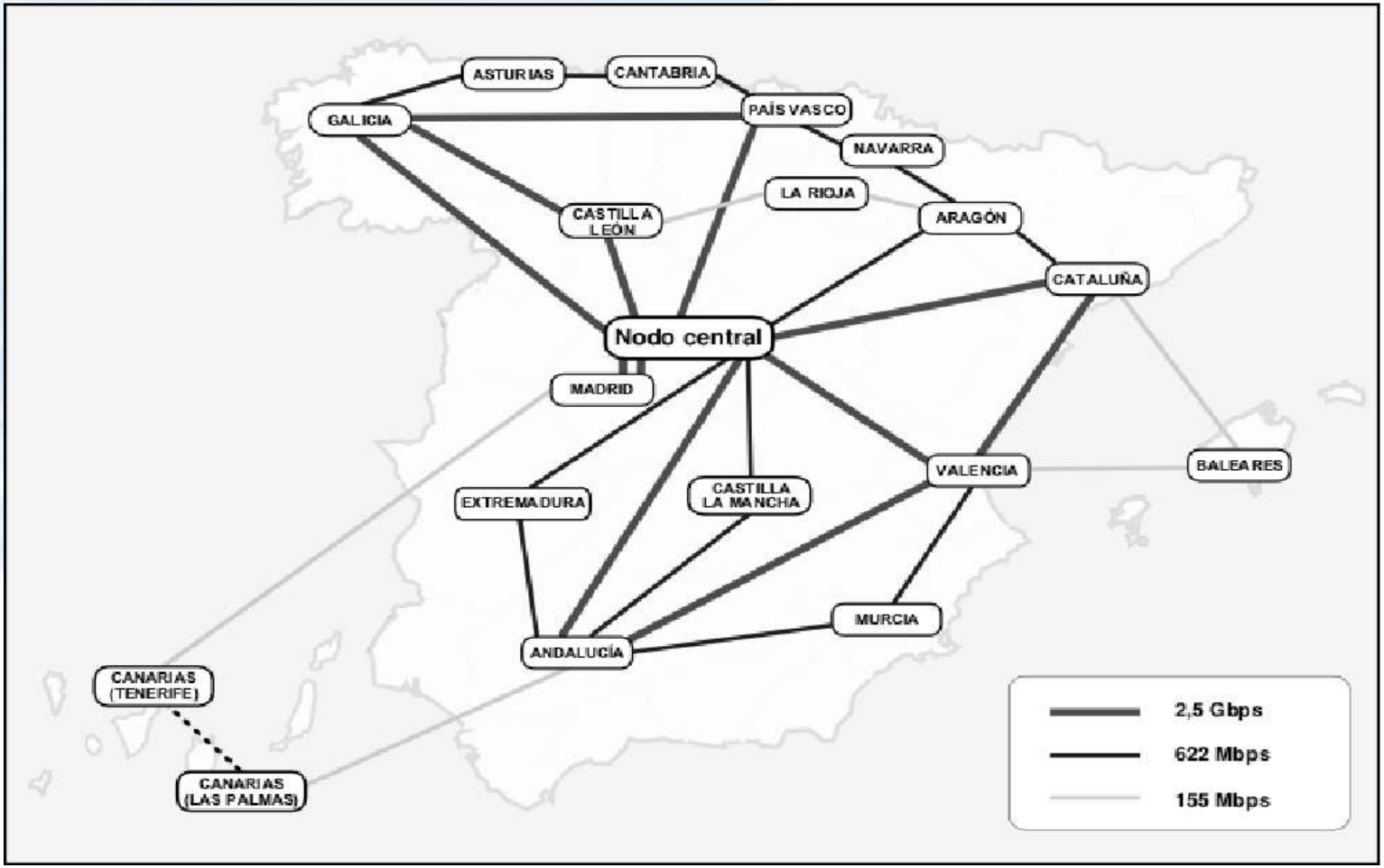
- IRISGrid
  - 16 sitios
    - 7 participaron en experimentos
  - 209 CPUs agregadas
- EGEE
  - 100 sitios
    - 7 participaron en experimentos
  - 404 CPUs agregadas
- En total...
  - 13 sitios (LCASAT-CAB pertenece a ambos testbeds)
  - 613 CPUs agregadas
- Limitación número de trabajos simultáneos enviados a mismo recurso a 4 (no saturar)

