

¿Qué puede hacer Grid por tí?

Matemáticas en acción
Noviembre 2006, Santander

Celso Martínez Rivero
Consejo Superior de Investigaciones Científicas
Instituto de Física de Cantabria, IFCA
(Centro Mixto CSIC-Universidad de Cantabria)

1. E-Ciencia
 2. ¿Qué es el Grid?
 3. Aplicaciones beneficiándose del Grid.
 4. Conclusiones
-

1. E-Ciencia

2. ¿Qué es el Grid?

3. Aplicaciones beneficiándose del Grid.

4. Conclusiones

- e-Ciencia
 - la e-Ciencia es ante todo colaboración
 - si no hay colaboración, no hay e-Ciencia
 - es más importante la colaboración que la e-Ciencia

- Cómo colaboran
 - Ordenadores
 - Ingenieros y Científicos

- La ciencia del siglo XXI está muy fragmentada
 - Los proyectos son muy complejos
 - Los proyectos multidisciplinares son demasiado complejos
 - Los investigadores, están demasiado repartidos por todo el mundo

- Colaborar obliga a
 - Modelizar
 - Compartir
 - Comprender
- Colaborar en e-Ciencia:
 - **Modelizar**: Diseño, Simulación
 - **Compartir**: Información, Computación
 - **Comprender**: Conocimiento (Knowledge)
- Sin fórmulas:
 - el impacto de la e-Ciencia es proporcional a la modelización, y se potencia si somos capaces de compartir recursos y conocimiento

- Proyectos de LHC
 - Modelizar: Simulación + Reconstrucción + Análisis
 - Compartir: Información (Datasets, Software) + Recursos (Tiers)
 - **Comprender**: un canal de Física
- Proyectos Multidisciplinares
 - Simulaciones en LHC y en misión Planck:
 - Modelos MC
 - Compartir recursos (EGEE)
 - **Comprender**: origen del cosmos
 - Data Mining en Meteorología
 - Modelos SOM
 - Compartir recursos (CrossGrid)
 - **Comprender**: predicción meteorológica

1. E-Ciencia

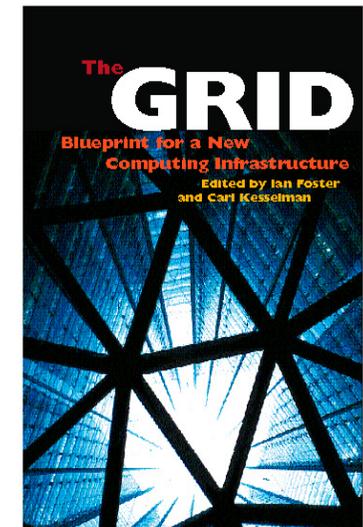
2. ¿Qué es el Grid?

3. Aplicaciones beneficiándose del Grid.

4. Conclusiones

¿ Qué es un “Grid” ?

- Una propuesta en analogía con la red eléctrica (“electrical grid”):
 - Recursos de computación distribuidos, fiables y accesibles desde cualquier punto
 - Interacción sencilla, sin que el usuario perciba la complejidad de la infraestructura:
 - Envía su trabajo mediante un “interface de usuario”
 - GRID localiza los recursos necesarios
 - Gestiona la identificación y autoriza el uso de los mismos dentro de una “Organización Virtual”
 - Organiza un acceso eficiente a los datos
 - Ejecuta el trabajo, e informa al usuario



El “problema” del Grid

- Compartir de forma flexible, segura y coordinada, los recursos entre un conjunto dinámico de individuos e instituciones
 - Del artículo “The Anatomy of the Grid: Enabling Scalable Virtual Organizations”

- Posibilitar comunidades (“organizaciones virtuales”) para compartir recursos distribuidos geográficamente, de cara a lograr un objetivo común, utilizando protocolos abiertos, y en ausencia de
 - Un sitio y/o un control central
 - Unas relaciones de confianza-propiedad ya existentes

(Breve paréntesis)

➤ Semejanzas entre el Web y el Grid:

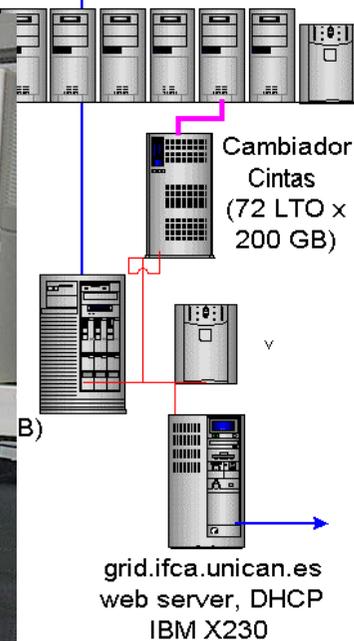
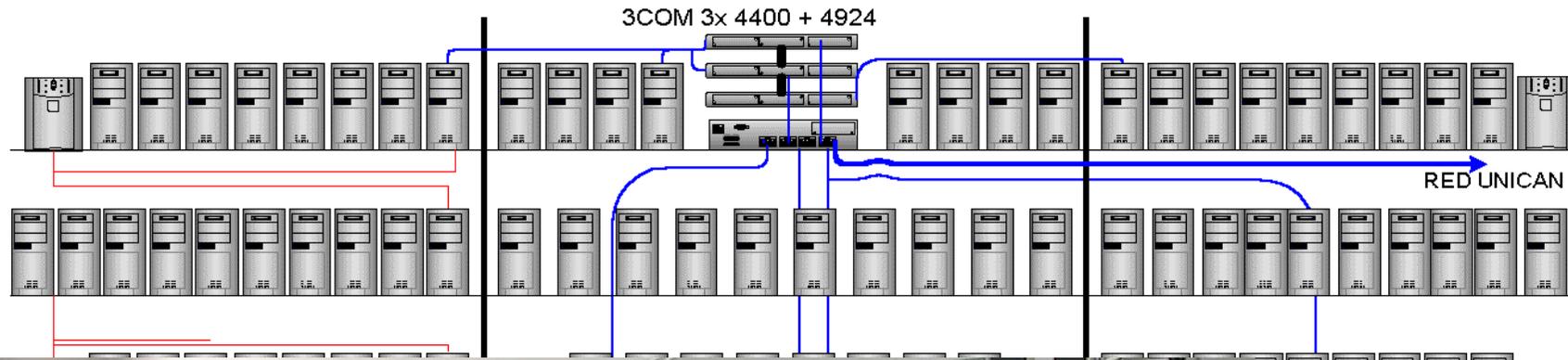
- El World Wide Web, inventado en el CERN, permite el acceso a información distribuída
- El Grid permite el acceso a computación distribuída.



Un poco de historia...

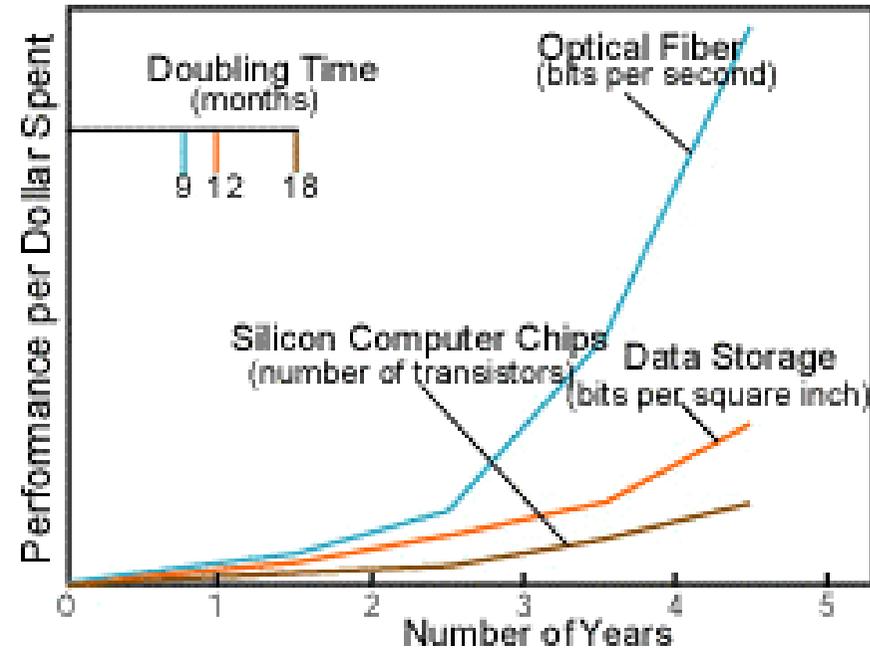
- **Agregación de recursos:**
 - Sistema SHIFT (“granjas” RISC Unix, L.Robertson, CERN)
 - Clusters de PCs (Sistemas Beowulf, “fábricas” de PC...)
- **Recursos distribuidos compartidos:**
 - Sistema Condor (M.Livny)
 - gestión de tiempo inactivo en sistemas Linux de la red local
 - Red Entropía, Programa SETI
 - Sistemas “Peer to Peer”
- **1995: Supercomputing '95**
 - Experiencia I-WAY: 17 centros USA conectados a 155Mbps
 - Primeras iniciativas Grid:
 - NASA Information Power Grid
 - Iniciativa de la NSF con los centros NCSA y SDC
 - Advanced Strategic Computing Initiative (DoE)

Clusters 200 CPUs < 200k\$



Evolución: la clave está en la red

- “Ley” de Moore
 - La potencia se duplica cada 18 meses
 - Seguro hasta 10 GHz
 - Pero problemas de acceso a memoria, saltos de instrucciones y disipación de calor! (charla de M.Valero)
 - Procesadores: “aún” de silicio
 - El más “popular”: Intel 32bits (IA32) .
 - Los de “élite”, 64bit
- Almacenamiento :
 - Discos convencionales:
 - Serial ATA: 600MB/s, convergencia con SCSI
 - Cinta Magnética:
 - LTO : crecimiento hasta 1TB /cartucho
- Mejora de la red:
 - Se duplica la capacidad de la red cada 9 meses
 - 10 GB ethernet



- Sistemas Operativos
 - Windows 2000/XP
 - Unix, en particular Linux
- Programación
 - Seguimos con C, incluso Fortran (90), y sobre todo :
 - Programación orientada a objeto: C++ y Java
 - El rendimiento es importante
 - La portabilidad es igualmente importante
 - Re-utilización y mantenimiento son también claves
 - Herramientas:
 - Compiladores
 - Debuggeadores
 - Modelado (UML)
 - Repositorios de código (CVS)
- Librerías Científicas
 - NAG, Mathematica, MatLab...

HTC vs HPC

- High Throughput Computing :
 - Maximizar numero de ejecuciones del programa
 - Ejemplo: 1.000.000 simulaciones independientes
 - Usemos 1000 procesadores, y procesemos 1000 simulaciones en cada uno
 - El resultado final se logra por agregación
 - Funciona en un entorno cluster
 - Ejemplos: Condor, colas batch: LSF, PBS
 - **Funciona en un entorno GRID si existe un "Resource Broker"**

- High Performance Computing:
 - Minimizar el tiempo de respuesta
 - Ejemplo: Entrenar una red neuronal con una arquitectura compleja NN
 - Usar 1000 procesadores, cada uno calcula el error sobre una parte de los datos, pero...
 - Para el siguiente paso, se necesita la información de todos los procesadores
 - Necesita una descripción "paralelizada" (p.ej. MPI) + topología (master-esclavo)
 - Cercano al "METACOMPUTING"
 - **Funcionará en un entorno GRID ?**

¿Hay diferencia con “clusters” o “intranets”?

