

# Grid Computing

Ramón Jesús Millán Tejedor

[www.ramonmillan.com](http://www.ramonmillan.com)

## 1. Introducción

Las tecnologías *grid* permiten que los ordenadores compartan a través de Internet u otras redes de telecomunicaciones no sólo información, sino también poder de cálculo (**grid computing**) y capacidad de almacenamiento (**grid data**). Es decir, en el *grid* no sólo se comparten contenidos, sino también capacidad de procesamiento, aplicaciones e incluso dispositivos totalmente heterogéneos (sensores, redes, ordenadores, etc.).

El término *grid computing* viene a raíz de la analogía con la red eléctrica (*electric power grid*): nos podemos enchufar al *grid* para obtener potencia de cálculo sin preocuparnos de dónde viene, al igual que hacemos cuando enchufamos un aparato eléctrico. Este innovador paradigma de computación distribuida es propuesto por **Lan Foster** y **Carl Kesselman** a mediados de los años 90, como una revolucionaria técnica para resolver problemas complejos entre diversas organizaciones optimizando costes y tiempo.

Un caso típico de entorno *grid* sería el descrito a continuación. Imaginemos a un científico que quiere ejecutar un programa creado por un colega. Si contase con una infraestructura como la de *grid*, no necesitaría instalar dicho programa en su máquina. En lugar de ello, solicitaría al *grid* que lo ejecutase remotamente en la computadora que almacena ese programa. En el caso de que la máquina estuviese ocupada, el sistema *grid* buscará automáticamente a través de Internet una copia del programa deseado en otras computadoras desocupadas, sin importar que dichas máquinas estuviesen situadas en otro punto del planeta, y lo ejecutaría allí.

Evidentemente, hacer esto no es nada trivial, y surgen preguntas como: ¿Cómo decido qué recursos forman parte del *grid* o no? ¿Cómo decido a qué recurso se dirige un programa o problema con-



creto? ¿Cómo divido un programa que va a ejecutarse en estas condiciones? ¿Cómo sabe una organización que sus recursos no están siendo abusados o empleados de forma maliciosa?... La computación *grid*, precisamente, se centra en solucionar todas estas cuestiones.

## 2. Evolución de la computación

En la computación “tradicional”, basada en **sistemas centralizados**, una organización tenía que hacerse cargo de todas sus necesidades computacionales utilizando sus propios recursos, empleando para ello grandes y costosos servidores con una enorme potencia de cálculo. Es más, no podía hacer paralelismo ni balanceo de carga entre sus distintos servidores, pudiendo haber casos de recursos sobrecargados y ociosos al mismo tiempo; además de tener otros inconvenientes como: falta de escalabilidad, equipos muy caros o falta de fiabilidad y robustez.

Parte de estos problemas son resueltos gracias a los **sistemas paralelos**, enmarcados habitualmente dentro de una única organización. En general, la mayoría de los trabajos sobre computación paralela y optimización están centrados en dos tipos de sistemas paralelos:

- **Sistemas multiprocesador de memoria compartida.**

En este caso, los servicios ofrecidos por el sistema operativo son suficientes para desarrollar programas paralelos: se pueden usar lenguajes secuenciales con bibliotecas de *threads* o llamadas al sistema para escribir aplicaciones *multithread* o multiproceso que utilicen los distintos procesadores del sistema operativo. Otra opción consiste en usar lenguajes paralelos como Java.

- **Sistemas distribuidos basados en clusters o redes de área local.**

En los sistemas distribuidos existe una cantidad enorme de aspectos a considerar, como la red de conexión (Ethernet, ATM, etc.), las distintas topologías (anillo, árbol, malla, etc.), los modelos de programación (paso de mensajes, RPC, memoria compartida distribuida, etc.), las bibliotecas de comunicación (MPI, PVM, sockets, etc.), lenguajes paralelos (Java, C++, etc.), *middleware* (CORBA, DCOM, RMI, etc.), e incluso tecnologías de Internet (XML, SOAP, Servicios Web, etc.).