Testbed: Más cifras

Nombre	Sitio	Proc.	Rec.	Velocidad	DRMS	Infraestructura
heraclito	RedIRIS	Intel Celeron	1	700MHz	Fork	IRISGrid
platon	RedIRIS	2 × Intel PIII	1	1.4GHz	Fork	IRISGrid
descartes	RedIRIS	Intel P4	1	2.6GHz	Fork	IRISGrid
socrates	RedIRIS	Intel P4	1	2.6GHz	Fork	IRISGrid
aquila	DACYA-UCM	Intel PIII	1	700MHz	Fork	IRISGrid
cepheus	DACYA-UCM	Intel PIII	1	600MHz	Fork	IRISGrid
cygnus	DACYA-UCM	Intel P4	1	2.5GHz	Fork	IRISGrid
hydrus*	DACYA-UCM	Intel P4	1	2.5GHz	-	IRISGrid
babieca	LCASAT-CAB	Alpha EV67	30	450MHz	PBS	IRISGrid
bw	CESGA	Intel P4	80	1.2GHz	Fork	IRISGrid
llucalcari	IMEDEA	AMD Athlon	14	800MHz	PBS	IRISGrid
augusto	DIF-UM	4 × Intel Xeon	1	2.4GHz	Fork	IRISGrid
caligula	DIF-UM	4 × Intel Xeon	1	2.4GHz	Fork	IRISGrid
claudio	DIF-UM	4 × Intel Xeon	1	2.4GHz	Fork	IRISGrid
lxsrv1	BIFI-UNIZAR	Intel P4	50	3.2GHz	SGE	IRISGrid
ce00	LCASAT-CAB	Intel P4	8	2.8GHz	PBS	EGEE
mallarme	CNB	4 × Intel Xeon	8	2GHz	PBS	EGEE
lcg02	CIEMAT	Intel P4	6	2.8GHz	PBS	EGEE
grid003	FT-UAM	Intel P4	49	2.6GHz	PBS	EGEE
gtbcg12	IFCA	2 × Intel PIII	34	1.3GHz	PBS	EGEE
lcg2ce	IFIC	AMD Athlon	117	1.4GHz	PBS	EGEE
lcgce02	PIC	Intel P4	69	2.8GHz	PBS	EGEE

Testbed: Conexiones

Centros interconectados por RedIRIS – Red académica y científica española



Testbed: *Middleware* Grid básico

Principales características y configuraciones de cada componente (Situación de partida):

Componente Globus	**IRISGrid**	**EGEE**
Infraestructura de Seguridad	IRISGrid CA y generación manual de archivos de autorización	DATAGRID-ES CA y generación automática de archivos de autorización
Gestión de Recursos	GRAM con directorios de usuario compartidos (clusters)	GRAM sin compartición de directorios de usuario (clusters)
Servicios de Información	IRISGrid GIIS Y GRIS locales, con esquema MDS	CERN BDII y GRIS locales, con esquema GLUE
Gestión de Datos	GASS y GridFTP	GASS y GridFTP

Testbed: Modificaciones (I)

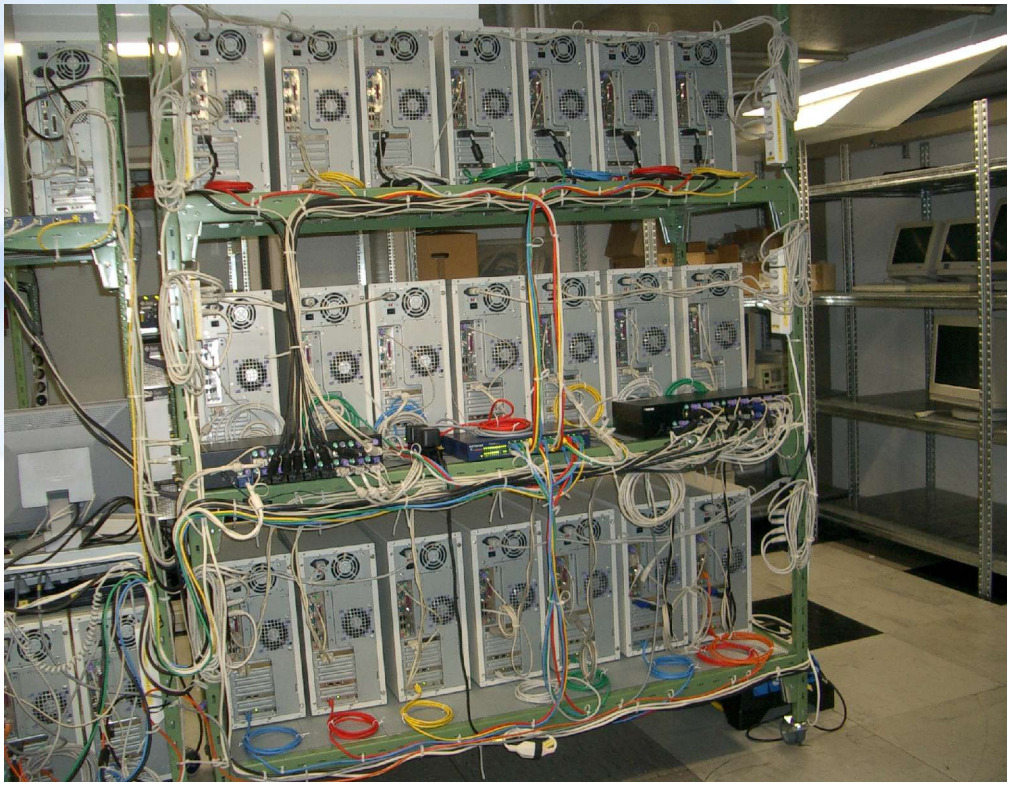
- Infraestructura y sistema de planificación
- **Infraestructura de Seguridad**
  - Autenticación mediante certificados de usuarios emitidos por DATAGRID-ES
    - Añadir nivel de confianza a esta CA en recursos IRISGrid
  - Autorización explícita al usuario de pruebas
- **Servicios de Información y Selector de Recursos**
  - Selector que combina información del estado
    - Esquema MDS – IRISGrid
    - Esquema GLUE – EGEE
  - Acceso homogéneo a servidores de información GISS y BDII
    - Ambos usan LDAP
    - Diferencia: BDII es persistente y se actualiza periódicamente