-
- Las organizaciones virtuales en el Grid comparten recursos distribuidos geográficamente, utilizando protocolos abiertos, y en ausencia de un sitio y/o un control central
 - Acceso “transparente” a recursos heterogéneos y distribuidos geográficamente
 - ¡Posibilidad de atender grandes demandas de computación!

 - La base del entorno de computación Grid: middleware **Globus**
 - Estándar de facto, desarrollado inicialmente en ANL (I.Foster) y USC (C. Kesselman)

¿Hay diferencia con “clusters” o “intranets”?

CLUSTER

- Solicitar cuenta/password al administrador
- Desarrollar el código
- Usar un repositorio
- Compilar usando MPI y almacenar el ejecutable
- Establecer los links a los datos
- Enviar el script a una cola (PBS, LSF...)
- Chequear el job status
- Recuperar el resultado de la cola
- Mostrar el resultado.

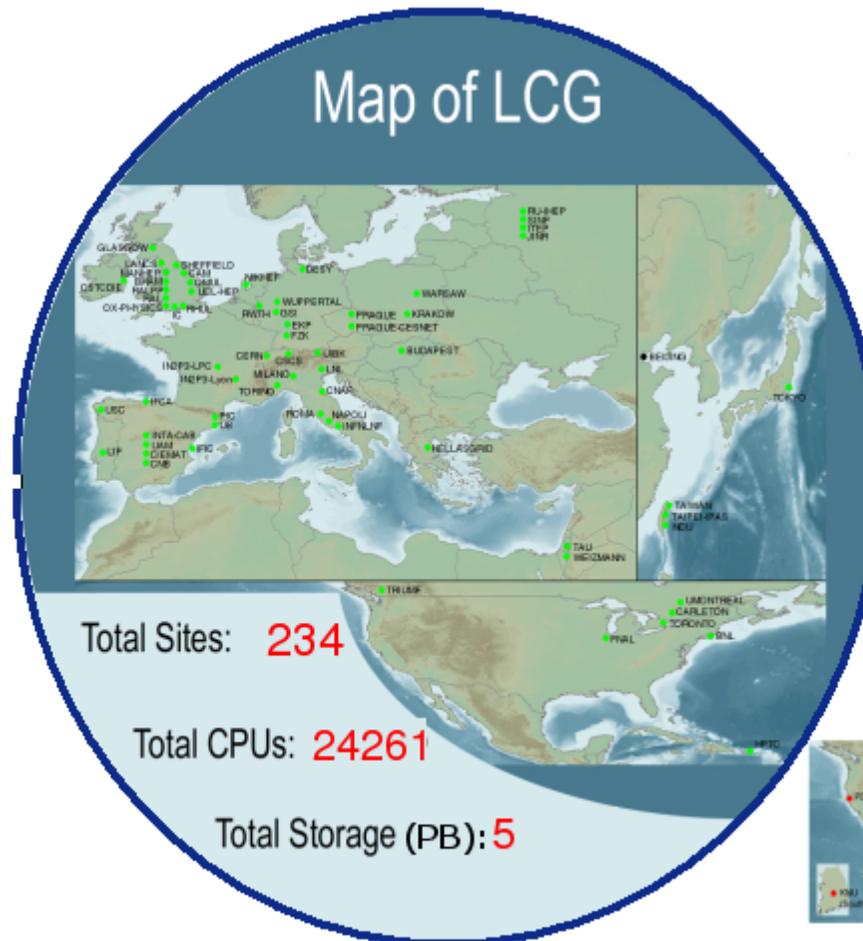
GRID

- Conseguir un certificado, registrarse en una Organización Virtual.
- Desarrollar el código
- Usar un repositorio CVS compartido con otros.
- Compilar usando MPI, almacenar el ejecutable en un Directorio Virtual (SE)
- Definir los lfn's de los datos (almacenados en SE's)
- Submitir al RB desde un desktop Grid
- Chequear el jobstatus
- Guardar el resultado en un directorio Virtual
- Mostrar el resultado en un Grid Desktop

¿Cómo puede ayudar el Grid?

- Desarrollando un testbed que permita ejecutar eficazmente nuestras aplicaciones paralelas o distribuidas
- Estableciendo las herramientas necesarias para mantener la seguridad a la par que la rapidez en la ejecución de los trabajos.
- Enviando de la forma más óptima los trabajos, teniendo en cuenta los requerimientos de datos de entrada así como de necesidad de CPU
- Facilitando el uso de herramientas online y visualización interactiva de los resultados parciales.