En cambio, en la **computación grid** intervienen varias organizaciones, cada una con sus propios

recursos computacionales, luego la complejidad se multiplica respecto a los sistemas distribuidos convencionales. Es importante aclarar que aunque el *grid* y el **P2P (Peer-To-Peer)** parecen tener el mismo objetivo final (la organización coordinada de los recursos compartidos dentro de comunidades virtuales), se centran en distintas comunidades, por lo cual tienen distintos requerimientos y siguen caminos evolutivos distintos. Mientras los sistemas *grid* proporcionan varios servicios sofisticados a comunidades relativamente pequeñas y se centran en la integración de recursos muy potentes para proporcionar grandes calidades de servicio dentro de un entorno de confianza limitada, los famosos sistemas P2P tratan con muchos más participantes pero ofrecen servicios más limitados y especializados, están menos preocupados por la calidad del servicio y hacen menos asunciones en cuanto a las relaciones de confianza entre recursos y usuarios.

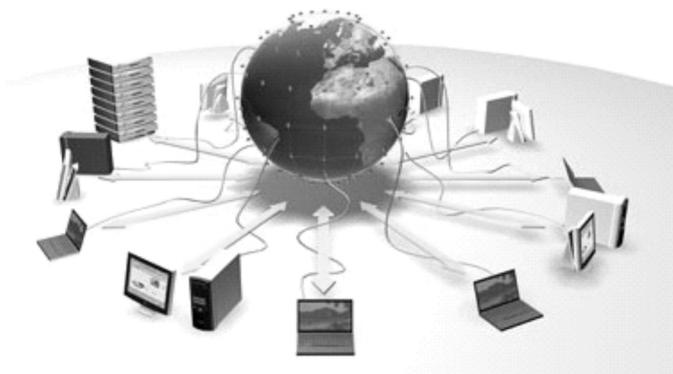


Figura 1. Concepto de grid computing.

## 3. Funcionamiento de la computación grid

El *grid* descansa sobre un software, denominado *middleware*, que asegura la comunicación transparente entre diferentes dispositivos repartidos por todo el mundo. La infraestructura *grid* integra un motor de búsqueda que no sólo encontrará los datos que el usuario necesite, sino también las herramientas para analizarlos y la potencia de cálculo necesaria para utilizarlas. Al final del proceso, el *grid* distribuirá las tareas de computación a cualquier lugar de la red en la que haya capacidad disponible y enviará los resultados al usuario.

El objetivo final del *grid* es poder utilizar **recursos remotos** que nos permitan realizar tareas que no podríamos abordar en nuestra máquina o centro de

trabajo. La idea va más allá del simple intercambio de ficheros, se trata del acceso directo a software, ordenadores y datos remotos, así como el acceso y control de otros dispositivos (sensores, telescopios, etc.). Los recursos son agrupados dinámicamente para resolver problemas concretos, formando organizaciones virtuales. La existencia de **conexiones de red rápidas y fiables** es un requisito indispensable para poder exportar el *grid* a escala mundial y esto es algo que ahora por fin es viable, gracias a la proliferación de las redes de banda ancha (xDSL, HFC, LMDS, UMTS/HSDPA, satélite, etc.).

El verdadero interés del *grid* radica en el **uso eficiente de los recursos**. Se necesitan mecanismos para repartir el trabajo de forma automática y eficiente entre una gran cantidad de recursos, reduciendo las colas de espera de los distintos usuarios. En principio, tendremos información sobre los diferentes trabajos que se han enviado y, ya que todo se está ejecutando en ordenadores, podemos calcular cuál sería la asignación óptima de recursos.