Testbed: Modificaciones (y II)

Servicios de Ejecución

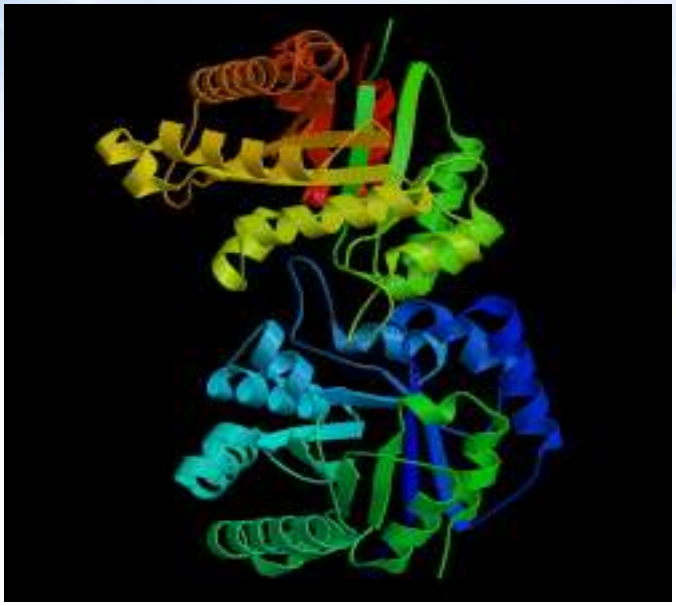
Middleware Grid de EGEE no requiere compartición de directorios de usuarios en *cluster*

Wrapper debe realizar transferencia explícita de ficheros entre *front-end* y nodos del *cluster*



Resultados: Primeras cifras

- Aplicación Bioinformática
  - Análisis de familia de 80 proteínas ortólogas de la enzima *Triosa Fosfata Isomerasa*
- Experimentos realizados en diferentes días de la semana
- 5 ejecuciones
- **Tiempo medio de respuesta: 43.37 minutos**



Resultados: Tiempos

Tiempos de transferencia y ejecución / Máquina

Incluyen sobrecarga inducida por *Middleware* Grid

	Tiempo de Ejecución		Tiempo de Transferencia		
Recurso	Media	Dev.	Media	Dev.	Sitio
heraclito	2146	57	150	107	RedIRIS
platon	919	275	67	50	RedIRIS
descartes	611	33	48	30	RedIRIS
socrates	647	103	51	27	RedIRIS
aquila	1895	235	143	93	DACYA-UCM
cepheus	2022	112	64	24	DACYA-UCM
cygnus	755	74	33	20	DACYA-UCM
babieca	1798	28	131	131	LCASAT-CAB
bw	697	176	123	54	CESGA
llucalcari	1567	168	169	92	IMEDEA
augusto	1200	233	89	61	DIF-UM
caligula	1242	233	153	109	DIF-UM
claudio	1228	184	187	131	DIF-UM
lxsrv1	687	190	152	92	BIFI-UNIZAR
ce00	929	267	220	80	LCASAT-CAB
mallarme	945	191	123	68	CNB
lcg02	932	342	158	115	CIEMAT
grid003	739	63	90	67	FT-UAM
gtbcg12	1002	52	261	105	IFCA
lcg2ce	889	226	208	113	IFIC
lcgce02	777	179	98	68	PIC