LCG testbed



Collaborating with LCG

NorduGrid

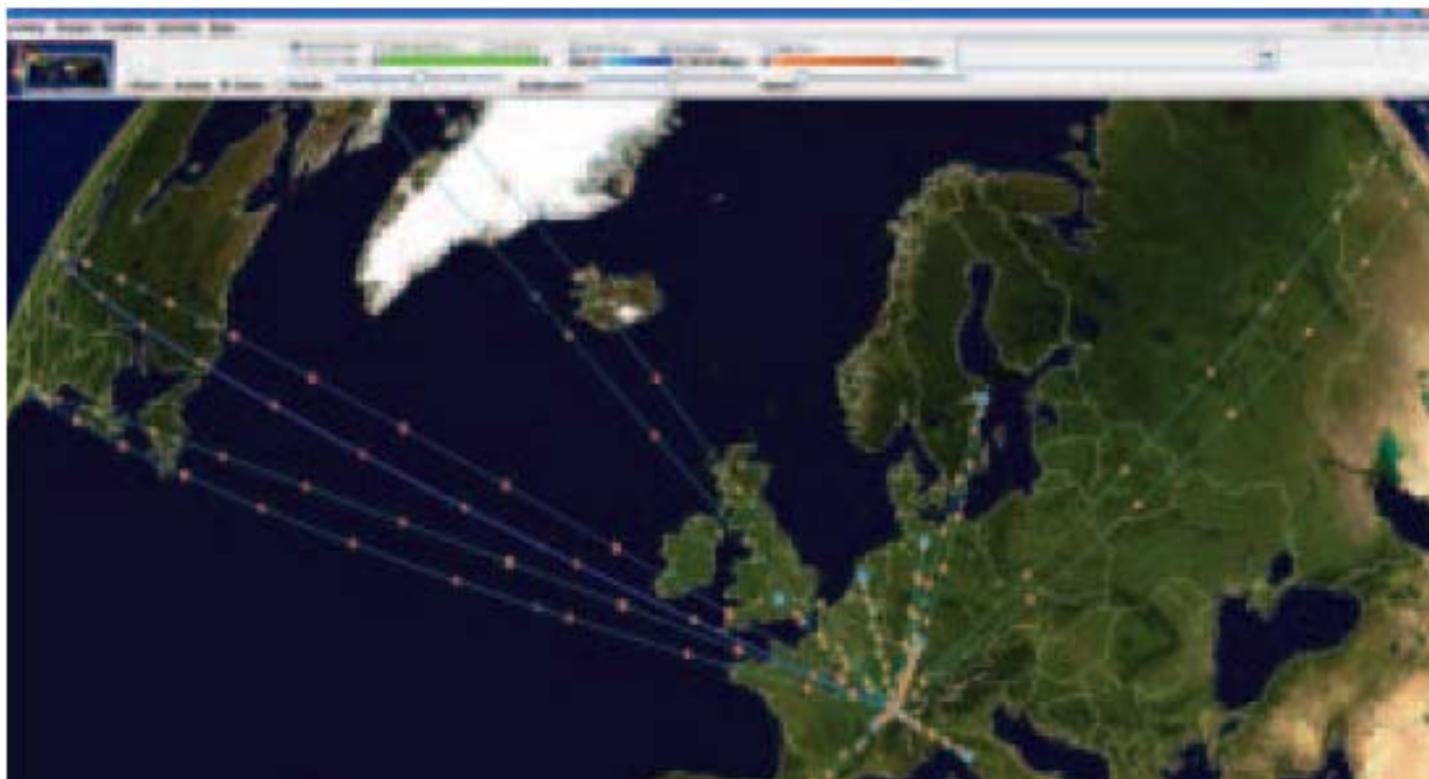


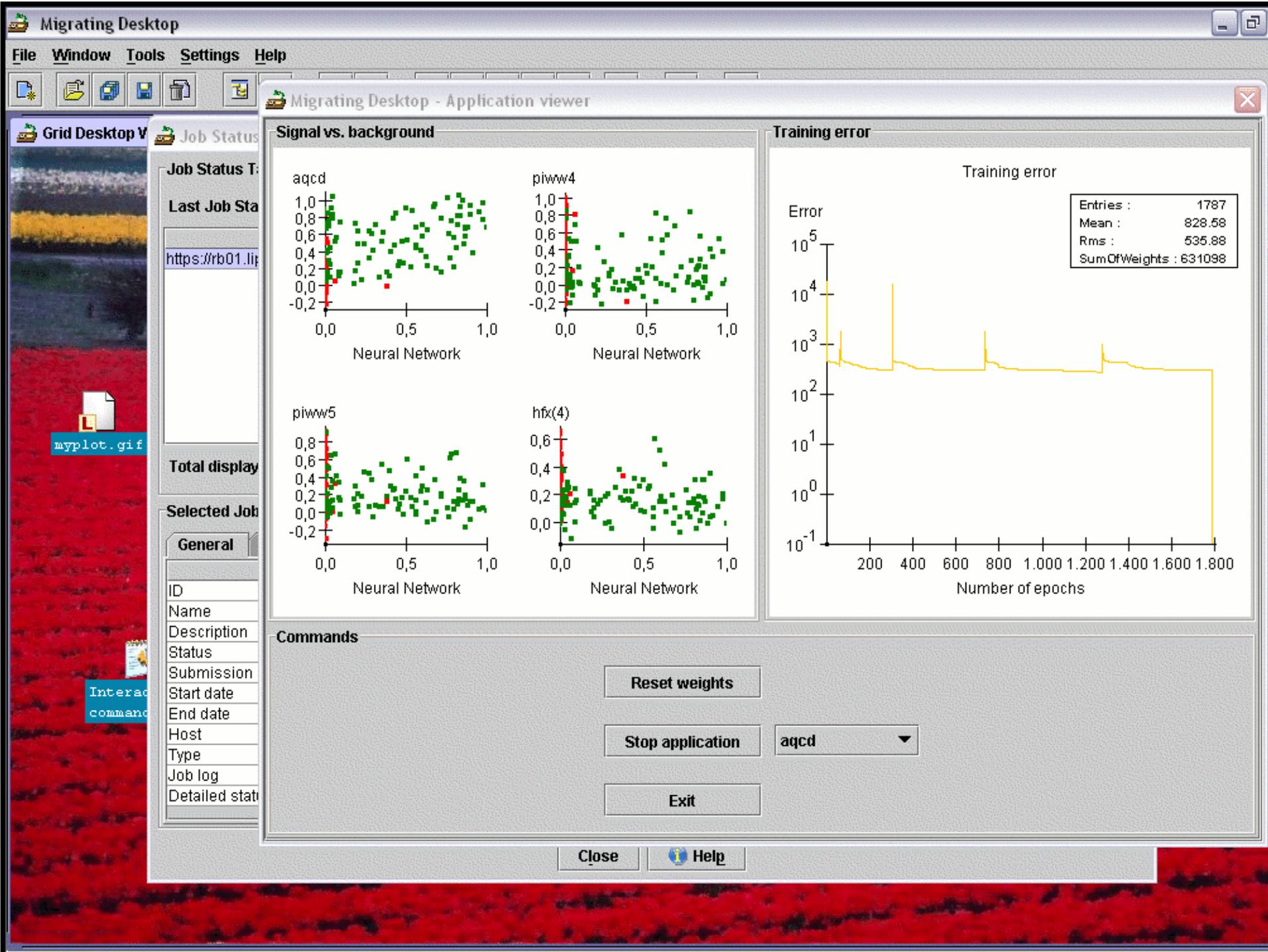
Grid3



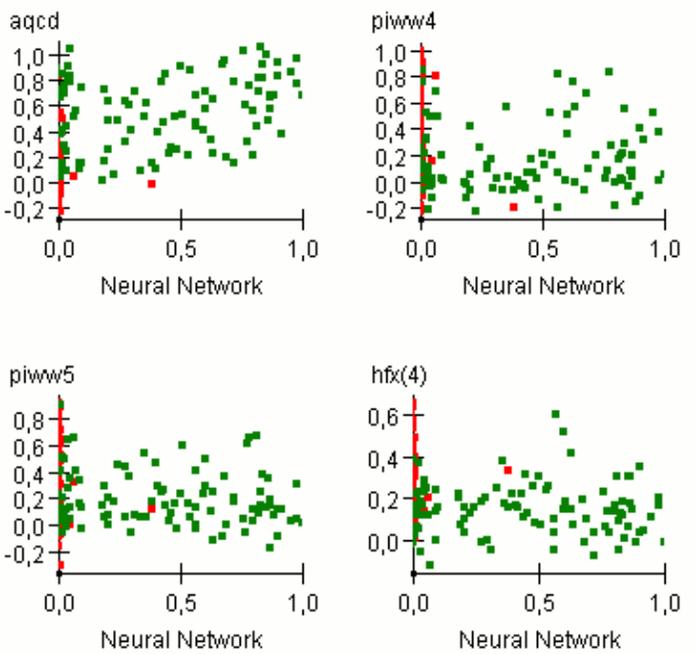
LCG Testbed en funcionamiento:

- Camino seguido por datos enviados desde el CERN hacia diversos sitios elegidos previamente en una prueba de funcionamiento.

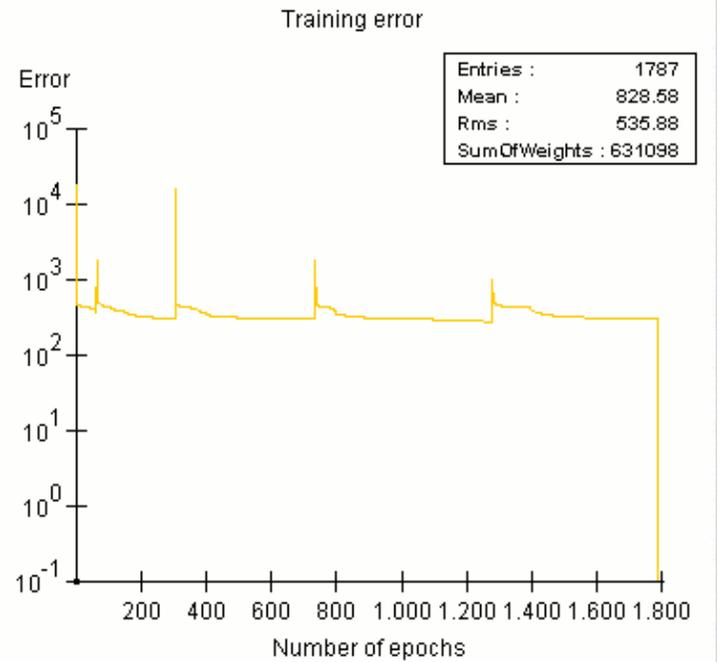




Signal vs. background



Training error



Commands

Reset weights

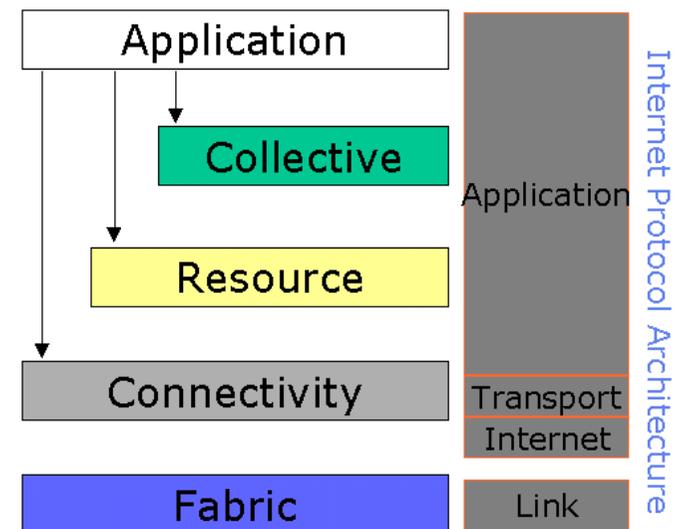
Stop application aqcd

Exit

Close Help

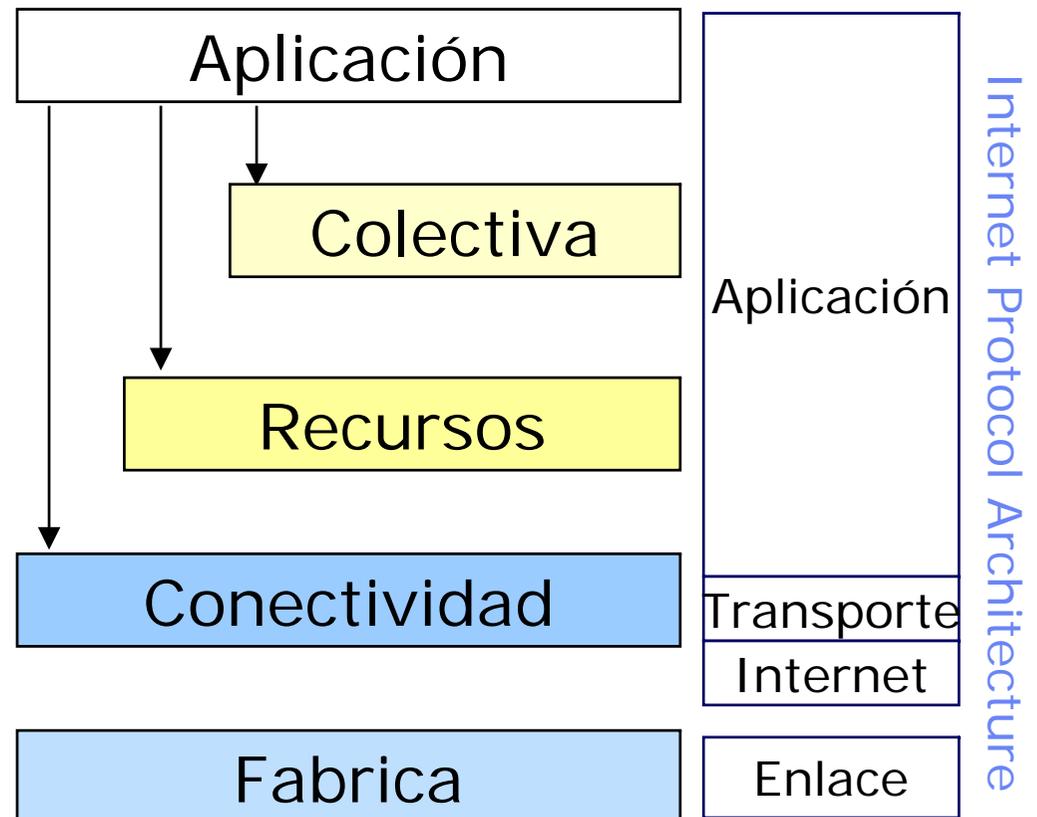
El "Middleware"

- Del mismo modo que Internet es posible gracias a la existencia de un estándar como el protocolo TCP/IP, y el WWW gracias al protocolo http y el lenguaje HTML, el Grid cuenta con un estándar "de facto": el middleware *Globus*
 - Desarrollado por ANL (I.Foster) y USC (C.Kesselman)
 - Arquitectura en "capas"
 - Protocolos:
 - Seguridad, Gestión de Recursos, Información y Transferencia



Arquitectura Grid en capas (Analogía con Internet)

- “Coordinando múltiples recursos”: servicios de infraestructura ubicuos, servicios distribuidos específicos de aplicaciones
- “Compartiendo recursos individuales”: negociación del acceso, control del uso
- “Hablando a los objetos”: comunicación (protocolos Internet) y seguridad
- “Control local”: Acceso y control de los recursos locales



Capa de Conectividad:

- Comunicación
 - Protocolos de Internet: IP, DNS, routing, etc.
- Seguridad: "Grid Security Infrastructure" (GSI)
 - Uniformidad en: autenticación y autorización
 - Un solo punto de firma
 - Tecnología de llave pública, SSL, X.509, GSS-API
 - Infraestructura de Soporte: Autoridades de Certificación, gestión de certificados & llaves



Capa de Recursos:

- Grid Resource Allocation Management (GRAM)
 - Asignación remota, reserva, monitorización, control de recursos de computación
- Protocolo GridFTP (extensión de FTP)
 - Acceso a datos y transporte de alto rendimiento
- Grid Resource Information Service (GRIS)
 - Acceso a la información sobre la estructura de los recursos y su estado
- Reserva de red, monitorización, control

Capa Colectiva :

- Servicios de metadirectorios (Index Servers)
 - Vistas específicas sobre colecciones dinámicas de recursos construidas por una comunidad
 - Gestores de recursos (Resource Brokers)
 - Descubrimiento de recursos y asignación
- Catálogos de replicas de datos
- Servicios de replica
- Co-reserva y co-asignación de servicios