Puesto que los recursos que son compartidos pertenecen a personas muy distintas, la **seguridad** es esencial, y se centra en los siguientes aspectos: política de accesos (qué es lo que se va a compartir, a quién se le permite el acceso, y bajo qué condiciones), autenticación (mecanismos para garantizar la identidad de un usuario o de un recurso concreto) y autorización (procedimiento para averiguar si una determinada operación es consistente con las relaciones que se han definido previamente de cara a compartir recursos). Otro aspecto muy importante es la **estandarización**: todas las aplicaciones que se ejecuten en un *grid* deben poder funcionar en cualquier otro.

## 4. Arquitectura del grid

Habitualmente se describe la arquitectura del *grid* en términos de “capas”, ejecutando cada una de ellas una determinada función. Las capas más altas son las más cercanas al usuario y las inferiores las más próximas a las redes de computación, distinguiendo entre:

### ■ Capa de aplicación.

Formada por todas las aplicaciones de los usuarios, portales y herramientas de desarrollo que soportan esas aplicaciones. Es la capa que ve el usuario y que proporciona el llamado *service-ware*, que recoge las funciones generales de gestión, tales como la contabilidad del uso del *grid* que hace cada usuario.

### ■ Capa de *middleware*.

Responsable de proporcionar herramientas que permiten que los distintos recursos participen de forma coordinada y segura en un entorno *grid* unificado.

### ■ Capa de recursos.

Constituida por los recursos que son parte del *grid*: ordenadores, supercomputadoras, sistemas de almacenamiento, catálogos electrónicos de datos, bases de datos, sensores, etc.

### ■ Capa de red.

Encargada de asegurar la conexión entre los recursos que forman el *grid*.

Para poder hacer todo lo anterior, las aplicaciones que se desarrollen para ser ejecutadas en un ordenador concreto, tendrán que adaptarse para poder invocar los servicios adecuados y utilizar los protocolos correctos. Sin embargo, una vez adaptadas al *grid*, miles de usuarios podrán usar las mismas aplicaciones, utilizando las capas de *middleware* para adaptarse a los posibles cambios en el tejido del *grid*.

## 4.1. *Middleware*

De todas estas capas, la más interesante es el ***middleware***, el auténtico cerebro del *grid*, que se ocupa de las siguientes funciones:

- Encontrar el lugar conveniente para ejecutar la tarea solicitada por el usuario.
- Optimizar el uso de recursos, que pueden estar muy dispersos.
- Organizar el acceso eficiente a los datos.
- Autenticar los diferentes elementos.
- Ejecutar las tareas.
- Monitorizar el progreso de los trabajos en ejecución.
- Gestionar automáticamente la recuperación frente a fallos.
- Avisar cuando se haya terminado la tarea y devolver los resultados.

El *middleware* está formado por muchos programas software; algunos de estos programas actúan como **agentes** (*agents*) y otros como **intermediarios** (*brokers*), negociando entre sí, de forma automática, en representación de los usuarios del *grid* y de los proveedores de recursos. Un elemento fundamental del *middleware* son los **metadatos** (datos sobre los datos), que contienen, entre otras cosas, toda la información sobre el formato de los datos y

dónde se almacenan (a veces en varios sitios distintos). Los agentes individuales presentan los metadatos referidos a los usuarios, datos y recursos. Por otro lado, los intermediarios se encargan de las negociaciones entre máquinas para la autenticación y autorización de los usuarios, de definir los acuerdos de acceso a los datos y recursos y, en caso de que corresponda, el pago por los mismos. Cuando queda establecido el acuerdo, un intermediario planifica las tareas de cómputo y supervisa las transferencias de datos necesarias para acometer cada trabajo concreto. Al mismo tiempo, una serie de agentes supervisores especiales optimizan las rutas a través de la red y monitorizan la calidad del servicio.

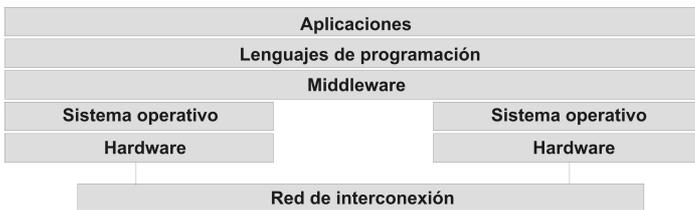


Figura 2. Concepto de middleware.