El Sentido de la Métrica

- Métricas importantes para evaluación de recursos
  - Impacto de estrategias
  - Rendimiento individual
  - Influencia de red de interconexión
- ¿Por qué Media y Desviación Típica?
  - Dinamismo inherente al Grid
  - Desviación Típica de Transferencia
    - Dinamismo del entorno
    - Calculada sobre mismo recurso a lo largo del tiempo

Tiempos de Ejecución

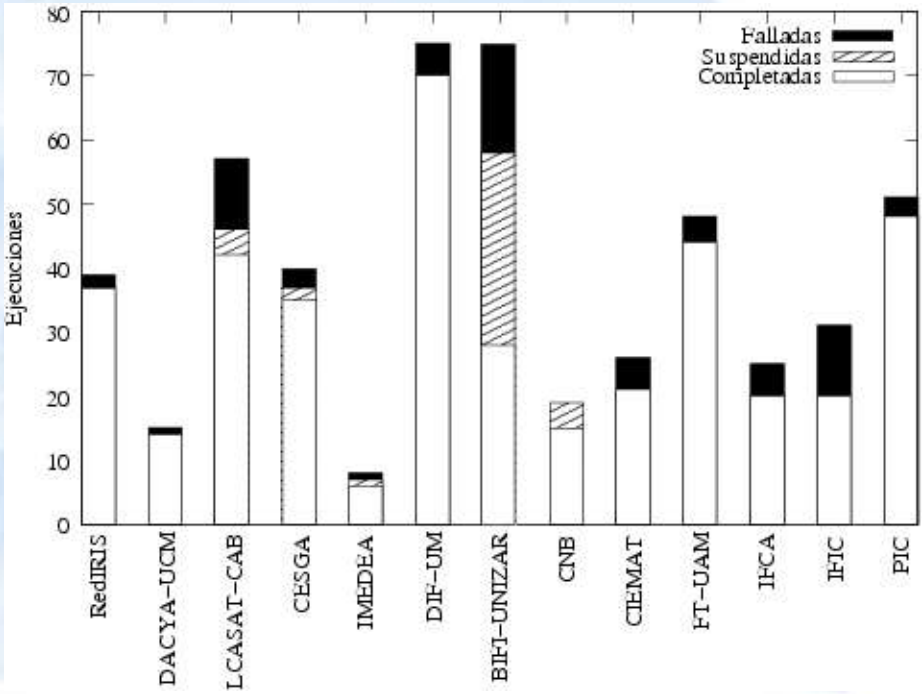
- Recursos con procesadores más rápidos
  - Tiempo medio ejecución más bajo
  - *lxsrv1, bw, grid003, lcgce02*
- Desviación mayor
  - Sobrecarga en sistema de colas
  - Nodo *front-end* más lento
  - *lcg02, ce00, lcg2ce*
- Nodos SMP
  - Uso competitivo de sus recursos compartidos
  - Mayor variabilidad en tiempo cuando todas las CPUs son usadas simultáneamente
  - *platon, augusto, caligula, claudio*

Tiempos de Transferencia

- Sitios bien conectados con cliente (*DACYA-UCM*)
  - Media menor
  - *DACYA-UCM, RedIRIS, FT-UAM*
  - Excepción: *Front-end* lento
    - *LCASAT-CAB, IFIC, CIEMAT*
- Sitios con peor conectividad con cliente
  - Mayor tiempo medio y variabilidad
  - *IFCA, IMEDEA, BIFI-UNIZAR, DIF-UM*

Tan real como...

Planificación agregada de las 5 ejecuciones:



Migración en caso de fallo

La bola de cristal del Grid

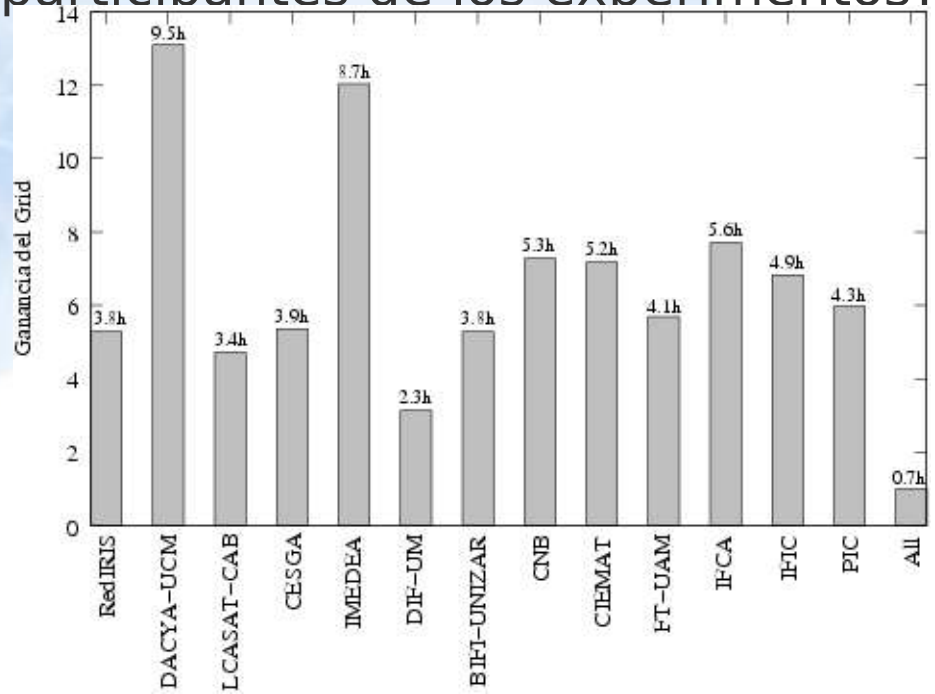
¿Predicciones? ¿Nos Beneficia el Grid?

Ganancia Grid

$$S_{Site} = \frac{T_{Site}}{T_{Grid}},$$

Tgrid: Tiempo ejecución en Grid - *Tsite*: Idem en sitio (usando *makespan* óptimo)

¿Y los participantes de los experimentos?

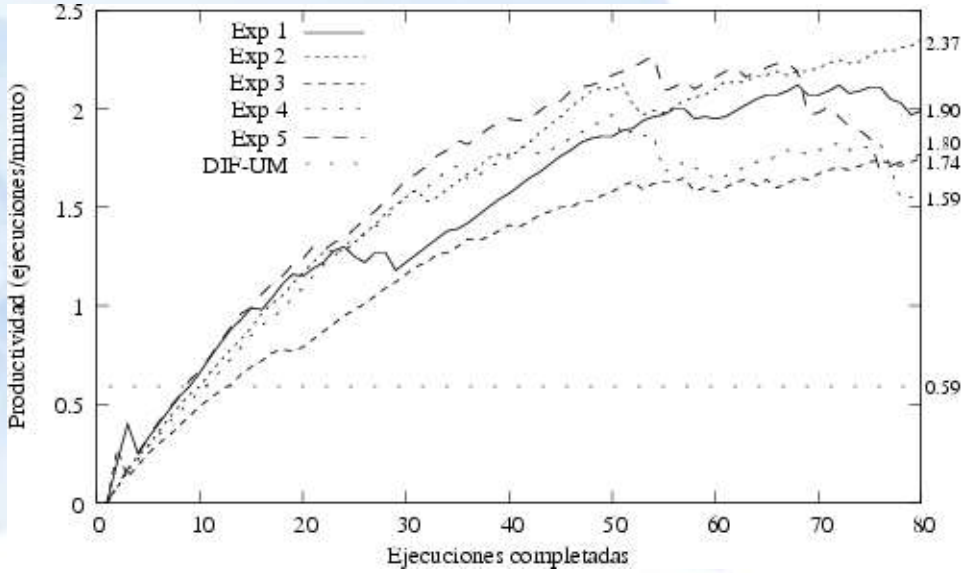


Ganancia + Tiempo de respuesta de cada site

¿Y todo el Grid?

Productividad dinámica del Grid

Frente a sitio con más potencia computacional (*DIF-UM*)



¿Qué hemos demostrado?

- **Se puede aplicar principio “extremo a extremo” en parte cliente**
  - *Middleware* Grid de usuario
  - Sistema planificación y ejecución de usuario propuesto puede trabajar con *middleware* Grid básico
    - En cualquier infraestructura
    - De forma débilmente acoplada
- Análisis similar YA había sido realizado desde recursos
  - **Principio “extremo a extremo” aplicable en ambos lados**
- Facilidad de integración en diferentes entornos
  - **Arquitectura descentralizada + principio “extremo a extremo” son idóneas para Grid**

Gracias a...

... aquellas instituciones que participan en IRISGrid y EGEE, en particular a las que colaboraron en las simulaciones (aportando recursos y atendiendo al equipo de investigación).

... ustedes, por su atención ;-).