1. E-Ciencia

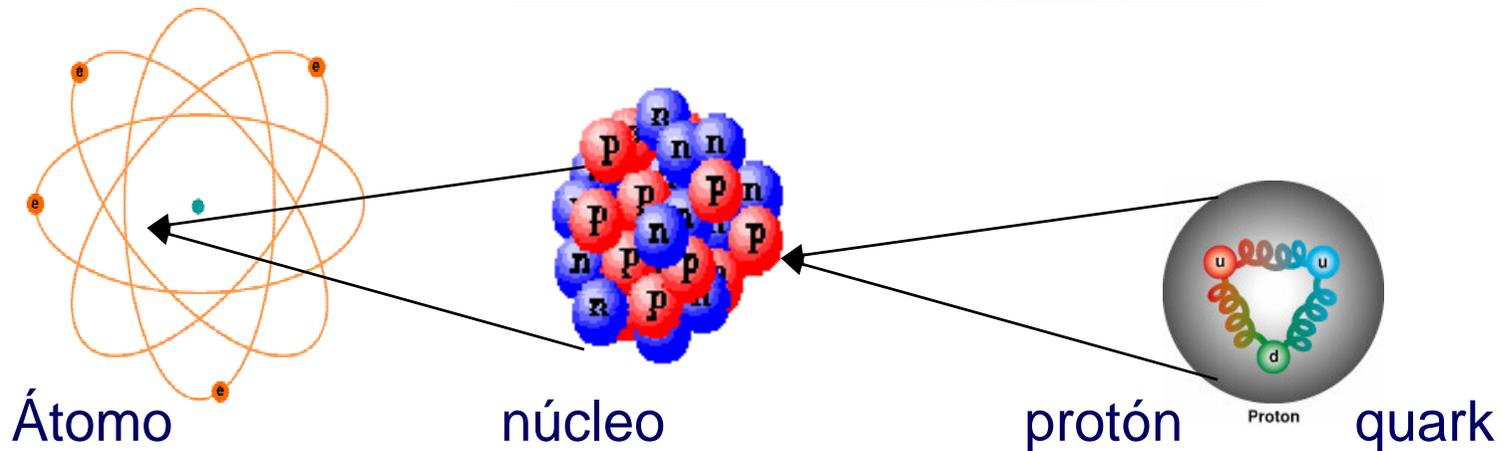
2. ¿Qué es el Grid?

3. Aplicaciones beneficiándose del Grid.

4. Conclusiones

Ejemplo práctico: Búsquedas Fundamentales en Física de Partículas:

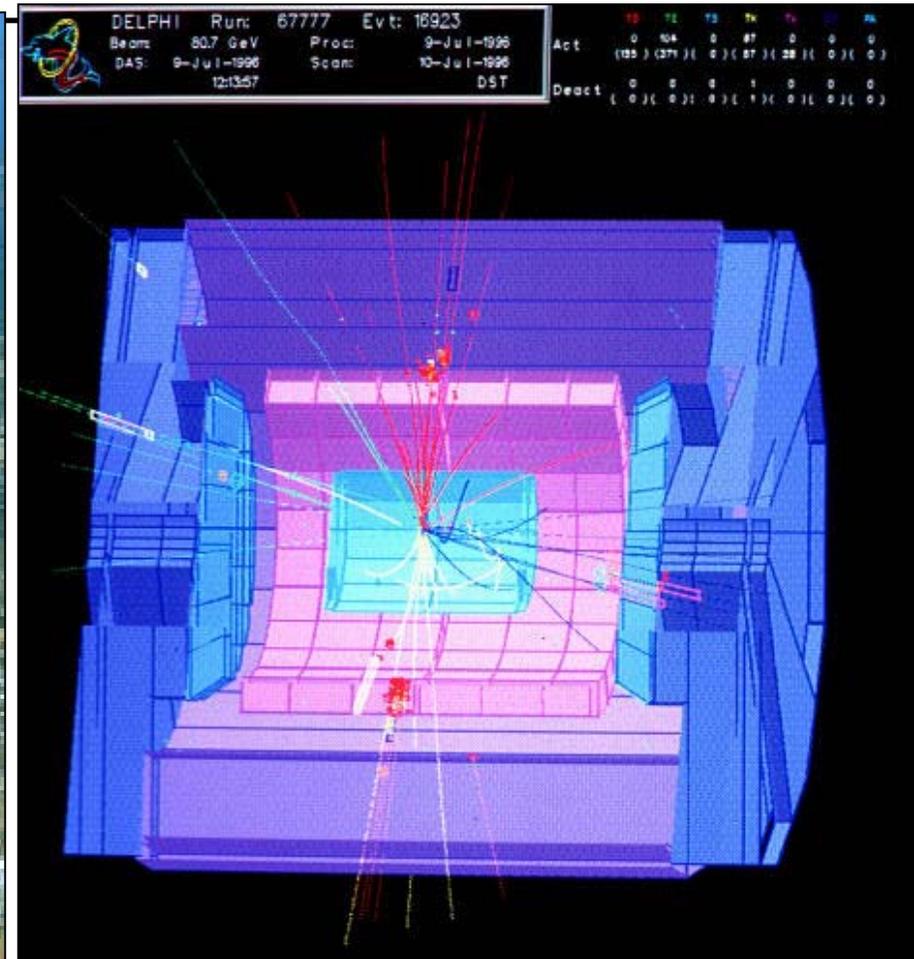
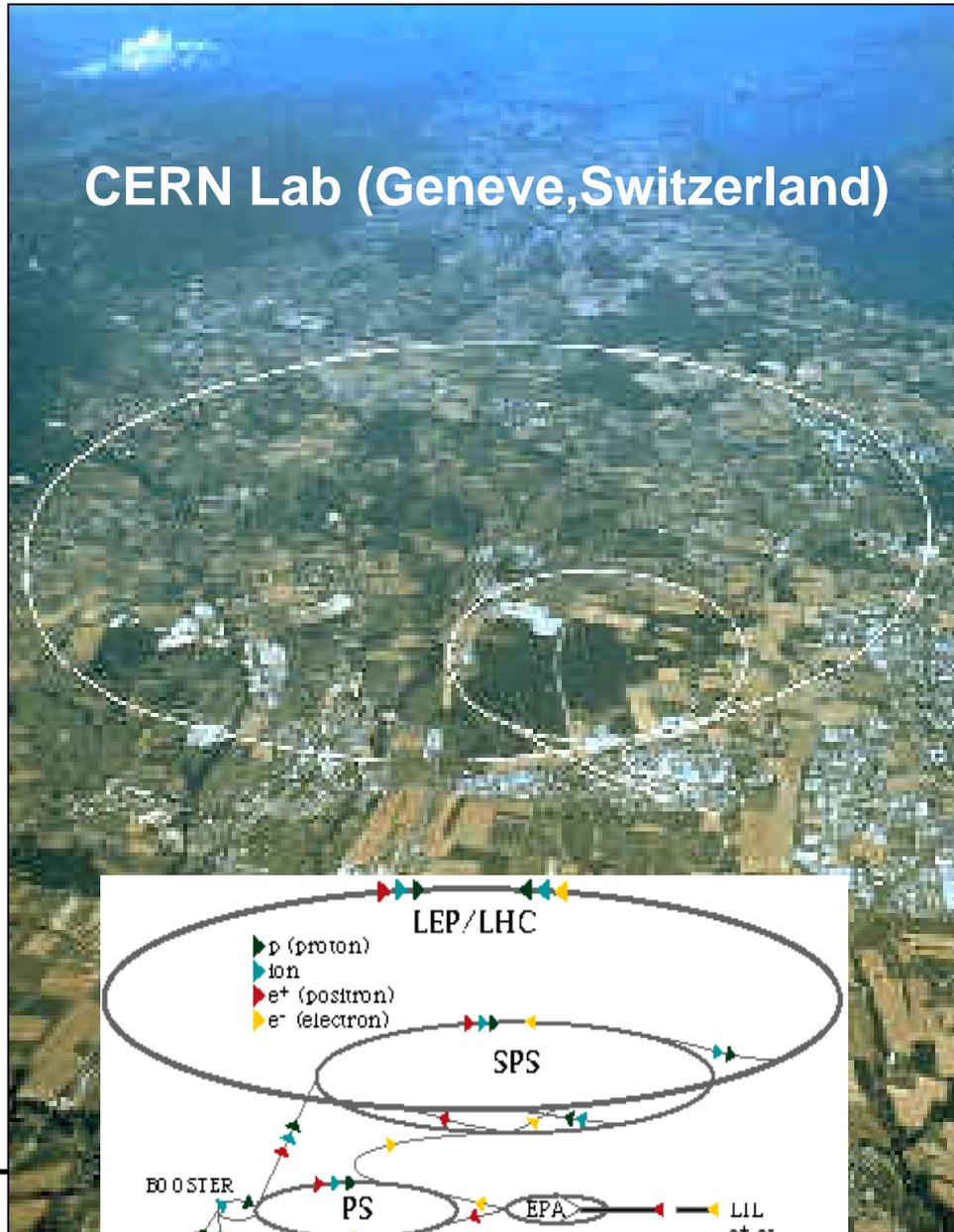
- Física de Partículas: Estudia los constituyentes de la materia y sus fuerzas.



El origen de la masa de todas las partículas está relacionada con una partícula fundamental predicha pero aún no descubierta: **el bosón de Higgs**

Aceleradores y Detectores

CERN Lab (Geneve, Switzerland)



LEP (final en el 2000)
Large Electron Positron Collider
 $E_{cm} = 200 \text{ GeV}$, e^+e^- collisions
Búsqueda Higgs hasta $M=115 \text{ GeV}$

LHC Large Hadron Collider

27 Km de imanes (campo de 8.4 Tesla)

Helio Super-fluido refrigerado a 1.9°K

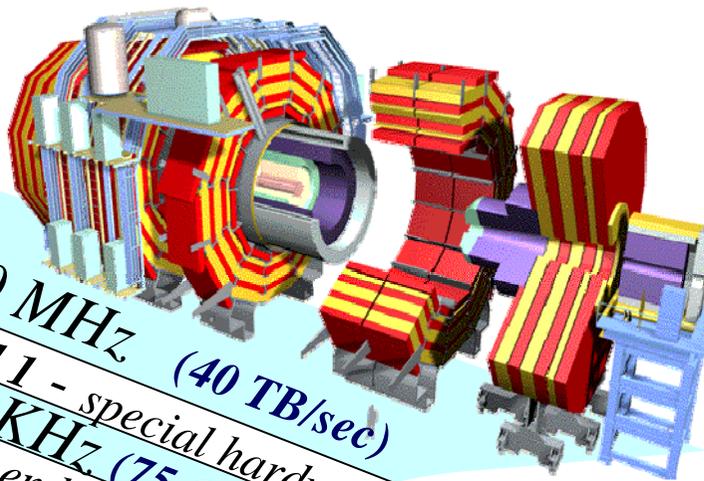
Dos haces opuestos de protones

Energía de colisión 7 + 7 TeV

¡La mayor estructura superconductora del mundo!



¿Realmente se necesita el Grid?