## 5. Estandarización

Actualmente, los estándares de *grid* los desarrolla el **OGF (Open Grid Forum)** [[www.ggf.org](http://www.ggf.org)], organización nacida de la integración del **GGF (Global Grid Forum)** y la **EGA (Enterprise Grid Alliance)**. El Foro es una comunidad de usuarios, desarrolladores y vendedores liderando los esfuerzos de estandarización mundial del *grid computing*. La comunidad consiste en miles de individuos de la industria e investigación, representando más de 400 organizaciones en más de 50 países. El trabajo del OGF es llevado a cabo a través de grupos de trabajo organizados en comunidades, que desarrollan estándares y especificaciones en cooperación con otras organizaciones de estandarización, desarrolladores de software y usuarios.

La arquitectura de sistemas *grid* basada en Servicios Web, conocida como **OGSA (Open Grid Services Architecture)** [[www.globus.org/ogsa](http://www.globus.org/ogsa)], aparece como la referencia clave para los proyectos en desarrollo *grid*. Los Servicios Web son una arquitectura del W3C (World Wide Web Consortium) [[www.w3c.es](http://www.w3c.es)] para el desarrollo de servicios avanzados a través de interfaces y protocolos consistentes y estándar, como: XML (*eXtensible Markup Language*), SOAP (*Simple Object Access Protocol*), WSDL (*Web Service Description Language*) y UDDI (*Universal Description, Discovery and Integration*). La imple-

mentación más famosa de Servicios Web es .NET, un proyecto de Microsoft para crear una nueva plataforma de desarrollo de software con énfasis en la transparencia de redes, independiente de la plataforma, y que permite un rápido desarrollo de aplicaciones.

La principal implementación de OGSA es el **Globus Toolkit**, una infraestructura de código abierto gratuita desarrollada por la Globus Alliance [[www.globus.org](http://www.globus.org)], que proporciona un conjunto de herramientas de programación Java (librerías, servicios y API) para construir fácil y rápidamente sistemas y aplicaciones *grid* basándose en sus servicios y capacidades básicas, tales como la seguridad, la localización y gestión de los recursos y las comunicaciones. Los principales proyectos relacionados con *grid* se están desarrollando en base al Globus Toolkit.

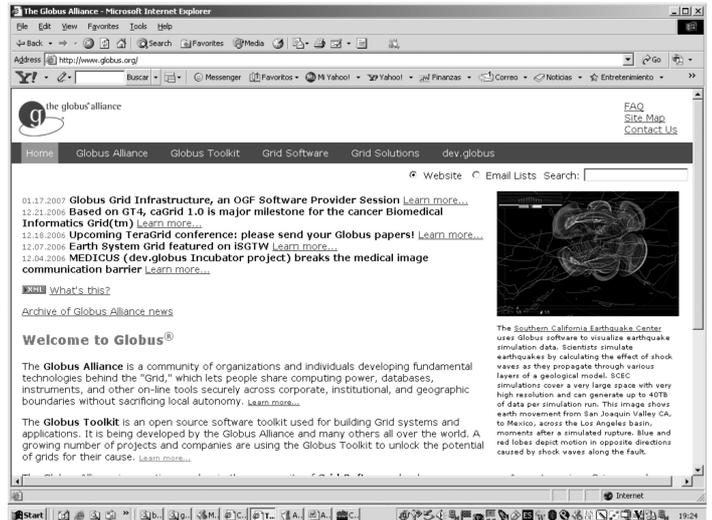


Figura 3. Portal Web de la Globus Alliance.