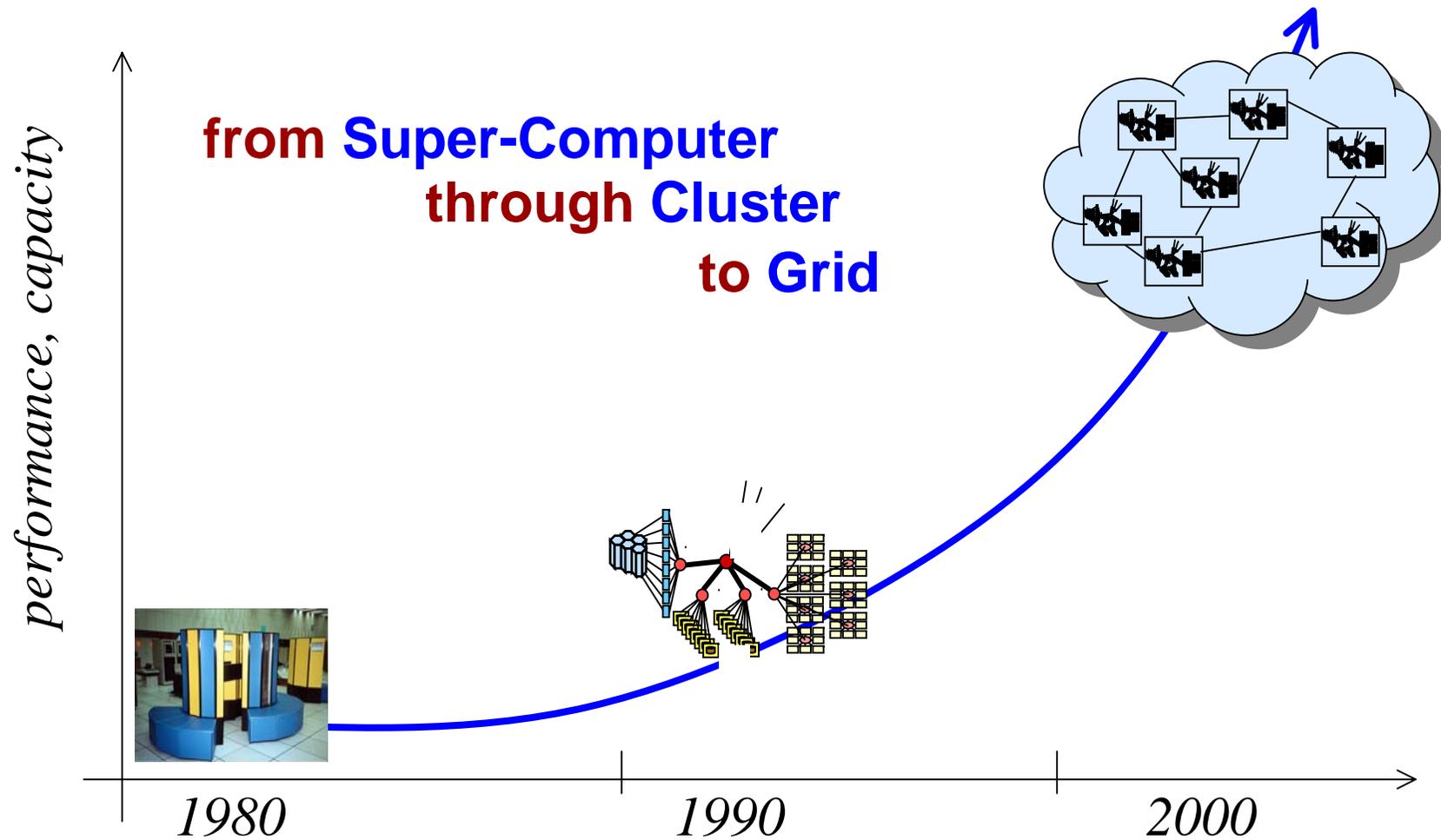
*Necesidades de Almacenamiento:
10 PetaBytes (= 10.000.000 Gbytes)
Necesidades de Procesado:
Equivalente a 200.000 PCs*

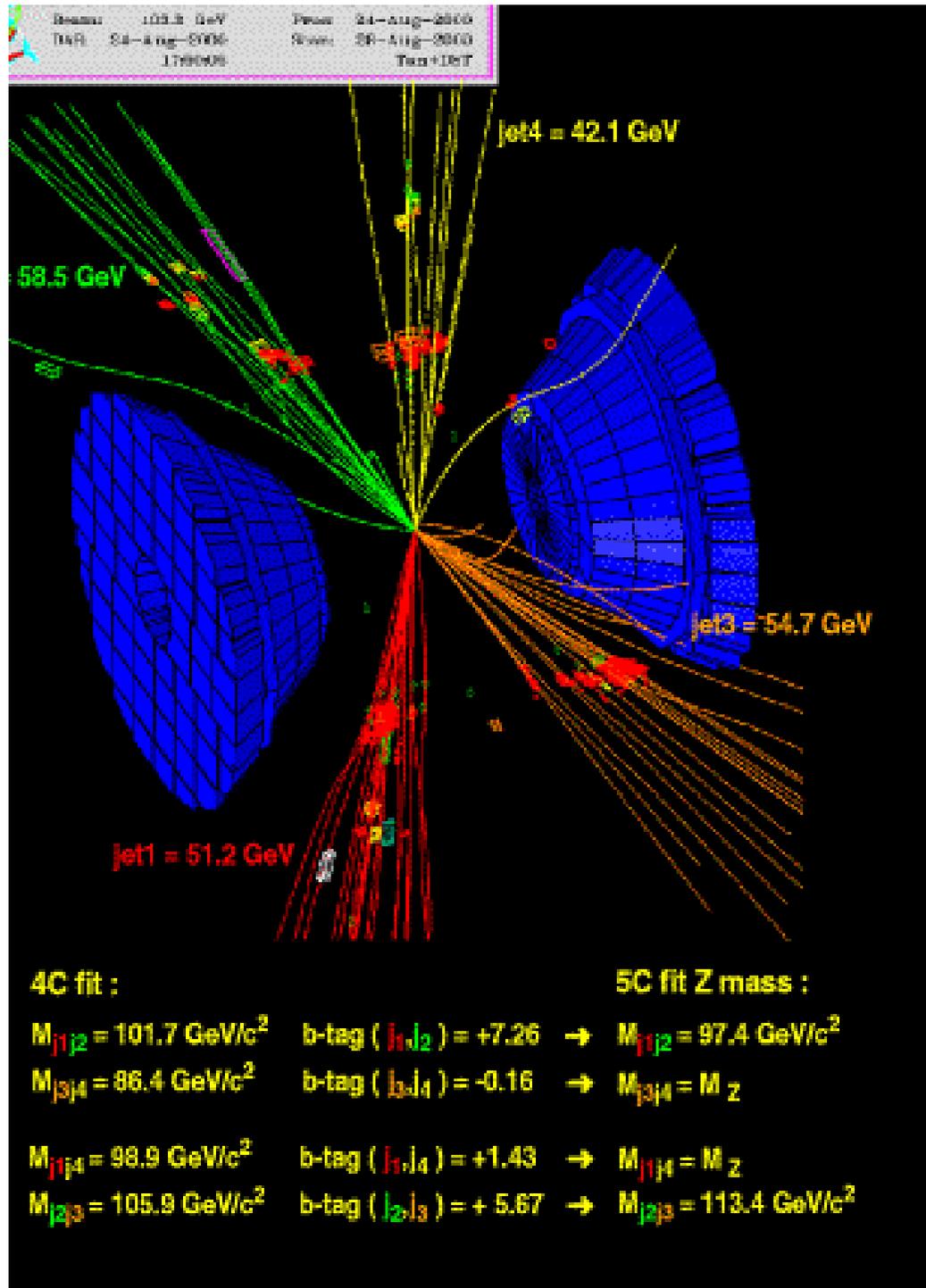


40 MHz (40 TB/sec)
level 1 - special hardware
75 KHz (75 GB/sec)
level 2 - embedded processors
5 KHz (5 GB/sec)
level 3 - PCs
100 Hz
(100 MB/sec)
data recording &
offline analysis

Europa: 267 instituciones, 4603 usuarios
Resto: 208 instituciones, 1632 usuarios

Para LHC, el Grid es el siguiente paso en la evolución a nivel de computación



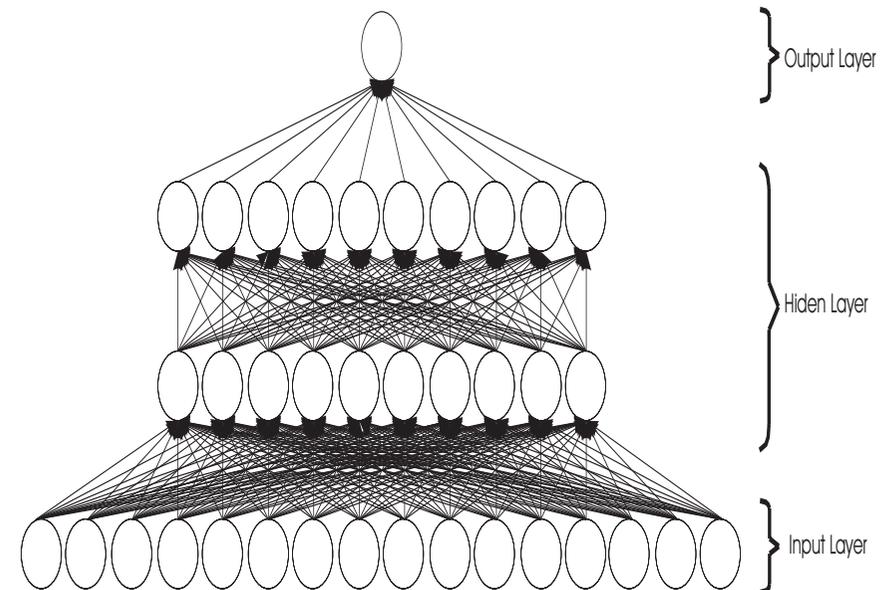


NEURAL NETWORKS
 imprescindibles
 para optimizar la
 búsqueda.

redes neuronales artificiales (ANN)

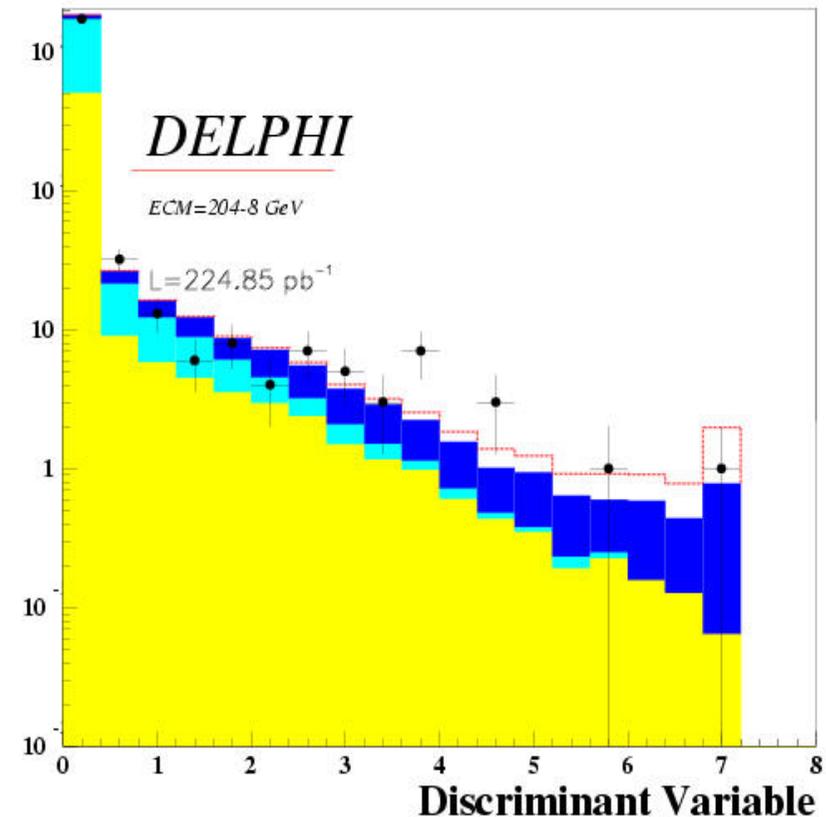
➤ Condiciones:

- Entrenar una ANN con 400k sucesos simulados de LEP, 20000 de los cuales son de señal.
- **Uso de una arquitectura 16-10-10-1 (270 pesos)**
 - **16 input variables**
 - **2 hidden layers with 10 nodes each**
 - **1 output layer, 1=signal, 0=background**
- 1000 épocas de entrenamiento.
- Tamaños similares para las muestras de test



Valor final de la ANN:

- Entrenadas en muestras de MonteCarlo.
 - Higgs generado a diversas masas.
 - Todo tipo de Fondos
 - 10x estadística de datos reales.
- Aplicada sobre datos reales para ordenar en S/B los candidatos a bosón de Higgs
- Proceso de entrenamiento:
 - Minimización del error.
 - Proceso Iterativo
 - Computing Intensive: Entre horas y días para cada entrenamiento

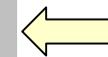
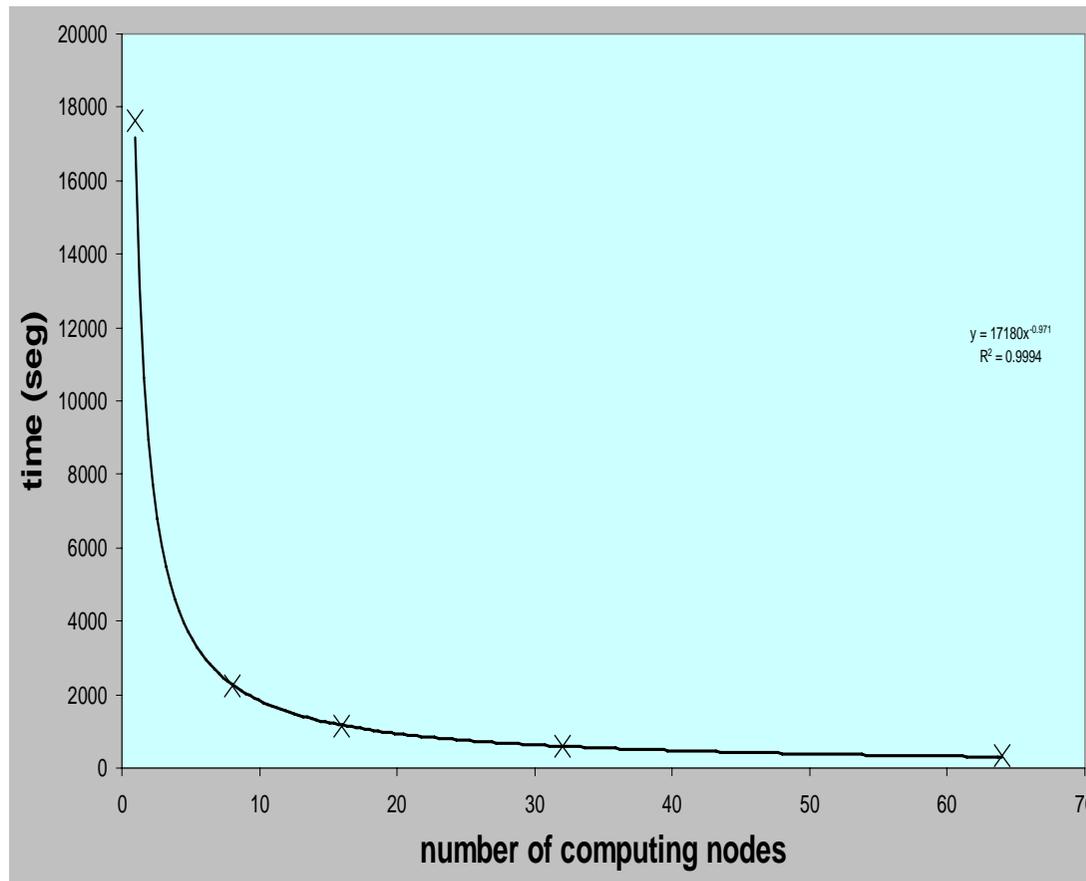


Algoritmo de procesamiento Paralelo

1. El Master lee el fichero de metadata y asigna los datos a los worker nodes. Asimismo asigna pesos random.
2. En cada paso:
 - i. El Master envía los pesos actualizados a los workers.
 - ii. Los workers calculan el error y el gradiente sobre sus datos.
 - iii. El Master suma los errores y gradientes y calcula los nuevos pesos a lo largo de la dirección correspondiente del espacio multidimensional.
 - iv. Se calcula el error sobre una muestra de test para chequear la convergencia.
3. Se repite el paso anterior hasta que converge o se alcanzan el número máximo de pasos.
4. La salida final está constituida por los pesos finales y la función que se obtiene a partir de ellos para aplicar directamente a los datos.

BENEFICIOS DEL GRID

Las técnicas de paralelización reducen el tiempo de espera



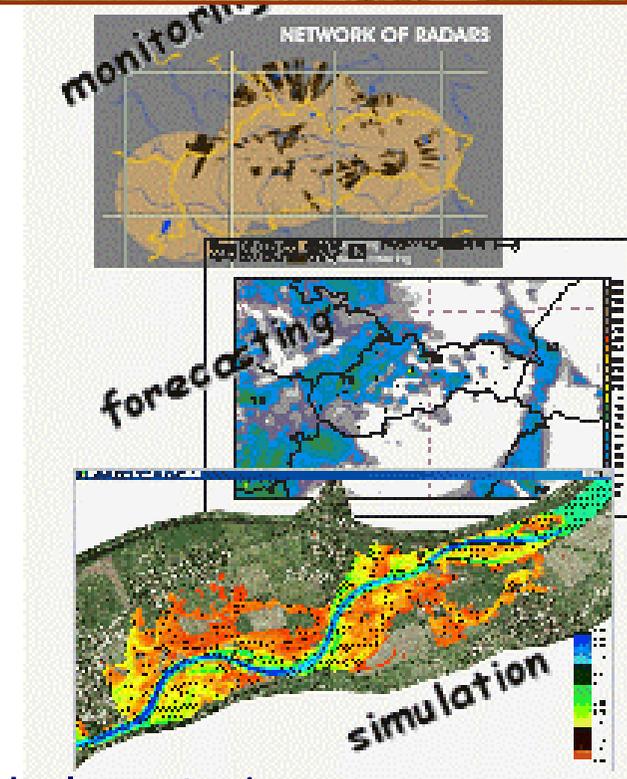
Escala 1/N

412980 events,
16 variables
16-10-10-1 architecture
1000 epochs for training

Reducción de
tiempo: ¡de **5 horas**
a **5 min**
usando **60 nodos!**

2º ejemplo: Prevención de inundaciones

- Acceso en tiempo real a una cantidad de recursos grande
 - Hacer simulaciones hidrológicas
 - Resultados cuantitativos de precipitaciones a partir de datos meteorológicos
 - Los resultados son más exactos cuanto más resolución (~1Km)
 - La correspondiente simulación hidráulica
 - A partir de estos resultados se simula el comportamiento de las cuencas de los ríos y embalses con esa cantidad de precipitación
 - Hay que hacer barridos de parámetros (modelización del suelo, etc...) lo que implica repetir la simulación hidráulica varias veces.

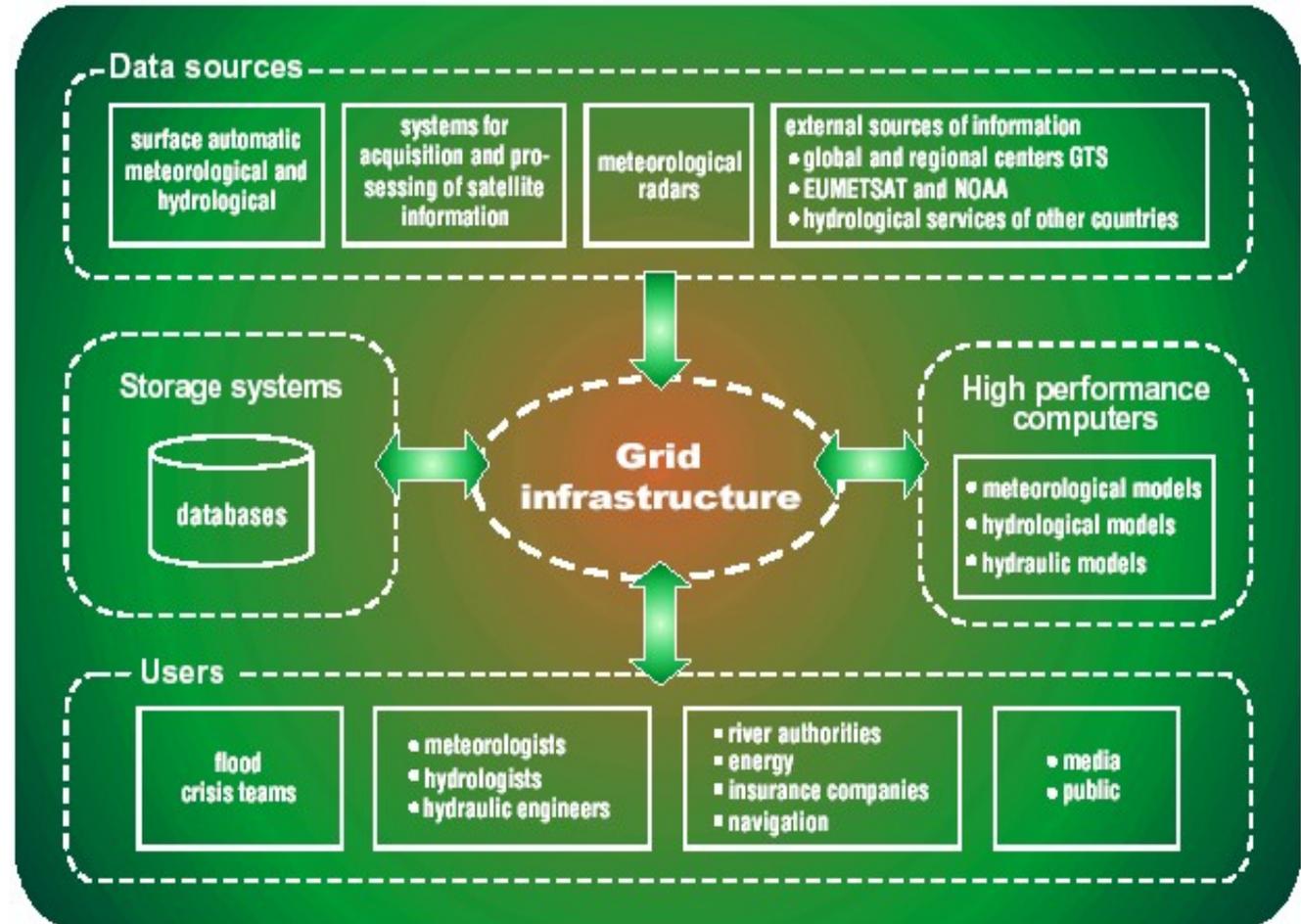


Cascada de simulaciones que necesitan del input de la anterior y de la intervención del usuario para cambiar los parámetros de la simulación

Prevención de inundaciones

➤ Se necesita el Grid en modo interactivo (simulación de “qué ocurriría si..”)