## 6. Aplicaciones del grid

Las instituciones y organismos más interesadas en el desarrollo del *grid* son, principalmente, las que comparten un objetivo común y que, para poder alcanzarlo, lo más efectivo es compartir sus recursos: Gobiernos y organizaciones internacionales (respuesta a desastres, planificación urbana, etc.), sanidad (análisis rápido de imágenes médicas complejas, etc.), educación (creación de aulas virtuales, teleconferencias, etc.), empresas y grandes corporaciones (cálculos complejos, reuniones virtuales, etc.).

Los beneficios del *grid*, gracias a la integración de recursos distribuidos, están teniendo repercusión en muchísimos campos, de entre los que cabe destacar: medicina (imágenes, diagnosis y tratamiento), ingeniería genética y biotecnología (estudios en genómica y

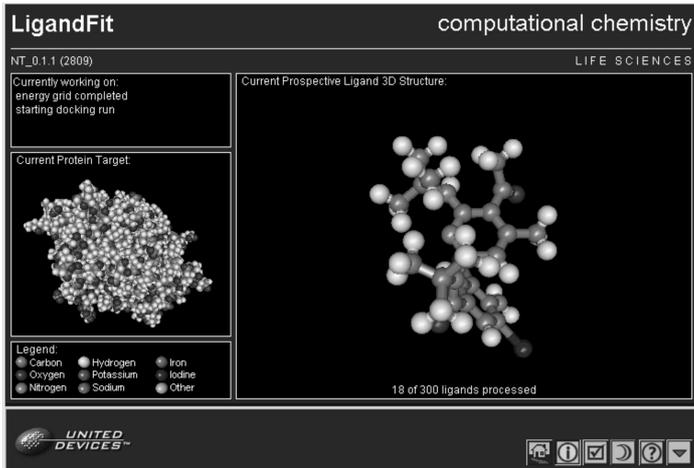


Figura 4. Aplicación del grid en la bioinformática.

proteómica), nanotecnología (diseño de nuevos materiales a escala molecular), ingeniería (diseño, simulación, análisis de fallos y acceso remoto a instrumentos de control), y recursos naturales y medio ambiente (previsión meteorológica, observación del planeta, modelos y predicción de sistemas complejos).

### 6.1. El grid en los centros académicos y de investigación

La computación distribuida empezó a ser escalada a niveles globales con la madurez de Internet en los años 90. Dos proyectos en particular han demostrado que el concepto es totalmente viable y eficiente, incluso más de lo que los expertos auguraban en un principio: Distributed.net y SETI@home.

**Distributed.net** [[www.distributed.net](http://www.distributed.net)] emplea miles de ordenadores distintos para crackear códigos de encriptación (RC5-64, CSC, DES-III, DES-II-1, DES-II-1, RC5-56, etc.). Fundado en 1997, el proyecto ha crecido hasta abarcar hoy en día más de 60.000 usuarios alrededor de todo el mundo. El poder de cómputo de Distributed.net ha ido creciendo hasta llegar a ser el equivalente a más de 160.000 computadoras PII 266 MHz trabajando 24 horas al día, 7 días a la semana y 365 días al año.

**SETI@home** [[boinc.astroseti.org](http://boinc.astroseti.org)] ha sido el proyecto de computación distribuida más popular de la historia, cuyo objetivo era la búsqueda de vida extraterrestre mediante la detección de su tecnología de comunicaciones, buscando patrones que demuestren inteligencia en las ondas de radio procedentes del espacio. Para ello, cualquier persona que quisiera colaborar podía descargarse un salvapantallas gratuito (de este modo aprovechaba los ciclos del ordenador sólo cuando éste no estaba en uso), que instala-

do en su ordenador analiza señales del espacio captadas con el radiotelescopio de Arecibo, en Puerto Rico. El software se hizo público el 17 de mayo de 1999 y, desde entonces hasta su finalización el 15 de diciembre de 2005, más de 5 millones de voluntarios han instalado el programa en su ordenador, se han conseguido un total acumulado de dos millones de años de tiempo de CPU y se han analizado alrededor de 50 TB de datos, convirtiendo al proyecto en el mayor computador virtual de la historia de la humanidad por análisis realizados. En la actualidad, SETI@Home sigue en funcionamiento pero integrado en BOINC (*Berkeley Open Infrastructure for Network Computing*) [[boinc.berkeley.edu](http://boinc.berkeley.edu)], un proyecto basado en recursos de redes abiertas con los mismos principios que el proyecto original. La nueva infraestructura continuará con la búsqueda de señales de radio extraterrestres, pero ahora la potencia de CPU de los participantes se dedicará también a investigar sobre otras tareas, como el cambio climático, la astronomía y la cura de enfermedades.