- Conectar de forma instantánea expertos, datos y recursos informáticos necesarios para tomar decisiones rápidas.
- Sistema de alerta altamente automatizado basado en simulaciones hidro-meteorológicas.



Prevención de inundaciones

● Acceso a través de portal Web

- Envío de trabajos.
- **Visualización de resultados**



Authentication
[login](#)
[logout](#)
[check Login State](#)

Chronology
[bmit ALADIN](#)
[ADIN vis](#)
[ADIN animation](#)
[\(experimental\)](#)

Chronology
[bmit HSPF](#)

Analytics
[bmit FESWMS](#)
[SWMS vis](#)

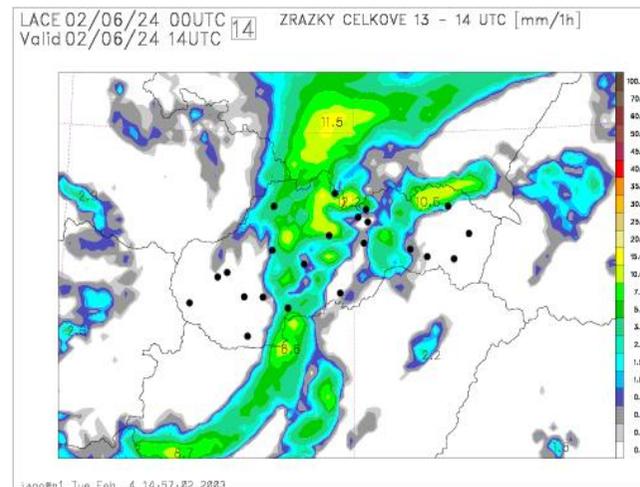
Management
[n Job](#)
[bmit Job](#)
[bmit Test Job](#)
[t Jobs](#)

Management
[t Files](#)

Web X Server

File Transfers
[t File](#)
[t File](#)
[dFTP Put File](#)
[dFTP Get File](#)
[J Party Transfer](#)

Precipitation forecast for 24.06.2002 00 + 14 [View](#) [Prev](#) [Next](#)

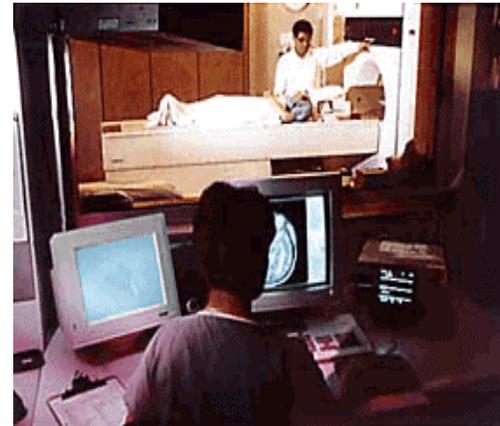


Precipitation from model ALADIN/SLOVAKIA
 integration: 24-06-2002 00 UTC
 period: +13 to +14 h

	[kg]	[km ²]	[kg/m ²]
A: Horna Kysuca	1.19678E+09	451	2.7
B: Dolna Kysuca	1.13437E+09	356	3.2
C: Bystrica	6.96173E+08	219	3.2
D: Varinka	6.27708E+08	168	3.7
E: Hricov-Nosice (prave)	8.82573E+08	385	2.3
F: Hricov-Nosice (lave)	5.30734E+08	255	2.1
G: Rajcianka	1.24472E+09	410	3.0

3^{er} ejemplo: Enfermedades vasculares:

- Intervención quirúrgica para instalar un bypass
- Planificación de la intervención usando una visualización 3D de la estructura arterial del paciente particular
- Simulación inmersiva en 3D para ver las distintas posibilidades de colocar el bypass

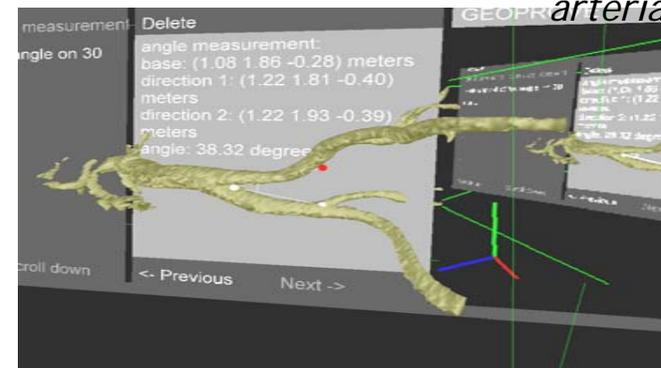


Observación



Stenosis

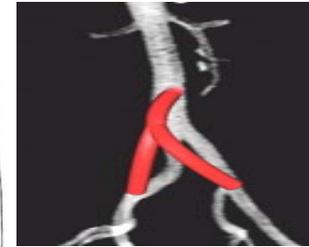
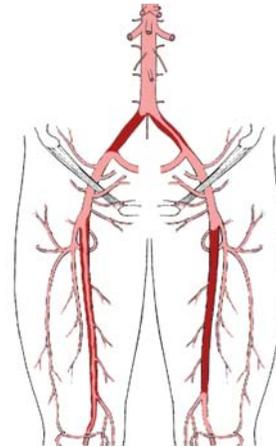
(estrechamiento de una arteria)



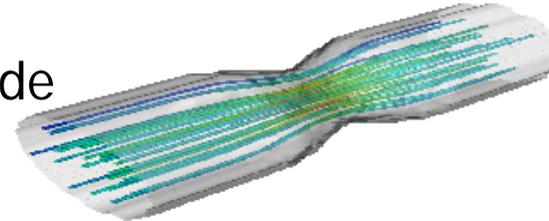
Vista de la estructura arterial en un entorno 3D inmersivo

Objetivo: Simular una reconstrucción vascular

- **Método:** Realidad Virtual interactiva Entorno de:
 - Visualización de datos
 - Definición de intervenciones propuestas
 - Ver los resultados de cada simulación
- Código avanzado de simulación de flúidos



Estructura Arterial a partir de varias simulaciones de bypasses



Simulación de flúidos

- Se necesita el Grid en modo interactivo (el cirujano no espera...)
 - Distribuir simulaciones distintas para cada bypass propuesto

BENEFICIOS DEL GRID

- Los esquemas previos hubiesen requerido grandes recursos dedicados para uso interactivo y disponibilidad local de los datos de input.
- En un entorno Grid el usuario se beneficia del uso de recursos distribuidos reduciéndose la transferencia de grandes volúmenes de datos y encontrando más recursos disponibles.
- En ciertas ocasiones es necesario respuestas inmediatas -ejemplos anteriores- por lo que se necesita un Grid Interactivo!



Interactive European Grid Project

Project acronym
int.eu.grid

Contract number
031857

Instrument
I3

Duration
2 years
may '06-
april '08

Interactive: because researchers need answers in seconds, not in hours.

Grid: easy, intuitive, transparent BUT distributed, powerful, open.

For e-Science: Collaboration.



“Convertir de forma transparente el ordenador del investigador en un supercomputador usando recursos distribuídos ”

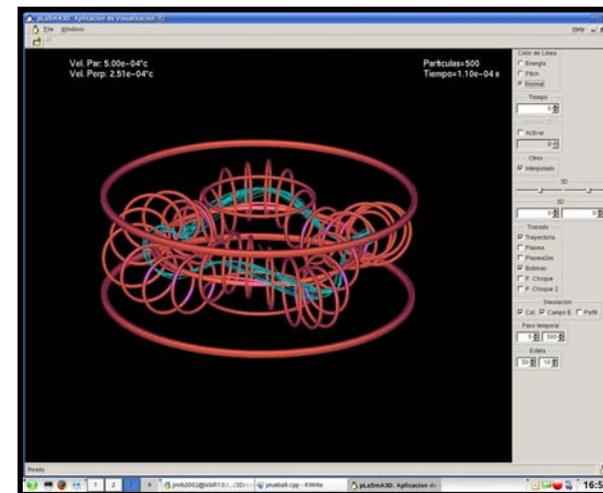
<http://www.interactive-grid.eu>

Coordinator: CSIC, Jesús Marco, IFCA, Santander, SPAIN [marco@ifca.unican.es]

CSIC-IFCA Coord - Spain	LIP Portugal	PSNC Poland	FKG Germany	UAB Spain	CYFRONET Poland	GUP Austria	TCD Ireland	CESGA Spain	II SAS Slovakia	ICM Poland	BIFI Spain	HLRS Germany	

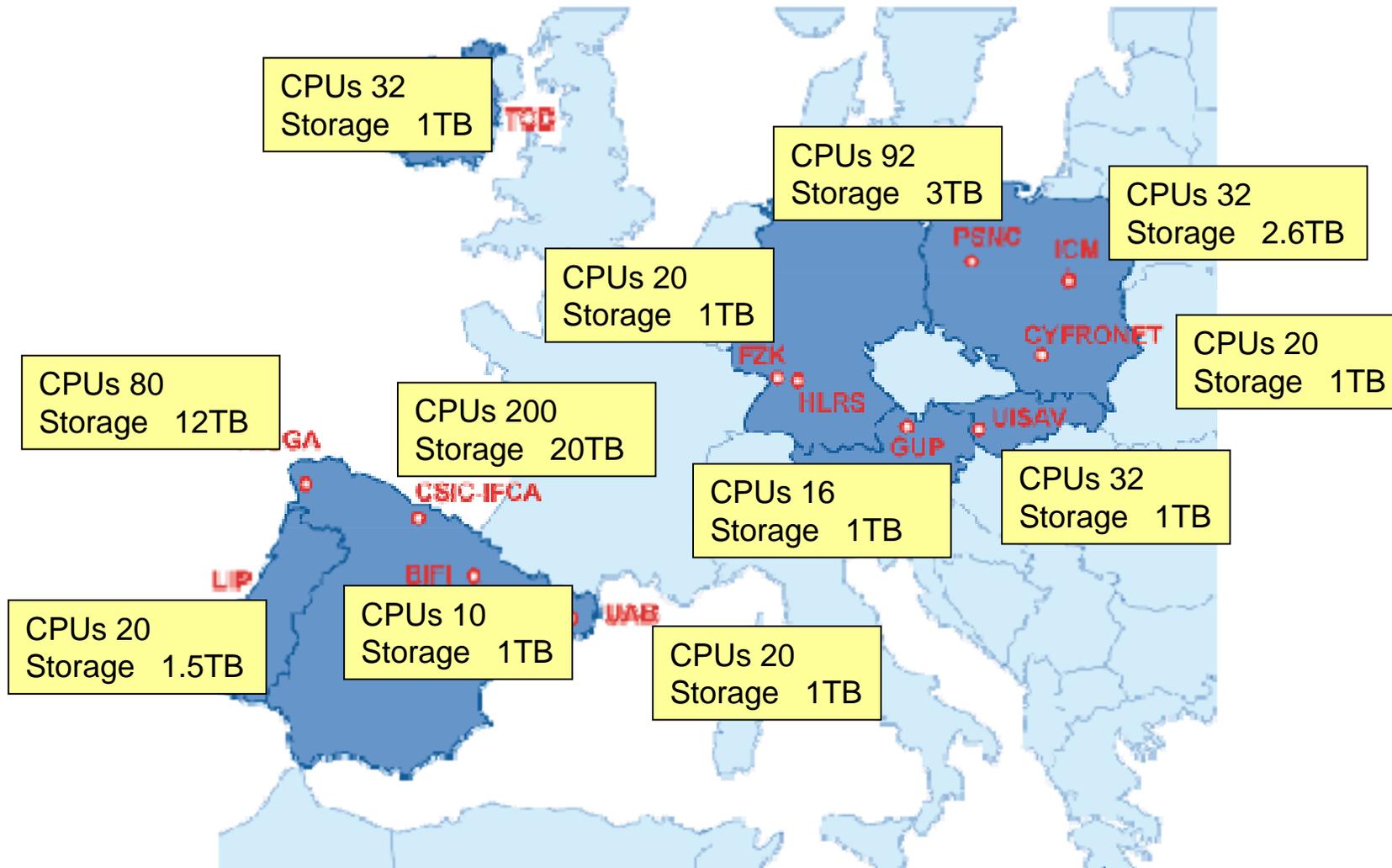
¡Un buen reto!

- Los investigadores deben convencerse de que pueden:
 - Transferir y procesar Tbytes de datos en minutos.
 - Trabajar con algoritmos más complejos sobre grandes estadísticas, testarlos y ajustarlos así como usar técnicas de visualización más potentes.
 - Colaborar a través de la Red de forma eficiente, tanto compartiendo información como discutiendo o presentando remotamente los resultados en entornos avanzados de videoconferencia.



Infrastructure

~700 CPUs and ~ 50 TB online storage



The int.eu.grid Architecture



Migrating
Desktop

Cross
Broker

Resource Discovery
Resource Selection
Job Launching

SERVICES

Information
Index

Replica
Manager

Internet

Globus + DataGrid

CE

WN

WN

WN

Globus + DataGrid

CE

WN

WN

WN

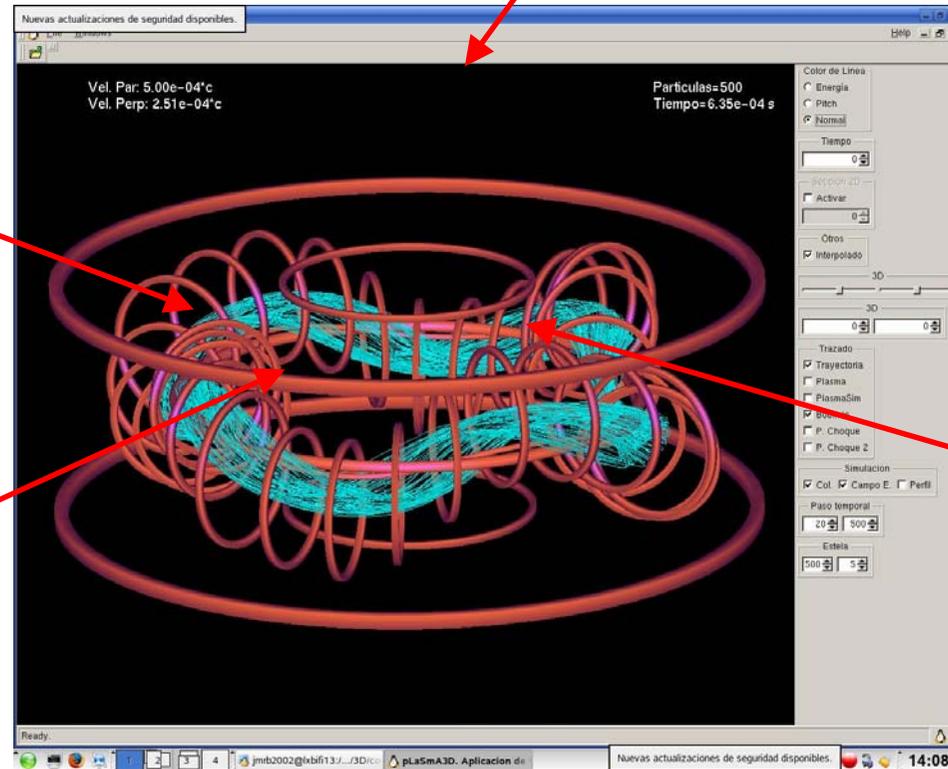
Visualización del Plasma en dispositivos de Fusión Nuclear

Frame $t = 2.43 \times 10^{-4}$ s

Partícula n_0

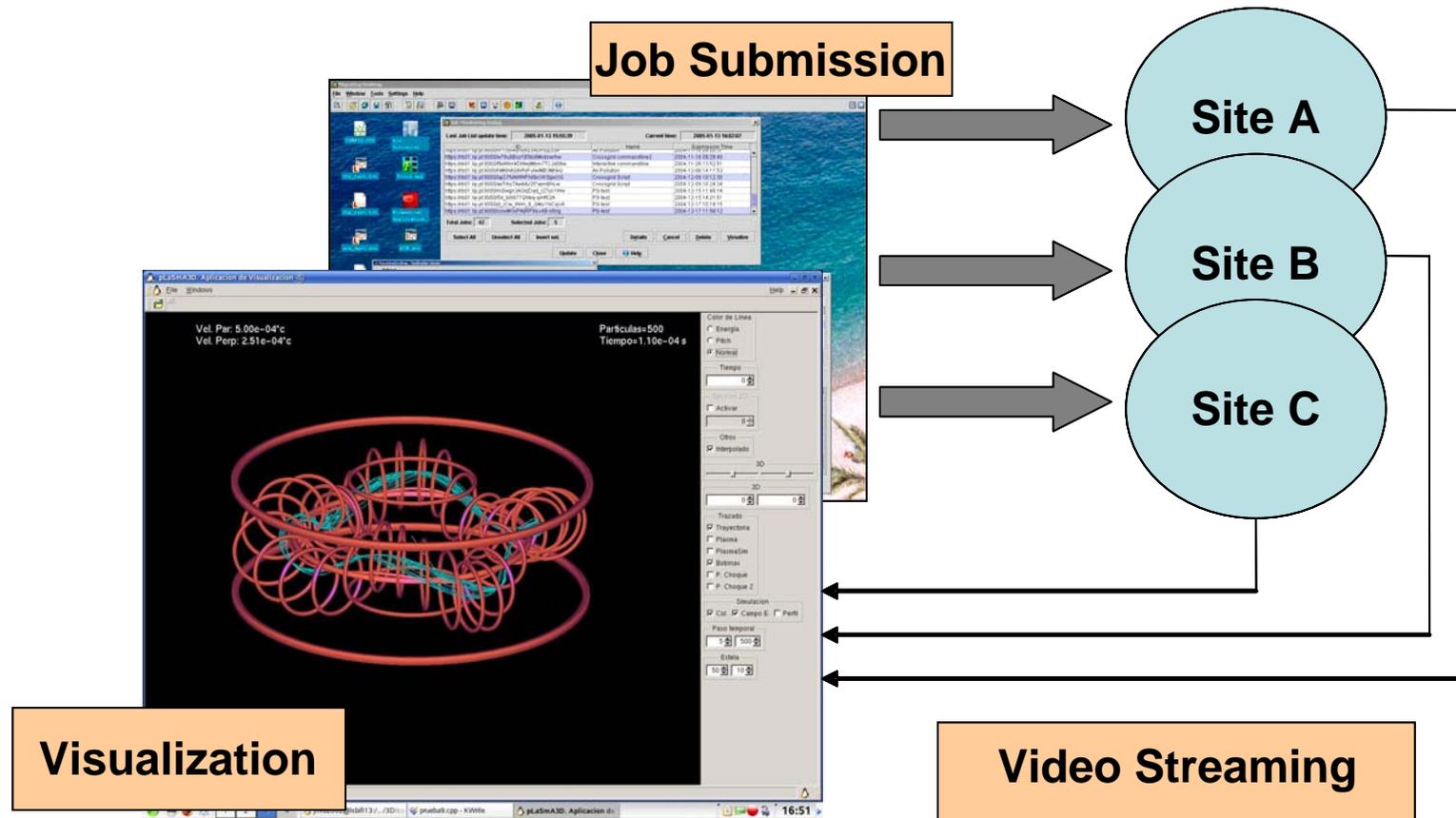
Partícula n_1

Partícula n_2



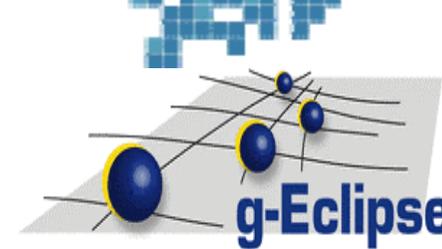
Visualización del Plasma en dispositivos de Fusión Nuclear

~ 4 HORAS



1. E-Ciencia
 2. ¿Qué es el Grid?
 3. Aplicaciones beneficiándose del Grid.
 4. Conclusiones
-

Proyectos Grid en e-Ciencia: muchas ideasy logos!



Mi resumen del GRID

- La idea del Grid es tan buena como difícil de llevar a la práctica
 - Poner al alcance de la comunidad científica recursos de distintos centros con estándares abiertos y una red adecuada de forma transparente:
Sentar al usuario delante de un ordenador de 1000 GHz/ 100 Tbytes
- **Teóricamente tiene sentido:**
 - La red YA está disponible (y evoluciona incluso más rápido que las CPUs!)
 - El middleware básico existe (para una comunicación segura compartiendo recursos)
- **El punto clave: “COMPARTIR y COLABORAR”**
 - Esto es EXTREMADAMENTE difícil, ¿por qué molestarse?
 - Por qué no usar simplemente los recursos de nuestro Instituto? (clusters, centros de SC)?

Veremos un par de ejemplos ahora:

- Conexión a un UI:
 - Iniciar la clave de seguridad
 - Escribir un script sencillo
 - Observar qué máquinas hay disponibles en el testbed EGEE
 - Lanzar el script

- Ejemplo más elaborado de uso de Migrating Desktop:
 - Observar el entrenamiento de una ANN en dos casos:
 - 1 CPU
 - 10 CPU's distribuidas por el testbed de CrossGrid
 - Ver un ejemplo completamente interactivo corriendo en el mismo testbed.