El sector más involucrado en todo el mundo en la puesta a punto de plataformas *grid* en el desarrollo de aplicaciones adaptadas a esta nueva tecnología es, sin lugar a dudas, el de la investigación. En España las principales universidades e instituciones de investigación han participado intensamente en diversos proyectos sobre *grid*, tanto nacionales como internacionales, entre otros: "IRISGrid" [[irisgrid.rediris.es](http://irisgrid.rediris.es)], "Damien" [[www.hlrs.de/organization/pds/projects/damien/](http://www.hlrs.de/organization/pds/projects/damien/)], "HealthGrid" [[www.healthgrid.org](http://www.healthgrid.org)], "CrossGrid" [[www.eu-crossgrid.org](http://www.eu-crossgrid.org)], "Enabling Grids for e-Science in Europe" [[www.eu-egee.org](http://www.eu-egee.org)], etc. Existen ya varios agentes *grid* disponibles para colaborar activamente en este tipo de proyectos, por ejemplo, en Grid.org [[www.grid.org](http://www.grid.org)] de United Devices, cualquiera puede

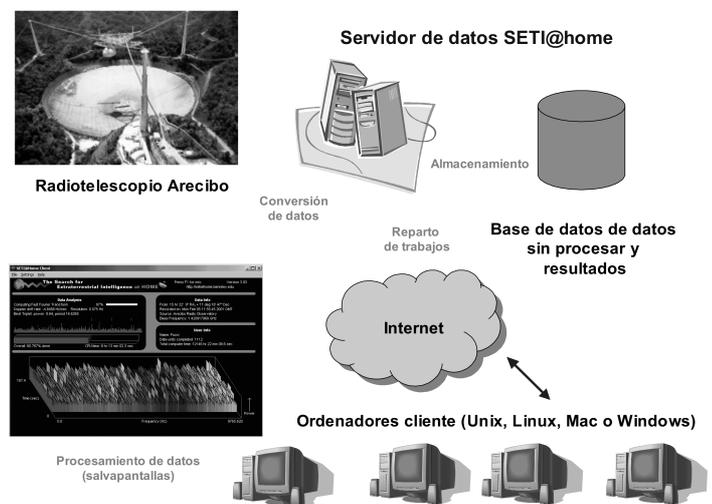


Figura 5: Proyecto SETI@home.

bajarse uno, instalarlo y configurarlo, para colaborar activamente en el desarrollo de esta fascinante tecnología y, a la vez, desempeñar una labor altamente solidaria (por ejemplo, ayudando al descubrimiento de nuevos fármacos y vacunas contra algunas de las enfermedades más terribles de nuestra Era).

## 6.2. El grid en la empresa

Las conexiones de ordenadores en *grid* están dejando de ser sólo una herramienta de las tecnologías de la información para unir recursos informáticos en grades redes de computación de centros académicos y de investigación, para empezar a jugar un papel más relevante en el desarrollo y prestación de servicios en las empresas, que consideran esta forma de computación como una tecnología estratégica para sostener sus futuras operaciones de negocios, caracterizadas por la necesidad de resolver problemas computacionales cada vez más complejos. De hecho, según una encuesta realizada en Norteamérica por Forrester Research [[www.forrester.com](http://www.forrester.com)] en 2005, el 37% de las 149 empresas sondeadas de diversos sectores aseguraban estar desplegando, haber ya implementado o estar adoptando en fase piloto algún tipo de tecnología *grid*. Para Europa, IDC [[www.idcresearch.com](http://www.idcresearch.com)] aportaba hace algo más de un año la cifra de 1.800 millones de dólares como el volumen de negocio que, en 2008, generará el *grid computing* sólo en el segmento de servidores.

Los inicios del desarrollo de la tecnología *grid* se enfocaron, básicamente, en asuntos técnicos, como mantener el flujo de datos unido dentro de una estructura federada, formada por un gran número de ordenadores de centros de investigación interconectados. Esta puesta en práctica ha probado el valor de la tecnología *grid*, y le ha dado la credibilidad necesaria para abrir ahora una nueva fase de desarrollo en la que cobrarán un papel más decisivo las empresas.

Para llevar el *grid* a este nuevo escenario, los grupos de trabajo del OGF (*Open Grid Forum*) están

desarrollando las especificaciones y las mejoras prácticas para su aplicación en general y en el ámbito específico de cada industria. Por ejemplo, hay un grupo de *grid* para las ciencias de la salud que observa los asuntos relacionados con la integración de las tecnologías de la información con estas ciencias dentro de una infraestructura de computación distribuida. Dentro de estos grupos el objetivo se encuentra en el desarrollo de modos de uso y prácticas para cada industria en particular. Un servicio relacionado con el ámbito sanitario podría usar las especificaciones generales en la arquitectura de la red de computación en *grid*, pero también tendría que alcanzar los estándares necesarios para este ámbito y que son exclusivos del mismo.

**United Devices** [[www.ud.com](http://www.ud.com)] es una de las empresas pioneras en productos y servicios *grid*, cuya idea es desarrollar programas que acaben por cubrir todo tipo de áreas de interés para empresas y usuarios, como aplicaciones financieras y de biotecnología, y establecer modelos de negocio en los que se compre la potencia de cálculo de otros usuarios para poder resolver los distintos problemas. Entre sus productos tenemos por ejemplo **Grid MP**, una potente plataforma *grid* basada en protocolos e interfaces estándar (TCP/IP, HTTP, Servicios Web, etc.), para que las empresas puedan emplear sus aplicaciones en un entorno *grid* formado por sus tradicionales plataformas PC ejecutando sistemas operativos Linux o Windows. Mediante Grid MP es mucho más sencillo para las empresas adaptar sus aplicaciones al *grid* que mediante el Globus Toolkit, además de ofrecer servicios mucho más potentes, si bien tiene la desventaja de que no es de código abierto y es necesario por lo tanto comprar el producto. El líder farmacéutico Novartis AG [[www.novartis.com](http://www.novartis.com)], por ejemplo, realizó una inversión de 400.000 dólares en tecnología *grid* basándose en Grid MP, que le ha supuesto un ahorro de más de 2 millones de dólares, consiguiendo mediante la integración de 2.700 ordenadores personales con procesador Intel Pentium 4 una potencia de cálculo superior a los 5 Teraflops, que ha sido fundamental para acelerar el desarrollo de sus nuevos fármacos.

## 7. Referencias

- **Rajkumar Buyya**. "High performance cluster computing". Prentice-Hall, 1999.
- **G. Coulouris, J. Dollimore, T. Kindberg**. "Distributed systems. Concepts and design". Addison-Wesley, 2000.
- **Ignacio García**. "e-Business colaborativo: cómo implantar software libre, servicios Web y el grid computing para ahorrar costes y mejorar las comunicaciones en su empresa". Fundación Confemetal, 2003.
- **J. Joseph y C. Fellenstein**. "Grid computing". Prentice-Hall, 2003.
- **Philip Kerr**. "The grid". Warner Books, 1997.
- **Ramón J. Millán**. "Domine las redes P2P (*Peer-To-Peer*)". Creaciones Copyright, 2006.
- **Daniel Minoli**. "A networking approach to grid computing". Wiley, 2005.